Oracle® Automatic Storage Management

Administrator's Guide

11*g* Release 2 (11.2)

E18951-03

February 2012

ORACLE®

Oracle Automatic Storage Management Administrator's Guide, 11*g* Release 2 (11.2)

E18951-03

Contents

3 Administering Oracle ASM Instances

4 Administering Oracle ASM Disk Groups

9 Administering Oracle ASM with Oracle Enterprise Manager

10 Administering Oracle ACFS with Oracle Enterprise Manager

11 Oracle ASM Configuration Assistant

12   Oracle ASM Command-Line Utility

13 Oracle ACFS Command-Line Tools

A   Creating Database Files in Oracle ASM Using SQL*Plus

B   Oracle ACFS Advanced Topics

Glossary

Index

List of Examples

List of Tables

Preface

The *Oracle Automatic Storage Management Administrator's Guide* describes how to administer Oracle Automatic Storage Management (Oracle ASM) for Oracle databases. This Preface contains the following topics:

- Audience
- Documentation Accessibility
- Related Documents
- Conventions

Audience

The audience for this book includes system administrators, database administrators, and storage administrators. The *Oracle Automatic Storage Management Administrator's Guide* is intended for database and storage administrators who perform the following tasks:

- Administer and manage Oracle ASM
- Configure and administer Oracle ASM

To use this document, you should be familiar with basic Oracle Database concepts and administrative procedures. Also, you might want to review the documentation for Oracle Clusterware and Oracle Real Application Clusters (Oracle RAC). See "Related Documents" on page xxvi.

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 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.

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an

otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

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

Related Documents

For more information, refer to the following Oracle resources:

- *Oracle Database 2 Day DBA*
- *Oracle Database 2 Day + Real Application Clusters Guide*
- *Oracle Database Administrator's Guide*
- *Oracle Database Concepts*
- *Oracle Database Net Services Administrator's Guide*
- *Oracle Clusterware Administration and Deployment Guide*
- *Oracle Real Application Clusters Administration and Deployment Guide*
- Platform-specific guides, including Oracle Database, Oracle Grid Infrastructure, and Oracle Real Application Clusters installation guides

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.

What's New in Oracle Automatic Storage Management?

This chapter describes the new features for Oracle Automatic Storage Management (Oracle ASM).

This chapter contains the following topics:

- Oracle Automatic Storage Management 11g Release 2 (11.2.0.3) New Features
- Oracle Automatic Storage Management 11g Release 2 (11.2.0.2) New Features
- Oracle Automatic Storage Management 11g Release 2 (11.2.0.1) New Features
- Oracle Automatic Storage Management 11g Release 1 (11.1) New Features

> **See Also:** *Oracle Database New Features Guide* for a complete description of the new features in Oracle Database 11*g* release 2 (11.2)

Oracle Automatic Storage Management 11*g* Release 2 (11.2.0.3) New Features

> **Note:** This functionality is available starting with Oracle Automatic Storage Management 11g Release 2 (11.2.0.3).

This section describes the Oracle Automatic Storage Management (Oracle ASM) 11*g* release 2 (11.2.0.3) new features:

- Read-write Oracle ACFS snapshots

  This feature supports read-write snapshots on Oracle ACFS file systems. For information about Oracle ACFS snapshots, refer to "About Oracle ACFS Snapshots" on page 5-8.

  The `acfsutil snap` commands support read-write snapshots. For information about these commands, refer to "Oracle ACFS Command-Line Utilities for Multiple Environments" on page 13-95.

- Support for Oracle ACFS Security and Encryption on Windows

  This feature supports Oracle ACFS Security and Encryption on Windows platforms. For information about security and encryption functionality, refer to "Oracle ACFS Security" on page 5-13 and "Oracle ACFS Encryption" on page 5-15.

- Support for Oracle ACFS Replication and Tagging on Windows

  This feature supports Oracle ACFS Replication and Tagging on Windows platforms. For information about replication and tagging functionality, refer to "Oracle ACFS Replication" on page 5-17 and "Oracle ACFS Tagging" on page 5-19.

- Support for additional file types

  Oracle ACFS supports, without snapshots, RMAN backups, archive logs, and Data Pump dumpsets. For more information, refer to "Overview of Oracle ACFS" on page 5-1.

Oracle Automatic Storage Management 11*g* Release 2 (11.2.0.2) New Features

> **Note:** This functionality is available starting with Oracle Automatic Storage Management 11g Release 2 (11.2.0.2). For release 2 (11.2.0.2), Oracle ACFS encryption, replication, security, and tagging are available on Linux.

This section describes the Oracle Automatic Storage Management (Oracle ASM) 11*g* release 2 (11.2.0.2) new features:

- Oracle ACFS security

  The Oracle ACFS security feature provides realm-based security for Oracle ACFS file systems.

  Oracle ACFS security enables you to create realms to specify security policies for users and groups to determine access on file system objects. The Oracle ACFS security feature provides a finer-grained access control on top of the access control provided by the operating system.

  Oracle ACFS security can use the encryption feature to protect the contents of realm-secured files stored in Oracle ACFS file systems.

  > **Note:** Oracle ACFS security is not supported if the file system has been replicated with Oracle ACFS replication.

  For more information, refer to "Oracle ACFS Security" on page 5-13, "Securing Oracle ACFS File Systems" on page 13-5, and "Oracle ACFS Command-Line Tools for Security" on page 13-64.

- Oracle ACFS encryption

  The Oracle ACFS encryption feature enables you to encrypt data stored on disk (data-at-rest).

  Oracle ACFS encryption protects data in an Oracle ACFS file system in encrypted format to prevent unauthorized use of data in the case of data loss or theft.

  > **Note:** Encryption is not supported if the file system has been replicated with Oracle ACFS replication.

For more information, refer to "Oracle ACFS Encryption" on page 5-15, "Encrypting Oracle ACFS File Systems" on page 13-10, and "Oracle ACFS Command-Line Tools for Encryption" on page 13-89.

- Oracle ACFS replication

  This feature enables replication of Oracle ACFS file systems across the network to a remote site, providing disaster recovery capability for the file system.

  This release of the feature supports only one standby file system for each primary file system.

  Note:

  - Oracle ACFS replication is not supported if the file system has been encrypted with Oracle ACFS encryption.

  - Oracle ACFS replication is not supported if the file system has been prepared for Oracle ACFS security.

  For more information, refer to "Oracle ACFS Replication" on page 5-17, "Replicating Oracle ACFS File Systems" on page 13-13, and "Oracle ACFS Command-Line Tools for Replication" on page 13-53.

- Oracle ACFS tagging

  This feature provides a method for relating a group of files based on a common naming attribute assigned to these files called a tag name. Oracle ACFS Replication can use this tagging feature to select specific files with a unique tag name for replication to a different remote cluster site. This tagging option reduces the need to replicate an entire Oracle ACFS file system.

  For more information, refer to "Oracle ACFS Tagging" on page 5-19, "Tagging Oracle ACFS File Systems" on page 13-11, and "Oracle ACFS Command-Line Tools for Tagging" on page 13-50.

- New configuration wizard for cluster configuration outside of software installation

  This feature provides a wizard to configure an Oracle Grid Infrastructure environment after the installation has completed and has been tested.

  For more information, refer to "Configuring Oracle Grid Infrastructure with the Configuration Wizard" on page 3-24.

- Oracle ACFS functionality on the Solaris operating system

  This feature provides Oracle ACFS functionality on the Solaris operating system.

  For more information, refer to "Oracle ACFS Command-line Tools for the Solaris Environment" on page 13-27.

- Oracle ACFS functionality on the AIX operating system

  This feature provides Oracle ACFS functionality on the AIX operating system.

  For more information, refer to "Oracle ACFS Command-line Tools for the AIX Environment" on page 13-34.

- Oracle ASM Configuration Assistant enhancements

  These enhancements to ASMCA provide support for Oracle ACFS file system security and encryption.

For more information, refer to "Managing Security and Encryption for Oracle ACFS File Systems with Oracle ASM Configuration Assistant" on page 11-13.

Oracle Automatic Storage Management 11*g* Release 2 (11.2.0.1) New Features

This section describes the Oracle Automatic Storage Management (Oracle ASM) 11*g* release 2 (11.2.0.1) new features:

- Oracle Automatic Storage Management Cluster File System (Oracle ACFS)

  Oracle Automatic Storage Management Cluster File System (Oracle ACFS) is a new multi-platform, scalable file system, and storage management design that extends Oracle ASM technology to support all of your application data in both single host and cluster configurations. Additionally, Oracle ACFS provides snapshot functionality for a point-in-time copy of an Oracle ACFS system.

  For more information, see "Overview of Oracle ACFS" on page 5-1.

- Oracle ASM Dynamic Volume Manager

  Oracle ASM Dynamic Volume Manager (Oracle ADVM) provides volume management services and a standard disk device driver interface to clients. File systems and other disk-based applications send I/O requests to Oracle ADVM volume devices as they would to other storage devices on a vendor operating system.

  Oracle Automatic Storage Management Cluster File System (Oracle ACFS) is layered on Oracle ASM through the Oracle ADVM interface. With the addition of Oracle ADVM, Oracle ASM becomes a complete storage solution for both database and non-database file needs.

  For more information, see "Overview of Oracle ASM Dynamic Volume Manager" on page 5-20.

- Oracle Automatic Storage Management Cluster File System Snapshots

  An Oracle ACFS Snapshot is a point-in-time copy of a file system that can provide up to 63 snapshot images. Oracle ACFS Snapshot performs fast creation of persistent Oracle ACFS images at a specific point in time with very low overhead leveraging the Copy-on-Write (COW) technology.

  Even as the file system changes, the snapshot does not, giving you the ability to view the file system as it was at the time the snapshot was created. Initially, snapshots are read-only, which preserves the point in time they captured.

  For more information, see "About Oracle ACFS Snapshots" on page 5-8.

- Oracle ASM Configuration Assistant (ASMCA)

  Oracle ASM Configuration Assistant provides a GUI interface for installing and configuring Oracle ASM instances, disk groups, volumes, and file systems.

  In addition, the ASMCA command-line interface provides functionality for installing and configuring Oracle ASM instances, disk groups, volumes, and file systems in a non-GUI utility.

  For more information about using Oracle ASM Configuration Assistant, see Chapter 11, "Oracle ASM Configuration Assistant".

- ASMCMD Enhancements

This feature extends Oracle ASM Command Line Utility (ASMCMD) to provide full functionality, so that any operation that can be performed with SQL commands can be performed with ASMCMD. The added functionality includes the following areas:

- Disk, failure group, and disk group operations

- Disk group attribute operations

- Oracle ASM File Access Control user and group operations

- Template operations

- Oracle ASM instance operations

- File operations

- Oracle ASM volume operations

Changes were also made to standardize ASMCMD command-line and command options.

For more information about ASMCMD commands, see "Types of ASMCMD Commands" on page 12-1.

- Intelligent Data Placement

The Intelligent Data Placement feature enables you to specify disk regions on Oracle ASM disks to ensure that frequently accessed data is placed on the outermost (hot) tracks which provide higher performance.

For more information, see "Intelligent Data Placement" on page 4-16.

- Specifying the Sector Size for Disk Drives

Oracle ASM provides the capability to specify a sector size of 512 bytes or 4096 kilobytes with the SECTOR_SIZE disk group attribute when creating disk groups. Oracle ASM provides support for 4 KB sector disk drives without a performance penalty.

For information about setting the sector size for disk drives, see "Specifying the Sector Size for Drives" on page 4-9.

> **See Also:** *Oracle Database SQL Language Reference* for information about the disk group attributes and the CREATE DISKGROUP SQL statement

- Disk Group Rename

The renamedg tool enables you to change the name of a cloned disk group.

For information about renaming disk groups with renamedg, see "Renaming Disks Groups" on page 4-47.

- Oracle ASM File Access Control

Oracle ASM on Linux and UNIX platforms implements access control on its files to isolate different database instances from each other and prevent accidental access that could lead to the corruption of data files. Oracle ASM implements new SQL statements and ASMCMD commands to grant, modify, and deny file permissions. The new security model and syntax is consistent with those that have been implemented for the objects represented in an Oracle database.

This feature enables multiple database instances to store their Oracle ASM files in the same disk group and be able to consolidate multiple databases while

preventing an unauthorized database instance from accessing and overwriting files of a different instance.

For more information about Oracle ASM File Access Control, see "Managing Oracle ASM File Access Control for Disk Groups" on page 4-40, "Managing Oracle ASM File Access Control with Oracle Enterprise Manager" on page 9-16, and "ASMCMD File Access Control Commands" on page 12-53.

- Oracle Cluster Registry (OCR) and Voting Files stored in Oracle ASM

  This feature enables storage of Oracle Cluster Registry (OCR) and the voting file in Oracle ASM disk groups. This functionality enables Oracle ASM to provide a unified storage solution, storing all the data for the clusterware and the database, without the need for third-party volume managers or cluster file systems.

  For more information, see "Oracle Cluster Registry and Voting Files in Oracle ASM Disk Groups" on page 4-11.

  > **See Also:** *Oracle Clusterware Administration and Deployment Guide* for information about administering voting files and Oracle Cluster Registry (OCR)

- Oracle Restart

  Oracle Restart enables the startup and automatic restart of single-instance Oracle Database through the Cluster Ready Services (CRS) component of Oracle Clusterware on a single server. This provides higher availability and automated management of single-instance Oracle Database and Oracle ASM instances.

  For more information, see "Using Oracle Restart" on page 3-13.

  > **See Also:**
  >
  > - *Oracle Database Administrator's Guide* for information about configuring and administering Oracle Restart
  >
  > - *Oracle Real Application Clusters Administration and Deployment Guide* for information about automatically restarting single-instance databases residing on an Oracle RAC node

- Oracle Enterprise Manager Support of Oracle ACFS

  Oracle Enterprise Manager has enhanced Oracle ASM support to enable you to manage Oracle Automatic Storage Management Cluster File System (Oracle ACFS) technology and Oracle ASM Dynamic Volume Manager. These features operate as a comprehensive management solution to extend Oracle ASM technology to support all your application data files in database and non-database and in both single host and cluster configurations.

  Oracle Enterprise Manager supports the following:

  – Create, modify, and delete Oracle ACFS

  – Monitor space usage and performance features of Oracle ACFS

  – Support Oracle ASM Dynamic Volume Manager features as required by Oracle ACFS

  For more information about using Oracle Enterprise Manager to administer Oracle ASM, see Chapter 10, "Administering Oracle ACFS with Oracle Enterprise Manager".

- Oracle Enterprise Manager Database Control Enhancements for Improved Oracle ASM Manageability

  Oracle Enterprise Manager has been enhanced to manage Oracle ASM features, such as Oracle ASM File Access Control, OCR and voting files in Oracle ASM, disk resync, Oracle ASM rolling migrations, disk regions, and Oracle ASM manageability and infrastructure.

  For more information about using Oracle Enterprise Manager to administer Oracle ASM, see Chapter 9, "Administering Oracle ASM with Oracle Enterprise Manager".

- Oracle Enterprise Manager Oracle ASM Support Workbench

  Oracle Enterprise Manager Oracle ASM Support Workbench has been enhanced to help diagnose and package incidents to Oracle Support Services for Oracle ASM instances. This feature extends the benefit of Oracle Enterprise Manager Support Workbench to Oracle ASM by helping you package all necessary diagnostic data for incidents.

  For more information about using Oracle Enterprise Manager to administer Oracle ASM, see "Oracle ASM Support Workbench" on page 9-25.

  > **See Also:**
  > - *Oracle Database Administrator's Guide* for information about viewing problems with Oracle Enterprise Manager Support Workbench
  > - *Oracle Database 2 Day DBA* for information about accessing Oracle Enterprise Manager Support Workbench home page and viewing problems using Oracle Enterprise Manager Support Workbench

Oracle Automatic Storage Management 11*g* Release 1 (11.1) New Features

This section describes the Oracle Automatic Storage Management (Oracle ASM) 11*g* release 1 (11.1) new features:

- This book is new for Oracle Database 11*g* release 1 (11.1) and it is the primary information source for Oracle Automatic Storage Management features.

- Oracle ASM Fast Mirror Resync

  Oracle ASM fast mirror resync quickly resynchronizes Oracle ASM disks within a disk group after transient disk path failures if the disk drive media is not corrupted. Any failures that render a failure group temporarily unavailable are considered transient failures. Disk path malfunctions, such as cable disconnections, host bus adapter or controller failures, or disk power supply interruptions, can cause transient failures.

  For more information about Oracle ASM fast mirror resync, see "Oracle ASM Fast Mirror Resync" on page 4-28.

- Oracle ASM Rolling Upgrade

  You can now place an Oracle ASM Cluster in rolling upgrade mode, which enables you to operate with mixed Oracle ASM versions starting with Oracle Database 11*g* release 1 (11.1) and later. Consequently, Oracle ASM nodes can be independently upgraded or patched without affecting database availability.

For more information about Oracle ASM rolling upgrade, see "Using Oracle ASM Rolling Upgrade" on page 3-24.

- New SYSASM Privilege and OSASM operating system group for Oracle ASM Administration

  This feature introduces a new SYSASM privilege that is specifically intended for performing Oracle ASM administration tasks. Using the SYSASM privilege instead of the SYSDBA privilege provides a clearer division of responsibility between Oracle ASM administration and database administration.

  OSASM is a new operating system group that is used exclusively for Oracle ASM. Members of the OSASM group can connect as SYSASM using operating system authentication and have full access to Oracle ASM.

  For more information about the SYSASM privilege, see "Authentication for Accessing Oracle ASM Instances" on page 3-25.

- Oracle ASM Scalability and Performance Enhancements

  Oracle ASM file extent management has been enhanced to improve performance and to use significantly less system global area (SGA) memory to store file extents. When Oracle ASM files increase in size, the size of each new extent also increases automatically so that fewer extent pointers are required to describe the file. This feature improves performance when accessing Oracle ASM files that are 20 GB and larger, up to 128 TB. Very large databases (VLDBs) often require these large file sizes.

  For more information about scalability and performance enhancements, see "Extents" on page 1-7.

- Oracle ASM Command Line Utility (ASMCMD) Commands and Options

  ASMCMD has the following new commands: lsdsk, md_backup, md_restore and remap. In addition, you can use new options for the ls and lsdg commands.

  For more information about new and enhanced ASMCMD commands, "Types of ASMCMD Commands" on page 12-1.

- Enhancements to SQL statements to manage Oracle ASM disk groups

  The ALTER DISKGROUP, CREATE DISKGROUP, and DROP DISKGROUP SQL statements have been enhanced with additional syntax options, including:

  – Syntax that sets various attributes of a disk group

  – Syntax for checking the consistency of disk groups, disks, and files in an Oracle ASM environment

  – Syntax options when mounting a disk group.

  – Syntax that takes a disk offline for repair and then brings it back online.

  – Syntax that drops a disk group that can no longer be mounted by an Oracle ASM instance.

  For more information about administering disk groups with the ALTER DISKGROUP, CREATE DISKGROUP, and DROP DISKGROUP SQL statements, see Chapter 4, "Administering Oracle ASM Disk Groups".

  > **See Also:** *Oracle Database SQL Language Reference* for information about the disk group attributes and the CREATE DISKGROUP SQL statement

- New Attributes for Disk Group Compatibility

  To enable the new Oracle ASM features, you can use two new disk group compatibility attributes, COMPATIBLE.RDBMS and COMPATIBLE.ASM. These attributes specify the minimum software version that is required to use disk groups for the database and for Oracle ASM respectively. This feature enables heterogeneous environments with disk groups from both Oracle Database 10*g* and Oracle Database 11*g*.

  For more information about disk group compatibility, see "Disk Group Compatibility" on page 4-33.

- Oracle ASM Preferred Read Failure Groups

  This feature is useful in extended clusters where remote nodes have asymmetric access with respect to performance. The feature enables more efficient use of network resources by eliminating the use of the network to perform read operations.

  Oracle ASM in Oracle Database 10*g* always reads the primary copy of a mirrored extent set. In Oracle Database 11*g*, when you configure Oracle ASM failure groups it might be more efficient for a node to read from a failure group that is closest to the node, even if it is a secondary extent. You can configure your database to read from a particular failure group extent by configuring preferred read failure groups.

  For more information about preferred read failure groups, see "Preferred Read Failure Groups" on page 4-30.

- Oracle ASM Fast Rebalance

  Rebalance operations that occur while a disk group is in RESTRICTED mode eliminate the lock and unlock extent map messaging between Oracle ASM instances in Oracle RAC environments, improving overall rebalance throughput.

  For more information about Oracle ASM fast rebalance, see "About Restricted Mode" on page 3-16.

1

Introduction to Oracle Automatic Storage Management

This chapter describes Oracle Automatic Storage Management (Oracle ASM) concepts and provides an overview of Oracle ASM features. This chapter contains the following topics:

- Overview of Oracle Automatic Storage Management
- Understanding Oracle ASM Concepts
- Understanding Oracle ASM Disk Group Administration

For a list of the terms that are used in the *Oracle Automatic Storage Management Administrator's Guide* and their definitions, refer to the Glossary in this guide.

> **See Also:** The Oracle Cloud Storage page on the Oracle Technology Network Web site at
> http://www.oracle.com/technetwork/database/cloud-storage/index.html for more information about Oracle ASM

Overview of Oracle Automatic Storage Management

Oracle ASM 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 ASM is Oracle's recommended storage management solution that provides an alternative to conventional volume managers, file systems, and raw devices.

Oracle ASM uses **disk group**s to store data files; an Oracle ASM disk group is a collection of disks that Oracle ASM manages as a unit. Within a disk group, Oracle ASM exposes a file system interface for Oracle database files. The content of files that are stored in a disk group is evenly distributed to eliminate hot spots and to provide uniform performance across the disks. The performance is comparable to the performance of raw devices.

You can add or remove disks from a disk group while a database continues to access files from the disk group. When you add or remove disks from a disk group, Oracle ASM automatically redistributes the file contents and eliminates the need for downtime when redistributing the content. For information about administering disk groups, see Chapter 4, "Administering Oracle ASM Disk Groups".

The Oracle ASM volume manager functionality provides flexible server-based mirroring options. The Oracle ASM normal and high redundancy disk groups enable two-way and three-way mirroring respectively. You can use external redundancy to

enable a Redundant Array of Independent Disks (RAID) storage subsystem to perform the mirroring protection function.

Oracle ASM also uses the Oracle Managed Files (OMF) feature to simplify database file management. OMF automatically creates files in designated locations. OMF also names files and removes them while relinquishing space when tablespaces or files are deleted.

Oracle ASM reduces the administrative overhead for managing database storage by consolidating data storage into a small number of disk groups. The smaller number of disk groups consolidates the storage for multiple databases and provides for improved I/O performance.

Oracle ASM files can coexist with other storage management options such as raw disks and third-party file systems. This capability simplifies the integration of Oracle ASM into pre-existing environments.

Oracle Automatic Storage Management Cluster File System (Oracle ACFS) is a multi-platform, scalable file system, and storage management technology that extends Oracle ASM functionality to support customer files maintained outside of Oracle Database. The Oracle ASM Dynamic Volume Manager (Oracle ADVM) provides volume management services and a standard disk device driver interface to clients. For information about Oracle ACFS and Oracle ADVM, see Chapter 5, "Introduction to Oracle ACFS".

Oracle Enterprise Manager includes a wizard that enables you to migrate non-Oracle ASM database files to Oracle ASM. Oracle ASM also has easy to use management interfaces such as SQL*Plus, the Oracle ASM Command Line Utility (ASMCMD) command-line interface, Oracle ASM Configuration Assistant, and Oracle Enterprise Manager. For information about using Oracle Enterprise Manager, see Chapter 9, "Administering Oracle ASM with Oracle Enterprise Manager". For information about Oracle ASM Configuration Assistant, see Chapter 11, "Oracle ASM Configuration Assistant". For information about the ASMCMD command-line interface, see Chapter 12, "Oracle ASM Command-Line Utility".

> **See Also:** *Oracle Database Administrator's Guide* for information about Oracle Database structure and storage

Understanding Oracle ASM Concepts

This section describes concepts for the key Oracle ASM components and it contains the following topics:

- About Oracle ASM Instances
- About Oracle ASM Disk Groups
- About Mirroring and Failure Groups
- About Oracle ASM Disks
- About Oracle ASM Files

For information about preparing your storage environment, see Chapter 2, "Considerations for Oracle ASM Storage".

About Oracle ASM Instances

An Oracle ASM instance is built on the same technology as an Oracle Database instance. An Oracle ASM instance has a System Global Area (SGA) and background processes that are similar to those of Oracle Database. However, because Oracle ASM

performs fewer tasks than a database, an Oracle ASM SGA is much smaller than a database SGA. In addition, Oracle ASM has a minimal performance effect on a server. Oracle ASM instances mount disk groups to make Oracle ASM files available to database instances; Oracle ASM instances do not mount databases. For information about managing an Oracle ASM instance, see Chapter 3, "Administering Oracle ASM Instances".

Oracle ASM is installed in the Oracle Grid Infrastructure home before Oracle Database is installed in a separate Oracle home. Oracle ASM and database instances require shared access to the disks in a disk group. Oracle ASM instances manage the metadata of the disk group and provide file layout information to the database instances.

Oracle ASM metadata is the information that Oracle ASM uses to control a disk group and the metadata resides within the disk group. Oracle ASM metadata includes the following information:

- The disks that belong to a disk group

- The amount of space that is available in a disk group

- The filenames of the files in a disk group

- The location of disk group data file **extent**s

- A redo log that records information about atomically changing metadata blocks

- Oracle ADVM volume information

Oracle ASM instances can be clustered using Oracle Clusterware; there is one Oracle ASM instance for each cluster node. If there are several database instances for different databases on the same node, then the database instances share the same single Oracle ASM instance on that node.

If the Oracle ASM instance on a node fails, then all of the database instances on that node also fail. Unlike a file system driver failure, an Oracle ASM instance failure does not require restarting the operating system. In an Oracle RAC environment, the Oracle ASM and database instances on the surviving nodes automatically recover from an Oracle ASM instance failure on a node.

Figure 1–1 shows a single node configuration with one Oracle ASM instance and multiple database instances. The Oracle ASM instance manages the metadata and provides space allocation for the Oracle ASM files. When a database instance creates or opens an Oracle ASM file, it communicates those requests to the Oracle ASM instance. In response, the Oracle ASM instance provides file extent map information to the database instance.

In Figure 1–1, there are two disk groups: one disk group has four disks and the other has two disks. The database can access both disk groups. The configuration in Figure 1–1 shows multiple database instances, but only one Oracle ASM instance is needed to serve the multiple database instances.

Figure 1–1   Oracle ASM for Single-Instance Oracle Databases

Single Instance Database Server

Oracle Database 1 Instance	Oracle Database 2 Instance	Oracle Database 3 Instance

ASM Instance

ASM Disk Groups

Disk Group A

Disk Group B
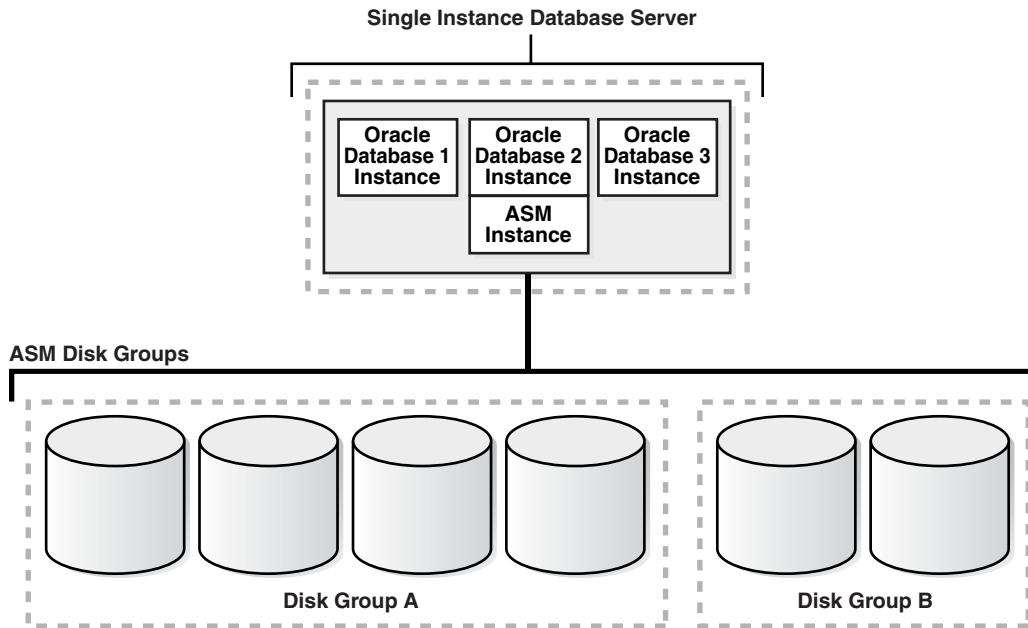
Figure 1–2 shows an Oracle ASM cluster in an Oracle RAC environment where Oracle ASM provides a clustered pool of storage. There is one Oracle ASM instance for each node serving multiple Oracle RAC or single-instance databases in the cluster. All of the databases are consolidated and share the same two Oracle ASM disk groups.

Figure 1–2   Oracle ASM Cluster Configuration with Oracle RAC

Oracle RAC

Oracle Database Instance	Oracle Database Instance	Oracle Database Instance	Oracle Database Instance	Oracle Database Instance
ASM Instance	ASM Instance	ASM Instance	ASM Instance	ASM Instance

ASM Clustered Pool of Storage

Disk Group A
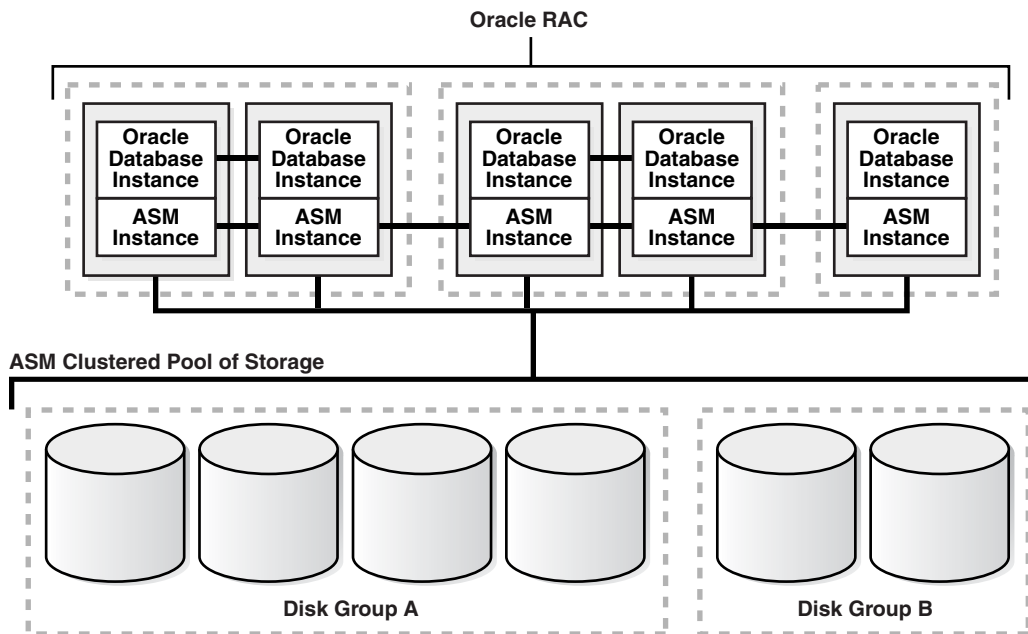
Disk Group B

A clustered storage pool can be shared by multiple single-instance Oracle Databases as shown in Figure 1–3. In this case, multiple databases share common disk groups. A shared Oracle ASM storage pool is achieved by using Oracle Clusterware. However, in such environments an Oracle RAC license is not required.

To share a disk group among multiple nodes, you must install Oracle Clusterware on all of the nodes, regardless of whether you install Oracle RAC on the nodes. Oracle ASM instances that are on separate nodes do not need to be part of an Oracle ASM cluster. However, if the Oracle ASM instances are not part of an Oracle ASM cluster, they cannot communicate with each other. Multiple nodes that are not part of an Oracle ASM cluster cannot share a disk group.

Figure 1–3   Oracle ASM Cluster with Single-Instance Oracle Databases
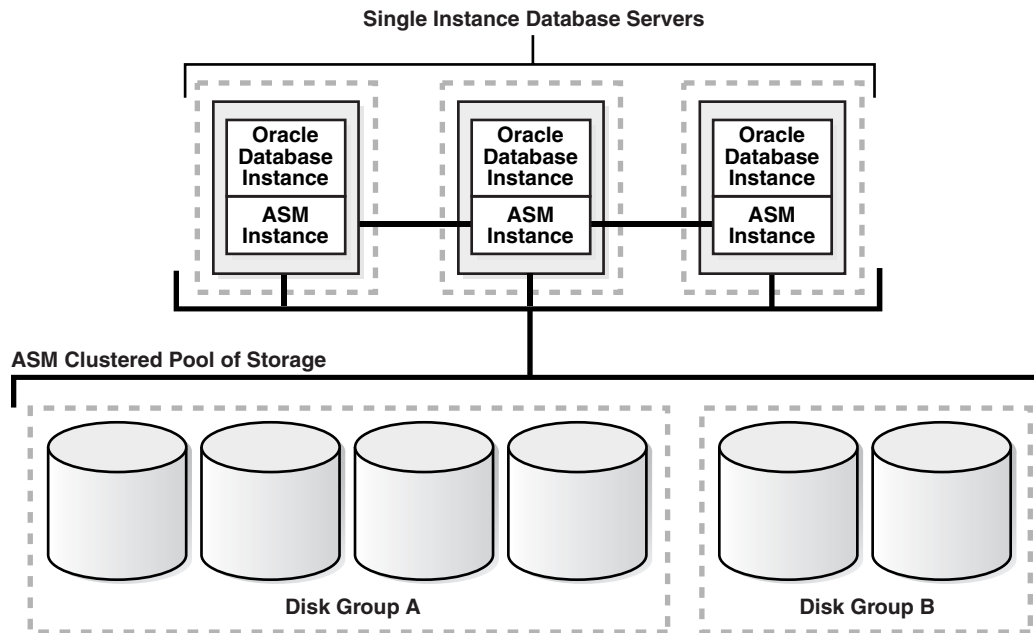
About Oracle ASM Disk Groups

A disk group consists of multiple disks and is the fundamental object that Oracle ASM manages. Each disk group contains the metadata that is required for the management of space in the disk group. Disk group components include disks, files, and allocation units.

Files are allocated from disk groups. Any Oracle ASM file is completely contained within a single disk group. However, a disk group might contain files belonging to several databases and a single database can use files from multiple disk groups. For most installations you need only a small number of disk groups, usually two, and rarely more than three. For more information about managing disk groups, see Chapter 4, "Administering Oracle ASM Disk Groups".

About Mirroring and Failure Groups

Mirroring protects data integrity by storing copies of data on multiple disks. When you create a disk group, you specify an Oracle ASM disk group type based on one of the following three redundancy levels:

- **Normal** for 2-way mirroring

- **High** for 3-way mirroring

- **External** to not use Oracle ASM mirroring, such as when you configure hardware RAID for redundancy

The redundancy level controls how many disk failures are tolerated without dismounting the disk group or losing data. The disk group type determines the mirroring levels with which Oracle creates files in a disk group. For information about disk group types and templates, see "Managing Disk Group Templates" on page 7-15.

Oracle ASM mirroring is more flexible than traditional RAID mirroring. For a disk group specified as NORMAL redundancy, you can specify the redundancy level for each file. For example, two files can share the same disk group with one file being mirrored while the other is not.

When Oracle ASM allocates an extent for a mirrored file, Oracle ASM allocates a primary copy and a mirror copy. Oracle ASM chooses the disk on which to store the mirror copy in a different failure group than the primary copy. Failure groups are used to place mirrored copies of data so that each copy is on a disk in a different failure group. The simultaneous failure of all disks in a failure group does not result in data loss.

You define the failure groups for a disk group when you create an Oracle ASM disk group. After a disk group is created, you cannot alter the redundancy level of the disk group. If you omit the failure group specification, then Oracle ASM automatically places each disk into its own failure group, except for disk groups containing disks on Oracle Exadata cells. Normal redundancy disk groups require at least two failure groups. High redundancy disk groups require at least three failure groups. Disk groups with external redundancy do not use failure groups.

For more information about mirroring and failure groups, see "Mirroring, Redundancy, and Failure Group Options" on page 4-24.

About Oracle ASM Disks

Oracle ASM disks are the storage devices that are provisioned to Oracle ASM disk groups. Examples of Oracle ASM disks include:

- A disk or partition from a storage array
- An entire disk or the partitions of a disk
- Logical volumes
- Network-attached files (NFS)

When you add a disk to a disk group, you can assign an Oracle ASM disk name or Oracle ASM assigns the Oracle ASM disk name automatically. This name is different from the path name used by the operating system. In a cluster, a disk may be assigned different operating system device names on different nodes, but the disk has the same Oracle ASM disk name on all of the nodes. In a cluster, an Oracle ASM disk must be accessible from all of the instances that share the disk group.

Oracle ASM spreads the files proportionally across all of the disks in the disk group. This allocation pattern maintains every disk at the same capacity level and ensures that all of the disks in a disk group have the same I/O load. Because Oracle ASM load balances among all of the disks in a disk group, different Oracle ASM disks should not share the same physical drive.

Allocation Units

Every Oracle ASM disk is divided into allocation units (AU). An allocation unit is the fundamental unit of allocation within a disk group. A file extent consists of one or more allocation units. An Oracle ASM file consists of one or more file extents.

When you create a disk group, you can set the Oracle ASM allocation unit size with the AU_SIZE disk group attribute. The values can be 1, 2, 4, 8, 16, 32, or 64 MB, depending on the specific disk group compatibility level. Larger AU sizes typically provide performance advantages for data warehouse applications that use large sequential reads.

For information about specifying the allocation unit size for a disk group, see "Specifying the Allocation Unit Size" on page 4-9 on page 4-33. For information about disk group compatibility attributes, see "Disk Group Compatibility" on page 4-33.

About Oracle ASM Files

Files that are stored in Oracle ASM disk groups are called Oracle ASM files. Each Oracle ASM file is contained within a single Oracle ASM disk group. Oracle Database communicates with Oracle ASM in terms of files. This is similar to the way Oracle Database uses files on any file system. You can store the various file types in Oracle ASM disk groups, including:

- Control files

- Data files, temporary data files, and data file copies

- SPFILEs

- Online redo logs, archive logs, and Flashback logs

- RMAN backups

- Disaster recovery configurations

- Change tracking bitmaps

- Data Pump dumpsets

Oracle ASM automatically generates Oracle ASM file names as part of file creation and tablespace creation. Oracle ASM file names begin with a plus sign (+) followed by a disk group name. You can specify user-friendly aliases for Oracle ASM files and create a hierarchical directory structure for the aliases.

The following sections describe the Oracle ASM file components:

- Extents

- Oracle ASM Striping

- File Templates

Extents

The contents of Oracle ASM files are stored in a disk group as a set, or collection, of **extent**s that are stored on individual disks within disk groups. Each extent resides on an individual disk. Extents consist of one or more allocation units (AU). To accommodate increasingly larger files, Oracle ASM uses variable size extents.

Variable size extents enable support for larger Oracle ASM data files, reduce SGA memory requirements for very large databases, and improve performance for file create and open operations. The initial extent size equals the disk group allocation unit size and it increases by a factor of 4 or 16 at predefined thresholds. This feature is automatic for newly created and resized data files when specific disk group compatibility attributes are set to 11.1 or higher. For information about compatibility attributes, see "Disk Group Compatibility" on page 4-33.

The extent size of a file varies as follows:

- Extent size always equals the disk group AU size for the first 20000 extent sets (0 - 19999).

- Extent size equals 4*AU size for the next 20000 extent sets (20000 - 39999).

- Extent size equals 16*AU size for the next 20000 and higher extent sets (40000+).

Figure 1–4 shows the Oracle ASM file extent relationship with allocation units. The first eight extents (0 to 7) are distributed on four Oracle ASM disks and are equal to the AU size. After the first 20000 extent sets, the extent size becomes 4*AU for the next 20000 extent sets (20000 - 39999). This is shown as bold rectangles labeled with the extent set numbers 20000 to 20007, and so on. The next increment for an Oracle ASM extent is 16*AU (not shown in Figure 1–4).

Figure 1–4 Oracle ASM File Allocation in a Disk Group
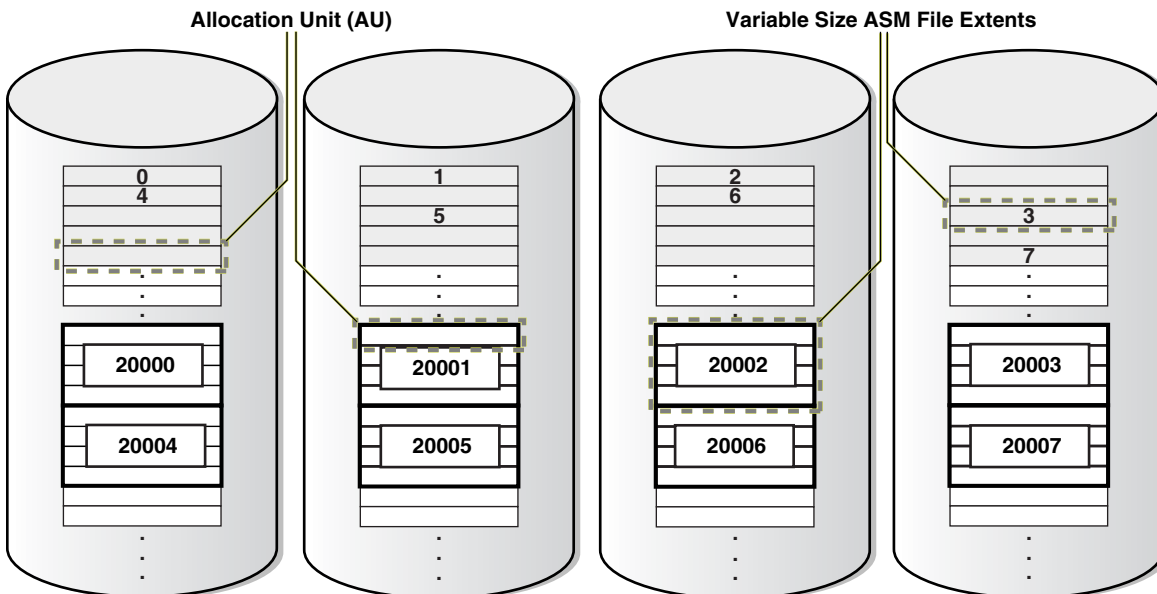
Oracle ASM Striping

Oracle ASM striping has two primary purposes:

- To balance loads across all of the disks in a disk group

- To reduce I/O latency

Coarse-grained striping provides load balancing for disk groups while fine-grained striping reduces latency for certain file types by spreading the load more widely.

To stripe data, Oracle ASM separates files into stripes and spreads data evenly across all of the disks in a disk group. The fine-grained stripe size always equals 128 KB in any configuration; this provides lower I/O latency for small I/O operations. The coarse-grained stripe size is always equal to the AU size (not the data extent size).

Figure 1–5 and Figure 1–6 are illustrations of Oracle ASM file striping. In both illustrations, the allocation unit size has been set to 1 M (AU_SIZE = 1M) for the disk group which consists of 8 disks. The Oracle ASM instance is release 11.2 and the disk group compatibility attributes for ASM and RDBMS have been set to 11.2, so variable extents are shown in the graphic after the first 20,000 extents. For the first 20,000 extents, the extent size is 1 M and equals one allocation unit (AU). For the next 20,000 extents, the extent size is 4 M and equals 4 AUs.

To identify the stripe chunks of the file, they have been labeled A..X (24 letters) using different fonts for successive series of A..X until all the chunks have been identified.

In Figure 1–5, the file is striped in 128 K chunks (labeled A..X) with each 128 K chunk stored in an extent, starting at the first extent in disk 1, then the first extent in disk 2, and then continuing in a round-robin pattern through all the disks until the entire file has been striped. As shown in this example, the striping chunks first fill up the first extent of each disk, then the second extent of each disk, and so on until the entire file has been striped.

Figure 1–5   Oracle ASM Fine-Grained Striping
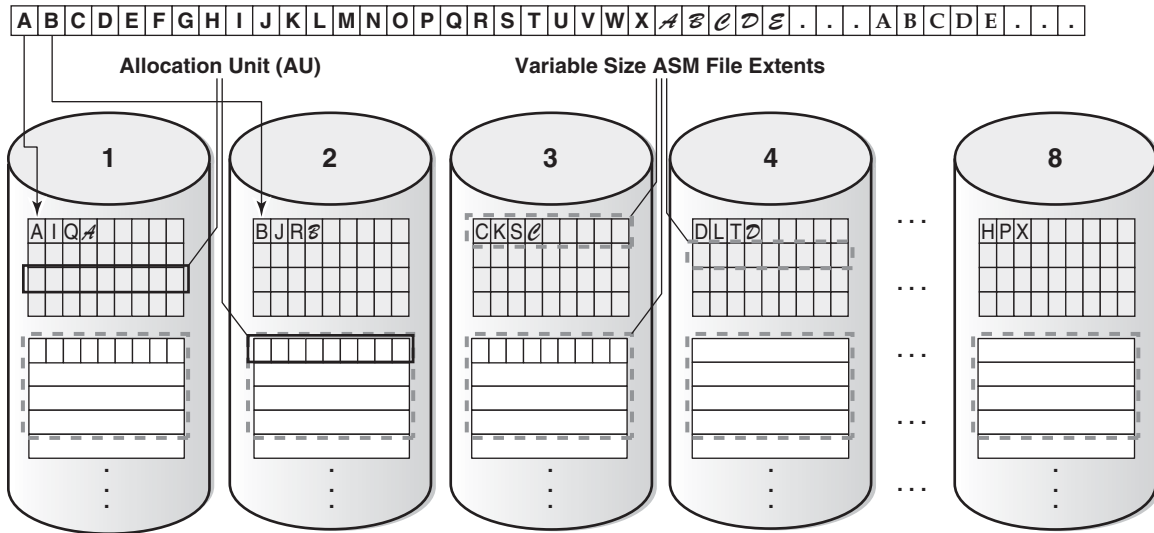
In Figure 1–6, the file is striped in 1 M chunks (labeled A..X) with each 1 M chunk stored uniquely in an extent, starting at the first extent in disk 1, then the first extent in disk 2, and then continuing in a round-robin pattern through all the disks until the entire file has been striped. For the first 20,000 extents where the AU equals the extent size (1 M), the stripe equals the extent size and allocation unit size.

For the variable extents, where an extent is composed of multiple allocation units, the file stripe is located in an AU of the extent. The striping chunks are placed in the allocation units of the first extents of all the disks before the striping continues to the next extent.

Figure 1–6 Oracle ASM Coarse-Grained Striping
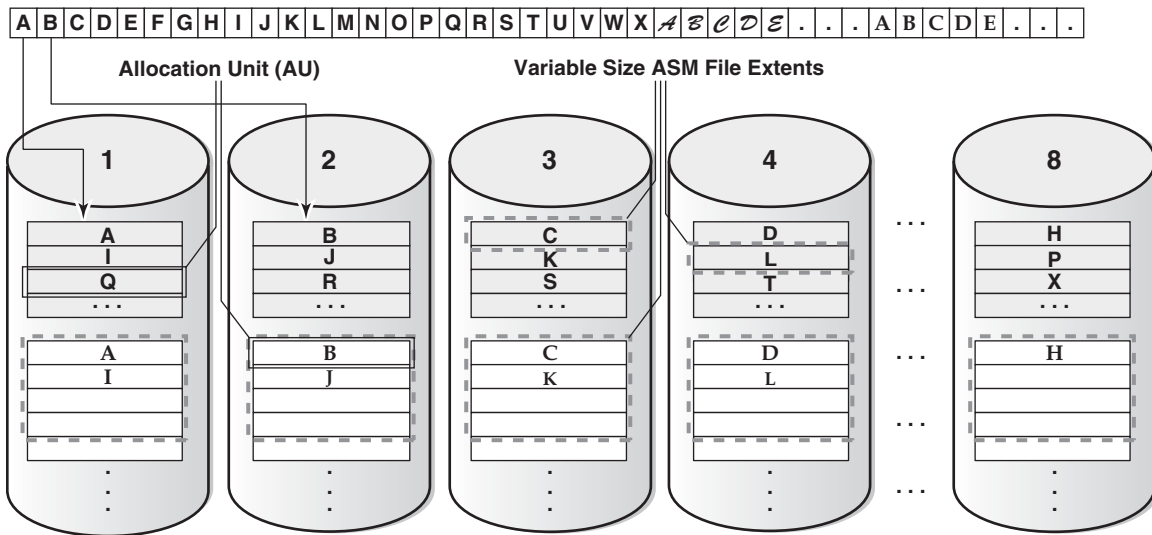
File Templates

Templates are collections of attribute values that are used to specify disk regions, file mirroring, and **striping** attributes for an Oracle ASM file when it is created. When creating a file, you can include a template name and assign desired attributes based on an individual file rather than the file type.

A default template is provided for every Oracle file type, but you can customize templates to meet unique requirements. Each disk group has a default template associated with each file type.

For more information about Oracle ASM templates, see "Managing Disk Group Templates" on page 7-15.

Understanding Oracle ASM Disk Group Administration

This section describes Oracle ASM disk group administration and it contains the following topics:

- About Discovering Disks

- About Mounting and Dismounting Disk Groups

- About Adding and Dropping Disks

- About Online Storage Reconfigurations and Dynamic Rebalancing

About Discovering Disks

The disk discovery process locates the operating system names for disks that Oracle ASM can access. Disk discovery finds all of the disks that comprise a disk group to be mounted. The set of discovered disks also includes disks that could be added to a disk group.

An Oracle ASM instance requires an `ASM_DISKSTRING` initialization parameter value to specify its discovery strings. Only path names that the Oracle ASM instance has permission to open are discovered. The exact syntax of a discovery string depends on the platform, ASMLib libraries, and whether Oracle Exadata disks are used. The path names that an operating system accepts are always usable as discovery strings.

For information about the ASM_DISKSTRING initialization parameter, see "ASM_ DISKSTRING" on page 3-7. For information about disk discovery, see "Oracle ASM Disk Discovery" on page 4-20.

About Mounting and Dismounting Disk Groups

A disk group must be mounted by a local Oracle ASM instance before database instances can access the files in the disk group. Mounting the disk group requires discovering all of the disks and locating the files in the disk group that is being mounted.

You can explicitly dismount a disk group. Oracle reports an error if you attempt to dismount a disk group without the force option when any of the disk group files are open. It is possible to have disks fail in excess of the Oracle ASM redundancy setting. If this happens, then the disk group is forcibly dismounted. If the disk group is forcibly dismounted, a database cannot access files in the disk group.

For more information about disk groups, see "Mounting and Dismounting Disk Groups" on page 4-44.

About Adding and Dropping Disks

You can add a disk to an existing disk group to add space and to improve throughput. The specified discovery string identifies the disk or disks that you could add. The disks that you add must be discovered by every Oracle ASM instance using its ASM_ DISKSTRING initialization parameter. After you add a disk, Oracle ASM rebalancing operations move data onto the new disk. To minimize the rebalancing I/O, it is more efficient to add multiple disks at the same time.

You can drop a disk from a disk group if it fails or to re-purpose capacity. Use the Oracle ASM disk name to drop a disk, not the discovery string device name. If an error occurs while writing to a disk, then Oracle ASM drops the disk automatically.

For more information about altering disk group membership, see "Altering Disk Groups" on page 4-12.

About Online Storage Reconfigurations and Dynamic Rebalancing

Rebalancing a disk group moves data between disks to ensure that every file is evenly spread across all of the disks in a disk group. When all of the files are evenly dispersed, all of the disks are evenly filled to the same percentage; this ensures load balancing. Rebalancing does not relocate data based on I/O statistics nor is rebalancing started based on I/O statistics. Oracle ASM rebalancing operations are controlled by the size of the disks in a disk group.

Oracle ASM automatically initiates a rebalance after storage configuration changes, such as when you add, drop, or resize disks. The power setting parameter determines the speed with which rebalancing operations occur.

You can manually start a rebalance to change the power setting of a running rebalance. A rebalance is automatically restarted if the instance on which the rebalancing is running stops. Databases can remain operational during rebalancing operations.

You can minimize the impact on database performance with the setting of the POWER_ LIMIT initialization parameter. For more information about the power limit setting, see "ASM_POWER_LIMIT" on page 3-8. For more information about disk rebalancing, see "Manually Rebalancing Disk Groups" on page 4-18.

2

Considerations for Oracle ASM Storage

This chapter discusses some points to consider about the storage subsystem before you configure Oracle Automatic Storage Management (Oracle ASM). When preparing your storage to use Oracle ASM, first determine the storage option for your system and then prepare the disk storage for your specific operating system environment.

When configuring your system's storage, you must consider the initial capacity of the system and your plans for future growth. Oracle ASM simplifies the task of accommodating growth. However, your growth plans can affect choices, such as the size of the Oracle ASM disks. You must also consider that I/O performance depends on the interconnect between the storage and host, not just the storage disks. As you scale up the number of nodes in a cluster, you must also scale up the storage subsystem.

This chapter contains the following topics:

- Storage Resources for Disk Groups

- Oracle ASM and Multipathing

- Recommendations for Storage Preparation

Storage Resources for Disk Groups

You can create an Oracle ASM disk group using one of the following storage resources:

- Disk Partition

  A disk **partition** can be the entire disk drive or a section of a disk drive. However, the Oracle ASM disk cannot be in a partition that includes the partition table because the partition table would be overwritten.

- **Logical Unit Number (LUN)**

  A LUN is a disk presented to a computer system by a storage array. Oracle recommends that you use hardware RAID functionality to create LUNs. Storage hardware RAID 0+1 or RAID5, and other RAID configurations, can be provided to Oracle ASM as Oracle ASM disks.

- Logical Volume

  A logical **volume** is supported in less complicated configurations where a logical volume is mapped to a LUN, or a logical volume uses disks or raw partitions. Logical volume configurations are not recommended by Oracle because they create a duplication of functionality. Oracle also does not recommended using logical **volume manager**s for mirroring because Oracle ASM provides mirroring.

- **Network File System (NFS)**

An Oracle ASM disk group can be created from NFS files, including Oracle Direct NFS (dNFS), as well as whole disks, partitions, and LUNs. The NFS files that are provisioned to a disk group may be from multiple NFS servers to provide better load balancing and flexible capacity planning.

Direct NFS can be used to store data files, but is not supported for Oracle Clusterware files. To install Oracle Real Application Clusters (Oracle RAC) on Windows using Direct NFS, you must also have access to a shared storage method other than NFS for Oracle Clusterware files.

> **See Also:** *Oracle Grid Infrastructure Installation Guide* for your operating system for information about Oracle Direct NFS

Notes:

- Oracle ASM Dynamic Volume Manager (Oracle ADVM) volumes and Oracle Automatic Storage Management Cluster File System (Oracle ACFS) file systems are currently not supported on disk groups that have been created from NFS or Common Internet File System (CIFS) files. However, Oracle ACFS file systems may be exported as NFS or CIFS file systems to network clients.

- Mounting loopback file systems over Oracle ACFS files is not supported.

- Block or raw devices are not supported by Oracle Universal Installer (OUI) or Database Configuration Assistant (DBCA). However, manually configured raw or block devices are supported by Oracle, but not recommended.

The procedures for preparing storage resources for Oracle ASM are:

1. Identify or create the storage devices for Oracle ASM by identifying all of the storage resource device names that you can use to create an Oracle ASM disk group. For example, on Linux systems without ASMLib, device names are typically presented from the `/dev` directory with the `/dev/device_name_identifier` name syntax.

2. Change the ownership and the permissions on storage device resources.

   For example, the following steps are required on Linux systems:

   - Change the user and group ownership of devices, such as `grid:asmadmin`

     For information about Oracle ASM privileges, see "About Privileges for Oracle ASM" on page 3-26.

   - Change the device permissions to read/write

   > **Note:** To ensure that ownership and permission settings are persistent, use ASMLib or `udev` to ensure that the disks do not revert to root ownership when the systems reboot.

After you have configured Oracle ASM, ensure that disk discovery has been configured correctly by setting the `ASM_DISKSTRING` initialization parameter. For information about the `ASM_DISKSTRING` parameter, see "ASM_DISKSTRING" on page 3-7.

> **Note:** Setting the ownership to `oracle:dba` is one example that corresponds to the default settings. A nondefault installation may require different settings. In general, the owner of the disk devices should be the same as the owner of the Oracle binary software. The group ownership should be OSDBA of the Oracle ASM instance, which is defined at installation. For information about Oracle ASM privileges, see "About Privileges for Oracle ASM" on page 3-26.

For detailed information about preparing disks for an Oracle ASM installation, refer to your platform-specific installation guide for Oracle Database, Oracle Clusterware, and Oracle Real Application Clusters (Oracle RAC).

> **See Also:** Oracle Exadata documentation for information about preparing Oracle Exadata storage

Oracle ASM and Multipathing

Multipathing solutions provide failover by using redundant physical path components. These components include adapters, cables, and switches that reside between the server and the storage subsystem. If one or more of these components fails, then applications can still access their data, eliminating a single point of failure with the **Storage Area Network (SAN)**, Host Bus Adapter, interface cable, or host port on a multiported storage array.

Multipathing is a software technology implemented at the operating system device driver level. Multipathing creates a pseudo device to facilitate the sharing and balancing of I/O operations across all of the available I/O paths. Multipathing also improves system performance by distributing the I/O load across all available paths, providing a higher level of data availability through automatic failover and failback.

Although Oracle ASM is not designed with multipathing functionality, Oracle ASM does operate with multipathing technologies. Multipathing technologies are available from many sources. Storage vendors offer multipathing products to support their specific storage products, while software vendors usually develop multipathing products to support several server platforms and storage products.

> **See Also:** Your storage or software vendor multipathing documentation for more information about multipathing options for specific platforms and storage products

With Oracle ASM, you can ensure the discovery of a multipath disk by setting the value of the `ASM_DISKSTRING` initialization parameter to a pattern that matches the pseudo devices that represents the multipath disk. When I/O is sent to the pseudo device, the multipath driver intercepts it and provides load balancing to the underlying subpaths.

If Oracle ASM discovers multiple paths to the same disk device, Oracle ASM then raises an error. Because a single disk can appear multiple times in a multipath configuration, you must configure Oracle ASM to discover only the multipath disk.

When using **ASMLib** with Oracle ASM on Linux, you can ensure the discovery of the multipath disk by configuring Oracle ASM to scan the multipath disk first or to exclude the single path disks when scanning.

For information about disk discovery, see "Oracle ASM Disk Discovery" on page 4-20.

> **See Also:**
>
> - Articles at My Oracle Support (`https://support.oracle.com`) for information about Oracle ASM and Multipathing
> - Your platform-specific installation guide for information about configuring multipathing for your system

Recommendations for Storage Preparation

The following are guidelines for preparing storage for use with Oracle ASM:

- Configure two disk groups, one for data and the other for the fast recovery area.

  > **See Also:**
  >
  > - *Oracle Database Backup and Recovery User's Guide* for information about configuring the fast recovery area
  > - *Oracle Database Administrator's Guide* for information about specifying a fast recovery area

- A minimum of four LUNs (Oracle ASM disks) of equal size and performance is recommended for each disk group.

- Ensure that all Oracle ASM disks in a disk group have similar storage performance and availability characteristics. In storage configurations with mixed speed drives, such as 10K and 15K RPM, I/O performance is constrained by the slowest speed drive.

- Oracle ASM data distribution policy is capacity-based. Ensure that Oracle ASM disks in a disk group have the same capacity to maintain balance.

- Create external redundancy disk groups when using high-end storage arrays. High-end storage arrays generally provide hardware RAID protection. Use Oracle ASM mirroring redundancy when not using hardware RAID, or when you need host-based volume management functionality, such as mirroring across storage systems. You can use Oracle ASM mirroring in configurations when mirroring between geographically-separated sites (extended clusters).

- Minimize I/O contention between Oracle ASM disks and other applications by dedicating disks in Oracle ASM disk groups.

- Choose a hardware RAID stripe size that is a power of 2 and less than or equal to the size of the Oracle ASM allocation unit.

- For Linux, use the Oracle ASMLib feature to provide consistent device naming and permission persistency.

  > **See Also:**
  >
  > - The Oracle Cloud Storage page on the Oracle Technology Network Web site at `http://www.oracle.com/technetwork/database/cloud-storage/index.html` for more information about Oracle ASM
  > - The Oracle ASMLib page on the Oracle Technology Network Web site at `http://www.oracle.com/technetwork/topics/linux/asmlib/index-101839.html` for information about ASMLib

3

Administering Oracle ASM Instances

This chapter describes how to administer Automatic Storage Management (Oracle ASM) instances. It explains how to configure Oracle ASM instance parameters and how to set Oracle Database parameters for use with Oracle ASM. The chapter also describes Oracle ASM upgrading, patching, and authentication for Oracle ASM instance access. You can also use procedures in this chapter to migrate a database to use Oracle ASM.

Administering an Oracle ASM instance is similar to administering an Oracle Database instance, but the process requires fewer procedures. You can use Oracle Enterprise Manager and SQL*Plus to perform Oracle ASM instance administration tasks.

Oracle ASM is installed in the Oracle Grid Infrastructure home separate from the Oracle Database home. Only one Oracle ASM instance is supported on a server. When managing an Oracle ASM instance, the administration activity must be performed in the Oracle Grid Infrastructure home.

This chapter contains the following topics:

- Operating with Different Releases of Oracle ASM and Database Instances Simultaneously
- Configuring Initialization Parameters for Oracle ASM Instances
- Managing Oracle ASM Instances
- Using Oracle ASM Rolling Upgrade
- Patching Oracle ASM Instances in Oracle RAC Environments
- Authentication for Accessing Oracle ASM Instances
- Migrating a Database to Use Oracle ASM

For a description of an Oracle ASM instance, see "About Oracle ASM Instances" on page 1-2. For information about using Oracle Enterprise Manager to administer Oracle ASM, see Chapter 9, "Administering Oracle ASM with Oracle Enterprise Manager".

Operating with Different Releases of Oracle ASM and Database Instances Simultaneously

Oracle Automatic Storage Management (Oracle ASM) in Oracle Database 11g Release 2 (11.2) supports 11g Release 2 (11.2) or older software versions of Oracle database instances, including Oracle Database 10*g*. For compatibility between Oracle Clusterware and Oracle ASM, the Oracle Clusterware release must be greater than or equal to the Oracle ASM release.

Notes:

- An Oracle ASM instance must be at 11g Release 2 (11.2) to support an 11g Release 2 (11.2) Oracle Database.

- See *Oracle Exadata documentation* for information about the Oracle Database versions that Oracle ASM 11g Release 2 (11.2) supports when Oracle Exadata storage is present.

There are additional compatibility considerations when using disk groups with different releases of Oracle ASM and database instances. For information about disk group compatibility attributes settings, see "Disk Group Compatibility" on page 4-33.

When using different software versions, the database instance supports Oracle ASM functionality of the earliest release in use. For example, a 10.1 database instance operating with an 11.2 Oracle ASM instance supports only Oracle ASM 10.1 features.

The V$ASM_CLIENT view contains the SOFTWARE_VERSION and COMPATIBLE_VERSION columns with information about the software version number and instance compatibility level.

- The SOFTWARE_VERSION column of V$ASM_CLIENT contains the software version number of the database or Oracle ASM instance for the selected disk group connection.

- The COMPATIBLE_VERSION column contains the setting of the COMPATIBLE parameter of the database or Oracle ASM instance for the selected disk group connection.

You can query the V$ASM_CLIENT view on both Oracle ASM and database instances. For an example showing a query on the V$ASM_CLIENT view, see Example 6–4, "Viewing disk group clients with V$ASM_CLIENT" on page 6-3. For more information about the V$ASM_CLIENT and V$ASM_* views, see "Views Containing Oracle ASM Disk Group Information" on page 6-1.

Configuring Initialization Parameters for Oracle ASM Instances

This section discusses initialization parameter files and parameter settings for Oracle ASM instances. To install and initially configure an Oracle ASM instance, use Oracle Universal Installer (OUI) and Oracle ASM Configuration Assistant (ASMCA). Refer to your platform-specific *Oracle Grid Infrastructure Installation Guide* for details about installing and configuring Oracle ASM.

After an Oracle ASM instance has been installed on a single-instance Oracle Database or in an Oracle Real Application Clusters (Oracle RAC) environment, the final Oracle ASM configuration can be performed. Only a few Oracle ASM-specific instance initialization parameters must be configured. The default values are usually sufficient.

> **See Also:** The Oracle Cloud Storage page on the Oracle Technology Network Web site at http://www.oracle.com/technetwork/database/cloud-storage/index.html for more information about Oracle ASM best practices

This section contains the following topics:

- Initialization Parameter Files for an Oracle ASM Instance

- Backing Up, Copying, and Moving an Oracle ASM Initialization Parameter File

- Setting Oracle ASM Initialization Parameters

- Oracle ASM Parameter Setting Recommendations
- Setting Database Initialization Parameters for Use with Oracle ASM

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

Initialization Parameter Files for an Oracle ASM Instance

When installing Oracle ASM in an Oracle Restart (standalone) configuration, Oracle Universal Installer (OUI) creates a separate server parameter file (SPFILE) and password file for the Oracle ASM instance. The ASM SPFILE is stored in a disk group during installation.

When installing Oracle ASM in a clustered Oracle ASM environment, OUI creates a single, shared SPFILE for Oracle ASM in a disk group.

When upgrading an Oracle ASM instance, if the ASM SPFILE was originally in a shared file system, then the upgraded Oracle ASM instance retains the SPFILE in the same location. If the original Oracle ASM instance used a PFILE, after an upgrade the instance continues to use a PFILE.

You can use an SPFILE or a text-based initialization parameter file (PFILE) as the Oracle ASM instance parameter file. If you use an SPFILE in a clustered Oracle ASM environment, then you must place the SPFILE in a disk group, on a shared raw device, or on a cluster file system. Oracle recommends that the Oracle ASM SPFILE is placed in a disk group. You cannot use a new alias created on an existing Oracle ASM SPFILE to start the Oracle ASM instance

If you do not use a shared Oracle Grid Infrastructure home, then the Oracle ASM instance can use a PFILE. The same rules for file name, default location, and search order that apply to database initialization parameter files also apply to Oracle ASM initialization parameter files.

When an Oracle ASM instance searches for an initialization parameter file, the search order is:

1. The location of the initialization parameter file specified in the Grid Plug and Play (GPnP) profile

2. If the location has not been set in the GPnP profile, then the search order changes to:

   a. SPFILE in the Oracle ASM instance home

      For example, the SPFILE for Oracle ASM has the following default path in the Oracle Grid Infrastructure home in a Linux environment:

      `$ORACLE_HOME/dbs/spfile+ASM.ora`

   b. PFILE in the Oracle ASM instance home

> **Note:** A PFILE or SPFILE is required if your configuration uses non-default initialization parameters for the Oracle ASM instance.

You can administer Oracle ASM initialization parameter files with SQL*Plus, Oracle Enterprise Manager, ASMCA, and ASMCMD commands. For information about Oracle Enterprise Manager, see "Configuring Oracle ASM Initialization Parameters with Oracle Enterprise Manager" on page 9-3. For information about ASMCA, see Chapter 11, "Oracle ASM Configuration Assistant". For information about ASMCMD commands, see "ASMCMD Instance Management Commands" on page 12-10.

> **See Also:**
>
> - *Oracle Database Administrator's Guide* for more information about creating and maintaining initialization parameter files
>
> - *Oracle Database 2 Day DBA* for information about viewing and modifying initialization parameters
>
> - *Oracle Database SQL Language Reference* for information about creating an SPFILE with the `CREATE SPFILE` SQL statement

Backing Up, Copying, and Moving an Oracle ASM Initialization Parameter File

You can back up, copy, or move an Oracle ASM SPFILE with the ASMCMD `spbackup`, `spcopy` or `spmove` commands. For information about these ASMCMD commands, see "spbackup" on page 12-15, "spcopy" on page 12-16, and "spmove" on page 12-18.

You can also use the SQL `CREATE SPFILE` to create an Oracle ASM SPFILE when connected to the Oracle ASM instance.

You can copy and move an Oracle ASM PFILE with the commands available on the specific platform, such as `cp` for Linux.

After copying or moving an SPFILE or PFILE, you must restart the instance with the SPFILE or PFILE in the new location to use that SPFILE or PFILE.

If the `COMPATIBLE.ASM` disk group attribute is set to 11.2 or greater for a disk group, you can create, copy, or move an Oracle ASM SPFILE into the disk group.

For example, after upgrading an Oracle ASM instance from 11*g* release 1 (11.1) to 11*g* release 2 (11.2), you could place the Oracle ASM SPFILE in a disk group that has `COMPATIBLE.ASM` set to 11.2.

In the following steps, assume an Oracle ASM 11*g* release 2 (11.2) instance is using a PFILE stored in `$ORACLE_HOME/dbs/asmspfile.ora`. You can use the SQL `CREATE SPFILE` statement to create an SPFILE from a PFILE stored in a local or shared file system. If a PFILE does not exist, then it could be created with the SQL `CREATE PFILE` statement.

To create an SPFILE in a disk group, perform the following steps:

1. Connect to the Oracle ASM instance.

   For example:

   ```
   $ sqlplus / as sysasm
   ```

2. Create an SPFILE in a disk group that has `COMPATIBLE.ASM` set to 11.2 with the SQL `CREATE SPFILE` statement.

   For example, create an Oracle ASM SPFILE from the existing `PFILE`.

   ```
   SQL> CREATE SPFILE = '+DATA/asmspfile.ora'
           FROM PFILE = '$ORACLE_HOME/dbs/asmpfile.ora';
   ```

The `CREATE SPFILE` statement also updates the Grid Plug and Play (GPnP) profile. You can check the location of the Oracle ASM SPFILE in the GPnP profile with the ASMCMD `spget` command. See "spget" on page 12-18.

3. Restart the Oracle ASM instance so that the instance reads the SPFILE in the new location.

   For information on shutting down and starting up an Oracle ASM instance, see "Starting Up an Oracle ASM Instance" on page 3-13 and "Shutting Down an Oracle ASM Instance" on page 3-16.

For information about disk group compatibility attributes, see "Disk Group Compatibility" on page 4-33. For information about upgrading an Oracle ASM instance, see "Upgrading an Oracle ASM Instance in an Oracle Restart Configuration with Oracle Universal Installer" on page 3-18.

> **See Also:**
>
> - *Oracle Database Administrator's Guide* for more information about creating and maintaining initialization parameter files
> - *Oracle Database SQL Language Reference* for information about creating an SPFILE with the `CREATE SPFILE` SQL statement
> - *Oracle Real Application Clusters Installation Guide* for information about Grid Plug and Play (GPnP)

Setting Oracle ASM Initialization Parameters

There are several initialization parameters that you must set for an Oracle ASM instance. You can set these parameters with Oracle ASM Configuration Assistant (ASMCA). You can also set some parameters after database creation using Oracle Enterprise Manager or SQL `ALTER SYSTEM` or `ALTER SESSION` statements.

The `INSTANCE_TYPE` initialization parameter is the only required parameter in the Oracle ASM instance parameter file. The Oracle ASM* parameters use suitable defaults for most environments. You cannot use parameters with names that are prefixed with Oracle ASM* in database instance parameter files.

Some database initialization parameters are also valid for an Oracle ASM instance initialization file. In general, Oracle ASM selects the appropriate defaults for database parameters that are relevant to an Oracle ASM instance.

For information about setting Oracle ASM parameters with Oracle Enterprise Manager, see "Configuring Oracle ASM Initialization Parameters with Oracle Enterprise Manager" on page 9-3.

Automatic Memory Management for Oracle ASM

Automatic memory management automatically manages the memory-related parameters for both Oracle ASM and database instances with the `MEMORY_TARGET` parameter. Automatic memory management is enabled by default on an Oracle ASM instance, even when the `MEMORY_TARGET` parameter is not explicitly set. The default value used for `MEMORY_TARGET` is acceptable for most environments. This is the only parameter that you must set for complete Oracle ASM memory management. Oracle strongly recommends that you use automatic memory management for Oracle ASM.

If you do not set a value for `MEMORY_TARGET`, but you do set values for other memory related parameters, Oracle internally calculates the optimum value for `MEMORY_TARGET` based on those memory parameter values. You can also increase `MEMORY_TARGET`

dynamically, up to the value of the MEMORY_MAX_TARGET parameter, just as you can do for the database instance.

Although it is not recommended, you can disable automatic memory management by either setting the value for MEMORY_TARGET to 0 in the Oracle ASM parameter file or by running an ALTER SYSTEM SET MEMORY_TARGET=0 statement. When you disable automatic memory management, Oracle reverts to auto shared memory management and automatic PGA memory management. To revert to Oracle Database 10*g* release 2 (10.2) functionality to manually manage Oracle ASM SGA memory, also run the ALTER SYSTEM SET SGA_TARGET=0 statement. You can then manually manage Oracle ASM memory using the information in "Oracle ASM Parameter Setting Recommendations" on page 3-6, that discusses Oracle ASM memory-based parameter settings. Unless specified, the behaviors of the automatic memory management parameters in Oracle ASM instances behave the same as in Oracle Database instances.

Notes:

- For a Linux environment, automatic memory management cannot work if /dev/shm is not available or is undersized. For more information, see *Oracle Database Administrator's Reference for Linux and UNIX-Based Operating Systems*. For information about platforms that support automatic memory management, see *Oracle Database Administrator's Guide*.

- The minimum MEMORY_TARGET for Oracle ASM is 256 MB. If you set MEMORY_TARGET to 100 MB, then Oracle increases the value for MEMORY_TARGET to 256 MB automatically.

See Also:

- *Oracle Database Administrator's Guide* for more information about the functionality of automatic memory management for database instances, which varies from Oracle ASM

- *Oracle Database Concepts* for an overview of memory management methods

Oracle ASM Parameter Setting Recommendations

This section contains information about the following parameters for Oracle ASM:

- ASM_DISKGROUPS
- ASM_DISKSTRING
- ASM_POWER_LIMIT
- ASM_PREFERRED_READ_FAILURE_GROUPS
- DB_CACHE_SIZE
- DIAGNOSTIC_DEST
- INSTANCE_TYPE
- LARGE_POOL_SIZE
- PROCESSES
- REMOTE_LOGIN_PASSWORDFILE
- SHARED_POOL_SIZE

- *Oracle Database Administrator's Guide* for more information about creating and maintaining an initialization parameter file

- *Oracle Database 2 Day DBA* for information about viewing and modifying initialization parameters

ASM_DISKGROUPS

The ASM_DISKGROUPS initialization parameter specifies a list of the names of disk groups that an Oracle ASM instance mounts at startup. Oracle ignores the value that you set for ASM_DISKGROUPS when you specify the NOMOUNT option at startup or when you run the ALTER DISKGROUP ALL MOUNT statement. The default value of the ASM_DISKGROUPS parameter is a NULL string. For information about disk groups that are mounted at startup time, see "About Mounting Disk Groups at Startup" on page 3-16.

The ASM_DISKGROUPS parameter is dynamic. If you are using a server parameter file (SPFILE), then you do not have to manually alter the value of ASM_DISKGROUPS. Oracle ASM automatically adds a disk group to this parameter when the disk group is successfully created or mounted. Oracle ASM also automatically removes a disk group from this parameter when the disk group is dropped or dismounted.

The following is an example of setting the ASM_DISKGROUPS parameter dynamically:

```
SQL> ALTER SYSTEM SET ASM_DISKGROUPS = DATA, FRA;
```

When using a text initialization parameter file (PFILE), you may edit the initialization parameter file to add the name of any disk group so that it is mounted automatically at instance startup. You must remove the name of any disk group that you no longer want automatically mounted.

The following is an example of the ASM_DISKGROUPS parameter in the initialization file:

```
ASM_DISKGROUPS = DATA, FRA
```

> **Note:** Issuing the ALTER DISKGROUP...ALL MOUNT or ALTER DISKGROUP...ALL DISMOUNT commands does not affect the value of ASM_DISKGROUPS.

For additional information about mounting Oracle ASM disk groups, see "Mounting and Dismounting Disk Groups" on page 4-44.

> **See Also:** *Oracle Database Reference* for more information about the ASM_DISKGROUPS initialization parameter

ASM_DISKSTRING

The ASM_DISKSTRING initialization parameter specifies a comma-delimited list of strings that limits the set of disks that an Oracle ASM instance discovers. The discovery strings can include wildcard characters. Only disks that match one of the strings are discovered. The same disk cannot be discovered twice.

The discovery string format depends on the Oracle ASM library and the operating system that are in use. Pattern matching is supported. Refer to your operating system-specific installation guide for information about the default pattern matching.

For example, on a Linux server that does not use ASMLib, to limit the discovery process to only include disks that are in the /dev/rdsk/mydisks directory, set the ASM_DISKSTRING initialization parameter to:

```
/dev/rdsk/mydisks/*
```

The asterisk is required. To limit the discovery process to only include disks that have a name that ends in `disk3` or `disk4`, set `ASM_DISKSTRING` to:

```
/dev/rdsk/*disk3,/dev/rdsk/*disk4
```

The `?` character, when used as the first character of a path, expands to the Oracle home directory. Depending on the operating system, when you use the `?` character elsewhere in the path, it is a wildcard for one character.

The default value of the `ASM_DISKSTRING` parameter is a `NULL` string. A `NULL` value causes Oracle ASM to search a default path for all disks in the system to which the Oracle ASM instance has read and write access. The default search path is platform-specific. Refer to your operating system specific installation guide for more information about the default search path.

Oracle ASM cannot use a disk unless all of the Oracle ASM instances in the cluster can discover the disk through one of their own discovery strings. The names do not have to be the same on every node, but all disks must be discoverable by all of the nodes in the cluster. This may require dynamically changing the initialization parameter to enable adding new storage.

For additional information about discovering disks, see "Oracle ASM Disk Discovery" on page 4-20.

> **See Also:**
> - *Oracle Exadata documentation* for information about the Oracle ASM discovery string format for Oracle Exadata
> - *Oracle Database Reference* for more information about the `ASM_DISKSTRING` initialization parameter

ASM_POWER_LIMIT

The `ASM_POWER_LIMIT` initialization parameter specifies the default power for disk rebalancing in a disk group. The range of values is `0` to `1024`. The default value is `1`. A value of `0` disables rebalancing. Higher numeric values enable the rebalancing operation to complete more quickly, but might result in higher I/O overhead and more rebalancing processes.

- For disk groups that have the disk group ASM compatibility set to `11.2.0.2` or greater (for example, `COMPATIBLE.ASM = 11.2.0.2`), the operational range of values is `0` to `1024` for the rebalance power.

- For disk groups that have the disk group ASM compatibility set to less than `11.2.0.2`, the operational range of values is `0` to `11` inclusive. If the value for `ASM_POWER_LIMIT` is larger than `11`, a value of `11` is used for these disk groups.

You can also specify the power of the rebalancing operation in a disk group with the `POWER` clause of the SQL `ALTER DISKGROUP ... REBALANCE` statement. The range of allowable values for the `POWER` clause is the same for the `ASM_POWER_LIMIT` initialization parameter. If the value of the `POWER` clause is specified larger than `11` for a disk group with ASM compatibility set to less than `11.2.0.2`, then a warning is displayed and a `POWER` value equal to `11` is used for rebalancing.

For information about the `ASM_POWER_LIMIT` unitization parameter, and the `POWER` clause, refer to "Manually Rebalancing Disk Groups" on page 4-18 and "Tuning Rebalance Operations" on page 4-19. For information about disk group compatibility, see "Disk Group Compatibility" on page 4-33.

See Also:

- *Oracle Database Reference* for more information about the `ASM_POWER_LIMIT` initialization parameter

- *Oracle Database SQL Language Reference* for more information about the `POWER` clause of the SQL `ALTER DISKGROUP REBALANCE` statement

ASM_PREFERRED_READ_FAILURE_GROUPS

The `ASM_PREFERRED_READ_FAILURE_GROUPS` initialization parameter value is a comma-delimited list of strings that specifies the failure groups that should be preferentially read by the given instance. The `ASM_PREFERRED_READ_FAILURE_GROUPS` parameter setting is instance specific. The default value is `NULL`. This parameter is generally used for clustered Oracle ASM instances and its value can be different on different nodes.

For example:

`diskgroup_name1.failure_group_name1, ...`

For more information about `ASM_PREFERRED_READ_FAILURE_GROUPS`, refer to "Preferred Read Failure Groups" on page 4-30.

See Also:

- *Oracle Real Application Clusters Administration and Deployment Guide* for more information about configuring preferred disks in extended clusters

- *Oracle Database Reference* for more information about the `ASM_PREFERRED_READ_FAILURE_GROUPS` initialization parameter

DB_CACHE_SIZE

You do not have to set a value for the `DB_CACHE_SIZE` initialization parameter if you use automatic memory management. The setting for the `DB_CACHE_SIZE` parameter determines the size of the buffer cache. This buffer cache is used to store metadata blocks. The default value for this parameter is suitable for most environments.

See Also:

- *Oracle Database Administrator's Guide* for more information about setting the `DB_CACHE_SIZE` initialization parameter

- *Oracle Database Performance Tuning Guide* for more information about memory configuration

- *Oracle Database Reference* for more information about the `DB_CACHE_SIZE` parameter

DIAGNOSTIC_DEST

The `DIAGNOSTIC_DEST` initialization parameter specifies the directory where diagnostics for an instance are located. The default value for an Oracle ASM instance is the `$ORACLE_BASE` directory for the Oracle Grid Infrastructure installation.

Example 3–1 shows an example of the diagnostic directory for an Oracle ASM instance.

Example 3–1   Sample diagnostic directory for an Oracle ASM instance

```
$ ls $ORACLE_BASE/diag/asm/+asm/+ASM
alert  cdump  hm  incident  incpkg  ir  lck  metadata  stage  sweep  trace
```

See Also:

- *Oracle Database Administrator's Guide* for more information about the `DIAGNOSTIC_DEST` initialization parameter and the fault diagnosability infrastructure.

- *Oracle Database Reference* for more information about the `DIAGNOSTIC_DEST` parameter

INSTANCE_TYPE

The `INSTANCE_TYPE` initialization parameter is optional for an Oracle ASM instance in an Oracle Grid Infrastructure home.

The following is an example of the `INSTANCE_TYPE` parameter in the initialization file:

```
INSTANCE_TYPE = ASM
```

See Also: *Oracle Database Reference* for more information about the `INSTANCE_TYPE` parameter

LARGE_POOL_SIZE

You do not have to set a value for the `LARGE_POOL_SIZE` initialization parameter if you use automatic memory management.

The setting for the `LARGE_POOL_SIZE` parameter is used for large allocations. The default value for this parameter is suitable for most environments.

See Also:

- *Oracle Database Administrator's Guide* for more information about setting the `LARGE_POOL_SIZE` initialization parameter

- *Oracle Database Performance Tuning Guide* for more information about memory configuration

- *Oracle Database Reference* for more information about the `LARGE_POOL_SIZE` parameter

PROCESSES

The `PROCESSES` initialization parameter affects Oracle ASM, but the default value is usually suitable. However, if multiple database instances are connected to an Oracle ASM instance, you can use the following formula:

```
PROCESSES = 50 + 50*n
```

where `n` is the number database instances connecting to the Oracle ASM instance.

See Also:

- *Oracle Database Administrator's Guide* for more information about setting the `PROCESSES` initialization parameter

- *Oracle Database Reference* for more information about the `PROCESSES` parameter

REMOTE_LOGIN_PASSWORDFILE

The `REMOTE_LOGIN_PASSWORDFILE` initialization parameter specifies whether the Oracle ASM instance checks for a password file. This parameter operates the same for Oracle ASM and database instances.

> **See Also:**
>
> - *Oracle Database Administrator's Guide* for more information about setting the `REMOTE_LOGIN_PASSWORDFILE` initialization parameter
>
> - *Oracle Database Reference* for more information about the `REMOTE_LOGIN_PASSWORDFILE` parameter

SHARED_POOL_SIZE

You do not have to set a value for the `SHARED_POOL_SIZE` initialization parameter if you use automatic memory management. The setting for the `SHARED_POOL_SIZE` parameter determines the amount of memory required to manage the instance. The setting for this parameter is also used to determine the amount of space that is allocated for extent storage. The default value for this parameter is suitable for most environments.

> **See Also:**
>
> - *Oracle Database Administrator's Guide* for more information about setting the `SHARED_POOL_SIZE` initialization parameter
>
> - *Oracle Database Performance Tuning Guide* for more information about memory configuration
>
> - *Oracle Database Reference* for more information about the `SHARED_POOL_SIZE` parameter

Setting Database Initialization Parameters for Use with Oracle ASM

When you do not use automatic memory management in a database instance, the SGA parameter settings for a database instance may require minor modifications to support Oracle ASM. When you use automatic memory management, the sizing data discussed in this section can be treated as informational only or as supplemental information to help determine the appropriate values that you should use for the SGA. Oracle highly recommends using automatic memory management.

> **See Also:**
>
> - *Oracle Database Administrator's Guide* for information about managing memory allocation in an Oracle Database instance
>
> - *Oracle Database Performance Tuning Guide* for more information about memory configuration and use

The following are configuration guidelines for SGA sizing on the database instance:

- `PROCESSES` initialization parameter—Add 16 to the current value

- `LARGE_POOL_SIZE` initialization parameter—Add an additional 600K to the current value

- `SHARED_POOL_SIZE` initialization parameter—Aggregate the values from the following queries to obtain the current database storage size that is either on Oracle ASM or stored in Oracle ASM. Next, determine the redundancy type and calculate the `SHARED_POOL_SIZE` using the aggregated value as input.

  ```
  SELECT SUM(bytes)/(1024*1024*1024) FROM V$DATAFILE;
  ```

```
SELECT SUM(bytes)/(1024*1024*1024) FROM V$LOGFILE a, V$LOG b
      WHERE a.group#=b.group#;
SELECT SUM(bytes)/(1024*1024*1024) FROM V$TEMPFILE
      WHERE status='ONLINE';
```

– For disk groups using external redundancy, every 100 GB of space needs 1 MB of extra shared pool plus 2 MB

– For disk groups using normal redundancy, every 50 GB of space needs 1 MB of extra shared pool plus 4 MB

– For disk groups using high redundancy, every 33 GB of space needs 1 MB of extra shared pool plus 6 MB

See Also:

- *Oracle Database Administrator's Guide* for information about managing memory allocation in an Oracle Database instance

- *Oracle Database Performance Tuning Guide* for more information about memory configuration and use

Managing Oracle ASM Instances

This section describes how to administer Oracle ASM instances under the following topics:

- Administering Oracle ASM Instances with Server Control Utility

- Using Oracle Restart

- Starting Up an Oracle ASM Instance

- Shutting Down an Oracle ASM Instance

- Upgrading an Oracle ASM Instance in an Oracle Restart Configuration with Oracle Universal Installer

- Downgrading an Oracle ASM Instance in an Oracle Restart Configuration

- Out of Place Upgrades

- Configuring Oracle Grid Infrastructure with the Configuration Wizard

- Active Session History Sampling for Oracle ASM

Administering Oracle ASM Instances with Server Control Utility

In addition to the Oracle ASM administration procedures that this section describes, you can use Server Control Utility (SRVCTL) in clustered Oracle ASM environments to perform the following Oracle ASM administration tasks:

- Add and remove the Oracle ASM CRS resource in Oracle Cluster Registry (OCR)

- Enable, disable, start, and stop Oracle ASM instances

- Display the Oracle ASM instance configuration and status

    See Also: The *Oracle Real Application Clusters Administration and Deployment Guide* for information about administering Oracle ASM instances with SRVCTL

Using Oracle Restart

Oracle Restart improves the availability of your Oracle database. When you install the Oracle Grid Infrastructure for a standalone server, it includes both Oracle ASM and Oracle Restart. Oracle Restart runs out of the Oracle Grid Infrastructure home, which you install separately from Oracle Database homes.

Oracle Restart provides managed startup and restart of a single-instance (non-clustered) Oracle Database, Oracle ASM instance, service, listener, and any other process running on the server. If an interruption of a service occurs after a hardware or software failure, Oracle Restart automatically takes the necessary steps to restart the component.

With Server Control Utility (SRVCTL) you can add a component, such as an Oracle ASM instance, to Oracle Restart. You then enable Oracle Restart protection for the Oracle ASM instance. With SRVCTL, you also remove or disable Oracle Restart protection.

> **See Also:**
>
> - *Oracle Database Administrator's Guide* for information about configuring and administering Oracle Restart
> - *Oracle Real Application Clusters Administration and Deployment Guide* for information about automatically restarting single-instance databases residing on an Oracle RAC node
> - *Oracle Grid Infrastructure Installation Guide* for information about installation of Oracle Grid Infrastructure

Starting Up an Oracle ASM Instance

This section describes how to start Oracle ASM instances under the following topics:

- Connecting To and Starting Up an Oracle ASM Instance
- Starting Up an Oracle ASM instance with an Incorrect SPFILE Path
- About Mounting Disk Groups at Startup
- About Restricted Mode

Connecting To and Starting Up an Oracle ASM Instance

You start an Oracle ASM instance similarly to the way in which you start an Oracle database instance with some minor differences.

When starting an Oracle ASM instance, note the following:

- To connect to a local Oracle ASM instance with SQL*Plus, set the `ORACLE_SID` environment variable to the Oracle ASM system identifier (SID).

  The default Oracle ASM SID for a single-instance database is `+ASM`, and the default SID for Oracle ASM for an Oracle RAC node is `+ASMnode_number` where `node_number` is the number of the node. The `ORACLE_HOME` environment variable must be set to the Grid Infrastructure home where Oracle ASM was installed.

  > **Note:** Oracle recommends that you do not change the default Oracle ASM SID name.

- The initialization parameter file must contain the following entry:

```
INSTANCE_TYPE = ASM
```

This parameter indicates that an Oracle ASM instance, not a database instance, is starting.

- When you run the `STARTUP` command, rather than trying to mount and open a database, this command attempts to mount Oracle ASM disk groups.

  For information about disk groups that are mounted at startup time, see "About Mounting Disk Groups at Startup" on page 3-16.

  After the Oracle ASM instance has started, you can mount disk groups with the `ALTER DISKGROUP...MOUNT` command. See "Mounting and Dismounting Disk Groups" on page 4-44 for more information.

- The associated Oracle database instance does not have to be running when you start the associated Oracle ASM instance.

The following list describes how Oracle ASM interprets SQL*Plus `STARTUP` command parameters.

- `FORCE` Parameter

  Issues a `SHUTDOWN ABORT` to the Oracle ASM instance before restarting it.

  If an Oracle Automatic Storage Management Cluster File System (Oracle ACFS) file system is currently mounted on Oracle ADVM volumes, the file system should first be dismounted. Otherwise, applications encounter I/O errors and Oracle ACFS user data and metadata may not be written to storage before the Oracle ASM storage is fenced. For information about dismounting an Oracle ACFS file system, see "Deregistering, Dismounting, and Disabling Volumes and Oracle ACFS File Systems" on page 13-18.

- `MOUNT` or `OPEN` Parameter

  Mounts the disk groups specified in the `ASM_DISKGROUPS` initialization parameter. This is the default if no command parameter is specified.

- `NOMOUNT` Parameter

  Starts up the Oracle ASM instance without mounting any disk groups.

- `RESTRICT` Parameter

  Starts up an instance in restricted mode that enables access only to users with both the `CREATE SESSION` and `RESTRICTED SESSION` system privileges. You can use the `RESTRICT` clause in combination with the `MOUNT`, `NOMOUNT`, and `OPEN` clauses.

  > **See Also:** "About Restricted Mode" on page 3-16 for more information

  In restricted mode, database instances cannot use the disk groups. In other words, databases cannot open files that are in that disk group. Also, the disk group cannot be mounted by any other instance in the cluster. Mounting the disk group in restricted mode enables only one Oracle ASM instance to mount the disk group. This mode is useful to mount the disk group for repairing configuration issues.

The following is a sample SQL*Plus session for starting an Oracle ASM instance.

```
SQLPLUS /NOLOG
SQL> CONNECT SYS AS SYSASM
Enter password: sys_password
Connected to an idle instance.
```

```
SQL> STARTUP
ASM instance started

Total System Global Area    71303168 bytes
Fixed Size                   1069292 bytes
Variable Size               45068052 bytes
ASM Cache                   25165824 bytes
ASM disk groups mounted
```

For more information about user authentication, see "Authentication for Accessing Oracle ASM Instances" on page 3-25.

> **See Also:**
>
> - *Oracle Database Administrator's Guide* for more information about using environment variables to select instances
> - *Oracle Database Administrator's Guide* for more information about starting up and shutting down Oracle instances
> - *Oracle Real Application Clusters Administration and Deployment Guide* for information about starting an Oracle ASM instance with `SRVCTL` in Oracle RAC
> - *Oracle Clusterware Administration and Deployment Guide* for information about Oracle Clusterware Cluster subcomponent processes and background processes
> - *Oracle Database Concepts* for information about Oracle database background processes
> - *Oracle Database Reference* for a description of the Oracle background processes

Starting Up an Oracle ASM instance with an Incorrect SPFILE Path

If the SPFILE path in the GPNP profile is incorrect, you can start the Oracle ASM instance as follows:

1. Create a PFILE with one line in it that identifies the path to the SPFILE.

   For example:

   Create the `/u01/oracle/dbs/spfileasm_init.ora` file that contains:

   ```
   SPFILE='+DATA/asm/asmparameterfile/asmspfile.ora'
   ```

2. Start up the instance using the initialization parameter file.

   For example:

   ```
   SQL> STARTUP PFILE=/u01/oracle/dbs/spfileasm_init.ora
   ```

3. After the instance is running, use the ASMCMD `spset` command to update the SPFILE path in the GPNP profile. See "spset" on page 12-19.

   For example:

   ```
   ASMCMD> spset +DATA/asm/asmparameterfile/asmspfile.ora
   ```

   > **See Also:** *Oracle Database Administrator's Guide* for more information about using `STARTUP` with a non-default server parameter file

About Mounting Disk Groups at Startup

At startup, the Oracle ASM instance attempts to mount the following disk groups:

- Disk groups specified in the `ASM_DISKGROUPS` initialization parameter

- Disk group used by Cluster Synchronization Services (CSS) for voting files

- Disk groups used by Oracle Clusterware for Oracle Cluster Registry (OCR)

- Disk group used by the Oracle ASM instance to store the ASM server parameter file (SPFILE)

If no disk groups are found in the previous list, then the Oracle ASM instance does not mount any disk groups at startup. After the Oracle ASM instance has started, you can mount disk groups with the `ALTER DISKGROUP...MOUNT` command. For more information, see "Mounting and Dismounting Disk Groups" on page 4-44.

About Restricted Mode

You can use the `STARTUP RESTRICT` command to control access to an Oracle ASM instance while you perform maintenance. When an Oracle ASM instance is active in this mode, all of the disk groups that are defined in the `ASM_DISKGROUPS` parameter are mounted in `RESTRICTED` mode. This prevents databases from connecting to the Oracle ASM instance. In addition, the restricted clause of the `ALTER SYSTEM` statement is disabled for the Oracle ASM instance. The `ALTER DISKGROUP` *diskgroup* `MOUNT` statement is extended to enable Oracle ASM to mount a disk group in restricted mode.

When you mount a disk group in `RESTRICTED` mode, the disk group can only be mounted by one instance. Clients of Oracle ASM on that node cannot access that disk group while the disk group is mounted in `RESTRICTED` mode. The `RESTRICTED` mode enables you to perform maintenance tasks on a disk group in the Oracle ASM instance without interference from clients.

Rebalance operations that occur while a disk group is in `RESTRICTED` mode eliminate the lock and unlock extent map messaging that occurs between Oracle ASM instances in an Oracle RAC environment. This improves the overall rebalance throughput. At the end of a maintenance period, you must explicitly dismount the disk group and remount it in normal mode.

Shutting Down an Oracle ASM Instance

The Oracle ASM shutdown process is initiated when you run the `SHUTDOWN` command in SQL*Plus. Before you run this command, ensure that the `ORACLE_SID` environment variable is set to the Oracle ASM SID so that you can connect to the local Oracle ASM instance. The default Oracle ASM SID for a single-instance database is `+ASM`, and the default SID for Oracle ASM for an Oracle RAC node is `+ASM`*node_number* where *node_number* is the number of the node. The `ORACLE_HOME` environment variable must be set to the Grid Infrastructure home where Oracle ASM was installed.

Oracle strongly recommends that you shut down all database instances that use the Oracle ASM instance and dismount all file systems mounted on Oracle ASM Dynamic Volume Manager (Oracle ADVM) volumes before attempting to shut down the Oracle ASM instance.

If Oracle Cluster Registry (OCR) or voting files are stored in a disk group, the disk group can only be dismounted by shutting down the Oracle ASM instance as part of shutting down the clusterware on a node. To shut down the clusterware, run `crsctl stop crs`.

See Also:

- *Oracle Database Administrator's Guide* for more information about using environment variables to select instances

- *Oracle Database Administrator's Guide* for more information about starting up and shutting down Oracle instances

- *Oracle Clusterware Administration and Deployment Guide* for information about managing voting files, Oracle Cluster Registry (OCR), and Oracle Local Registry (OLR)

To shut down an Oracle ASM instance, perform the following steps:

```
SQLPLUS /NOLOG
SQL> CONNECT SYS AS SYSASM
Enter password: sys_password
Connected.
SQL> SHUTDOWN NORMAL
```

For more information about user authentication, see "Authentication for Accessing Oracle ASM Instances" on page 3-25.

The following list describes the SHUTDOWN modes and the behavior of the Oracle ASM instance in each mode.

- NORMAL Clause

  Oracle ASM waits for any in-progress SQL to complete before performing an orderly dismount of all of the disk groups and shutting down the Oracle ASM instance. Before the instance is shut down, Oracle ASM waits for all of the currently connected users to disconnect from the instance. If any database instances are connected to the Oracle ASM instance, then the SHUTDOWN command returns an error and leaves the Oracle ASM instance running. NORMAL is the default shutdown mode.

- IMMEDIATE or TRANSACTIONAL Clause

  Oracle ASM waits for any in-progress SQL to complete before performing an orderly dismount of all of the disk groups and shutting down the Oracle ASM instance. Oracle ASM does not wait for users currently connected to the instance to disconnect. If any database instances are connected to the Oracle ASM instance, then the SHUTDOWN command returns an error and leaves the Oracle ASM instance running. Because the Oracle ASM instance does not contain any transactions, the TRANSACTIONAL mode behaves the same as IMMEDIATE mode.

- ABORT Clause

  The Oracle ASM instance immediately shuts down without the orderly dismount of disk groups. This causes recovery to occur upon the next Oracle ASM startup.

  If any database instance is connected to the Oracle ASM instance, then the database instance aborts.

  If any Oracle Automatic Storage Management Cluster File System (Oracle ACFS) file systems are currently mounted on Oracle ADVM volumes, those file systems should first be dismounted. Otherwise, applications encounter I/O errors and Oracle ACFS user data and metadata may not be written to storage before the Oracle ASM storage is fenced. For information about dismounting an Oracle ACFS file system, see "Deregistering, Dismounting, and Disabling Volumes and Oracle ACFS File Systems" on page 13-18. For more information about user

authentication on Oracle ASM instance, see "Authentication for Accessing Oracle ASM Instances" on page 3-25.

Upgrading an Oracle ASM Instance in an Oracle Restart Configuration with Oracle Universal Installer

This section discusses the process to upgrade an Oracle ASM instance to an Oracle Restart 11*g* release 2 (11.2) configuration. The recommended practice is to upgrade an Oracle ASM instance with Oracle Universal Installer (OUI). OUI automatically defaults to upgrade mode when it detects an Oracle ASM instance at a previous release level.

This section discusses the following topics:

- Upgrading an Oracle ASM Instance from 11.1 to 11.2 in an Oracle Restart Configuration with Oracle Universal Installer

- Upgrading an Oracle ASM Instance from 11.2.0.x to 11.2.0.3 in an Oracle Restart Configuration with Oracle Universal Installer

---

Notes:

- For information about performing an Oracle ASM Rolling Upgrade from 11.1.0.6 to 11.2.0.1, refer to Oracle Database Readme.

- For information about upgrading Oracle ASM from version 11.2.0.1 to 11.2.0.2, see "Out of Place Upgrades" on page 3-24.

- In an Oracle Clusterware configuration, the Oracle ASM instance is upgraded as part of the Oracle Clusterware upgrade.

- If you are making any changes to Oracle software, Oracle recommends that you create a backup of the Oracle software.

See Also:

- *Oracle Grid Infrastructure Installation Guide* for information about installing and upgrading Oracle Grid Infrastructure

- Articles at My Oracle Support (`https://support.oracle.com`) for information about upgrading Oracle ASM

For information about copying and moving an Oracle ASM instance initialization parameter file after upgrading, see "Backing Up, Copying, and Moving an Oracle ASM Initialization Parameter File" on page 3-4.

Note: The procedures described in this section upgrade the Oracle ASM instance and Oracle Restart (standalone) configuration. To upgrade Oracle Database and Oracle Enterprise Manager, see *Oracle Database Upgrade Guide*.

Upgrading an Oracle ASM Instance from 11.1 to 11.2 in an Oracle Restart Configuration with Oracle Universal Installer

The following procedure describes how to upgrade an Oracle ASM instance from 11*g* release 1 (11.1) to 11*g* release 2 (11.2) in an Oracle Restart (standalone) configuration. In this scenario:

- The Oracle ASM and Oracle Database 11*g* release 1 (11.1) instances exist in separate homes.

- The Oracle Grid Infrastructure 11*g* release 2 (11.2) is to be installed in a separate home and the Oracle ASM instance 11*g* release 2 (11.2) is to be set up as an Oracle Restart (single-instance) configuration.

1. Shut down the Oracle Enterprise Manager agent, Oracle Database instances, Oracle ASM instance, and the listener in the older database and Oracle ASM homes.

   - Run `emctl stop dbconsole` to stop the Oracle Enterprise Manager agent.

   - Connect to the database instances with SQL*Plus as a privileged user and run the `SHUTDOWN` command.

   - Connect to the Oracle ASM instance with SQL*Plus as a privileged user and run the `SHUTDOWN` command.

   - Run `lsnrctl` and enter the `STOP` command to stop the listener.

   For information about shutting down an Oracle ASM instance, see "Shutting Down an Oracle ASM Instance" on page 3-16.

   > **See Also:**
   >
   > - Oracle Enterprise Manager manuals and online help for information about starting and stopping the Oracle Enterprise Manager agent
   >
   > - *Oracle Database Administrator's Guide* for more information about starting up and shutting down Oracle instances
   >
   > - *Oracle Database Net Services Administrator's Guide* for information about configuring a listener

2. Start the Oracle Grid Infrastructure OUI and select the **Upgrade Oracle Grid Infrastructure** option.

3. Complete the screens in the OUI installer and run the scripts as prompted by the OUI installer.

   For example, on Linux you must run the `root.sh` script as the `root` user.

   ```
   # GRID_HOME/root.sh
   ```

4. Confirm that the listener and Oracle ASM instance are running in the Oracle Grid Infrastructure home and ensure that the Oracle Database instance and Oracle Enterprise Manager agent are running in the old database home.

   - Confirm that the listener is running.

     Otherwise, start the listener with Server Control Utility (SRVCTL).

     For example:

     ```
     $ srvctl start listener
     ```

- Confirm that the Oracle ASM instance is running.

  For example:

  ```
  $ srvctl status listener
  $ srvctl status asm
  ```

  Otherwise, start the Oracle ASM instance with SRVCTL.

  For example:

  ```
  $ srvctl start asm
  ```

- Ensure that the database instances are running. Otherwise, connect to the database instances with SQL*Plus as a privileged user and run the STARTUP command.

- Ensure that the Oracle Enterprise Manager agent is running. Otherwise start the Oracle Enterprise Manager agent with emctl start dbconsole.

Upgrading an Oracle ASM Instance from 11.2.0.x to 11.2.0.3 in an Oracle Restart Configuration with Oracle Universal Installer

The following procedure describes how to upgrade an Oracle ASM instance from 11*g* release 2 (11.2.0.*x*) to 11*g* release 2 (11.2.0.3) in an Oracle Restart (standalone) configuration. In this scenario:

- The Oracle ASM and Oracle Database 11*g* release 2 instances exist in separate homes.

- The Oracle Grid Infrastructure 11*g* release 2 (11.2.0.3) is to be installed in a separate home and the Oracle ASM instance 11*g* release 2 (11.2.0.3) is to be set up as an Oracle Restart (single-instance) configuration.

1. Shut down the Oracle Enterprise Manager agent, Oracle Database instances, and the listener in the older database home.

   - Run emctl stop dbconsole to stop the Oracle Enterprise Manager agent.

   - Connect to the database instances with SQL*Plus as a privileged user and run the SHUTDOWN command.

     See Also:

     - Oracle Enterprise Manager manuals and online help for information about starting and stopping the Oracle Enterprise Manager agent

     - *Oracle Database Administrator's Guide* for more information about starting up and shutting down Oracle instances

2. Start the Oracle Grid Infrastructure OUI and select the **Upgrade Oracle Grid Infrastructure** option.

3. Complete the screens in the OUI installer and run the scripts as prompted by the OUI installer.

   For example, on Linux you must run the rootupgrade.sh script as the root user:

   ```
   # GRID_HOME/rootupgrade.sh
   ```

4. Confirm that the listener and Oracle ASM instance are running in the Oracle Grid Infrastructure home and ensure that the Oracle Database instance and Oracle Enterprise Manager agent are running in the old database home.

- Confirm that the listener is running.

  Otherwise, start the listener with Server Control Utility (SRVCTL).

  For example:

  ```
  $ srvctl start listener
  ```

- Confirm that the Oracle ASM instance is running.

  For example:

  ```
  $ srvctl status listener
  $ srvctl status asm
  ```

  Otherwise, start the Oracle ASM instance with SRVCTL.

  For example:

  ```
  $ srvctl start asm
  ```

- Ensure that the database instances are running. Otherwise, connect to the database instances with SQL*Plus as a privileged user and run the STARTUP command.

- Ensure that the Oracle Enterprise Manager agent is running. Otherwise start the Oracle Enterprise Manager agent with emctl start dbconsole.

Downgrading an Oracle ASM Instance in an Oracle Restart Configuration

This section discusses the process to downgrade an Oracle ASM instance that has been upgraded to an Oracle Restart (standalone) configuration.

Notes:

- In an Oracle Clusterware configuration, the Oracle ASM instance is downgraded as part of the Oracle Clusterware downgrade.

- If you are making any changes to Oracle software, Oracle recommends that you create a backup of the Oracle software.

See Also:

- *Oracle Grid Infrastructure Installation Guide* for information about downgrading Oracle Clusterware

- Articles at My Oracle Support (https://support.oracle.com) for information about downgrading Oracle ASM

The following procedure describes how to downgrade an Oracle ASM instance from Oracle 11*g* release 2 (11.2) to Oracle 11*g* release 1 (11.1). In this scenario, the Oracle ASM instance was previously upgraded from an Oracle 11*g* release 1 (11.1) home to an Oracle Restart (standalone) 11*g* release 2 (11.2) configuration. The Oracle 11*g* release 1 (11.1) home was not removed.

1. Determine disk group compatibility attribute settings.

   If compatibility attributes have been advanced, then the disk groups must be re-created using compatibility attributes that allow access by the downgraded Oracle ASM and Oracle Database instances. A new disk group must be created

with the old compatibility attributes and then you must restore the database files that were in the disk group.

When you revert to a new disk group with the old compatibility attribute settings, the latest Oracle ASM features might not be available. For example, if you revert the disk group compatibility to a pre-11.2 value, Oracle ACFS functionality is not available.

Copy or move an Oracle ASM SPFILE in a disk group to the file system before reverting disk group compatibility. Check the initialization parameters to ensure they are compatible with Oracle ASM 11*g* release 1 (11.1).

For information about reverting disk group compatibility, see "Reverting Disk Group Compatibility" on page 4-39. For information about moving data files between disk groups, see "Moving Data Files Between Oracle ASM Disk Groups Using RMAN" on page 8-9.

2. Downgrade any client databases from 11*g* release 2 (11.2) down to 11*g* release 1 (11.1).

> **See Also:** *Oracle Database Upgrade Guide* for information about downgrading an Oracle Database and Oracle Enterprise Manager

3. Shut down the Oracle Enterprise Manager agent, Oracle Database instance, Oracle ASM instance, and the listener in the database and Oracle ASM homes.

- Run `emctl stop dbconsole` to stop the Oracle Enterprise Manager agent.

- Connect to the database instances with SQL*Plus as a privileged user and run the `SHUTDOWN` command.

- Shut down the Oracle ASM instance with Server Control Utility (SRVCTL).

  ```
  $ srvctl stop asm
  ```

- Stop the listener with SRVCTL.

  ```
  $ srvctl stop listener
  ```

For information about shutting down an Oracle ASM instance, see "Shutting Down an Oracle ASM Instance" on page 3-16.

> **See Also:**
>
> - Oracle Enterprise Manager manuals and online help for information about starting and stopping the Oracle Enterprise Manager agent
>
> - *Oracle Database Administrator's Guide* for more information about starting up and shutting down Oracle instances
>
> - *Oracle Database Net Services Administrator's Guide* for information about configuring a listener
>
> - *Oracle Real Application Clusters Administration and Deployment Guide* for information about Server Control Utility (SRVCTL)

4. Deconfigure the Oracle Restart 11*g* release 2 (11.2) configuration.

Run the `roothas.pl` script as `root`.

For example, on Linux:

```
# GRID_HOME/crs/install/roothas.pl -deconfig
```

The Oracle 11*g* release 2 (11.2) `inittab` and `init*` scripts should be removed with the deconfiguration of Oracle Clusterware.

5. Unload the Oracle ACFS drivers.

   For example, on Linux run `acfsload stop` as `root`.

   ```
   # GRID_HOME/bin/acfsload stop
   ```

   For information about Oracle ACFS driver resource management, see "Oracle ACFS Drivers Resource Management" on page B-3.

6. Re-create the Oracle ASM 11*g* release 1 (11.1) resources.

   Run `localconfig` as `root` to add the resources to the Oracle ASM 11*g* release 1 (11.1) home.

   For example, on Linux:

   ```
   # ORACLE_ASM_11.1_HOME/bin/localconfig add
   ```

   If `localconfig add` fails, then use the `reset` option followed by the `ORACLE_HOME` to reset the existing resources.

   For example, on Linux:

   ```
   # localconfig reset ORACLE_ASM_11.1_HOME
   ```

7. Confirm that the Oracle ASM PFILE and `listerner.ora` files are present in the Oracle ASM 11*g* release 1 (11.1) home.

   If the Oracle ASM 11*g* release 1 (11.1) home has not been removed, then the files should be available.

8. Configure additional configuration files in the Oracle ASM 11*g* release 1 (11.1) home.

   For example, update files in the `/etc` directory on the Linux computer.

   Update the Oracle ASM entry in `/etc/oratab` to point to the Oracle ASM 11*g* release 1 (11.1) home, as shown in the following example:

   ```
   +ASM:/ORACLE_ASM_11.1_HOME/product/11.1.0/asm_1:N
   ```

   The 11*g* release 2 (11.2) `inittab` and `init*` scripts should be removed with the deconfiguration of Oracle Clusterware.

9. Ensure that the listener, Oracle ASM instance, Oracle Database instance, and Oracle Enterprise Manager agent are running in the 11*g* release 1 (11.1) Oracle ASM and database homes.

   - Start the listener with `lsnrctl` and enter the `START` option.

     If necessary, start Network Configuration Assistant (NETCA) in the Oracle ASM 11*g* release 1 (11.1) home with `netca`. Follow the prompts in the wizard to reconfigure the listener.

   - Connect to the Oracle ASM instance with SQL*Plus as a privileged user and run the `STARTUP` command.

   - Connect to the database instances with SQL*Plus as a privileged user and run the `STARTUP` command.

   - Start the Oracle Enterprise Manager agent with `emctl start dbconsole`.

Oracle Enterprise Manager may require configuration after the Oracle ASM instance has been downgraded.

Out of Place Upgrades

With an out-of-place upgrade, the installer installs the newer version of Oracle Grid Infrastructure in a separate Oracle Grid Infrastructure home.

An in-place upgrade of Oracle Grid Infrastructure 11g release 2 is not supported. For example, an upgrade of Oracle Grid Infrastructure 11.2.0.1 to 11.2.0.2 must be an out of place upgrade.

> **See Also:** *Oracle Grid Infrastructure Installation Guide* for information about installing Oracle Grid Infrastructure, out of place upgrades, and performing rolling upgrades of Oracle Grid Infrastructure and Oracle ASM

Configuring Oracle Grid Infrastructure with the Configuration Wizard

The Oracle Grid Infrastructure configuration wizard can update the configuration of an Oracle Grid Infrastructure environment after the software has been installed. The configuration wizard accepts your input, validates the input, and populates the configuration data into the CRSCONFIG_PARAMS file. If additional scripts must be run, the configuration wizard directs you to run those scripts.

> **See Also:** *Oracle Clusterware Administration and Deployment Guide* for information about the Oracle Grid Infrastructure configuration wizard.

Active Session History Sampling for Oracle ASM

Active Session History sampling is now available on Oracle ASM instances. This activity is exposed in the dynamic V$ACTIVE_SESSION_HISTORY view. Active Session History sampling requires a diagnostic pack license for the Oracle ASM instance.

> **See Also:**
>
> ■ *Oracle Database Performance Tuning Guide* for more information about gathering performance statistics
>
> ■ *Oracle Database Reference* for a description of the V$ACTIVE_ SESSION_HISTORY view

Using Oracle ASM Rolling Upgrade

> **Note:** For information about performing an Oracle ASM Rolling Upgrade from 11.1.0.6 to 11.2.0.1, refer to Oracle Database Readme

Oracle ASM rolling upgrade enables you to independently upgrade or patch clustered Oracle ASM nodes without affecting database availability, thus providing greater uptime. Rolling upgrade means that some features of a clustered Oracle ASM environment continue to function when one or more of the nodes in the cluster uses different software versions. Oracle recommends that you perform an Oracle ASM rolling upgrade when performing an Oracle Clusterware rolling upgrade. Note that

Oracle ASM is upgraded with Oracle Clusterware for Oracle 11g Release 2 (11.2) or later as both are in the Oracle Grid Infrastructure 11g Release 2 (11.2) home.

To perform a rolling upgrade from Oracle 11g Release 1 (11.1) to Oracle 11g Release 2 (11.2), your environment must be prepared. Oracle Clusterware must be fully upgraded to the next patch or release version before you start the Oracle ASM rolling upgrade. In addition, you should prepare your Oracle Clusterware in a rolling upgrade manner to ensure high availability and maximum uptime. Note that the rolling upgrade to Oracle 11g Release 2 (11.2) moves the Oracle ASM instance to Oracle Grid Infrastructure 11g Release 2 (11.2) home.

You can upgrade a single Oracle ASM instance with Oracle Universal Installer (OUI). For information, see

> **Notes:**
>
> - Rolling upgrades only apply to clustered Oracle ASM instances, and you can only perform rolling upgrades on environments with Oracle Database 11*g* or later. In other words, you cannot use this feature to upgrade from Oracle Database 10*g* to Oracle Database 11*g*.
>
> - See *Oracle Exadata documentation* for information about performing a rolling upgrading of an Oracle ASM instance when Oracle Exadata storage is present.

> **See Also:**
>
> - *Oracle Grid Infrastructure Installation Guide* for information about performing a rolling upgrade of Oracle ASM
>
> - *Oracle Database Upgrade Guide* for information about upgrading Oracle Database
>
> - *Oracle Database SQL Language Reference* for information about the rolling migration clause of the ALTER SYSTEM commands

Patching Oracle ASM Instances in Oracle RAC Environments

For Oracle RAC environments, ensure that your Oracle Clusterware version is at least equal to the version of the patch that you are applying to the Oracle Database. First apply the patch to the Oracle Grid Infrastructure home and then apply the patch to the Oracle Database home.

> **Note:** You must apply the patch to the Oracle Grid Infrastructure home before you apply it to the Oracle Database home.

Authentication for Accessing Oracle ASM Instances

An Oracle ASM instance does not have a data dictionary, so the only way to connect to an Oracle ASM instance is by using one of three system privileges, SYSASM, SYSDBA, or SYSOPER. There are three modes of connecting to Oracle ASM instances:

- Local connection using operating system authentication

- Local connection using password authentication

- Remote connection by way of Oracle Net Services using password authentication

  > **See Also:** Your operating system-specific *Oracle Grid Infrastructure Installation Guide* for information about how to ensure that the Oracle ASM and database instances have member disk access

This section describes the following topics:

- About Privileges for Oracle ASM
- Creating Users with the SYSASM Privilege
- Operating System Authentication for Oracle ASM
- Password File Authentication for Oracle ASM

The Oracle ASM and database instances must have read/write operating system access rights to disk groups. For example, the Oracle ASM instance and the database instance must have identical read and write permissions for the disks that comprise the related Oracle ASM disk group. For Linux and UNIX systems, this is typically provided through shared Linux and UNIX group membership (OSASM group). On Windows systems, the Oracle ASM service must be run as Administrator. For information about file permissions and Oracle ASM File Access Control, see "Managing Oracle ASM File Access Control for Disk Groups" on page 4-40.

About Privileges for Oracle ASM

During Oracle ASM installation, you can use one operating system group for all users or divide system privileges so that database administrators, storage administrators, and database operators each have distinct operating system privilege groups.

Whether you create separate operating system privilege groups or use one group to provide operating system authentication for all system privileges, you should use SYSASM to administer an Oracle ASM instance. The SYSDBA privilege cannot be used to administer an Oracle ASM instance. If you use the SYSDBA privilege to run administrative commands on an Oracle ASM instance, the operation results in an error. The SYSDBA privilege is intended to be used by the database to access disk groups.

Oracle also recommends the use of a less privileged user, such as ASMSNMP with SYSDBA privileges that is created during installation, for monitoring the Oracle ASM instance.

Operating system authentication using membership in the group or groups designated as OSDBA, OSOPER, and OSASM is valid on all Oracle platforms. Connecting to an Oracle ASM instance as SYSASM grants you full access to all of the available Oracle ASM disk groups and management functions.

This section contains these topics:

- Using One Operating System Group for Oracle ASM Users
- Using Separate Operating System Groups for Oracle ASM Users
- The SYSASM Privilege for Administering Oracle ASM
- The SYSDBA Privilege for Managing Oracle ASM Components

For information about privileges and Oracle ACFS, see "Oracle ACFS and File Access and Administration Security" on page 5-11.

Using One Operating System Group for Oracle ASM Users

If you do not want to divide the privileges for system access into separate operating system groups, then you can designate one operating system group as the group whose members are granted access as OSDBA, OSOPER, and OSASM for Oracle ASM privileges. The default operating system group name for all of these is usually `dba` and that group is typically chosen for the default configuration.

Table 3–1 shows an example of a Linux deployment without separated privileges for Oracle ASM users.

Table 3–1   One operating system group and one set of privileges for all Oracle ASM users

Role/Software Owner	User	Group/Privilege
Oracle ASM administrator/Oracle Grid Infrastructure home	oracle	dba/SYSASM, SYSDBA, SYSOPER
Database administrator 1/Database home 1	oracle	dba/SYSASM, SYSDBA, SYSOPER
Database administrator 2/Database home 2	oracle	dba/SYSASM, SYSDBA, SYSOPER
Operating system disk device owner	oracle	dba

Using Separate Operating System Groups for Oracle ASM Users

You can designate separate operating system groups as the operating system authentication groups for privileges on Oracle ASM. The following list describes the separate operating system authentication groups for Oracle ASM and the privileges that their members are granted.

- OSASM group

  This group is granted the SYSASM privilege, which provides full administrative privileges for the Oracle ASM instance. For example, the group could be `asmadmin`.

- OSDBA for Oracle ASM group

  This group is granted the SYSDBA privilege on the Oracle ASM instance, which grants access to data stored on Oracle ASM. This group has a subset of the privileges of the OSASM group.

  When you implement separate administrator privileges, choose an OSDBA group for the Oracle ASM instance that is different than the group that you select for the database instance, such as `dba`. For example, the group could be `asmdba`.

- OSOPER for Oracle ASM group

  This group is granted the SYSOPER privilege on the Oracle ASM instance, which provides operations such as startup, shutdown, mount, dismount, and check disk group. This group has a subset of the privileges of the OSASM group. For example, the group could be `asmoper`.

When you implement separate Oracle ASM and database administrator duties, this configuration requires different group and different software owners. Implicitly this implementation requires that the OSASM and OSDBA are different groups. For this configuration, you must create an OSDBA for Oracle ASM group and a database instance must be a member of that group to access the Oracle ASM instance.

In an installation that has been configured as Oracle Grid Infrastructure, the Oracle ASM user, such as `grid`, does not have to be a member of the Oracle Database OSDBA group, such as `dba1` or `dba2`, because the Oracle Clusterware database agent runs as the database owner and can use SYSDBA to connect to the database.

However, in an Oracle Restart configuration, the Oracle ASM user (`grid`) must be a member of the OSDBA group (`dba1`, `dba2`, ...) of every database. This requirement is necessary because Oracle Restart software runs as the Oracle ASM user (`grid`) and this user must be able to start and stop the databases using the `CONNECT / AS SYSDBA` authentication.

Additionally, the owner of the operating system disk devices should be the same as the owner of the Oracle ASM software.

Table 3–2 shows an example of a Linux deployment using separate operating system privilege groups for Oracle ASM users.

Table 3–2   Separated operating system groups and privileges for Oracle ASM users

Role/Software Owner	User	Group/Privilege
Oracle ASM administrator/Oracle Grid Infrastructure home	grid	asmadmin (OSASM)/SYSASM
		asmdba (OSDBA for ASM)/SYSDBA
		asmoper (OSOPER for ASM)/SYSOPER
		dba1, dba2, ... (OSDBA for the databases when in an Oracle Restart configuration)
Database administrator 1/Database home 1	oracle1	asmdba (OSDBA for ASM)/SYSDBA
		oper1 (OSOPER for database 1)/SYSOPER
		dba1 (OSDBA for database 1)/SYSDBA
Database administrator 2/Database home 2	oracle2	asmdba (OSDBA for ASM)/SYSDBA
		oper2 (OSOPER for database 2)/SYSOPER
		dba2 (OSDBA for database 2)/SYSDBA
Operating system disk device owner	grid	asmadmin (OSASM)

The SYSASM Privilege for Administering Oracle ASM

SYSASM is a system privilege that enables the separation of the SYSDBA database administration privilege from the Oracle ASM storage administration privilege. Access to the SYSASM privilege is granted by membership in an operating system group that is designated as the OSASM group. This is similar to SYSDBA and SYSOPER privileges, which are system privileges granted through membership in the groups designated as the OSDBA and OSOPER operating system groups. You can designate one group for all of these system privileges, or you can designate separate groups for each operating system privilege.

You can also grant the SYSASM privilege with password file authentication, as discussed in "Password File Authentication for Oracle ASM" on page 3-30.

To connect locally as SYSASM using password authentication with SQL*Plus, use the following statement:

```
sqlplus SYS AS SYSASM
...
Enter password:
```

To connect remotely as SYSASM using password authentication with SQL*Plus, use the following statement:

```
sqlplus sys@\"myhost.mydomain.com:1521/+ASM\" AS SYSASM
...
Enter password:
```

In the previous example, `+ASM` is the service name of the Oracle ASM instance.

To connect locally as SYSASM to an Oracle ASM instance using operating system authentication with SQL*Plus, use the following statement:

```
sqlplus / AS SYSASM
```

The SYSDBA Privilege for Managing Oracle ASM Components

You can connect as SYSDBA to use SQL*Plus or ASMCMD commands to manage Oracle ASM components associated with the database. When running SQL or ASMCMD operations with the SYSDBA privilege, connect to the database instance rather than the Oracle ASM instance.

Connecting as SYSDBA to the database instance has a limited set of Oracle ASM privileges. For example, you cannot create a disk group when connected with the SYSDBA privilege.

When connected as SYSDBA to the database instance, the Oracle ASM operations are limited to:

- Create and delete files, aliases, directories, and templates

- Examine various Oracle ASM instance views

- Operate on files that were created by this user or only access files to which another user had explicitly granted access

- Granting Oracle ASM File Access Control to other users

Creating Users with the SYSASM Privilege

When you are logged in to an Oracle ASM instance as SYSASM, you can use the combination of CREATE USER and GRANT SQL statements to create a user who has the SYSASM privilege. You also can revoke the SYSASM privilege from a user using the REVOKE command, and you can drop a user from the password file using the DROP USER command.

> **Note:** These commands update the password file for the local Oracle ASM instance only.

The following example describes how to perform these SQL operations for the user identified as *new_user*:

```
REM create a new user, then grant the SYSASM privilege
SQL> CREATE USER new_user IDENTIFIED by new_user_passwd;
SQL> GRANT SYSASM TO new_user;

REM connect the user to the ASM instance
SQL> CONNECT new_user AS SYSASM;
Enter password:

REM revoke the SYSASM privilege, then drop the user
SQL> REVOKE SYSASM FROM new_user;
SQL> DROP USER new_user;
```

When you revoke the last privilege of a user in an Oracle ASM password file, the user is not automatically deleted as is done in the Oracle Database password file. You need to run DROP USER to delete a user with no privileges in an Oracle ASM password file.

For information about creating a user with Oracle ASM command-line utility (ASMCMD), see "orapwusr" on page 12-14. For information about creating a user with

Oracle Enterprise Manager, see "Managing Oracle ASM Users with Oracle Enterprise Manager" on page 9-4.

Operating System Authentication for Oracle ASM

Membership in the operating system group designated as the OSASM group provides operating system authentication for the SYSASM system privilege. OSASM is provided exclusively for Oracle ASM. Initially, only the user that installs ASM is a member of the OSASM group, if you use a separate operating system group for that privilege. However, you can add other users. Members of the OSASM group are authorized to connect using the SYSASM privilege and have full access to Oracle ASM, including administrative access to all disk groups that are managed by that Oracle ASM instance.

On Linux and UNIX systems, dba is the default operating system group designated as OSASM, OSOPER, and OSDBA for Oracle ASM. On Windows systems, ora_dba is the default name designated as OSASM, OSOPER, and OSDBA.

SQL*Plus commands, ASMCMD commands, and ASMCA use operating system authentication.

> **See Also:**
>
> - *Oracle Database Administrator's Guide* for more information about using operating system authentication
>
> - *Oracle Grid Infrastructure Installation Guide* for information about installation of the Oracle Grid Infrastructure

Password File Authentication for Oracle ASM

Password file authentication for Oracle ASM can work both locally and remotely. To enable password file authentication, you must create a password file for Oracle ASM. A password file is also required to enable Oracle Enterprise Manager to connect to Oracle ASM remotely.

If you select the Oracle ASM storage option, then ASMCA creates a password file for Oracle ASM with initial users (SYS and ASMSNMP) when ASMCA configures the Oracle ASM disk groups. To add other users to the password file, you can use the CREATE USER and GRANT commands as described previously in the section titled "About Privileges for Oracle ASM" on page 3-26.

If you configure an Oracle ASM instance without using ASMCA, then you must manually create a password file and grant the SYSASM privilege to user SYS.

SQL*Plus commands and Oracle Enterprise Manager use password file authentication.

> **See Also:**
>
> - *Oracle Database Administrator's Guide* for information about creating and maintaining a password file
>
> - *Oracle Database SQL Language Reference* for information about the CREATE USER and GRANT commands
>
> - *Oracle Database Security Guide* for information about database security
>
> - *Oracle Database Reference* for information about the V$PWFILE_USERS view which lists users who have been granted SYSASM, SYSDBA, and SYSOPER privileges as derived from the password file.

Migrating a Database to Use Oracle ASM

With a new installation of Oracle Database and Oracle ASM, you can initially create your database and select the Oracle ASM storage option. If you have an existing Oracle database that stores database files in the operating system file system or on raw devices, then you can migrate some or all of your data files to Oracle ASM storage.

Oracle provides several methods for migrating your database to Oracle ASM. Using Oracle ASM enables you to realize the benefits of automation and simplicity in managing your database storage. To migrate to Oracle ASM, you can use the methods described in the following sections:

- Using Oracle Enterprise Manager to Migrate Databases to Oracle ASM

- Using Oracle Recovery Manager to Migrate Databases to Oracle ASM

- Best Practices White Papers on Migrating to Oracle ASM

> **Note:** You must upgrade to at least Oracle Database 10*g* before migrating your database to Oracle ASM.

Using Oracle Enterprise Manager to Migrate Databases to Oracle ASM

Oracle Enterprise Manager enables you to perform cold and hot database migration with a GUI. You can access the migration wizard from the Oracle Enterprise Manager Home page under the Change Database heading.

For more information about using Oracle Enterprise Manager to upgrade to Oracle ASM, see Chapter 9, "Administering Oracle ASM with Oracle Enterprise Manager".

Using Oracle Recovery Manager to Migrate Databases to Oracle ASM

You can use Oracle Recovery Manager (RMAN) to manually migrate to Oracle ASM. You can also use RMAN to migrate a single tablespace or data file to Oracle ASM.

For more information, see Chapter 8, "Performing Oracle ASM Data Migration with RMAN".

Best Practices White Papers on Migrating to Oracle ASM

The Oracle Maximum Availability Architecture (MAA) Web site provides excellent best practices technical white papers based on different scenarios, such as:

- Minimal Downtime Migration to Oracle ASM

- Platform Migration using Transportable Tablespaces

- Platform Migration using Transportable Database

> **See Also:** For information about Oracle ASM best practices for migrating to Oracle ASM from environments that do not use Oracle ASM, refer to the documentation at the MAA link on Oracle Technology Network:
>
> http://www.oracle.com/technetwork/database/features/availabi lity/maa-096107.html

4

Administering Oracle ASM Disk Groups

This chapter describes how to administer Oracle Automatic Storage Management (Oracle ASM) disk groups. This information includes how to create, alter, drop, mount, and dismount Oracle ASM disk groups. The database instances that use Oracle ASM can continue operating while you administer disk groups.

The examples in this chapter use SQL statements. These examples assume that SQL*Plus is run from the Oracle grid home where Oracle ASM is installed and the Oracle environmental variables are set to this home. The examples also assume that the Oracle ASM instance is running. This chapter contains the following topics:

- Disk Group Attributes
- Creating Disk Groups
- Altering Disk Groups
- Oracle ASM Disk Discovery
- Managing Capacity in Disk Groups
- Oracle ASM Mirroring and Disk Group Redundancy
- Performance and Scalability Considerations for Disk Groups
- Disk Group Compatibility
- Managing Oracle ASM File Access Control for Disk Groups
- Mounting and Dismounting Disk Groups
- Checking the Internal Consistency of Disk Group Metadata
- Dropping Disk Groups
- Renaming Disks Groups

For information about starting up an Oracle ASM instance, refer to "Starting Up an Oracle ASM Instance" on page 3-13.

For information about administering Oracle ASM disk groups with Oracle Enterprise Manager, refer to Chapter 9, "Administering Oracle ASM with Oracle Enterprise Manager".

For information about administering Oracle ASM disk groups with Oracle ASM Configuration Assistant (ASMCA), refer to Chapter 11, "Oracle ASM Configuration Assistant".

For information about administering Oracle ASM disk groups with ASMCMD, refer to Chapter 12, "Oracle ASM Command-Line Utility".

> **See Also:** The Oracle Cloud Storage page on the Oracle Technology Network Web site at
> http://www.oracle.com/technetwork/database/cloud-storage/index.html for more information about Oracle ASM

Disk Group Attributes

Disk group attributes are parameters that are bound to a disk group, rather than an Oracle ASM instance.

Disk group attributes can be set when a disk group is created or altered, unless otherwise noted in the following list.

- ACCESS_CONTROL.ENABLED

  This attribute can only be set when altering a disk group.

  For information about the `ACCESS_CONTROL.ENABLED` attribute, see "Setting Disk Group Attributes for Oracle ASM File Access Control" on page 4-42.

- ACCESS_CONTROL.UMASK

  This attribute can only be set when altering a disk group.

  For information about the `ACCESS_CONTROL.UMASK` attribute, see "Setting Disk Group Attributes for Oracle ASM File Access Control" on page 4-42.

- AU_SIZE

  This attribute can only be set when creating a disk group.

  For information about allocation unit size and extents, see "Allocation Units" on page 1-6, "Extents" on page 1-7, and "Specifying the Allocation Unit Size" on page 4-9. For an example of the use of the `AU_SIZE` attribute, see Example 4–1, "Creating the DATA disk group" on page 4-6.

- CELL.SMART_SCAN_CAPABLE

  This attribute is only applicable to Oracle Exadata storage.

  > **See Also:** *Oracle Exadata documentation*

- COMPATIBLE.ASM

  For information about the `COMPATIBLE.ASM` attribute, see "COMPATIBLE.ASM" on page 4-35.

- COMPATIBLE.RDBMS

  For information about the `COMPATIBLE.RDBMS` attribute, see "COMPATIBLE.RDBMS" on page 4-35.

- COMPATIBLE.ADVM

  For information about the `COMPATIBLE.ADVM` attribute, see "COMPATIBLE.ADVM" on page 4-35 and "Overview of Oracle ASM Dynamic Volume Manager" on page 5-20.

- CONTENT.TYPE

  The `CONTENT.TYPE` attribute identifies the disk group type: `data`, `recovery`, or `system`. The type value determines the distance to the nearest neighbor disk in the failure group where Oracle ASM mirrors copies of the data. The default value is `data` which specifies a distance of 1 to the nearest neighbor disk. A value of

`recovery` specifies a distance of 3 to the nearest neighbor disk and a value of `system` specifies a distance of 5.

The attribute can be specified when creating or altering a disk group. If `CONTENT.TYPE` attribute is set or changed using `ALTER DISKGROUP`, then the new configuration does not take effect until a disk group rebalance is explicitly run.

The `CONTENT.TYPE` attribute is only valid for disk groups that are set to normal or high redundancy. The `COMPATIBLE.ASM` attribute must be set to `11.2.0.3` or higher to enable the `CONTENT.TYPE` attribute for the disk group.

The attribute can be set with SQL, ASMCMD, or ASMCA. For example:

```
SQL> CREATE DISKGROUP data NORMAL REDUNDANCY
        FAILGROUP controller1 DISK
        '/devices/diska1' NAME diska1,
...
        ATTRIBUTE
        'compatible.asm' = '11.2.0.3',
        'content.type' = 'recovery',
...

SQL> ALTER DISKGROUP data SET ATTRIBUTE 'content.type' = 'data';
```

This attribute is primarily intended for use with Oracle Exadata storage.

> **See Also:** *Oracle Exadata documentation*

- DISK_REPAIR_TIME

  This attribute can only be set when altering a disk group.

  For information about the `DISK_REPAIR_TIME` attribute, see "Oracle ASM Fast Mirror Resync" on page 4-28.

- IDP.BOUNDARY and IDP.TYPE

  These attributes are intended for use with Oracle Exadata storage.

  > **See Also:** *Oracle Exadata documentation*

- SECTOR_SIZE

  This attribute can only be set when creating a disk group.

  For information about the `SECTOR_SIZE` attribute, see "Specifying the Sector Size for Drives" on page 4-9.

- STORAGE.TYPE

  The `STORAGE.TYPE` disk group attribute specifies the type of the disks in the disk group. The possible values are `AXIOM`, `ZFSSA`, and `OTHER`. If the attribute is set to `AXIOM` or `ZFSSA`, then all disks in the disk group must be of that type. If the attribute is set to `OTHER`, then any types of disks can be in the disk group.

  If the `STORAGE.TYPE` disk group attribute is set to `AXIOM` or `ZFSSA`, then functionality for Hybrid Columnar Compression (HCC) can be enabled for Pillar Axiom or ZFS storage.

  > **Note:** ZFS storage must be added through Direct NFS (dNFS) and Pillar Axiom storage must be added as SCSI/Fiber Channel.

To set the `STORAGE.TYPE` attribute, the `COMPATIBLE.ASM` and `COMPATIBLE.RDBMS` disk group attributes must be set to `11.2.0.3` or higher. For information about disk group compatibility attributes, refer to "Disk Group Compatibility" on page 4-33.

The `STORAGE.TYPE` attribute can be set when creating a disk group or when altering a disk group. The attribute cannot be set when clients are connected to the disk group. For example, the attribute cannot be set when an Oracle ADVM volume is enabled on the disk group.

The attribute is not visible in the `V$ASM_ATTRIBUTE` view or with the ASMCMD `lsattr` command until the attribute has been set.

> **See Also:** *Oracle Database Concepts* for more information about Hybrid Columnar Compression. Hybrid Columnar Compression is a feature of certain Oracle storage systems.

In addition to the disk group attributes listed in this section, template attributes are also assigned to a disk group. For information about template attributes, see "Managing Disk Group Templates" on page 7-15.

You can display disk group attributes with the `V$ASM_ATTRIBUTE` view and the ASMCMD `lsattr` command. For an example of the use of the `V$ASM_ATTRIBUTE` view, see Example 6–1 on page 6-2. For information about the `lsattr` command, see "lsattr" on page 12-36.

Creating Disk Groups

This section contains information about creating disk groups. You can use the `CREATE DISKGROUP` SQL statement to create a disk group.

This section contains the following topics:

- Using the CREATE DISKGROUP SQL Statement
- Example: Creating a Disk Group
- Creating Disk Groups for a New Oracle Installation
- Specifying the Allocation Unit Size
- Specifying the Sector Size for Drives
- Oracle Cluster Registry and Voting Files in Oracle ASM Disk Groups

Using the CREATE DISKGROUP SQL Statement

The `CREATE DISKGROUP` SQL statement is used to create disk groups. When creating a disk group, you:

- Assign a unique name to the disk group.

  The specified disk group name is not case sensitive and is always converted to uppercase when stored internally.

  > **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.

- Specify the redundancy level of the disk group.

  For Oracle ASM to mirror files, specify the redundancy level as `NORMAL REDUNDANCY` (2-way mirroring by default for most file types) or `HIGH REDUNDANCY` (3-way mirroring for all files). Specify `EXTERNAL REDUNDANCY` if you do not want mirroring by Oracle ASM. For example, you might choose `EXTERNAL REDUNDANCY` to use storage array protection features.

  After a disk group is created, you cannot alter the redundancy level of the disk group. To change the redundancy level, you must create another disk group with the appropriate redundancy and then move the files to the new disk group.

  Oracle recommends that you create failure groups of equal size to maintain space balance and even distribution of mirror data.

  For more information about redundancy levels, refer to "Mirroring, Redundancy, and Failure Group Options" on page 4-24.

- Specify the disks that are to be formatted as Oracle ASM disks belonging to the disk group.

  The disks can be specified using operating system dependent wildcard characters in search strings that Oracle ASM then uses to find the disks. You can specify names for the disks with the `NAME` clause or use the system-generated names.

- Optionally specify the disks as belonging to specific failure groups.

  For information about failure groups, refer to "Understanding Oracle ASM Concepts" on page 1-2 and "Mirroring, Redundancy, and Failure Group Options" on page 4-24.

- Optionally specify the type of failure group.

  For information about `QUORUM` and `REGULAR` failure groups, refer to "Oracle Cluster Registry and Voting Files in Oracle ASM Disk Groups" on page 4-11.

- Optionally specify disk group attributes, such as software compatibility or allocation unit size.

Oracle ASM programmatically determines the size of each disk. If for some reason this is not possible, or to restrict the amount of space used on a disk, you can specify a `SIZE` clause for each disk. Oracle ASM creates operating system–independent names for the disks in a disk group that you can use to reference the disks in other SQL statements. Optionally, you can provide your own name for a disk using the `NAME` clause. Disk names are available in the `V$ASM_DISK` view.

Note: A disk cannot belong to multiple disk groups.

The Oracle ASM instance ensures that any disk in a newly created disk group is addressable and is not currently a member of another disk group. You must use `FORCE` only when adding a disk that was dropped with `FORCE`. If a disk is dropped with `NOFORCE`, then you can add it with `NOFORCE`. For example, a disk might have failed and was dropped from its disk group. After the disk is repaired, it is no longer part of any disk group, but Oracle ASM still recognizes that the disk had been a member of a disk group. You must use the `FORCE` flag to include the disk in a new disk group. In addition, the disk must be addressable, and the original disk group must not be mounted. Otherwise, the operation fails.

> **Note:** Use caution when using the `FORCE` option to add a previously used disk to a disk group; you might cause another disk group to become unusable.

The `CREATE DISKGROUP` statement mounts the disk group for the first time, and adds the disk group name to the `ASM_DISKGROUPS` initialization parameter if a server parameter file is being used. If a text initialization parameter file is being used and you want the disk group to be automatically mounted at instance startup, then you must remember to add the disk group name to the `ASM_DISKGROUPS` initialization parameter before you shut down and restart the Oracle ASM instance. You can also create disk groups with Oracle Enterprise Manager. Refer to "Creating Disk Groups" on page 9-5.

> **See Also:** The `CREATE DISKGROUP` SQL statement in the *Oracle Database SQL Language Reference*

Example: Creating a Disk Group

The following examples assume that the `ASM_DISKSTRING` initialization parameter is set to the `'/devices/*'` string. Oracle ASM disk discovery identifies disks in the `/devices` directory, including the following disks:

Controller 1:

```
/devices/diska1
/devices/diska2
/devices/diska3
/devices/diska4
```

Controller 2:
```
/devices/diskb1
/devices/diskb2
/devices/diskb3
/devices/diskb4
```

The SQL statement in Example 4–1 creates a disk group named `data` with normal redundancy consisting of two failure groups `controller1` or `controller2` with four disks in each failure group. The `data` disk group is typically used to store database data files.

Example 4–1   Creating the DATA disk group

```
CREATE DISKGROUP data NORMAL REDUNDANCY
  FAILGROUP controller1 DISK
    '/devices/diska1' NAME diska1,
    '/devices/diska2' NAME diska2,
    '/devices/diska3' NAME diska3,
    '/devices/diska4' NAME diska4
  FAILGROUP controller2 DISK
    '/devices/diskb1' NAME diskb1,
    '/devices/diskb2' NAME diskb2,
    '/devices/diskb3' NAME diskb3,
    '/devices/diskb4' NAME diskb4
  ATTRIBUTE 'au_size'='4M',
    'compatible.asm' = '11.2',
    'compatible.rdbms' = '11.2',
    'compatible.advm' = '11.2';
```

In Example 4–1, the `NAME` clauses enable you to explicitly assign names to the disks rather than the default system-generated names. The system-generated names are in the form *diskgroup_nnnn*, where *nnnn* is the disk number for the disk in the disk group. For ASMLib disks, the disk name defaults to the ASMLib name that is the user label of the disk; for example, `mydisk` is the default Oracle ASM disk name for `ORCL:mydisk`.

When creating the disk group in Example 4–1, the values of following disk group attributes were explicitly set:

- `AU_SIZE`

  Specifies the size of the allocation unit for the disk group. For information about allocation unit size and extents, see "Extents" on page 1-7.

  You can view the value of the `AU_SIZE` disk group attribute in the `ALLOCATION_UNIT_SIZE` column of the `V$ASM_DISKGROUP` view.

- `COMPATIBLE.ASM`

  Determines the minimum software version for any Oracle ASM instance that uses a disk group. For information about the `COMPATIBLE.ASM` attribute, see "COMPATIBLE.ASM" on page 4-35.

- `COMPATIBLE.RDBMS`

  Determines the minimum software version for any database instance that uses a disk group. For information about the `COMPATIBLE.RDBMS` attribute, see "COMPATIBLE.RDBMS" on page 4-35.

- `COMPATIBLE.ADVM`

  Determines whether the disk group can contain Oracle ASM volumes. For information about the `COMPATIBLE.ADVM` attribute, see "COMPATIBLE.ADVM" on page 4-35.

In Example 4–2, the `fra` disk group (typically created for the fast recovery area) is created with the default disk group attribute values. Names are not specified for the Oracle ASM disks and failure groups are not explicitly specified. This example assumes that `diskc1` through `diskc9` are present in the `/devices` directory.

Example 4–2   Creating the FRA disk group

```
CREATE DISKGROUP fra NORMAL REDUNDANCY
  DISK '/devices/diskc*';
```

> **See Also:** For information about using ASMLib when creating disk groups, refer to the Oracle ASMLib page on the Oracle Technology Network Web site at
> http://www.oracle.com/technetwork/topics/linux/asmlib/index-101839.html

Creating Disk Groups for a New Oracle Installation

This section describes the basic procedure to follow when creating disk groups during a new installation of Oracle Restart and Oracle Database. This information also applies to an Oracle Grid Infrastructure installation.

The procedure assumes that the `data` disk group is used for the storage of the database data files and the `fra` disk group is used for storage of the fast recovery area files. Detailed information about installation with Oracle Universal Installer (OUI) and

database creation with Database Configuration Assistant (DBCA) is available in the installation guides for your specific operating system.

1. Install Oracle Restart with OUI, following the screen prompts.

   During the installation, create the `data` disk group for storing database files such as the data and control files.

   This OUI process is similar to creating a disk group with Oracle ASM Configuration Assistant (ASMCA). For information about using ASMCA to create disk groups, see "Managing Disk Groups with Oracle ASM Configuration Assistant" on page 11-6.

   Note that the `data` disk group is the disk group used for storing Oracle Cluster Registry (OCR) and voting files in an Oracle Grid Infrastructure installation. See "Oracle Cluster Registry and Voting Files in Oracle ASM Disk Groups" on page 4-11

2. After Oracle Restart is installed, use ASMCA to create the `fra` disk group for storing the fast recovery area files.

   Create the `fra` disk group to hold the fast recovery area files.

   At this time, you can also update the `data` disk group if necessary. For information about using ASMCA to create or alter disk groups, see "Managing Disk Groups with Oracle ASM Configuration Assistant" on page 11-6.

   You can also create the `fra` disk group with SQL*Plus or ASMCMD commands run from the Oracle Restart home. For information, see "Using the CREATE DISKGROUP SQL Statement" on page 4-4 and "mkdg" on page 12-43.

   > **See Also:**
   >
   > - *Oracle Database Backup and Recovery User's Guide* for information about configuring the fast recovery area
   > - *Oracle Database Administrator's Guide* for information about specifying a fast recovery area

3. Install the Oracle Database software with OUI, following the screen prompts.

4. After the database software has been installed, run DBCA to create a database, following the screen prompts.

   During the creation of the database, make the appropriate selections to use Oracle ASM for storage of data files and fast recovery area files. When prompted:

   - Store database data files in the `data` disk group
   - Store fast recovery area files in the `fra` disk group

   > **See Also:**
   >
   > - *Oracle Grid Infrastructure Installation Guide* for your operating system for detailed information about installation of Oracle Grid Infrastructure and Oracle Restart
   > - *Oracle Database Installation Guide* for your operating system for detailed information about installation of an Oracle Database

Specifying the Allocation Unit Size

Oracle recommends that the allocation unit (AU) size for a disk group be set to 4 megabytes (MB). In addition to this AU size recommendation, the operating system (OS) I/O size should be set to the largest possible size.

Some benefits of a 4 MB allocation unit are:

- Increased I/O through the I/O subsystem if the I/O size is increased to the AU size.

- Reduced SGA size to manage the extent maps in the database instance.

- Faster datafile initialization if the I/O size is increased to the AU size.

- Increased file size limits.

- Reduced database open time.

The allocation unit size is specified with the disk group attribute `AU_SIZE`. The AU size cannot be altered after a disk group is created. Example 4–1 on page 4-6 shows how the `AU_SIZE` is specified with the `CREATE DISKGROUP` SQL statement.

Specifying the Sector Size for Drives

You can use the optional `SECTOR_SIZE` disk group attribute with the `CREATE DISKGROUP` SQL statement to specify disks with the sector size set to the value of `SECTOR_SIZE` for the disk group. Oracle ASM provides support for 4 KB sector disk drives without negatively affecting performance. The `SECTOR_SIZE` disk group attribute can be set only during disk group creation.

The values for `SECTOR_SIZE` can be set to 512, 4096, or 4K if the disks support those values. The default value is platform dependent. The `COMPATIBLE.ASM` and `COMPATIBLE.RDBMS` disk group attributes must be set to 11.2 or higher to set the sector size to a value other than the default value.

> **Note:** Oracle Automatic Storage Management Cluster File System (Oracle ACFS) does not support 4 KB sector drives. There is a performance penalty for Oracle ACFS when using 4 KB sector disk drives in 512 sector emulation mode.

The following validations apply to the sector size disk group attribute.

- Oracle ASM prevents disks of different sector sizes from being added to the same disk group. This validation occurs during `CREATE DISKGROUP`, `ALTER DISKGROUP ADD DISK`, and `ALTER DISKGROUP MOUNT` operations.

- If the `SECTOR_SIZE` attribute is explicitly specified when creating a disk group, then Oracle ASM attempts to verify that all disks discovered through disk search strings have a sector size equal to the specified value. If one or more disks were found to have a sector size different from the specified value, or if Oracle ASM was not able to verify a disk sector size, then the create operation fails.

  Oracle ASM also attempts to verify disk sector size during the mount operation and the operation fails if one or more disks have a sector size different than the value of the `SECTOR_SIZE` attribute.

- If the `SECTOR_SIZE` attribute is not specified when creating a disk group and Oracle ASM can verify that all discovered disks have the same sector value, then that

value is assumed for the disk group sector size that is created. If the disks have different sector sizes, the create operation fails.

- When new disks are added to an existing disk group using the ALTER DISKGROUP .. ADD DISK SQL statement, you must ensure that the new disks to be added have the same value as the SECTOR_SIZE disk group attribute. If the new disks have different sector sizes, the alter operation fails.

- You can determine the sector size value that has either been assumed or explicitly set for a successful disk group creation by querying the V$ASM_ATTRIBUTE view or run the ASMCMD lsattr command. You can also query the SECTOR_SIZE column in the V$ASM_DISKGROUP view.

```
SQL> SELECT name, value FROM V$ASM_ATTRIBUTE
     WHERE name = 'sector_size' AND group_number = 1;
NAME                       VALUE
-------------------------- -----------------------
sector_size                512

SQL> SELECT group_number, sector_size FROM V$ASM_DISKGROUP
     WHERE group_number = 1;

GROUP_NUMBER SECTOR_SIZE
------------ -----------
           1         512
```

- Not all disks support all of the possible SECTOR_SIZE values. The sector size attribute setting must be compatible with the physical hardware.

As shown in Example 4–3, you can use the SECTOR_SIZE attribute with the CREATE DISKGROUP SQL statement to specify the sector size of the disk drive on which the Oracle ASM disk group is located.

Example 4–3   Creating a disk group of 4K sector size

```
CREATE DISKGROUP data NORMAL REDUNDANCY
FAILGROUP controller1 DISK
'/devices/diska1',
'/devices/diska2',
'/devices/diska3',
'/devices/diska4'
FAILGROUP controller2 DISK
'/devices/diskb1',
'/devices/diskb2',
'/devices/diskb3',
'/devices/diskb4'
ATTRIBUTE 'compatible.asm' = '11.2', 'compatible.rdbms' = '11.2',
          'sector_size'='4096';
```

> **See Also:**
>
> - *Oracle Database SQL Language Reference* for information about the disk group attributes and the CREATE DISKGROUP SQL statement
>
> - *Oracle Database SQL Language Reference* for information about the BLOCKSIZE keyword
>
> - *Oracle Database Reference* for details about the V$ASM_ATTRIBUTE and V$ASM_DISKGROUP views

Oracle Cluster Registry and Voting Files in Oracle ASM Disk Groups

You can store Oracle Cluster Registry (OCR) and voting files in Oracle ASM disk groups. The voting files and OCR are two important components of Oracle Clusterware.

Voting files manage information about node membership. OCR is a system that manages cluster and Oracle Real Application Clusters (Oracle RAC) database configuration information. A quorum failure group is a special type of failure group and disks in these failure groups do not contain user data. A quorum failure group is not considered when determining redundancy requirements in respect to storing user data. However, a quorum failure group counts when mounting a disk group. For information about failure groups, see "Oracle ASM Failure Groups" on page 4-26.

You can manage and monitor OCR and voting files with the following:

- The CRSCTL and `ocrconfig` command-line tools

  The CRSCTL and `ocrconfig` commands enable the placement of OCR storage and Cluster Synchronization Services (CSS) voting files inside the disk groups managed by Oracle ASM.

- `CREATE/ALTER DISKGROUP` SQL Statements

  The SQL keywords `QUORUM` and `REGULAR` enable the specification of disk and failure groups when creating or altering disk groups.

  See Example 4–4 on page 4-12.

- `V$ASM` views

  The `FAILGROUP_TYPE` column in both the `V$ASM_DISK` and `V$ASM_DISK_STAT` views specifies failure group type. The value for this column is `REGULAR` for regular failure groups and `QUORUM` for quorum failure groups.

  The `VOTING_FILE` column in both the `V$ASM_DISK` and `V$ASM_DISK_STAT` views specifies whether a disk contains a voting file. The value for this column is either `Y` if the disk contains a voting file or `N` if not.

  Note that the value of `USABLE_FILE_MB` in `V$ASM_DISKGROUP` and `V$ASM_DISKGROUP_STAT` does not consider any free space that is present in `QUORUM` disks because that space is not available for client data files.

  See "Views Containing Oracle ASM Disk Group Information" on page 6-1.

The `QUORUM` and `REGULAR` keywords provide an additional qualifier for failure group or disk specifications when creating or altering a disk group. `QUORUM` disks (or disks in `QUORUM` failure groups) cannot have client data files, whereas `REGULAR` disks (or disks in non-quorum failure groups) have no such restriction.

You can use these keywords before the keyword `FAILGROUP` if a failure group is being explicitly specified. If the failure group is implicitly implied, you can use these keywords (`QUORUM`/`REGULAR`) before the keyword `DISK`. When failure groups are explicitly specified, it is an error to specify these keywords (`QUORUM`/`REGULAR`) immediately before the keyword `DISK`. `REGULAR` is the default failure group type.

When performing operations on existing disks or failure groups, the qualifier specified in the SQL must match the qualifier that was specified when the disks or failure groups were added to the disk group.

Example 4–4 shows the creation of a disk group with a `QUORUM` failure group. For Oracle Clusterware files a minimum of three disk devices or three failure groups is required with a normal redundancy disk group. A `QUORUM` failure group is not

considered when determining redundancy requirements in respect to storing user data.

The COMPATIBLE.ASM disk group compatibility attribute must be set to 11.2 or greater to store OCR or voting files in a disk group.

Example 4–4   Using the QUORUM keyword

```
CREATE DISKGROUP ocr_data NORMAL REDUNDANCY
   FAILGROUP fg1 DISK '/devices/diskg1'
   FAILGROUP fg2 DISK '/devices/diskg2'
   QUORUM FAILGROUP fg3 DISK '/devices/diskg3'
   ATTRIBUTE 'compatible.asm' = '11.2.0.0.0';
```

> **See Also:**
>
> - *Oracle Grid Infrastructure Installation Guide* for information about configuring voting files and Oracle Cluster Registry (OCR)
>
> - *Oracle Clusterware Administration and Deployment Guide* for information about administering voting files and Oracle Cluster Registry (OCR) using CRSCTL and ocrconfig commands
>
> - *Oracle Database SQL Language Reference* for information about the CREATE DISKGROUP and ALTER DISKGROUP SQL statements
>
> - *Oracle Database Reference* for details about the V$ASM* dynamic performance views

Altering Disk Groups

You can use the ALTER DISKGROUP SQL statement to alter a disk group configuration. You can add, resize, or drop disks while the database remains online. Whenever possible, multiple operations in a single ALTER DISKGROUP statement are recommended. Grouping operations in a single ALTER DISKGROUP statement can reduce rebalancing operations.

Oracle ASM automatically rebalances when the configuration of a disk group changes. By default, the ALTER DISKGROUP statement does not wait until the operation is complete before returning. Query the V$ASM_OPERATION view to monitor the status of this operation.

Use the REBALANCE WAIT clause to cause the ALTER DISKGROUP statement processing to wait until the rebalance operation is complete before returning. This is especially useful in scripts. The statement also accepts a REBALANCE NOWAIT clause that invokes the default behavior of conducting the rebalance operation asynchronously in the background.

You can interrupt a rebalance running in wait mode by typing CTRL-C on most platforms. This causes the statement to return immediately with the message ORA-01013: user requested cancel of current operation, and then to continue the operation asynchronously. Typing CTRL-C does not cancel the rebalance operation or any disk add, drop, or resize operations.

To control the speed and resource consumption of the rebalance operation, you can include the REBALANCE POWER clause in statements that add, drop, or resize disks. Refer to "Manually Rebalancing Disk Groups" on page 4-18 for more information about this clause.

This section contains the following topics:

- Managing Volumes in a Disk Group

- Adding Disks to a Disk Group

- Dropping Disks from Disk Groups

- Intelligent Data Placement

- Resizing Disks in Disk Groups

- Undropping Disks in Disk Groups

- Manually Rebalancing Disk Groups

- Tuning Rebalance Operations

> **See Also:** The ALTER DISKGROUP SQL statement in the *Oracle Database SQL Language Reference*

Managing Volumes in a Disk Group

You can create an Oracle ASM Dynamic Volume Manager (Oracle ADVM) volume in a disk group. The volume device associated with the dynamic volume can then be used to host an Oracle ACFS file system.

The compatibility parameters COMPATIBLE.ASM and COMPATIBLE.ADVM must be set to 11.2 or higher for the disk group. See "Disk Group Compatibility Attributes" on page 4-34.

The ALTER DISKGROUP VOLUME SQL statements enable you to manage Oracle ADVM volumes, including the functionality to add, modify, resize, disable, enable, and drop volumes. The following are examples of the ALTER DISKGROUP VOLUME statement.

Example 4–5   Managing volumes with ALTER DISKGROUP VOLUME statements

```
SQL> ALTER DISKGROUP data ADD VOLUME volume1 SIZE 10G;
Diskgroup altered.

SQL> ALTER DISKGROUP data RESIZE VOLUME volume1 SIZE 15G;
Diskgroup altered.

SQL> ALTER DISKGROUP data DISABLE VOLUME volume1;
Diskgroup altered.

SQL> ALTER DISKGROUP data ENABLE VOLUME volume1;
Diskgroup altered.

SQL> ALTER DISKGROUP ALL DISABLE VOLUME ALL;
Diskgroup altered.

SQL> ALTER DISKGROUP data DROP VOLUME volume1;
Diskgroup altered.
```

If the volume is hosting an Oracle ACFS file system, then you cannot resize that volume with the SQL ALTER DISKGROUP statement. Instead you must use the acfsutil size command. For information, see "acfsutil size" on page 13-106.

For information about Oracle ADVM, see "Overview of Oracle ASM Dynamic Volume Manager" on page 5-20. For information about managing Oracle ADVM volumes with ASMCMD, see "ASMCMD Volume Management Commands" on page 12-61. For information about managing Oracle ADVM volumes with ASMCA, see "Managing Oracle ADVM Volumes with Oracle ASM Configuration Assistant" on page 11-9. For

information about managing Oracle ADVM volumes with Oracle Enterprise Manager, see "Managing Oracle ACFS with Oracle Enterprise Manager" on page 10-1.

> **See Also:** The `ALTER DISKGROUP` SQL statement in the *Oracle Database SQL Language Reference*

Adding Disks to a Disk Group

You can use the `ADD` clause of the `ALTER DISKGROUP` statement to add a disk or a failure group to a disk group. You can use the same syntax with the `ALTER DISKGROUP` statement that you use to add a disk or failure group with the `CREATE DISKGROUP` statement. For an example of the `CREATE DISKGROUP` SQL statement, refer to Example 4–1 on page 4-6. After you add new disks, the new disks gradually begin to accommodate their share of the workload as rebalancing progresses.

Oracle ASM behavior when adding disks to a disk group is best illustrated through"Adding Disks to a Disk Group with SQL Statements" on page 4-14. You can also add disks to a disk group with Oracle Enterprise Manager, described in "Adding Disks to Disk Groups" on page 9-7.

Adding Disks to a Disk Group with SQL Statements

The statements presented in this example demonstrate the interactions of disk discovery with the `ADD DISK` operation.

Assume that disk discovery identifies the following disks in directory `/devices`:

```
/devices/diska1 -- member of data1
/devices/diska2 -- member of data1
/devices/diska3 -- member of data1
/devices/diska4 -- member of data1
/devices/diska5 -- candidate disk
/devices/diska6 -- candidate disk
/devices/diska7 -- candidate disk
/devices/diska8 -- candidate disk

/devices/diskb1 -- member of data1
/devices/diskb2 -- member of data1
/devices/diskb3 -- member of data1
/devices/diskb4 -- member of data2

/devices/diskc1 -- member of data2
/devices/diskc2 -- member of data2
/devices/diskc3 -- member of data3
/devices/diskc4 -- candidate disk

/devices/diskd1 -- candidate disk
/devices/diskd2 -- candidate disk
/devices/diskd3 -- candidate disk
/devices/diskd4 -- candidate disk
/devices/diskd5 -- candidate disk
/devices/diskd6 -- candidate disk
/devices/diskd7 -- candidate disk
/devices/diskd8 -- candidate disk
```

You can query the `V$ASM_DISK` view to display the status of Oracle ASM disks. See "Views Containing Oracle ASM Disk Group Information" on page 6-1.

The following statement would fail because `/devices/diska1` - `/devices/diska4` already belong to the disk group `data1`.

```
ALTER DISKGROUP data1 ADD DISK
     '/devices/diska*';
```

The following statement successfully adds disks `/devices/diska5` through `/devices/diska8` to `data1`. Because no `FAILGROUP` clauses are included in the `ALTER DISKGROUP` statement, each disk is assigned to its own failure group. The `NAME` clauses assign names to the disks, otherwise they would have been assigned system-generated names.

```
ALTER DISKGROUP data1 ADD DISK
     '/devices/diska5' NAME diska5,
     '/devices/diska6' NAME diska6,
     '/devices/diska7' NAME diska7,
     '/devices/diska8' NAME diska8;
```

The following statement would fail because the search string matches disks that are contained in other disk groups. Specifically, `/devices/diska4` belongs to disk group `data1` and `/devices/diskb4` belongs to disk group `data2`.

```
ALTER DISKGROUP data1 ADD DISK
     '/devices/disk*4';
```

The following statement would successfully add `/devices/diskd1` through `/devices/diskd8` to disk group `data1`. This statement runs with a rebalance power of 5, and does not return until the rebalance operation is complete.

```
ALTER DISKGROUP data1 ADD DISK
     '/devices/diskd*'
      REBALANCE POWER 5 WAIT;
```

If `/devices/diskc3` was previously a member of a disk group that no longer exists, then you could use the `FORCE` option to add the disk as a member of another disk group. For example, the following use of the `FORCE` clause enables `/devices/diskc3` to be added to `data2`, even though it is a current member of `data3`. For this statement to succeed, `data3` cannot be mounted.

```
ALTER DISKGROUP data2 ADD DISK
     '/devices/diskc3' FORCE;
```

Dropping Disks from Disk Groups

To drop disks from a disk group, use the `DROP DISK` clause of the `ALTER DISKGROUP` statement. You can also drop all of the disks in specified failure groups using the `DROP DISKS IN FAILGROUP` clause.

When a disk is dropped, the disk group is rebalanced by moving all of the file extents from the dropped disk to other disks in the disk group. A drop disk operation might fail if not enough space is available on the other disks. The best approach is to perform both the add and drop operation with the same `ALTER DISKGROUP` statement. This has the benefit of rebalancing data extents only one time and ensuring that there is enough space for the rebalance operation to succeed.

> **Caution:** The `ALTER DISKGROUP...DROP DISK` SQL statement returns
> to SQL prompt before the drop and rebalance operations are complete.
> Do not reuse, remove, or disconnect the dropped disk until the
> `HEADER_STATUS` column for this disk in the `V$ASM_DISK` view changes
> to `FORMER`. You can query the `V$ASM_OPERATION` view to determine the
> amount of time remaining for the drop/rebalance operation to
> complete. For more information, refer to the *Oracle Database SQL
> Language Reference* and the *Oracle Database Reference*.

If you specify the `FORCE` clause for the drop operation, the disk is dropped even if
Oracle ASM cannot read or write to the disk. You cannot use the `FORCE` flag when
dropping a disk from an external redundancy disk group.

> **Caution:** A `DROP FORCE` operation leaves data at reduced redundancy
> until the subsequent rebalance operation completes. This increases
> your exposure to data loss if there is a subsequent disk failure during
> rebalancing. Use `DROP FORCE` with caution.

You can also drop disks from a disk group with Oracle Enterprise Manager. See
"Dropping Disks from Disk Groups" on page 9-11.

The statements in Example 4–6 demonstrate how to drop disks from the disk group
`data1` described in "Adding Disks to a Disk Group with SQL Statements" on page 4-14.
The first example drops `diska5` from disk group `data1`. The second example drops
`diska5` from disk group `data1`, and also illustrates how multiple actions are possible
with one `ALTER DISKGROUP` statement.

Example 4–6   Dropping disks from disk groups

```
ALTER DISKGROUP data1 DROP DISK diska5;

ALTER DISKGROUP data1 DROP DISK diska5
    ADD FAILGROUP failgrp1 DISK '/devices/diska9' NAME diska9;
```

Intelligent Data Placement

Intelligent Data Placement enables you to specify disk regions on Oracle ASM disks
for best performance. Using the disk region settings, you can ensure that frequently
accessed data is placed on the outermost (hot) tracks which have greater speed and
higher bandwidth. In addition, files with similar access patterns are located physically
close, reducing latency. Intelligent Data Placement also enables the placement of
primary and mirror extents into different hot or cold regions.

Intelligent Data Placement settings can be specified for a file or in disk group
templates. The disk region settings can be modified after the disk group has been
created. The disk region setting can improve I/O performance by placing more
frequently accessed data in regions furthest from the spindle, while reducing your cost
by increasing the usable space on a disk.

Intelligent Data Placement works best for the following:

- Databases with data files that are accessed at different rates. A database that
  accesses all data files in the same way is unlikely to benefit from Intelligent Data
  Placement.

- Disk groups that are more than 25% full. If the disk group is only 25% full, the management overhead is unlikely to be worth any benefit.

- Disks that have better performance at the beginning of the media relative to the end. Because Intelligent Data Placement leverages the geometry of the disk, it is well suited to JBOD (just a bunch of disks). In contrast, a storage array with LUNs composed of concatenated volumes masks the geometry from Oracle ASM.

The `COMPATIBLE.ASM` and `COMPATIBLE.RDBMS` disk group attributes must be set to 11.2 or higher to use Intelligent Data Placement.

Intelligent Data Placement can be managed with the `ALTER DISKGROUP ADD` or `MODIFY TEMPLATE` SQL statements and the `ALTER DISKGROUP MODIFY FILE` SQL statement.

- The `ALTER DISKGROUP TEMPLATE` SQL statement includes a disk region clause for setting *hot*/*mirrorhot* or *cold*/*mirrorcold* regions in a template:

```
ALTER DISKGROUP data ADD TEMPLATE datafile_hot
  ATTRIBUTE (
    HOT
    MIRRORHOT);
```

- The `ALTER DISKGROUP ... MODIFY FILE` SQL statement that sets disk region attributes for *hot*/*mirrorhot* or *cold*/*mirrorcold* regions:

```
ALTER DISKGROUP data MODIFY FILE '+data/orcl/datafile/users.259.679156903'
  ATTRIBUTE (
    HOT
    MIRRORHOT);
```

When you modify the disk region settings for a file, this action applies to new extensions of the file, but existing file contents are not affected until a rebalance operation. To apply the new Intelligent Data Placement policy for existing file contents, you can manually initiate a rebalance. A rebalance operation uses the last specified policy for the file extents. For information on the rebalance operation, see "Manually Rebalancing Disk Groups" on page 4-18.

Oracle ASM Configuration Assistant (ASMCA) supports Intelligent Data Placement with template creation during disk group alterations. See "Managing Disk Groups with Oracle ASM Configuration Assistant" on page 11-6.

Oracle Enterprise Manager supports Intelligent Data Placement from the **Templates** page launched from the disk group page. See "Managing Disk Group Templates with Oracle Enterprise Manager" on page 9-20.

To display information about Intelligent Data Placement regions, see "Viewing Disk Region Information" on page 6-5 and "iostat" on page 12-34.

Resizing Disks in Disk Groups

The `RESIZE` clause of `ALTER DISKGROUP` enables you to perform the following operations:

- Resize all disks in the disk group

- Resize specific disks

- Resize all of the disks in a specified failure group

If you do not specify a new size in the `SIZE` clause, then Oracle ASM uses the size of the disk as returned by the operating system. The new size is written to the Oracle ASM disk header and if the size of the disk is increasing, then the new space is immediately available for allocation. If the size is decreasing, rebalancing must

relocate file extents beyond the new size limit to available space below the limit. If the rebalance operation can successfully relocate all extents, then the new size is made permanent, otherwise the rebalance fails.

The following example resizes all of the disks in failure group `failgrp1` of disk group `data1`. If the new size is greater than disk capacity, the statement fails.

Example 4–7   Resizing disks in disk groups

```
ALTER DISKGROUP data1
    RESIZE DISKS IN FAILGROUP failgrp1 SIZE 100G;
```

Undropping Disks in Disk Groups

The `UNDROP DISKS` clause of the `ALTER DISKGROUP` statement enables you to cancel all pending drops of disks within disk groups. If a drop disk operation has completed, then this statement cannot be used to restore it. This statement cannot be used to restore disks that are being dropped as the result of a `DROP DISKGROUP` statement, or for disks that are being dropped using the `FORCE` clause.

The following example cancels the dropping of disks from disk group `data1`:

Example 4–8   Undropping disks in disk groups

```
ALTER DISKGROUP data1 UNDROP DISKS;
```

Manually Rebalancing Disk Groups

You can manually rebalance the files in a disk group using the `REBALANCE` clause of the `ALTER DISKGROUP` statement. This would normally not be required, because Oracle ASM automatically rebalances disk groups when their configuration changes. You might want to do a manual rebalance operation to control the speed of what would otherwise be an automatic rebalance operation.

The `POWER` clause of the `ALTER DISKGROUP ... REBALANCE` statement specifies the degree of parallelism, and thus the speed of the rebalance operation. It can be set to a minimum value of 0 which halts a rebalancing operation until the statement is either implicitly or explicitly re-run. A higher value increases the speed of the rebalance operation.

The default rebalance power is set by the `ASM_POWER_LIMIT` initialization parameter. The range of values for the `POWER` clause is the same for the `ASM_POWER_LIMIT` initialization parameter. For information about the `ASM_POWER_LIMIT` initialization parameter, refer to "ASM_POWER_LIMIT" on page 3-8.

The power level of an ongoing rebalance operation can be changed by entering the rebalance statement with a new level.

The `ALTER DISKGROUP...REBALANCE` command by default returns immediately so that you can run other commands while the rebalance operation takes place asynchronously in the background. You can query the `V$ASM_OPERATION` view for the status of the rebalance operation.

To cause the `ALTER DISKGROUP...REBALANCE` command to wait until the rebalance operation is complete before returning, add the `WAIT` keyword to the `REBALANCE` clause. The wait functionality is especially useful in scripts. The command also accepts a `NOWAIT` keyword, which invokes the default behavior of conducting the rebalance operation asynchronously. You can interrupt a rebalance running in wait mode by typing `CTRL-C` on most platforms. This causes the command to return immediately

with the message `ORA-01013: user requested cancel of current operation`, and then continue the rebalance operation asynchronously.

Additional rules for the rebalance operation include the following:

- An ongoing rebalance command is restarted if the storage configuration changes either when you alter the configuration, or if the configuration changes due to a failure or an outage. Furthermore, if the new rebalance fails because of a user error, then a manual rebalance may be required.

- The `ALTER DISKGROUP...REBALANCE` statement runs on a single node even if you are using Oracle Real Application Clusters (Oracle RAC).

- Oracle ASM can perform one disk group rebalance at a time on a given instance. If you have initiated multiple rebalances on different disk groups on a single node, then Oracle processes these operations in parallel on additional nodes if available; otherwise the rebalances are performed serially on the single node. You can explicitly initiate rebalances on different disk groups on different nodes in parallel.

- Rebalancing continues across a failure of the Oracle ASM instance performing the rebalance.

- The `REBALANCE` clause (with its associated `POWER` and `WAIT/NOWAIT` keywords) can also be used in `ALTER DISKGROUP` commands that add, drop, or resize disks.

> **Note:** Oracle restarts the processing of an ongoing rebalance operation if the storage configuration changes. If the next rebalance operation fails because of a user error, then a manual rebalance may be required.

The following example manually rebalances the disk group `data2`. The command does not return until the rebalance operation is complete.

Example 4–9   Manually rebalancing a disk group

```
ALTER DISKGROUP data2 REBALANCE POWER 5 WAIT;
```

For more information about rebalancing operations, refer to "Tuning Rebalance Operations" on page 4-19.

Tuning Rebalance Operations

If the `POWER` clause is not specified in an `ALTER DISKGROUP` statement, or when rebalance is implicitly run by adding or dropping a disk, then the rebalance power defaults to the value of the `ASM_POWER_LIMIT` initialization parameter. You can adjust the value of this parameter dynamically.

The higher the power limit, the more quickly a rebalance operation can complete. Rebalancing takes longer with lower power values, but consumes fewer processing and I/O resources which are shared by other applications, such as the database.

Oracle ASM tries to keep a rebalance I/O for each unit of power. Each I/O requires PGA memory for the extent involved in the relocation.

The default value of `1` minimizes disruption to other applications. The appropriate value is dependent on your hardware configuration, performance requirements, and availability requirements.

If a rebalance is in progress because a disk is manually or automatically dropped, then increasing the power of the rebalance shortens the time frame during which redundant copies of that data on the dropped disk are reconstructed on other disks.

The `V$ASM_OPERATION` view provides information for adjusting `ASM_POWER_LIMIT` and the resulting power of rebalance operations. The `V$ASM_OPERATION` view also gives an estimate in the `EST_MINUTES` column of the amount of time remaining for the rebalance operation to complete. You can see the effect of changing the rebalance power by observing the change in the time estimate.

The range of values for the `POWER` clause is the same for the `ASM_POWER_LIMIT` initialization parameter. For information about the `ASM_POWER_LIMIT` initialization parameter, refer to "ASM_POWER_LIMIT" on page 3-8.

For more information about rebalancing disk groups, refer to "Manually Rebalancing Disk Groups" on page 4-18.

Oracle ASM Disk Discovery

Disk discovery is the mechanism used to find the operating system names for disks Oracle ASM can access. It is used to find all the disks that comprise a disk group to be mounted, the disks an administrator wants to add to a disk group, or the disks the administrator might consider adding to a disk group. This section contains the following topics:

- How A Disk is Discovered

- Disk Discovery Rules

- Improving Disk Discovery Time

For additional information about disk discovery and the `ASM_DISKSTRING` initialization parameter, refer to "ASM_DISKSTRING" on page 3-7.

How A Disk is Discovered

When an Oracle ASM instance is initialized, Oracle ASM discovers and examines the contents of all of the disks that are in the paths that you designated with values in the `ASM_DISKSTRING` initialization parameter.

Disk discovery also occurs when you:

- Run the following SQL statements

  - Mount a disk group with `ALTER DISKGROUP ... MOUNT`

  - Online a disk with `ALTER DISKGROUP ... ONLINE DISK`

  - Add a disk to a disk group with `CREATE` or `ALTER DISKGROUP...ADD DISK`

  - Resize a disk in a disk group with `ALTER DISKGROUP...RESIZE DISK`

  - Query with `SELECT ... FROM V$ASM_DISKGROUP` or `V$ASM_DISK` views

- Run Oracle Enterprise Manager or Oracle ASM Configuration Assistant (ASMCA) operations that invoke the SQL statements previously listed

- Run ASMCMD commands that perform the same operations as the SQL statements previously listed

After Oracle ASM successfully discovers a disk, the disk appears in the `V$ASM_DISK` view. Disks that belong to a disk group, that is, disks that have a disk group name in the disk header, show a header status of `MEMBER`. Disks that were discovered, but that

have not yet been assigned to a disk group, have a status of either CANDIDATE or PROVISIONED. Disks that previously belonged to a disk group and were dropped cleanly from the disk group have a status of FORMER.

The PROVISIONED status implies that an additional platform-specific action has been taken by an administrator to make the disk available for Oracle ASM. For example, on Windows computers, the administrator might have used asmtool or asmtoolg to stamp the disk with a header. On Linux computers, the administrator might have used ASMLib to prepare the disk for Oracle ASM.

Example 4–10 shows a SQL query on V$ASM_DISK that displays the header status of a group of disks.

Example 4–10   Querying V$ASM_DISK for header status

```
SQL> SELECT name, header_status, path FROM V$ASM_DISK
     WHERE path LIKE '/devices/disk0%'

NAME       HEADER_STATUS PATH
---------  ------------- --------------------
           FORMER        /devices/disk02
           FORMER        /devices/disk01
           CANDIDATE     /devices/disk07
DISK06     MEMBER        /devices/disk06
DISK05     MEMBER        /devices/disk05
DISK04     MEMBER        /devices/disk04
DISK03     MEMBER        /devices/disk03
7 rows selected.
```

> **See Also:** *Oracle Database Reference* for information about the header status of an Oracle ASM disk that is displayed in the V$ASM_DISK view

Disk Discovery Rules

The rules for discovering Oracle ASM disks are as follows:

- Oracle ASM can discover up to 10,000 disks. That is, if more than 10,000 disks match the ASM_DISKSTRING initialization parameter, then Oracle ASM discovers only the first 10,000.

- Oracle ASM only discovers disk partitions. Oracle ASM does not discover partitions that include the partition table.

- From the perspective of the installation, candidate disks are those that have the CANDIDATE, PROVISIONED, or FORMER header status. These disks with a CANDIDATE, PROVISIONED, or FORMER status can be added to Oracle ASM disk groups without using the FORCE flag.

- When adding a disk, the FORCE option must be used if Oracle ASM recognizes that the disk was managed by Oracle. Such a disk appears in the V$ASM_DISK view with a status of FOREIGN. In this case, you can only add the disk to a disk group by using the FORCE keyword.

- MEMBER disks can usually be added to a disk group by specifying the FORCE flag, if the disks are not part of a currently mounted disk group.

In addition, Oracle ASM identifies the following configuration errors during discovery:

- Multiple paths to the same disk

In this case, if the disk is part of a disk group, then disk group mount fails. If the disk is being added to a disk group with the ADD DISK or CREATE DISKGROUP command, then the command fails. To correct the error, adjust the ASM_DISKSTRING value so that Oracle ASM does not discover multiple paths to the same disk. Or if you are using multipathing software, then ensure that you include only the pseudo-device name in the ASM_DISKSTRING value. See "Oracle ASM and Multipathing" on page 2-3.

■ Multiple Oracle ASM disks with the same disk header

This can be caused by having copied one disk onto another. In this case, the disk group mount operation fails.

Improving Disk Discovery Time

The value for the ASM_DISKSTRING initialization parameter is an operating system–dependent value that Oracle ASM uses to limit the set of paths that the discovery process uses to search for disks. When a new disk is added to a disk group, each Oracle ASM instance that has the disk group mounted must be able to discover the new disk using its ASM_DISKSTRING.

In many cases, the default value (NULL) is sufficient. Using a more restrictive value might reduce the time required for Oracle ASM to perform discovery, and thus improve disk group mount time or the time for adding a disk to a disk group. Oracle may dynamically change the ASM_DISKSTRING before adding a disk so that the new disk is discovered through this parameter.

The default value of ASM_DISKSTRING might not find all disks in all situations. If your site is using a third-party vendor ASMLib, then the vendor might have discovery string conventions that you must use for ASM_DISKSTRING. In addition, if your installation uses multipathing software, then the software might place pseudo-devices in a path that is different from the operating system default. See "Oracle ASM and Multipathing" on page 2-3 and consult the multipathing vendor documentation for details.

Managing Capacity in Disk Groups

When Oracle ASM provides redundancy, such as when you create a disk group with NORMAL or HIGH redundancy, you must have sufficient capacity in each disk group to manage a re-creation of data that is lost after a failure of one or two failure groups. After one or more disks fail, the process of restoring redundancy for all data requires space from the surviving disks in the disk group. If not enough space remains, then some files might end up with reduced redundancy.

Reduced redundancy means that one or more extents in the file are not mirrored at the expected level. For example, a reduced redundancy file in a high redundancy disk group has at least one file extent with two or fewer total copies of the extent instead of three. For unprotected files, data extents could be missing altogether. Other causes of reduced redundancy files are disks running out of space or an insufficient number of failure groups. The REDUNDANCY_LOWERED column in the V$ASM_FILE view provides information about files with reduced redundancy.

The following guidelines help ensure that you have sufficient space to restore full redundancy for all disk group data after the failure of one or more disks.

■ Normal redundancy disk group - It is best to have enough free space in your disk group to tolerate the loss of all disks in one failure group. The amount of free space should be equivalent to the size of the largest failure group.

- High redundancy disk group - It is best to have enough free space to cope with the loss of all disks in two failure groups. The amount of free space should be equivalent to the sum of the sizes of the two largest failure groups.

> **Note:** When you lose multiple disks from multiple failure groups, then you could lose both the primary and the redundant copies of your data. In addition, if you do not have enough capacity to restore redundancy, then Oracle ASM can continue to operate. However, if another disk fails, then the system may not be able to tolerate additional failures.

The `V$ASM_DISKGROUP` view contains the following columns that contain information to help you manage capacity:

- `REQUIRED_MIRROR_FREE_MB` indicates the amount of space that must be available in a disk group to restore full redundancy after the worst failure that can be tolerated by the disk group without adding additional storage. This requirement ensures that there are sufficient failure groups to restore redundancy. Also, this worst failure refers to a permanent failure where the disks must be dropped, not the case where the disks go offline and then back online.

  The amount of space displayed in this column takes the effects of mirroring into account. The value is computed as follows:

  – Normal redundancy disk group with more than two failure groups

    The value is the total raw space for all of the disks in the largest failure group. The largest failure group is the one with the largest total raw capacity. For example, if each disk is in its own failure group, then the value would be the size of the largest capacity disk.

  – High redundancy disk group with more than three failure groups

    The value is the total raw space for all of the disks in the two largest failure groups.

- `USABLE_FILE_MB` indicates the amount of free space, adjusted for mirroring, that is available for new files to restore redundancy after a disk failure. `USABLE_FILE_MB` is computed by subtracting `REQUIRED_MIRROR_FREE_MB` from the total free space in the disk group and then adjusting the value for mirroring. For example, in a normal redundancy disk group where by default the mirrored files use disk space equal to twice their size, if 4 GB of actual usable file space remains, then `USABLE_FILE_MB` equals roughly 2 GB. You can then add a file that is up to 2 GB.

- `TOTAL_MB` is the total usable capacity of a disk group in megabytes. The calculations for data in this column take the disk header overhead into consideration. The disk header overhead depends on the number of Oracle ASM disks and Oracle ASM files. This value is typically about 1% of the total raw storage capacity. For example, if the total LUN capacity provisioned for Oracle ASM is 100 GB, then the value in the `TOTAL_MB` column would be about 99 GB.

- `FREE_MB` is the unused capacity of the disk group in megabytes, without considering any data imbalance. There may be situations where the value in the `FREE_MB` column shows unused capacity but because one Oracle ASM disk is full, database writes fail because of the imbalance in the disk group. Ensure that you initiate a manual rebalance to force even data distribution which results in an accurate presentation of the values in the `FREE_MB` column.

With fine grain striping using 128 KB, the storage is preallocated to be eight times the AU size. The data file size may appear slightly larger on Oracle ASM than on a local file system because of the preallocation.

When you use Oracle ASM normal or high redundancy, the disk space utilization becomes more complex to measure because it depends on several variables.

> **Note:** The values in the `TOTAL_MB` and `FREE_MB` columns best describe space usage when you do not configure Oracle ASM mirroring, that is, when you use external redundancy.

The results from the following query show capacity metrics for a normal redundancy disk group that consists of six 1 GB (1024 MB) disks, each in its own failure group:

```
SQL> SELECT name, type, total_mb, free_mb, required_mirror_free_mb,
    usable_file_mb FROM V$ASM_DISKGROUP;

NAME        TYPE    TOTAL_MB    FREE_MB  REQUIRED_MIRROR_FREE_MB USABLE_FILE_MB
----------- ------  ----------  -------- ----------------------- --------------
DATA        NORMAL      6144     3768                       1024          1372
```

The `REQUIRED_MIRROR_FREE_MB` column shows that 1 GB of extra capacity must be available to restore full redundancy after one or more disks fail. The first three numeric columns in the query results are raw numbers. That is, they do not take redundancy into account. Only the last column is adjusted for normal redundancy. In the query output example for the `data` disk group, the calculation is as follows:

`(FREE_MB - REQUIRED_MIRROR_FREE_MB)` / 2 = `USABLE_FILE_MB`

`(3768 - 1024)` / 2 = `2744` / `2` = `1372`

Negative Values of USABLE_FILE_MB

Due to the relationship between `FREE_MB`, `REQUIRED_MIRROR_FREE_MB`, and `USABLE_FILE_MB`, `USABLE_FILE_MB` can become negative. Although this is not necessarily a critical situation, it does mean that:

- Depending on the value of `FREE_MB`, you may not be able to create new files.
- The next failure might result in files with reduced redundancy.

If `USABLE_FILE_MB` becomes negative, it is strongly recommended that you add more space to the disk group as soon as possible.

Oracle ASM Mirroring and Disk Group Redundancy

This section contains the following topics:

- Mirroring, Redundancy, and Failure Group Options
- Oracle ASM Recovery from Read and Write I/O Errors
- Oracle ASM Fast Mirror Resync
- Preferred Read Failure Groups

Mirroring, Redundancy, and Failure Group Options

If you specify mirroring for a file, then Oracle ASM automatically stores redundant copies of the file extents in separate failure groups. Failure groups apply only to

normal and high redundancy disk groups. You can define the failure groups for each disk group when you create or alter the disk group.

There are three types of disk groups based on the Oracle ASM redundancy level. Table 4–1 lists the types with their supported and default mirroring levels. The default mirroring levels indicate the mirroring level with which each file is created unless a different mirroring level is designated.

Table 4–1   Mirroring options for Oracle ASM disk group types

Disk Group Type	Supported Mirroring Levels	Default Mirroring Level
External redundancy	Unprotected (none)	Unprotected
Normal redundancy	Two-way, three-way, unprotected (none)	Two-way
High redundancy	Three-way	Three-way

The redundancy level controls how many disk failures are tolerated without dismounting the disk group or losing data. Each file is allocated based on its own redundancy, but the default comes from the disk group.

The redundancy levels are:

- External redundancy

  Oracle ASM does not provide mirroring redundancy and relies on the storage system to provide RAID functionality. Any write error causes a forced dismount of the disk group. All disks must be located to successfully mount the disk group.

- Normal redundancy

  Oracle ASM provides two-way mirroring by default, which means that all files are mirrored so that there are two copies of every **extent**. A loss of one Oracle ASM disk is tolerated. You can optionally choose three-way or unprotected mirroring.

  A file specified with `HIGH` redundancy (three-way mirroring) in a `NORMAL` redundancy disk group provides additional protection from a bad disk sector, not protection from a disk failure.

- High redundancy

  Oracle ASM provides triple mirroring by default. A loss of two Oracle ASM disks in different failure groups is tolerated.

If there are not enough online failure groups to satisfy the file mirroring (redundancy attribute value) specified in the disk group file type template, Oracle ASM allocates as many mirrors copies as possible and subsequently allocates the remaining mirrors when sufficient online failure groups are available. For information about specifying Oracle ASM disk group templates, see "Managing Disk Group Templates" on page 7-15.

Failure groups enable the mirroring of metadata and user data. System reliability can diminish if your environment has an insufficient number of failure groups.

This section contains these topics:

- Oracle ASM Failure Groups

- How Oracle ASM Manages Disk Failures

- Guidelines for Using Failure Groups

- Failure Group Frequently Asked Questions

Oracle ASM Failure Groups

Failure groups are used to store mirror copies of data. When Oracle ASM allocates an extent for a normal redundancy file, Oracle ASM allocates a primary copy and a secondary copy. Oracle ASM chooses the disk on which to store the secondary copy so that it is in a different failure group than the primary copy. Each copy is on a disk in a different failure group so that the simultaneous failure of all disks in a failure group does not result in data loss.

A failure group is a subset of the disks in a disk group, which could fail at the same time because they share hardware. The failure of common hardware must be tolerated. Four drives that are in a single removable tray of a large JBOD (Just a Bunch of Disks) array should be in the same failure group because the tray could be removed making all four drives fail at the same time. Drives in the same cabinet could be in multiple failure groups if the cabinet has redundant power and cooling so that it is not necessary to protect against failure of the entire cabinet. However, Oracle ASM mirroring is not intended to protect against a fire in the computer room that destroys the entire cabinet.

There are always failure groups even if they are not explicitly created. If you do not specify a failure group for a disk, then Oracle automatically creates a new failure group containing just that disk, except for disk groups containing disks on Oracle Exadata cells.

A normal redundancy disk group must contain at least two failure groups. A high redundancy disk group must contain at least three failure groups. However, Oracle recommends using several failure groups. A small number of failure groups, or failure groups of uneven capacity, can create allocation problems that prevent full use of all of the available storage.

Failure groups can be specified as regular or quorum failure groups. For information about quorum failure groups, see "Oracle Cluster Registry and Voting Files in Oracle ASM Disk Groups" on page 4-11.

> **See Also:** *Oracle Exadata documentation* for information about Oracle Exadata failure groups

How Oracle ASM Manages Disk Failures

Depending on the redundancy level of a disk group and how you define failure groups, the failure of one or more disks could result in either of the following:

- The disks are first taken offline and then automatically dropped. In this case, the disk group remains mounted and serviceable. In addition, because of mirroring, all of the disk group data remains accessible. After the disk drop operation, Oracle ASM performs a rebalance to restore full redundancy for the data on the failed disks.

- The entire disk group is automatically dismounted, which means loss of data accessibility.

Guidelines for Using Failure Groups

The following are guidelines for using failure groups:

- Each disk in a disk group can belong to only one failure group.

- Failure groups should all be of the same size. Failure groups of different sizes may lead to reduced availability.

- Oracle ASM requires at least two failure groups to create a normal redundancy disk group and at least three failure groups to create a high redundancy disk group.

Failure Group Frequently Asked Questions

This section discusses frequently asked questions about failure group under the following topics:

- How Many Failure Groups Should I Create?

- How are Multiple Failure Groups Recovered after Simultaneous Failures?

- When Should External, Normal, or High Redundancy Be Used?

How Many Failure Groups Should I Create?

Choosing the number of failure groups to create depends on the types of failures that must be tolerated without data loss. For small numbers of disks, such as fewer than 20, it is usually best to use the default failure group creation that puts every disk in its own failure group.

Using the default failure group creation for small numbers of disks is also applicable for large numbers of disks where your main concern is disk failure. For example, a disk group might be configured from several small modular disk arrays. If the system must continue operating when an entire modular array fails, then a failure group should consist of all of the disks in one module. If one module fails, then all of the data on that module is relocated to other modules to restore redundancy. Disks should be placed in the same failure group if they depend on a common piece of hardware whose failure must be tolerated with no loss of availability.

How are Multiple Failure Groups Recovered after Simultaneous Failures?

A simultaneous failure can occur if there is a failure of a piece of hardware used by multiple failure groups. This type of failure usually forces a dismount of the disk group if all disks are unavailable.

When Should External, Normal, or High Redundancy Be Used?

Oracle ASM mirroring runs on the database server and Oracle recommends to off load this processing to the storage hardware RAID controller by using external redundancy. You can use normal redundancy in the following scenarios:

- Storage system does not have RAID controller

- Mirroring across storage arrays

- Extended cluster configurations

In general, Oracle ASM mirroring is the Oracle alternative to third party logical volume managers. Oracle ASM mirroring eliminates the deployment of additional layers of software complexity in your Oracle database environment.

Oracle ASM Recovery from Read and Write I/O Errors

Read errors can be the result of a loss of access to the entire disk or media corruptions on an otherwise a healthy disk. Oracle ASM tries to recover from read errors on corrupted sectors on a disk. When a read error by the database or Oracle ASM triggers the Oracle ASM instance to attempt bad block remapping, Oracle ASM reads a good copy of the extent and copies it to the disk that had the read error.

- If the write to the same location succeeds, then the underlying allocation unit (sector) is deemed healthy. This might be because the underlying disk did its own bad block reallocation.

- If the write fails, Oracle ASM attempts to write the extent to a new allocation unit on the same disk. If this write succeeds, the original allocation unit is marked as unusable. If the write fails, the disk is taken offline.

One unique benefit on Oracle ASM based mirroring is that the database instance is aware of the mirroring. For many types of logical corruptions such as a bad checksum or incorrect System Change Number (SCN), the database instance proceeds through the mirror side looking for valid content and proceeds without errors. If the process in the database that encountered the read can obtain the appropriate locks to ensure data consistency, it writes the correct data to all mirror sides.

When encountering a write error, a database instance sends the Oracle ASM instance a *disk offline* message.

- If database can successfully complete a write to at least one extent copy and receive acknowledgment of the offline disk from Oracle ASM, the write is considered successful.

- If the write to all mirror side fails, database takes the appropriate actions in response to a write error such as taking the tablespace offline.

When the Oracle ASM instance receives a write error message from a database instance or when an Oracle ASM instance encounters a write error itself, the Oracle ASM instance attempts to take the disk offline. Oracle ASM consults the Partner Status Table (PST) to see whether any of the disk's partners are offline. If too many partners are offline, Oracle ASM forces the dismounting of the disk group. Otherwise, Oracle ASM takes the disk offline.

The ASMCMD `remap` command was introduced to address situations where a range of bad sectors exists on a disk and must be corrected before Oracle ASM or database I/O. For information about the `remap` command, see "remap" on page 12-48.

Oracle ASM Fast Mirror Resync

Restoring the redundancy of an Oracle ASM disk group after a transient disk path failure can be time consuming. This is especially true if the recovery process requires rebuilding an entire Oracle ASM failure group. Oracle ASM fast mirror resync significantly reduces the time to resynchronize a failed disk in such situations. When you replace the failed disk, Oracle ASM can quickly resynchronize the Oracle ASM disk extents.

> **Note:** To use this feature, the disk group compatibility attributes must be set to `11.1` or higher. For more information, refer to "Disk Group Compatibility" on page 4-33.

Any problems that make a failure group temporarily unavailable are considered transient failures that can be recovered by the Oracle ASM fast mirror resync feature. For example, transient failures can be caused by disk path malfunctions, such as cable failures, host bus adapter failures, controller failures, or disk power supply interruptions.

Oracle ASM fast resync keeps track of pending changes to extents on an `OFFLINE` disk during an outage. The extents are resynced when the disk is brought back online.

By default, Oracle ASM drops a disk in 3.6 hours after it is taken offline. You can set the `DISK_REPAIR_TIME` disk group attribute to delay the drop operation by specifying a time interval to repair the disk and bring it back online. The time can be specified in units of minutes (`m` or `M`) or hours (`h` or `H`). If you omit the unit, then the default unit is hours. The `DISK_REPAIR_TIME` disk group attribute can only be set with the `ALTER DISKGROUP` SQL statement.

If the attribute is not set explicitly, then the default value (`3.6h`) applies to disks that have been set to `OFFLINE` mode without an explicit `DROP AFTER` clause. Disks taken offline due to I/O errors do not have a `DROP AFTER` clause.

The default `DISK_REPAIR_TIME` attribute value is an estimate that should be adequate for most environments. However, ensure that the attribute value is set to the amount of time that you think is necessary in your environment to fix any transient disk error, and that you are willing to tolerate reduced data redundancy.

The elapsed time (since the disk was set to `OFFLINE` mode) is incremented only when the disk group containing the offline disks is mounted. The `REPAIR_TIMER` column of `V$ASM_DISK` shows the amount of time left (in seconds) before an offline disk is dropped. After the specified time has elapsed, Oracle ASM drops the disk. You can override this attribute with the `ALTER DISKGROUP OFFLINE DISK` statement and the `DROP AFTER` clause.

> **Note:** If a disk is offlined by Oracle ASM because of an I/O (write) error or is explicitly offlined using the `ALTER DISKGROUP`... `OFFLINE` statement without the `DROP AFTER` clause, then the value specified for the `DISK_REPAIR_TIME` attribute for the disk group is used.
>
> Altering the `DISK_REPAIR_TIME` attribute has no effect on offline disks. The new value is used for any disks that go offline after the attribute is updated. You can confirm this behavior by viewing the Oracle ASM alert log.

If an offline disk is taken offline for a second time, then the elapsed time is reset and restarted. If another time is specified with the `DROP AFTER` clause for this disk, the first value is overridden and the new value applies. A disk that is in `OFFLINE` mode cannot be dropped with an `ALTER DISKGROUP DROP DISK` statement; an error is returned if attempted. If for some reason the disk must be dropped (such as the disk cannot be repaired) before the repair time has expired, a disk can be dropped immediately by issuing a second `OFFLINE` statement with a `DROP AFTER` clause specifying `0h` or `0m`.

You can use `ALTER DISKGROUP` to set the `DISK_REPAIR_TIME` attribute to a specified hour or minute value, such as 4.5 hours or 270 minutes. For example:

```
ALTER DISKGROUP data SET ATTRIBUTE 'disk_repair_time' = '4.5h'
ALTER DISKGROUP data SET ATTRIBUTE 'disk_repair_time' = '270m'
```

After you repair the disk, run the SQL statement `ALTER DISKGROUP ONLINE DISK`. This statement brings a repaired disk group back online to enable writes so that no new writes are missed. This statement also starts a procedure to copy of all of the extents that are marked as stale on their redundant copies.

If a disk goes offline when the Oracle ASM instance is in rolling upgrade mode, the disk remains offline until the rolling upgrade has ended and the timer for dropping the disk is stopped until the Oracle ASM cluster is out of rolling upgrade mode. See . Examples of taking disks offline and bringing them online follow.

The following example takes disk `DATA_001` offline and drops it after five minutes.

```
ALTER DISKGROUP data OFFLINE DISK DATA_001 DROP AFTER 5m;
```

The next example takes the disk `DATA_001` offline and drops it after the time period designated by `DISK_REPAIR_TIME` elapses:

```
ALTER DISKGROUP data OFFLINE DISK DATA_001;
```

This example takes all of the disk in failure group `FG2` offline and drops them after the time period designated by `DISK_REPAIR_TIME` elapses. IF you used a `DROP AFTER` clause, then the disks would be dropped after the specified time:

```
ALTER DISKGROUP data OFFLINE DISK IN FAILGROUP FG2;
```

The next example brings all of the disks in failure group `FG2` online:

```
ALTER DISKGROUP data ONLINE DISK IN FAILGROUP FG2;
```

This example brings only disk `DATA_001` online:

```
ALTER DISKGROUP data ONLINE DISK DATA_001;
```

This example brings all of the disks in disk group `DATA` online:

```
ALTER DISKGROUP data ONLINE ALL;
```

Querying the `V$ASM_OPERATION` view while you run any of these types of `ALTER DISKGROUP ... ONLINE` statements displays the name and state of the current operation that you are performing. For example, the query:

```
SELECT GROUP_NUMBER, OPERATION, STATE FROM V$ASM_OPERATION;
```

Displays output similar to the following:

```
GROUP_NUMBER OPERA STAT
------------ ----- ----
           1 ONLIN RUN
```

An `OFFLINE` operation is not displayed in a `V$ASM_OPERATION` view query.

> **See Also:** *Oracle Database SQL Language Reference* for information about `ALTER DISKGROUP` and `CREATE DISKGROUP`

Preferred Read Failure Groups

When you configure Oracle ASM failure groups, it might be more efficient for a node to read from an extent that is closest to the node, even if that extent is a secondary extent. In other words, you can configure Oracle ASM to read from a secondary extent if that extent is closer to the node instead of Oracle ASM reading from the primary copy which might be farther from the node. Using the preferred read failure groups feature is most useful in extended clusters.

To use this feature, Oracle recommends that you configure at least one mirrored extent copy from a disk that is local to a node in an extended cluster. However, a failure group that is preferred for one instance might be remote to another instance in the same Oracle RAC database. The parameter setting for preferred read failure groups is instance specific.

Both the Oracle ASM clients and Oracle ASM require Oracle Database 11*g* release 1 (11.1) or higher to use preferred read failure groups.

> **Note:**   If you do not specify failure groups for a disk group, each disk in the disk group belongs to its own failure group. Oracle does not recommend that you configure multiple preferred read failure groups in a disk group for an Oracle ASM instance. For any given instance, if you specify multiple failure groups in the same disk group as preferred read, a warning message is written to the alert log.

> **See Also:**   *Oracle Real Application Clusters Administration and Deployment Guide* for information about configuring preferred read disks in extended clusters

Configuring and Administering Preferred Read Failure Groups

To configure this feature, set the `ASM_PREFERRED_READ_FAILURE_GROUPS` initialization parameter to specify a list of failure group names as preferred read disks. For more information about this initialization parameter, refer to "ASM_PREFERRED_READ_FAILURE_GROUPS" on page 3-9.

Set the parameter where `diskgroup` is the name of the disk group and `failuregroup` is the name of the failure group, separating these variables with a period. Oracle ASM ignores the name of a failure group that you use in this parameter setting if the failure group does not exist in the named disk group. You can append multiple values using commas as a separator as follows:

`ASM_PREFERRED_READ_FAILURE_GROUPS = diskgroup.failuregroup,...`

In an extended cluster, the failure groups that you specify with settings for the `ASM_PREFERRED_READ_FAILURE_GROUPS` parameter should only contain disks that are local to the instance. For normal redundancy disk groups, there should be only one failure group on each site of the extended cluster.

If there are multiple mirrored copies and you have set a value for the `ASM_PREFERRED_READ_FAILURE_GROUPS` parameter, then Oracle ASM first reads the copy that resides on a preferred read disk. If that read fails, then Oracle ASM attempts to read from the next mirrored copy that might not be on a preferred read disk.

Having multiple failure groups on one site can cause the loss of access to the disk group by the other sites if the site containing multiple failure groups fails. In addition, by having multiple failure groups on a site, an extent might not be mirrored to another site. This can diminish the read performance of the failure group on the other site.

For example, for a normal redundancy disk group, if a site contains two failure groups of a disk group, then Oracle ASM might put both mirror copies of an extent on the same site. In this configuration, Oracle ASM cannot protect against data loss from a site failure.

You should configure at most two failure groups on a site for a high redundancy disk group. If there are three sites in an extended cluster, for the same reason previously mentioned, then you should only create one failure group.

For a two-site extended cluster, a normal redundancy disk group only has two failure groups. In this case, you can only specify one failure group as a preferred read failure group for each instance.

You can use views to identify preferred read failure groups, such as the `V$ASM_DISK` view that shows whether a disk is a preferred read disk by the value in the `PREFERRED_READ` column. You can also use `V$ASM_DISK` to verify whether local disks in an extended

cluster are preferred read disks. Use the Oracle ASM disk I/O statistics to verify that read operations are using the preferred read disks that you configured.

If a disk group is not optimally configured for an extended cluster, then Oracle ASM records warning messages in the alert logs. To identify specific performance issues with Oracle ASM preferred read failure groups, use the `V$ASM_DISK_IOSTAT` view. This view displays disk I/O statistics for each Oracle ASM client. You can also query the `V$ASM_DISK_IOSTAT` view on a database instance. However, this query only shows the I/O statistics for the database instance. In general, optimal preferred read extended cluster configurations balance performance with disk group availability.

> **See Also:** *Oracle Database Reference* for details about the `V$ASM*` dynamic performance views

Performance and Scalability Considerations for Disk Groups

This section discusses the following considerations for evaluating disk group performance:

- Determining the Number of Disk Groups
- Performance Characteristics When Grouping Disks
- Oracle ASM Storage Limits

Determining the Number of Disk Groups

Use the following criteria to determine the number of disk groups to create:

- Disks in a given disk group should have similar size and performance characteristics. If you have several different types of disks in terms of size and performance, then create several disk groups that contain similar characteristics.
- Create separate disk groups for your database files and fast recovery area for backup files. This configuration allows fast recovery should a disk group failure occur.

For information about creating disk groups for a new installation, refer to "Creating Disk Groups for a New Oracle Installation" on page 4-7.

Performance Characteristics When Grouping Disks

Oracle ASM load balances the file activity by uniformly distributing file extents across all of the disks in a disk group. For this technique to be effective it is important that disks in a disk group be of similar performance characteristics. For example, the newest and fastest disks might reside in a disk group reserved for the database work area, and slower drives could reside in a disk group reserved for the fast recovery area.

There might be situations where it is acceptable to temporarily have disks of different sizes and performance characteristics coexist in a disk group. This would be the case when migrating from an old set of disks to a new set of disks. The new disks would be added and the old disks dropped. As the old disks are dropped, their storage is migrated to the new disks while the disk group is online.

Oracle ASM Storage Limits

Oracle ASM provides near unlimited capacity for future growth, but does have some storage limits. For example, Oracle ASM has the following limits on the number of disk groups, disks, and files:

- 63 disk groups in a storage system

- 10,000 Oracle ASM disks in a storage system

- 1 million files for each disk group

Without any Oracle Exadata Storage, Oracle ASM has these storage limits:

- 2 terabytes (TB) maximum storage for each Oracle ASM disk

- 20 petabytes (PB) maximum for the storage system

With all Oracle Exadata Storage, Oracle ASM has these storage limits:

- 4 PB maximum storage for each Oracle ASM disk

- 40 exabytes (EB) maximum for the storage system

The maximum size limit of a disk group equals the maximum disk size multiplied by the maximum number of disks in a disk group (10,000).

The maximum number of disks across all disk groups is 10,000. The 10,000 disks can be in one disk group or distributed across a maximum of 63 disk groups. This is a limitation on the number of Oracle ASM disks, not necessarily the number of spindles. A storage array could group multiple spindles into a LUN that is used as a single Oracle ASM disk. However Oracle ASM is currently limited to 2 TB in a single disk unless using Oracle Exadata storage.

File size limits are dependent on the value of the disk group compatibility attributes. Oracle ASM supports file sizes greater than 128 TB in any redundancy mode when the `COMPATIBLE.RDBMS` disk group attribute is set greater than `10.1`.

If `COMPATIBLE.RDBMS` is set to `10.1`, the file size limits are less. For example, with `COMPATIBLE.RDBMS` equal to `10.1` and the AU size equal to 1 MB, Oracle ASM file size limits are:

- External redundancy: 16 TB

- Normal redundancy: 5.8 TB

- High redundancy: 3.9 TB

> **Note:** Oracle Database supports data file sizes up to 128 TB depending on the file system. In addition, Oracle Database has a file size limit that is dependent on the `DB_BLOCK_SIZE` initialization parameter.

For information about Oracle ASM disk group compatibility attributes, see "Disk Group Compatibility" on page 4-33. For information about Oracle ASM file size limits, see Table 4–4 on page 4-40.

Disk Group Compatibility

This section describes disk group compatibility under the following topics:

- Overview of Disk Group Compatibility

- Disk Group Compatibility Attributes

- Setting Disk Group Compatibility Attributes

- Considerations When Setting Disk Group Compatibility in Replicated Environments

Overview of Disk Group Compatibility

Advancing the disk group compatibility settings enables you to use the new Oracle ASM features that are available in a later release. For example, a disk group with the disk group compatibility attributes set to 11.2 can take advantage of new Oracle 11*g* release 2 (11.2) features, such as Oracle ASM volumes in disk groups and Oracle ASM File Access Control. See Table 4–3 on page 4-38 for the features enabled for combinations of compatibility attribute settings.

The disk group compatibility feature also enables environments to interoperate when they use disk groups from both Oracle Database 10*g* and Oracle Database 11*g*. For example, disk group compatibility settings that are set to Oracle Database 10*g* enable an Oracle 10*g* client to access a disk group created with Oracle ASM 11*g*.

The disk group attributes that determine compatibility are COMPATIBLE.ASM, COMPATIBLE.RDBMS. and COMPATIBLE.ADVM. The COMPATIBLE.ASM and COMPATIBLE.RDBMS attribute settings determine the minimum Oracle Database software version numbers that a system can use for Oracle ASM and the database instance types respectively. For example, if the Oracle ASM compatibility setting is 11.2, and RDBMS compatibility is set to 11.1, then the Oracle ASM software version must be at least 11.2, and the Oracle Database client software version must be at least 11.1. The COMPATIBLE.ADVM attribute determines whether the Oracle ASM Dynamic Volume Manager feature can create an volume in a disk group.

When you create a disk group, you can specify the disk group compatibility attribute settings in the CREATE DISKGROUP SQL statement. The ALTER DISKGROUP SQL statement can update the compatible attribute settings for existing disk groups. If not specified when using the CREATE DISKGROUP SQL statement, 10.1 is the default setting for both the COMPATIBLE.ASM and COMPATIBLE.RDBMS attributes for Oracle ASM in Oracle Database 11*g*. The COMPATIBLE.ADVM attribute is empty if it is not set. See Table 4–2 on page 4-36 for examples of valid combinations of compatible attribute settings.

You can set and update disk group attributes with the ASMCMD setattr command. For information about the ASMCMD setattr command, see "setattr" on page 12-49.

Notes:

- The disk group compatibility settings determine whether your environment can use the latest Oracle ASM features.

- The disk group compatibility settings can only be advanced; you cannot revert to a lower compatibility setting. See "Reverting Disk Group Compatibility" on page 4-39.

- The COMPATIBLE.ASM attribute must be advanced before advancing other disk group compatibility attributes and its value must be greater than or equal to the value of other disk group compatibility attributes.

Disk Group Compatibility Attributes

The disk group compatibility attributes specify the disk group compatibility settings for Oracle ASM and database instances. These attributes are described under the following topics:

- COMPATIBLE.ASM
- COMPATIBLE.RDBMS
- COMPATIBLE.ADVM

COMPATIBLE.ASM

The value for the disk group `COMPATIBLE.ASM` attribute determines the minimum software version for an Oracle ASM instance that can use the disk group. This setting also affects the format of the data structures for the Oracle ASM metadata on the disk. The format of other file contents is determined by Oracle ASM Dynamic Volume Manager (Oracle ADVM) and the database instance.

For Oracle ASM in Oracle Database 11*g*, `10.1` is the default setting for the `COMPATIBLE.ASM` attribute when using the SQL `CREATE DISKGROUP` statement, the ASMCMD `mkdg` command, and Oracle Enterprise Manager **Create Disk Group** page. When creating a disk group with ASMCA, the default setting is `11.2`.

COMPATIBLE.RDBMS

The value for the disk group `COMPATIBLE.RDBMS` attribute determines the minimum `COMPATIBLE` database initialization parameter setting for any database instance that is allowed to use the disk group. Before advancing the `COMPATIBLE.RDBMS` attribute, ensure that the values for the `COMPATIBLE` initialization parameter for all of the databases that access the disk group are set to at least the value of the new setting for `COMPATIBLE.RDBMS`.

For example, if the `COMPATIBLE` initialization parameters of the databases are set to either `11.1` or `11.2`, then `COMPATIBLE.RDBMS` can be set to any value between `10.1` and `11.1` inclusively.

For Oracle ASM in Oracle Database 11*g*, `10.1` is the default setting for the `COMPATIBLE.RDBMS` attribute when using the SQL `CREATE DISKGROUP` statement, the ASMCMD `mkdg` command, ASMCA **Create Disk Group** page, and Oracle Enterprise Manager **Create Disk Group** page.

> **Note:** The database initialization parameter `COMPATIBLE` enables you to use a new release of Oracle Database, while at the same time guaranteeing backward compatibility with an earlier release. See *Oracle Database Reference* for more information about the `COMPATIBLE` initialization parameter.

COMPATIBLE.ADVM

The value for the disk group `COMPATIBLE.ADVM` attribute determines whether the disk group can contain Oracle ASM volumes. The value must be set to `11.2` or higher. Before setting this attribute, the `COMPATIBLE.ASM` value must be `11.2` or higher. Also, the Oracle ADVM volume drivers must be loaded in the supported environment.

By default, the value of the `COMPATIBLE.ADVM` attribute is empty until set.

For more information about Oracle ADVM, see "Overview of Oracle ASM Dynamic Volume Manager" on page 5-20.

Setting Disk Group Compatibility Attributes

This section discusses the settings of the disk group compatibility attributes and how to set the attribute values with the `CREATE DISKGROUP` or `ALTER DISKGROUP` SQL statement.

This section contains these topics:

- Valid Combinations of Compatibility Attribute Settings
- Using CREATE DISKGROUP with Compatibility Attributes

- Using ALTER DISKGROUP with Compatibility Attributes
- Viewing Compatibility Attribute Settings
- Features Enabled By Disk Group Compatibility Attribute Settings
- Reverting Disk Group Compatibility

You can also set the disk group compatibility settings with Oracle Enterprise Manager, Oracle ASM command-line utility (ASMCMD), and Oracle ASM Configuration Assistant (ASMCA). See Chapter 9, "Administering Oracle ASM with Oracle Enterprise Manager", Chapter 11, "Oracle ASM Configuration Assistant", and Chapter 12, "Oracle ASM Command-Line Utility".

> **Note:** Advancing the values for disk group compatibility attributes is an irreversible operation. See "Reverting Disk Group Compatibility" on page 4-39.

> **See Also:** *Oracle Database SQL Language Reference* for more information about the disk group compatibility SQL statements

Valid Combinations of Compatibility Attribute Settings

When setting the values for the disk group attributes, specify at least the major and minor versions of a valid Oracle Database release number. For example, you can specify compatibility as `'11.1'` or `'11.2'`; Oracle assumes that any missing version number digits are zeros.

Table 4–2 shows some valid combinations of the disk group compatibility attributes and the valid Oracle ASM and database instance versions for each combination.

Table 4–2   Examples of disk group compatibility attribute settings

COMPATIBLE.ASM	COMPATIBLE.RDBMS	COMPATIBLE.ADVM	ASM Instance Version	COMPATIBLE Setting for RDBMS Instance
10.1	10.1	n/a	>= 10.1	>= 10.1
11.1	10.1	n/a	>= 11.1	>= 10.1
11.2	11.1	11.2	>= 11.2	>= 11.1
11.2	11.2	11.2	>= 11.2	>= 11.2

These are some possible combinations of Oracle ASM and database releases:

- The database release is 11*g* release 2 (11.2) (database initialization parameter `COMPATIBLE` is set to `11.2`) and the Oracle ASM release is 11*g* release 2 (11.2). The `COMPATIBLE.ASM` and `COMPATIBLE.RDBMS` disk group attributes are set to the default value of `10.1` for a disk group.

  The Oracle ASM disk group functionality remains at 10*g*.

- The database release is 10*g* and the Oracle ASM release is 11.2. `COMPATIBLE.ASM` is set to `11.2` and `COMPATIBLE.RDBMS` is set to `10.1` for a disk group.

  The disk group features enabled for this combination of attribute settings are shown in Table 4–3 on page 4-38.

- The database release is 11.2 (database initialization parameter `COMPATIBLE` is set to `11.2`) and the Oracle ASM release is 11.2. All the disk group compatibility attributes are set to `11.2` for a disk group.

The disk group features enabled for this combination of attribute settings are shown in Table 4–3 on page 4-38.

Using CREATE DISKGROUP with Compatibility Attributes

You can specify the compatibility settings for a disk group with the CREATE DISKGROUP statement when creating the disk group.

The following example creates a normal redundancy disk group data1 with the Oracle ASM compatibility set to 11.2 and the RDBMS compatibility set to the default (the COMPATIBLE.RDBMS default is less than or equal to 11.2):

```
CREATE DISKGROUP data1 DISK '/dev/sd*'
     ATTRIBUTE 'compatible.asm' = '11.2';
```

The following example creates a normal redundancy disk group data2 with the ASM, RDBMS, and ADVM compatibility set to 11.2:

```
CREATE DISKGROUP data2 DISK '/dev/sd*'
     ATTRIBUTE 'compatible.asm' = '11.2', 'compatible.rdbms' = '11.2',
               'compatible.advm' = '11.2';
```

Using ALTER DISKGROUP with Compatibility Attributes

After a disk group has been created, you can use the ALTER DISKGROUP SQL statement to change the compatibility attributes. The ALTER DISKGROUP SQL statement ensures that Oracle can advance the compatibility of the specified disk group before committing the change.

All of the affected databases and file systems should be online when running ALTER DISKGROUP to ensure that advancing compatibility does not reduce the database and file system access. When advancing disk group compatibility, you must advance the COMPATIBLE.ASM attribute before the COMPATIBLE.RDBMS or COMPATIBLE.ADVM attribute to ensure a valid combination of compatible attribute settings as shown in Table 4–2. You can advance only one compatibility attribute in a single ALTER DISKGROUP statement.

The following example advances the Oracle ASM compatibility for disk group data3 to 11.2. An Oracle ASM instance must be at release 11.2 or higher to access the data3 disk group.

```
ALTER DISKGROUP data3 SET ATTRIBUTE 'compatible.asm' = '11.2';
```

The following example advances the COMPATIBLE.RDBMS and COMPATIBLE.ADVM compatibility attributes of the disk group data3 to 11.2. This example assumes that the value of COMPATIBLE.ASM is set to 11.2.

```
ALTER DISKGROUP data3 SET ATTRIBUTE 'compatible.rdbms' = '11.2',
ALTER DISKGROUP data3 SET ATTRIBUTE 'compatible.advm' = '11.2';
```

Viewing Compatibility Attribute Settings

You can view the disk group compatibility settings in the V$ASM_ATTRIBUTE view. However, the V$ASM_ATTRIBUTE view does not display any rows when the COMPATIBLE.ASM value is set to 10.1. Instead you can determine the values for the COMPATIBLE.ASM and COMPATIBLE.RDBMS disk group compatibility attributes with the COMPATIBILITY and DATABASE_COMPATIBILITY columns of the V$ASM_DISKGROUP view.

See Example 6–1, "Viewing disk group attributes with V$ASM_ATTRIBUTE" on page 6-2 for an example querying the V$ASM_ATTRIBUTE view.

You can also display the disk group compatibility attributes with the ASMCMD command lsattr. For information about lsattr, see "lsattr" on page 12-36.

See Also:

- *Oracle Database SQL Language Reference* for information about disk group attributes

- *Oracle Database Reference* for details on the `V$ASM_DISKGROUP` and `V$ASM_ATTRIBUTE` dynamic performance views

Features Enabled By Disk Group Compatibility Attribute Settings

Table 4–3 describes the features enabled by valid combinations of the disk group compatibility attribute settings.

Table 4–3   Features enabled by disk group compatibility attribute settings

Disk Group Features Enabled	COMPATIBLE.ASM	COMPATIBLE.RDBMS	COMPATIBLE.ADVM
Support for larger AU sizes (32 or 64 MB)	>= 11.1	>= 11.1	n/a
Attributes are displayed in the `V$ASM_ATTRIBUTE` view	>= 11.1	n/a	n/a
Fast mirror resync	>= 11.1	>= 11.1	n/a
Variable size extents	>= 11.1	>= 11.1	n/a
Exadata storage	>= 11.1.0.7	>= 11.1.0.7	n/a
Intelligent Data Placement	>= 11.2	>= 11.2	n/a
OCR and voting files in a disk group	>= 11.2	n/a	n/a
Sector size set to nondefault value	>= 11.2	>= 11.2	n/a
Oracle ASM SPFILE in a disk group	>= 11.2	n/a	n/a
Oracle ASM File Access Control	>= 11.2	>= 11.2	n/a
Volumes in disk groups	>= 11.2	n/a	>= 11.2
`ASM_POWER_LIMIT` value up to `1024`	>= 11.2.0.2	n/a	n/a
Encryption, replication, security, tagging on Linux systems	>= 11.2.0.2	n/a	>= 11.2.0.2
Encryption, replication, security, tagging on Windows systems	>= 11.2.0.3	n/a	>= 11.2.0.3
Read-write snapshots	>= 11.2.0.3	n/a	>= 11.2.0.3

The following list applies to Table 4–3.

- If encryption is configured for the first time on Oracle ASM 11*g* release 2 version 11.2.0.3 on Linux or if encryption parameters must be changed or a new volume encryption key must be created following a software upgrade to Oracle ASM 11*g* release 2 version 11.2.0.3 on Linux, then the disk group compatibility attributes for `ASM` and `ADVM` must be set to `11.2.0.3`.

- Oracle ASM features not explicitly listed in Table 4–3 do not require advancing the disk group compatibility attribute settings.

- The value of `COMPATIBLE.ASM` must always be greater than or equal to the value of `COMPATIBLE.RDBMS` and `COMPATIBLE.ADVM`.

- A value of not applicable (n/a) means that the setting of the attribute has no effect on the feature.

- For information about the rebalancing process and the `ASM_POWER_LIMIT` initialization parameter, refer to "ASM_POWER_LIMIT" on page 3-8.

Reverting Disk Group Compatibility

Advancing the values for disk group compatibility attributes is an irreversible operation. If you advance the disk group compatibility settings, you cannot change the values back to the previous settings. To revert to the previous values, you must create a new disk group with the old compatibility attribute settings and then restore the database files that were in the disk group to the new disk group.

When you revert to a new disk group with the old compatibility attribute settings, the latest Oracle ASM features might not be available. For example, if you revert the disk group compatibility to a pre-11.2 value, Oracle ACFS functionality is not available.

For example, you could perform the following procedure to revert a disk group to previous compatibility settings:

1. If the Oracle ASM SPFILE is in the disk group, move this SPFILE out of the disk group:

   a. Connect with SQL*Plus to the Oracle ASM instance.

   b. Create a PFILE in the file system.

      For example:

      ```
      SQL> CREATE PFILE '$ORACLE_HOME/dbs/asmspfile.ora' FROM SPFILE
      ```

2. If the OCR and voting files are in the disk group, move them out of this disk group.

   > **See Also:** The *Oracle Clusterware Administration and Deployment Guide* for information about administering OCR and voting files

3. Back up any files that must be saved.

   a. Back up the database files.

   b. If an Oracle ACFS file system is mounted on an Oracle ADVM volume on the disk group, the operating system files in the file system must be backed up or copied to a location outside the file system mount point.

4. Create a new disk group using SQL `CREATE DISKGROUP` specifying the previous values for the disk group attribute settings.

   For information about creating a disk group, see "Using the CREATE DISKGROUP SQL Statement" on page 4-4.

5. Restore the database files into the newly created disk group using Recovery Manager (RMAN).

   For information about moving data files between disk groups, see "Moving Data Files Between Oracle ASM Disk Groups Using RMAN" on page 8-9.

6. Drop the disk group with the advanced disk group compatibility settings using SQL `DROP DISKGROUP INCLUDING CONTENTS` to remove the disk group and its contents.

   This SQL statement also removes any Oracle ACFS file system and its contents.

   For information about dropping a disk group, see "Dropping Disk Groups" on page 4-46.

Considerations When Setting Disk Group Compatibility in Replicated Environments

If you advance disk group compatibility, then you could enable the creation of files that are too large to be managed by a previous Oracle database release. You must be aware of the file size limits because replicated sites cannot continue using the software from a previous release to manage these large files. The disk group compatibility settings should be the same for all replicated environments.

Table 4–4 shows the maximum Oracle ASM file sizes supported for `COMPATIBLE.RDBMS` settings when the `AU_SIZE` attribute is set to one megabyte for a disk group.

Table 4–4   Maximum Oracle ASM file sizes for disk groups with AU_SIZE equal to 1 MB

Redundancy	COMPATIBLE.RDBMS = 10.1	COMPATIBLE.RDBMS >= 11.1
External	16 TB	140 PB
Normal	5.8 TB	23 PB
High	3.9 TB	15 PB

Table 4–4 shows that Oracle Database 10*g* can only support a file size of up to 16 TB for external redundancy. If you advance the `COMPATIBILE.RDBMS` attribute to `11.1` or greater, then a file can grow beyond 16 TB. However, the larger size causes the file to be unusable in a replicated and disaster recovery site if the disaster recovery site has a disk group `COMPATIBLE.RDBMS` setting that is incompatible with the larger size.

For information about Oracle ASM storage sizes, see "Oracle ASM Storage Limits" on page 4-32.

> **See Also:**
>
> - *Oracle Database Upgrade Guide* for information about database compatibility
> - *Oracle Database Administrator's Guide* for information about the `COMPATIBLE` initialization parameter and irreversible compatibility
> - *Oracle Database Reference* for information about the `COMPATIBLE` initialization parameter

Managing Oracle ASM File Access Control for Disk Groups

Oracle ASM File Access Control provides optional protection for the content of Oracle ASM disk groups from accidental access by unauthorized Oracle ASM clients, such as an unauthorized database.

To set up Oracle ASM File Access Control, you must designate separate operating system groups as described in "Using Separate Operating System Groups for Oracle ASM Users" on page 3-27. Oracle ASM File Access Control is available for Linux and UNIX computers only.

You can manage Oracle ASM file access control with ASMCMD commands, Oracle Enterprise Manager, and SQL statements.

This section contains these topics:

- About Oracle ASM File Access Control
- Setting Disk Group Attributes for Oracle ASM File Access Control
- Using SQL Statements to Manage Oracle ASM File Access Control

For information about managing Oracle ASM File Access Control with ASMCMD commands, see "ASMCMD File Access Control Commands" on page 12-53.

For information about managing Oracle ASM File Access Control with Oracle Enterprise Manager, see "Managing Oracle ASM File Access Control with Oracle Enterprise Manager" on page 9-16.

For information about views that provide details about Oracle ASM file access control, see "Viewing Oracle ASM File Access Control Information" on page 6-4.

For information about controlling accessing to Oracle ASM instances, see "Authentication for Accessing Oracle ASM Instances" on page 3-25.

About Oracle ASM File Access Control

Oracle ASM File Access Control restricts the access of files to specific Oracle ASM clients that connect as SYSDBA. An Oracle ASM client is typically a database, which is identified as the user that owns the database instance home. Oracle ASM File Access Control uses this user name to identify a database. Oracle ASM File Access Control restricts access based on the operating system effective user identification number of a database owner. For example, in Table 3–2, " Separated operating system groups and privileges for Oracle ASM users" on page 3-28 the databases are identified as oracle1 and oracle2.

Oracle ASM uses file access control to determine the additional privileges that are given to a database that has been authenticated AS SYSDBA on the Oracle ASM instance. These additional privileges include the ability to modify and delete certain files, aliases, and user groups.

You can set up user groups to specify the list of databases that share the same access permissions to Oracle ASM files. User groups are lists of databases and any database that authenticates AS SYSDBA can create a user group. However, only the creator of a group can delete it or modify its membership list.

Each Oracle ASM file has three categories of privileges: owner, group, and other. Each category can have no permission, read-only permission, or read-write permission.

The file owner is usually the creator of the file and can assign permissions for the file in any of the owner, group, or other categories. The owner can also change the group associated with the file.

When administering Oracle ASM File Access Control, Oracle recommends that you connect as SYSDBA to the database instance that is the owner, or planned owner, of the files in the disk group.

To set up Oracle ASM File Access Control for files in a disk group, perform the following steps:

1. Alter a new or existing disk group to set the Oracle ASM File Access Control disk group attributes.

    For a newly-created disk group, you should set the disk group attributes before creating any files in the disk group.

    See "Setting Disk Group Attributes for Oracle ASM File Access Control" on page 4-42.

2. For files that exist in a disk group before setting the Oracle ASM File Access Control disk group attributes, you must explicitly set the permissions and ownership on those existing files.

Ensure that the user exists before setting ownership or permissions on a file. The file must be closed before setting the ownership or permissions.

See `ALTER DISKGROUP SET PERMISSION` and `ALTER DISKGROUP SET OWNERSHIP` in "Using SQL Statements to Manage Oracle ASM File Access Control" on page 4-43.

3. Optionally, you can create user groups that are groups of database users that share the same access permissions to Oracle ASM files.

See `ALTER DISKGROUP ADD USERGROUP` in "Using SQL Statements to Manage Oracle ASM File Access Control" on page 4-43.

Setting Disk Group Attributes for Oracle ASM File Access Control

To manage Oracle ASM File Access Control for a disk group, you must set the `ACCESS_CONTROL.ENABLED` and `ACCESS_CONTROL.UMASK` disk group attributes. You can set the attributes by altering the disk group with the `ALTER DISKGROUP` SQL statement or you can set the attributes with the ASMCMD `setattr` command. For information about the ASMCMD `setattr` command, see "setattr" on page 12-49.

When you set up file access control on an existing disk group, the files previously created remain accessible by everyone, unless you run the `ALTER DISKGROUP SET PERMISSION` SQL statement or the ASMCMD `chmod` command to restrict the permissions. For information about the ASMCMD `chmod` command, see "chmod" on page 12-54.

The `COMPATIBLE.ASM` and `COMPATIBLE.RDBMS` disk group attributes must be set to 11.2 or higher to enable Oracle ASM File Access Control. For information about disk group compatibility attributes, see "Disk Group Compatibility" on page 4-33.

The disk group attributes that control Oracle ASM File Access Control are the following:

- `ACCESS_CONTROL.ENABLED`

  This attribute determines whether Oracle ASM File Access Control is enabled for a disk group.

  The value can be `true` or `false`. The default is `false`.

  If the attribute is set to `true`, accessing Oracle ASM files is subject to access control. If `false`, any user can access every file in the disk group. All other operations behave independently of this attribute.

- `ACCESS_CONTROL.UMASK`

  This attribute determines which permissions are masked out on the creation of an Oracle ASM file for the user that owns the file, users in the same user group, and others not in the user group. This attribute applies to all files on a disk group.

  The values can be combinations of three digits {0 | 2 | 6} {0 | 2 | 6} {0 | 2 | 6}. The default is `066`.

  Setting to `0` masks out nothing. Setting to `2` masks out write permission. Setting to `6` masks out both read and write permissions.

  Before setting the `ACCESS_CONTROL.UMASK` disk group attribute, you must set the `ACCESS_CONTROL.ENABLED` attribute to `true` to enable Oracle ASM File Access Control.

Example 4–11 shows how to enable Oracle ASM File Access Control for a disk group using SQL*Plus. In this example, the permissions setting is `026` which enables

read-write access for the owner, read access for users in the group, and no access to others not in the group

Example 4–11   Setting up Oracle ASM File Access Control

```
ALTER DISKGROUP data1 SET ATTRIBUTE 'access_control.enabled' = 'true';
ALTER DISKGROUP data1 SET ATTRIBUTE 'access_control.umask' = '026';
```

Using SQL Statements to Manage Oracle ASM File Access Control

You can use the `ALTER DISKGROUP` SQL statement to manage file access control for Oracle ASM disk groups. These SQL statements are available for both database and Oracle ASM instances.

The SQL statements that support disk group access control are:

- `ALTER DISKGROUP ADD USERGROUP ... WITH MEMBER`

  Adds an Oracle ASM user group to a disk group. The user group name is limited to a maximum of 30 characters. The databases identified in the `MEMBER` clause must be in the disk group, as shown by `V$ASM_USER`, or the command returns an error. Any users authenticated as `SYSASM` or `SYSDBA` can create new user groups. For example:

  ```
  SQL> SELECT group_number, os_name FROM V$ASM_USER;

  GROUP_NUMBER OS_NAME
  ------------ ----------------------------------------------------------------
             1 oracle1
             1 oracle2
  ...

  SQL> ALTER DISKGROUP data ADD USERGROUP 'test_grp1'
       WITH MEMBER 'oracle1','oracle2';
  ```

- `ALTER DISKGROUP DROP USERGROUP`

  Drops an Oracle ASM user group from a disk group. Dropping a group might leave some files without a valid group. For those files to have a valid group, you must manually update the group associated with those files to a valid group.

  ```
  SQL> ALTER DISKGROUP data DROP USERGROUP 'test_grp1';
  ```

- `ALTER DISKGROUP MODIFY USERGROUP ADD MEMBER`

  Adds users to the specified user group. The users must be in the disk group, as shown by `V$ASM_USER`, or the command returns an error. Only the creator of the group or the Oracle ASM administrator can modify group membership.

  ```
  SQL> ALTER DISKGROUP data MODIFY USERGROUP 'test_grp2' ADD MEMBER 'oracle2';
  ```

- `ALTER DISKGROUP MODIFY USERGROUP DROP MEMBER`

  Removes users from the specified user group. If a member is not in the user group, then an error is returned. Only the creator of the group or the Oracle ASM administrator can modify group membership.

  ```
  SQL> ALTER DISKGROUP data MODIFY USERGROUP 'test_grp2' DROP MEMBER 'oracle2';
  ```

- `ALTER DISKGROUP ADD USER`

  Adds operating system (OS) users to an Oracle ASM disk group, so that these users can have access privileges on the disk group. The users must be existing

operating system users, and their user names must have a corresponding operating system user ID or system ID. If a user exists in the disk group, as shown by `V$ASM_USER`, then the command records an error and continues to add other users, if any.

The operating system user of a running database instance is automatically added to a disk group when the database instance accesses that disk group and creates files. However, for a database instance to read files in a disk group without creating any files, then you must use the `ADD USER` clause to add that database user to the disk group. Also, you can use this clause to add a database user to an existing disk group immediately after setting the Oracle ASM File Access Control disk group attributes and before creating new files.

```
SQL>  ALTER DISKGROUP DATA ADD USER 'oracle1';
```

- `ALTER DISKGROUP DROP USER`

    Drops operating system users from an Oracle ASM disk group. If a user is not in the disk group, then this command records an error and continues to drop other users, if any.

    If the user owns any files on the same Oracle ASM disk group, then this command fails with an error, unless the `CASCADE` keyword is specified. If the latter case, then the user is deleted, along with all the files that the user owns.

    If any files owned by the user are currently open, then the `DROP USER` command fails, and no files are deleted.

```
SQL>  ALTER DISKGROUP DATA DROP USER 'oracle1';
```

- `ALTER DISKGROUP SET PERMISSION`

    Modifies permissions of an Oracle ASM file. Setting `read only` permission to a file that has `read write` permission revokes the `write` permission. Only the file owner or the Oracle ASM administrator can change the permissions of a file. You cannot change the permissions on an open file.

```
SQL> ALTER DISKGROUP data SET PERMISSION OWNER=read write, GROUP=read only,
     OTHER=none FOR FILE '+data/controlfile.f';
```

- `ALTER DISKGROUP SET OWNERSHIP`

    Changes the owner or group of a file to the specified user or user group name, respectively. If the specified user or user group name does not exist, this command fails with an error. Only the owner of the file or the Oracle ASM administrator can run this command, and only the Oracle ASM administrator can change the owner. Also, the user group name must exist, and the owner of the file must be a member of that group. You cannot change the ownership of an open file.

```
SQL> ALTER DISKGROUP data SET OWNERSHIP OWNER='oracle1', GROUP='test_grp1'
     FOR FILE '+data/controlfile.f';
```

> **See Also:** The `ALTER DISKGROUP` SQL statement in the *Oracle Database SQL Language Reference*

Mounting and Dismounting Disk Groups

Disk groups that are specified in the `ASM_DISKGROUPS` initialization parameter are mounted automatically at Oracle ASM instance startup. This makes them available to all database instances running on the same node as Oracle ASM. The disk groups are dismounted at Oracle ASM instance shutdown. Oracle ASM also automatically

mounts a disk group when you initially create it, and dismounts a disk group if you drop it.

When a disk group is mounted, a disk group number is chosen. This number may change across disk group mounts. A disk group number is not recorded in any persistent structure, but the current value can be viewed in the `GROUP_NUMBER` column of the V$ASM views.

When you want to mount or dismount disk groups manually, use the `ALTER DISKGROUP...MOUNT` or `ALTER DISKGROUP...DISMOUNT` statement. You can mount or dismount disk groups by name, or specify `ALL`.

If you try to dismount a disk group that contains open files, the statement fails, unless you also specify the `FORCE` clause.

In a clustered Oracle ASM environment in `RESTRICTED` mode, a disk group is mounted in single-instance exclusive mode. No other Oracle ASM instance in that cluster can mount that disk group. In this mode the disk group is not usable by any Oracle ASM client. Use this mode to perform a fast rebalance.

The following SQL statement dismounts all disk groups that are currently mounted to the Oracle ASM instance:

```
ALTER DISKGROUP ALL DISMOUNT;
```

The following SQL statement mounts disk group `data1`:

```
ALTER DISKGROUP data1 MOUNT;
```

Mounting Disk Groups Using the FORCE Option

Oracle ASM provides a `MOUNT FORCE` option with `ALTER DISKGROUP` to enable Oracle ASM disk groups to be mounted in normal or high redundancy modes even though some Oracle ASM disks may be unavailable to the disk group at mount time. The default behavior without the `FORCE` option is to fail to mount a disk group that has damaged or missing disks.

The `MOUNT FORCE` option is useful in situations where a disk is temporarily unavailable and you want to mount the disk group with reduced redundancy while you correct the situation that caused the outage.

To successfully mount with the `MOUNT FORCE` option, Oracle ASM must be able to find at least one copy of the extents for all of the files in the disk group. In this case, Oracle ASM can successfully mount the disk group, but with potentially reduced redundancy.

The disks that Oracle ASM cannot access are placed in an offline mode. Oracle ASM then begins timing the period that these disks are in an offline mode. If the disk offline time period exceeds the timer threshold set by `DISK_REPAIR_TIME` disk group attribute, then those disks are permanently dropped from the disk group. You can change the offline timer after a disk is put in an offline state by using the `ALTER DISKGROUP OFFLINE` statement. For more information about setting the `DISK_REPAIR_TIME` disk group attribute, see "Oracle ASM Fast Mirror Resync" on page 4-28.

> **Note:** An Oracle ASM instance mounts an incomplete disk group differently depending on the specified compatibility. See "Features Enabled By Disk Group Compatibility Attribute Settings" on page 4-38.

In clustered Oracle ASM environments, if an Oracle ASM instance is not the first instance to mount the disk group, then using the MOUNT FORCE statement fails. This is because the disks have been accessed by another instance and the disks are not locally accessible.

If all disks are available, then using the FORCE option causes the MOUNT command to fail. This discourages unnecessary and improper use of the feature.

The following example shows how to use the FORCE option to force the mount of the data1 disk group:

```
ALTER DISKGROUP data1 MOUNT FORCE
```

> **See Also:** The *Oracle Database SQL Language Reference* for additional information about the ALTER DISKGROUP statement and the FORCE option

Checking the Internal Consistency of Disk Group Metadata

You can check the internal consistency of disk group metadata using the ALTER DISKGROUP statement with the CHECK keyword. You can use this statement to check specific files in a disk group, specific disks or all disks in a disk group, or specific failure groups within a disk group. The disk group must be mounted to perform these checks.

By default, the CHECK DISK GROUP clause verifies all of the metadata directories. Oracle ASM displays summary errors and writes the details about the errors in an alert log. The CHECK keyword performs the following operations:

- Verifies the consistency of the disk

- Cross checks all of the file extent maps and allocation tables for consistency

- Checks that the alias metadata directory and file directory are linked correctly

- Verifies that the alias directory tree is linked correctly

- Checks that Oracle ASM metadata directories do not have unreachable allocated blocks

The REPAIR | NOREPAIR clause specifies whether Oracle ASM should attempt to repair errors that are found during the check. The default is NOREPAIR. Use the NOREPAIR clause to receive alerts about inconsistencies and to suppress Oracle ASM from resolving the errors automatically. The following example statement checks for consistency in the metadata for all disks in the data1 disk group:

```
ALTER DISKGROUP data1 CHECK ALL;
```

> **See Also:** The *Oracle Database SQL Language Reference* for additional information about the CHECK clause syntax

Dropping Disk Groups

The DROP DISKGROUP statement enables you to delete an Oracle ASM disk group and optionally, all of its files. You can specify the INCLUDING CONTENTS clause if you also want to delete all files that are contained in the disk group. The default is EXCLUDING CONTENTS, which provides syntactic consistency and prevents you from dropping the disk group if it has any contents

The Oracle ASM instance must be started and the disk group must be mounted with none of the disk group files open, in order for the DROP DISKGROUP statement to succeed. The statement does not return until the disk group has been dropped.

When you drop a disk group, Oracle ASM dismounts the disk group and removes the disk group name from the ASM_DISKGROUPS initialization parameter if a server parameter file is being used. If a text initialization parameter file is being used, and the disk group is mentioned in the ASM_DISKGROUPS initialization parameter, then you must remove the disk group name from the ASM_DISKGROUPS initialization parameter before the next time that you shut down and restart the Oracle ASM instance.

The following statement deletes data1:

```
DROP DISKGROUP data1;
```

After ensuring that none of the files contained in data1 are open, Oracle ASM rewrites the header of each disk in the disk group to remove Oracle ASM formatting information. The statement does not specify INCLUDING CONTENTS, so the drop operation fails if the disk group contains any files.

If an Oracle Automatic Storage Management Cluster File System (Oracle ACFS) file system is mounted on a volume contained in the disk group, then the file system must be dismounted. If the file system has been registered, then it must be deregistered. The INCLUDING CONTENTS clause must be used to drop this disk group. All data in the file system is destroyed. To view the volumes and mount paths associated with a disk group, you can query the V$ASM_VOLUME view. For an example of a query on the V$ASM_VOLUME view, see Example 6–16 on page 6-8. For information about deregistering and dismounting Oracle ACFS file systems, see "Deregistering, Dismounting, and Disabling Volumes and Oracle ACFS File Systems" on page 13-18.

If you cannot mount a disk group but must drop it, you can use the FORCE option of the DROP DISKGROUP statement. This command enables you to remove the headers on disks that belong to a disk group that cannot be mounted by any Oracle ASM instances as shown in the following example:

```
DROP DISKGROUP data1 FORCE
```

The disk group on which you perform this operation should not be mounted anywhere in the cluster. When you use the FORCE option, the Oracle ASM instance does not attempt to verify that a disk group is being used by another Oracle ASM instance in the same storage subsystem.

> **Note:** Use the FORCE option with extreme caution.

You can also drop a disk group with Oracle Enterprise Manager. See "Dropping Disk Groups" on page 9-12.

Renaming Disks Groups

The renamedg tool enables you to change the name of a cloned disk group. The disk group must be dismounted on all nodes in the cluster before running renamedg on the disk group.

renamedg renames a disk group using a two-step process:

1. Phase one

    This phase generates a configuration file to be used in phase two.

2. Phase two

This phase uses the configuration file to perform the renaming of the disk group.

The syntax is:

```
renamedg {-help | help=true}

renamedg
    [phase={ one | two | both } ] dgname=diskgroup
    newdgname=newdiskgroup [config=configfile]
    [ asm_diskstring=discoverystring, discoverystring ... ]
    [ clean={true | false} ] [ check={true | false} ]
    [ confirm={true | false}] [ verbose={ true | false} ]
    [ keep_voting_files={true | false}]
```

- `phase={one | two | both}`

  Specifies the phase to be run. Allowed values are `one`, `two`, or `both`. This argument is optional. The default is `both`.

  Typically you would run both phases. If a problem occurs during the second phase, then you can re-run phase `two` using the generated configuration file.

- `dgname=diskgroup`

  Specifies the name of the disk group that to be renamed.

- `newdgname=newdiskgroup`

  Specifies the new name for the disk group.

- `config=configfile`

  Specifies the path to the configuration file to be generated during phase one or specifies the path to the configuration file to be used during phase two.

  This argument is optional. The default configuration file is named `renamedg_config` and is located in the directory in which the command is run. The single quotations may be required on some platforms.

- `asm_diskstring=discoverystring, discoverystring ...`

  Specifies the Oracle ASM discovery strings. The `asm_diskstring` value must be specified if the Oracle ASM disks are not in the default location for the platform. The single quotations may be required on some platforms, usually when wildcard characters are specified.

- `clean={true | false}`

  Specifies whether to tolerate errors that are otherwise ignored. The default is `true`.

- `check={true | false}`

  Specifies a boolean value that is used in the second phase. If `true`, then the tool prints the list of changes that are to be made to the disks. No writes are issued. It is an optional parameter that defaults to `false`.

- `confirm={true | false}`

  Specifies a boolean value that is used in the second phase. If `false`, then the tool prints the changes that are to be made and seeks confirmation before actually making the changes. It is an optional value that defaults to `false`. If check is set to `true`, then the value of this parameter is redundant.

- `verbose={true | false}`

Specifies verbose execution when `verbose=true`. The default is `false`.

- `keep_voting_files={true|false}`

  Specifies whether voting files are kept in the renamed disk group. The default is `false` which deletes the voting files from the renamed disk group.

  > **Note:** `renamedg` does not update resources, nor does `renamedg` update any file references within the database. Because of this behavior, the original disk group resource is not automatically deleted after the completion of phase `two`. The status of the old disk group resource can be checked with the Oracle Clusterware Control (CRSCTL) `crsctl stat res -t` command and then manually deleted with the Server Control Utility (SRVCTL) `srvctl remove diskgroup` command.

Example 4–12 shows several examples of the use of `renamedg`. The first example renames the `data` disk group to `new_data` using a disk string to locate the disks and the `verbose` option is enabled. The second example only creates a configuration file during the completion of phase `one` of the `renamedg` operation. The third example runs phase `two` of the `renamedg` operation using a configuration file generated from a phase `one` execution of `renamedg`.

Example 4–12   Using renamedg

```
$ renamedg dgname=data newdgname=new_data asm_diskstring='/devices/disk*'
      verbose=true

$ renamedg phase=one dgname=data newdgname=new_data
      asm_diskstring='/devices/disk*' config=/tmp/data2.conf verbose=true

$ renamedg phase=two dgname=data newdgname=new_data config=/tmp/data2.conf
      verbose=true
```

5

Introduction to Oracle ACFS

This chapter describes Oracle Automatic Storage Management Cluster File System (Oracle ACFS) concepts and provides an overview of Oracle ACFS features.

This chapter contains the following topics:

- Overview of Oracle ACFS
- Understanding Oracle ACFS Concepts
- Understanding Oracle ACFS Administration
- Overview of Oracle ASM Dynamic Volume Manager

> **See Also:**
> - "Views Containing Oracle ACFS Information" on page 6-6 for information about using views to obtain Oracle ACFS information
> - "Managing Volumes in a Disk Group" on page 4-13 for information about using SQL*Plus to administer volumes
> - Chapter 10, "Administering Oracle ACFS with Oracle Enterprise Manager" for information about using Oracle Enterprise Manager to administer Oracle ACFS
> - Chapter 11, "Oracle ASM Configuration Assistant" for information about using Oracle ASM Configuration Assistant to administer Oracle ACFS
> - "ASMCMD Volume Management Commands" on page 12-61 for information about volume management commands
> - Chapter 13, "Oracle ACFS Command-Line Tools" for information about Oracle ACFS operating system utilities

Overview of Oracle ACFS

Oracle Automatic Storage Management Cluster File System (Oracle ACFS) is a multi-platform, scalable file system, and storage management technology that extends Oracle Automatic Storage Management (Oracle ASM) functionality to support customer files maintained outside of Oracle Database. Oracle ACFS supports many database and application files, including executables, database trace files, database alert logs, application reports, BFILEs, and configuration files. Other supported files are video, audio, text, images, engineering drawings, and other general-purpose application file data.

Notes:

- Oracle ASM is the preferred storage manager for all database files. It has been specifically designed and optimized to provide the best performance for database file types. For a list of file types supported by Oracle ASM, see Table 7–1, " File types supported by Oracle ASM" on page 7-1.

- Oracle ACFS is the preferred file manager for non-database files. It is optimized for general purpose files.

- Oracle ACFS does not support any file type that can be directly stored in Oracle ASM, except where explicitly noted in the documentation.

  Not supported means Oracle Support Services does not take calls and development does not fix bugs associated with storing unsupported file types in Oracle ACFS.

- Starting with Oracle Automatic Storage Management 11*g* Release 2 (11.2.0.3), Oracle ACFS supports RMAN backups (`BACKUPSET` file type), archive logs (`ARCHIVELOG` file type), and Data Pump dumpsets (`DUMPSET` file type). Note that Oracle ACFS snapshots are not supported with these files.

- Oracle ACFS does not support files for the Oracle Grid Infrastructure home.

- Oracle ACFS does not support Oracle Cluster Registry (OCR) and voting files.

- Oracle ACFS functionality requires that the disk group compatibility attributes for `ASM` and `ADVM` be set to `11.2` or greater. For information about disk group compatibility, refer to "Disk Group Compatibility" on page 4-33.

An Oracle ACFS file system is a layer on Oracle ASM and is configured with Oracle ASM storage, as shown in Figure 5–1. Oracle ACFS leverages Oracle ASM functionality that enables:

- Oracle ACFS dynamic file system resizing

- Maximized performance through direct access to Oracle ASM disk group storage

- Balanced distribution of Oracle ACFS across Oracle ASM disk group storage for increased I/O parallelism

- Data reliability through Oracle ASM mirroring protection mechanisms

Figure 5–1   Oracle Automatic Storage Management Storage Layers
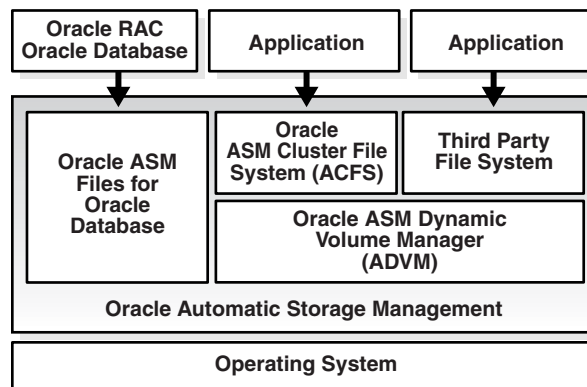
Oracle ACFS establishes and maintains communication with the Oracle ASM instance to participate in Oracle ASM state transitions including Oracle ASM instance and disk group status updates and disk group rebalancing. Oracle Automatic Storage Management with Oracle ACFS and Oracle ASM Dynamic Volume Manager (Oracle ADVM) delivers support for all customer data and presents a common set of Oracle storage management tools and services across multiple vendor platforms and operating system environments on both Oracle Restart (single-node) and cluster configurations. For an overview of Oracle ADVM, see "Overview of Oracle ASM Dynamic Volume Manager" on page 5-20.

Oracle ACFS is tightly coupled with Oracle Clusterware technology, participating directly in Clusterware cluster membership state transitions and in Oracle Clusterware resource-based high availability (HA) management. In addition, Oracle installation, configuration, verification, and management tools have been updated to support Oracle ACFS.

Oracle ACFS can be accessed and managed using native operating system file system tools and standard application programming interfaces (APIs). Oracle ACFS can be configured and managed using Oracle Enterprise Management tools. Oracle ACFS can be accessed using industry standard Network Attached Storage (NAS) File Access Protocols: Network File System (NFS) and Common Internet File System (CIFS).

In addition to sharing file data, Oracle ACFS provides additional storage management services including support for the Oracle Grid Infrastructure clusterwide mount registry, dynamic on-line file system resizing, and multiple space-efficient snapshots for each file system. For information about the mount registry, see "About the Oracle ACFS Mount Registry" on page 5-8.

Oracle ACFS contributes to the overall Oracle storage management by providing:

- A general-purpose single-node (standalone) and cluster file system solution that is integrated with Oracle ASM, Oracle Clusterware, and Oracle Enterprise Management technologies

- A common set of file system features across multiple vendor platforms and operating systems, offering an alternative to native operating system or third-party file system solutions

- Single-node and clusterwide shared Oracle Database homes and application data

- Uniform, coherent shared file access and clusterwide naming of all customer application files maintained outside of the database

Oracle ACFS accommodates large storage capacities and large numbers of cluster nodes. It efficiently manages large numbers of file systems, files, and supports both

small and large sized files with exabyte-capable file and file system capacities. Oracle ACFS provides optimized fast directory lookup for large directories with millions of files.

Oracle ACFS file systems are generally mounted on all Oracle Cluster Synchronization Services (CSS) cluster members. In the event of a member failure, another cluster member quickly recovers any outstanding metadata transactions on behalf of the failed member. Following recovery, access by other active cluster members and any remote client systems can resume.

See Chapter 6, "Using Views to Display Information" and Chapter 13, "Oracle ACFS Command-Line Tools".

Understanding Oracle ACFS Concepts

This section describes concepts for the key Oracle ACFS components and contains the following topics:

- About Oracle ACFS
- About Oracle ACFS and Oracle Database Homes
- About Oracle ASM Dynamic Volume Manager
- About the Oracle ACFS Driver Model
- About the Oracle ACFS Mount Model and Namespace
- About the Oracle ACFS Mount Registry
- About Oracle ACFS Snapshots
- About Oracle ACFS and Backup and Restore
- About Oracle ACFS Integration with Oracle ASM
- About Oracle ACFS and External Tables on Windows

About Oracle ACFS

Oracle ACFS is designed as a general-purpose, single-node and clusterwide file system that delivers support for customer file system application data maintained outside of Oracle Database. Users and applications can access and manage Oracle ACFS using native operating system file system application programming interfaces (APIs) and command-line interface (CLI) tools. Users can also manage Oracle ACFS with Oracle Enterprise Manager.

Oracle ACFS supports large files with 64-bit file and file system data structure sizes leading to exabyte capable file and file system capacities on 64 bit platforms. Variable extent-based storage allocation and high-performance directories contribute to fast performance and shared disk configurations that provide direct storage paths to Oracle ACFS file data from each cluster member. File system integrity and fast recovery is achieved with Oracle ACFS metadata checksums and journaling. Oracle ACFS is designed as a multi-node, shared file system model that delivers coherent, cached, direct storage paths to Oracle ACFS file data from each cluster member.

Oracle ACFS files systems are typically configured for clusterwide access. File systems, files, and directories are visible and accessible from all cluster members and can be referenced by users and applications using the same path names from any cluster member. This design enables simplified application deployments across cluster members and facilitates both multiple instance cluster applications and high availability (HA) failover of unmodified single-node applications.

Oracle ACFS presents single system file access semantics across cluster configurations. Applications and users on all cluster members are always presented with the same view of shared Oracle ACFS file data, supported by the Oracle ACFS clusterwide user and metadata cache coherency mechanism.

About Oracle ACFS and Oracle Database Homes

You can use an Oracle ACFS file system for an Oracle Database home file system for Oracle 11*g* Release 2 (11.2) or greater. The file system can be configured as any Oracle Database home, including a shared or non-shared Oracle Database home in Oracle RAC cluster configurations.

After the Oracle ACFS file system is created, the Oracle ACFS-based database home mount point location can be selected as the Oracle Database Home location by browsing to and then choosing the directory during the Oracle Universal Installer (OUI) Database Software installation.

> **Note:** Oracle Database data files are not supported on Oracle ACFS. These data files should be placed in Oracle ASM disk groups.

When installing Oracle Software, there must be a separate Oracle base (`ORACLE_BASE`) associated with each operating system user. For example, there should be a separate Oracle base for a grid user and a database user.

> **See Also:** *Oracle Database Installation Guide* for information about Optimal Flexible Architecture (OFA) recommendations for Oracle base and home directories

You can locate the Oracle Database base (`ORACLE_BASE` for database) directory and home (`ORACLE_HOME` for database) directory on an Oracle ACFS file system. The Oracle Database base (`ORACLE_BASE` for database) directory should not be the Oracle Grid Infrastructure base (`ORACLE_BASE` for grid) directory or should not be located under the Oracle Grid Infrastructure base directory (`ORACLE_BASE` for grid).

Note that the Oracle Grid Infrastructure base (`ORACLE_BASE` for grid) directory and home (`ORACLE_HOME` for grid) directory cannot be located on the Oracle ACFS file system because the Oracle ACFS file system cannot be created until Oracle Grid Infrastructure is installed.

One or more Oracle Database homes on Oracle ACFS can be created under the same mount point. Each Oracle Database home should be created using a separate Oracle ACFS file system located under the mount point.

After the installation of Grid Infrastructure Software and before the installation of the Oracle Database software with Oracle Universal Installer (OUI), you can create an Oracle ACFS file system to be configured for use as an Oracle Database Home.

You can also use the Oracle ASM Configuration Assistant (ASMCA) to create the Oracle file system or you can perform the general procedure for creating the file system. For information about using ASMCA, see "Creating an Oracle ACFS File System for a Database Home" on page 11-15. For information about using Oracle ACFS commands to create a file system, see Chapter 13, "Oracle ACFS Command-Line Tools".

> **Note:**   When an Oracle ACFS file system contains an Oracle database home or an Oracle database uses the file system for any file storage, the file system must have an individual file system resource, rather than being listed in the Oracle ACFS mount registry. You must use Server Control Utility (SRVCTL) commands to administer that file system. For information about SRVCTL, see *Oracle Real Application Clusters Administration and Deployment Guide*.

In an Oracle Grid Infrastructure clusterware configuration, run `srvctl add filesystem` to enable a file system to be automounted when an Oracle Database home is installed on the Oracle ACFS file system. That file system should not be added to the Oracle ACFS mount registry. The database owner must be specified with the `-u` option to allow that owner to mount and dismount the file system. Root privilege is required when using `srvctl add filesystem` to add a file system on Linux.

The `srvctl start filesystem` command is used to manually mount the Oracle ACFS file system.

Oracle ACFS file systems can be also configured for use as application homes. However, Oracle ACFS file systems cannot be used for an Oracle base directory or an Oracle Grid Infrastructure home that contains the software for Oracle Clusterware, Oracle ASM, Oracle ACFS, and Oracle ADVM components.

To reduce contention on an Oracle ACFS file system in an Oracle RAC environment where the Oracle Database home is shared on Oracle ACFS, Oracle Database auditing operating system files should be configured as node specific. For a node-specific setup, you must ensure that the `AUDIT_FILE_DEST` initialization parameter in the configuration file of each database instance points to a unique location rather than one location for all the database instances.

For example, if you have a database with the Oracle name set to `TEST` and you want to ensure that the location of `AUDIT_FILE_DEST` initialization parameter for database instance `TEST1` points to a node specific location for that instance, you can run the following SQL statement on database instance `TEST1`:

```
SQL> ALTER SYSTEM SET AUDIT_FILE_DEST='$ORACLE_BASE/admin/TEST1/adump'
     SCOPE=SPFILE SID='TEST1';
```

For other database instances, such as `TEST2` or `TEST3`, you can run one of the following statements on the appropriate database instance:

```
SQL> ALTER SYSTEM SET AUDIT_FILE_DEST='$ORACLE_BASE/admin/TEST2/adump'
     SCOPE=SPFILE SID='TEST2';
```

```
SQL> ALTER SYSTEM SET AUDIT_FILE_DEST='$ORACLE_BASE/admin/TEST3/adump'
     SCOPE=SPFILE SID='TEST3';
```

In the previous examples, `ORACLE_BASE` has been set to `/u01/app/acfsmounts` and that value could have been used in place of the `ORACLE_BASE` variable.

See Also:

- *Oracle Database Installation Guide* for your environment for information about the installation of Oracle Database software and storage options

- *Oracle Grid Infrastructure Installation Guide* for your environment for information about the installation of Oracle Database software and storage options

- *Oracle Database Administrator's Guide* for information about using Oracle Managed files

- *Oracle Database Reference* for information about the AUDIT_FILE_ DEST initialization parameter

About Oracle ASM Dynamic Volume Manager

The Oracle ASM Dynamic Volume Manager (Oracle ADVM) provides volume management services and a standard disk device driver interface to clients. File systems and other disk-based applications send I/O requests to Oracle ADVM volume devices as they would to other storage devices on a vendor operating system.

For more information about Oracle ADVM, see "Overview of Oracle ASM Dynamic Volume Manager" on page 5-20.

About the Oracle ACFS Driver Model

An Oracle ACFS file system is installed as a dynamically loadable vendor operating system (OS) file system driver and tool set that is developed for each supported operating system platform. The driver is implemented as a Virtual File System (VFS) and processes all file and directory operations directed to a specific file system.

> **Note:** Errors encountered by the drivers are written to the native operating system console and system event loggers. See "Understanding Oracle ACFS I/O Failure Console Messages" on page B-6.

About the Oracle ACFS Mount Model and Namespace

Oracle ACFS is designed as a hierarchical file system containing files and subdirectories organized into a tree-structured namespace with files at the leaf nodes. The namespace design is a single-file system naming model for both single-node and cluster configurations. This design enables each cluster member to present shared files to cluster applications using the same path names, simplifying multi-node application and user access, and overall file system administration. The Oracle ACFS mount model also accommodates node local mounts and cluster node subset mounts in cluster configurations to accommodate additional customer requirements.

With a primary focus upon the support of customer application files, Oracle ACFS is not for use as root file system of an operating system or boot storage. Otherwise, an Oracle ACFS file system may be mounted into the native operating system file system namespace using either the mount command line tool or an Oracle Enterprise Management tool.

About the Oracle ACFS Mount Registry

The Oracle ACFS mount registry supports both Oracle Restart and Oracle Grid Infrastructure cluster configurations. File systems that are to be mounted persistently (across reboots) can be registered with the Oracle ACFS mount registry. In cluster configurations, registered Oracle ACFS file systems are automatically mounted by the mount registry, similar to a clusterwide mount table. However, in Oracle Restart configurations the automatic mounting of registered Oracle ACFS file systems is not supported. For more information, see "Oracle ACFS and Oracle Restart" on page B-6.

By default, an Oracle ACFS file system that is inserted into the cluster mount registry is automatically mounted on all cluster members, including cluster members that are added after the registry addition. However, the cluster mount registry also accommodates single-node and multi-node (subset of cluster nodes) file system registrations. The mount registry actions for each cluster member mount only registered file systems that have been designated for mounting on that member.

The Oracle ACFS registry resource actions are designed to automatically mount a file system only one time for each Oracle Grid Infrastructure initialization to avoid potential conflicts with administrative actions to dismount a given file system.

For information about registering an Oracle ACFS file system using the `acfsutil` command, see "acfsutil registry" on page 13-104.

> **Note:** Do not register an Oracle ACFS file system that has had an individual file system Cluster Ready Services (CRS) resource added. For example, do not register a file system with `acfsutil registry` if the Oracle ACFS file system has been registered using `srvctl add filesystem`.

About Oracle ACFS Snapshots

An Oracle ACFS snapshot is an online, read-only or read-write, point in time copy of an Oracle ACFS file system. The snapshot copy is space-efficient and uses Copy-On-Write functionality. Before an Oracle ACFS file extent is modified or deleted, its current value is copied to the snapshot to maintain the point-in-time view of the file system.

Oracle ACFS snapshots are immediately available for use after they are created. The snapshots are created in the `.ACFS/snaps/` directory of the file system. They are always online while the file system is mounted. Consequently, an Oracle ACFS snapshot can support the online recovery of files inadvertently modified or deleted from a file system. With up to a total of 63 read-only, read-write, or combination of read-only and read-write snapshot views supported for each file system, flexible online file recovery solutions spanning multiple views can be employed. An Oracle ACFS snapshot can also be used as the source of a file system backup, as it can be created on demand to deliver a current, consistent, online view of an active file system.

Oracle ACFS read-write snapshots enable fast creation of an snapshot image that can be both read and written without impacting the state of the Oracle ACFS file system hosting the snapshot images. The read-write snapshots can be used for:

- Testing of new versions of application software on production file data reflected in the read-write snapshot image without modifying the original production file system

- Running test scenarios on a real data set without modifying the original production file system

To use Oracle ACFS read-write snapshots, the disk group compatibility attribute for `ADVM` must be set to `11.2.0.3.0` or higher. If you create a read-write snapshot on an existing Oracle ACFS file system from a version earlier than 11.2.0.3.0, then the file system is updated to the 11.2.0.3.0 format. After a file system has been updated to a higher version, a file system cannot be returned to an earlier version.

Oracle ACFS snapshot storage is maintained within the file system, eliminating the management of separate storage pools for file systems and snapshots. Oracle ACFS file systems can be dynamically resized to accommodate additional file and snapshot storage requirements.

You cannot modify security or encryption metadata in read-only snapshots. After a snapshot of a file system is taken, you can only enable or disable security or encryption in a read-write snapshot. No other alteration is permitted on Oracle ACFS security or encryption metadata in a snapshot. If a file was not secured by a security realm in the snapshot, it cannot be realm secured by adding the corresponding file in the active file system to a security realm. If a file was not encrypted in the snapshot, that file cannot be encrypted by encrypting the corresponding file in the active file system.

A new file created in a realm-secured directory in a read-write snapshot inherits the realm security attributes of the parent directory. If the realm protecting the new file has encryption turned on, the file is encrypted with the encryption parameters set in the realm. If the realm protecting the new file has encryption turned off, the file is decrypted. Files and directories in a read-write snapshot cannot be added to or removed from any security realm.

Files in a read-write snapshot can be encrypted, decrypted, or rekeyed if the operation target is a path specified for a file or directory of the read-write snapshot. However, if an encryption, decryption, or rekey operation is specified at the file system level, then the operation does not process files and directories of snapshots in the `.ACFS/snaps/` directory.

For information about Oracle ACFS security, refer to "Oracle ACFS Security" on page 5-13. For information about Oracle ACFS encryption, refer to "Oracle ACFS Encryption" on page 5-15.

Oracle ACFS snapshots are administered with the `acfsutil snap` commands. For information about the `acfsutil snap` commands, refer to "acfsutil snap create" on page 13-108, "acfsutil snap delete" on page 13-109, and "acfsutil snap info" on page 13-110.

You can also manage Oracle ACFS snapshots with Oracle Enterprise Manager. For information about using Oracle Enterprise Manager, see "Managing Oracle ACFS Snapshots with Oracle Enterprise Manager" on page 10-5.

> **Notes:**
>
> ■ The `link()` and `rename()` system calls fail if an attempt is made to link or rename a file in the Oracle ACFS file system and a file in any associated read-write snapshot, or vice versa. Any tools which use the `link()` and `rename()` system calls, such as `ln` and `mv`, also fail in the same scenario.
>
> ■ Oracle ACFS snapshots have not been optimized for use with database data files.

About Oracle ACFS and Backup and Restore

Oracle ACFS runs on operating system platforms as a native file system technology supporting native operating system file system application programming interfaces (APIs). Consequently, backup applications that access files using the native operating system file system interfaces are able to access and backup Oracle ACFS file systems and other native operating system file systems. Oracle ACFS snapshots can be dynamically created and used to present a consistent, on-line view of an active file system to a backup application.

Backup applications that use interfaces other than the standard operating system interfaces (read or write) are not supported with Oracle ACFS. For example, Windows backup applications that depend upon the presence of reparse points or the Windows Volume Shadow Copy Service (VSS) are not supported.

About Oracle ACFS Integration with Oracle ASM

Oracle ACFS is always configured with Oracle ASM storage and interfaces with Oracle ASM storage through a traditional device file. This device file is presented by the Oracle ASM Dynamic Volume Manager (Oracle ADVM) and is constructed using a dynamic volume file. The Oracle ADVM volume device file is created automatically following the creation of an Oracle ADVM volume. An Oracle ACFS file system is then bound to the Oracle ADVM device file during the file system creation. After an Oracle ACFS is configured and mounted, the file system inherits the Oracle ASM storage management features associated with an Oracle ADVM volume, including dynamic balanced distribution, mirroring and striping, and dynamic resizing.

The Oracle ACFS driver establishes communication with the Oracle ASM instance to receive Oracle ASM status information including Oracle ASM instance and disk group state transitions.

For information about Oracle ACFS and Oracle ASM operations, see "Oracle ACFS and Dismount or Shutdown Operations" on page 5-13.

About Oracle ACFS and External Tables on Windows

To access an external table stored on an Oracle ACFS file system on Windows, the external table must be created with the `DISABLE_DIRECTORY_LINK_CHECK` access parameter.

> **See Also:**
>
> - *Oracle Database Utilities* for information about the `DISABLE_DIRECTORY_LINK_CHECK` access parameter
> - *Oracle Database SQL Language Reference* for information about creating external tables

Understanding Oracle ACFS Administration

This section describes Oracle ACFS administration and contains the following topics:

- Oracle ACFS and File Access and Administration Security
- Oracle ACFS and Grid Infrastructure Installation
- Oracle ACFS and Grid Infrastructure Configuration
- Clusterware Resources and Oracle ACFS Administration
- Oracle ACFS and Dismount or Shutdown Operations

- Oracle ACFS Security
- Oracle ACFS Encryption
- Oracle ACFS Replication
- Oracle ACFS Tagging

Oracle ACFS and File Access and Administration Security

Oracle ACFS supports both traditional UNIX-style file access control classes (user, group, other) for Linux and UNIX platforms and the Windows Security Model including file access control lists (ACLs) for Windows platforms. Most Oracle ACFS administrative actions are performed by users with either root or Oracle ASM administration privileges for Linux and UNIX platforms and by users with Windows Administrative privileges on Windows platforms. General Oracle ACFS information for file systems can be accessed by any system user.

In support of Oracle ACFS administration, Oracle recommends that the Oracle ASM administrator role is given to a root privileged user, as many common Oracle ACFS file system management tasks including mount, umount, fsck, driver load, and driver unload are root privileged operations. Other privileged Oracle ACFS file system operations that do not require root privileges can be performed by the Oracle ASM administrator. If the Oracle ASM administrator role is not given to a root privileged user, access to Oracle ACFS file systems can be restricted with the `norootsuid` and `nodev` mount options.

Additional fine grain access control is provided for Oracle ACFS file systems with the security infrastructure feature. For information about Oracle ACFS security infrastructure, refer to "Oracle ACFS Security" on page 5-13. For information about Oracle ACFS encryption, refer to "Oracle ACFS Encryption" on page 5-15.

For information about Oracle ASM privileges, see "About Privileges for Oracle ASM" on page 3-26. For information about administering Oracle ACFS, see Chapter 10, "Administering Oracle ACFS with Oracle Enterprise Manager" and Chapter 13, "Oracle ACFS Command-Line Tools".

Oracle ACFS and Grid Infrastructure Installation

Oracle Grid Infrastructure includes Oracle Clusterware, Oracle ASM, Oracle ACFS, Oracle ADVM, and driver resources software components, which are installed into the Grid Infrastructure Home using the Oracle Universal Installation (OUI) tool.

For information about Oracle ACFS and Oracle Clusterware resources, see "Clusterware Resources and Oracle ACFS Administration" on page 5-12.

Oracle ACFS and Grid Infrastructure Configuration

After a Grid Infrastructure installation and Oracle Clusterware is operational, you can use Oracle ASM Configuration Assistant (ASMCA) to start the Oracle ASM instance and create Oracle ASM disk groups, Oracle ADVM volumes, and Oracle ACFS file systems. Alternatively, Oracle ASM disk groups and Oracle ADVM volumes can be created using SQL*Plus, ASMCMD command line tools, or Oracle Enterprise Manager. File systems can be created using operating system command-line tools or Oracle Enterprise Manager.

Oracle ACFS file systems are configured with Oracle ADVM based operating system storage devices that are created automatically following the creation of an Oracle ADVM dynamic volume file. After a volume file and its associated volume device file

are created, a file system can be created and bound to that operating system storage device. Following creation, an Oracle ACFS file system can be mounted, after which it is accessible to authorized users and applications executing file and file system operations.

For an example of the specific actions required to create a file system, see "Basic Steps to Manage Oracle ACFS Systems" on page 13-1. For information about managing Oracle ACFS file systems with ASMCA, see "Managing Oracle ACFS File Systems with Oracle ASM Configuration Assistant" on page 11-11. For information about managing Oracle ACFS file systems with Oracle Enterprise Manager, see "Managing Oracle ACFS with Oracle Enterprise Manager" on page 10-1.

Clusterware Resources and Oracle ACFS Administration

Oracle Clusterware resources support Oracle ACFS, Oracle Kernel Services Driver (OKS), Oracle ADVM startup, the Oracle ACFS cluster mount registry, and Oracle ACFS single file system startup, shutdown, and steady-state actions.

The following list summarizes Oracle ACFS resource-based management.

■   The Oracle ACFS, Oracle Kernel Services (OKS), and Oracle ADVM drivers are dynamically loaded when the Oracle ASM instance is started.

   –   Oracle ACFS

       This driver processes all Oracle ACFS file and directory operations.

   –   Oracle ADVM

       This driver provides block device services for Oracle ASM volume files that are used by file systems for creating file systems.

   –   Oracle Kernel Services Driver (OKS)

       This driver provides portable driver services for memory allocation, synchronization primitives, and distributed locking services to Oracle ACFS and Oracle ADVM.

   The drivers are managed as a single resource set.

   For information, see "Oracle ACFS Drivers Resource Management" on page B-3.

■   Oracle ACFS file systems listed in the Oracle ACFS mount registry are automatically mounted during Grid Infrastructure initialization and as new mount registry entries are created.

   The registry resource is used to manage activation of the Oracle ACFS mount registry and to support the mount and dismount actions for Oracle ACFS file systems listed in the Oracle ACFS mount registry.

   For information, see "Oracle ACFS Registry Resource Management" on page B-4.

■   Individual file systems are either manually mounted or dismounted using an Oracle ACFS or Oracle Clusterware command-line tool, or automatically mounted or dismounted based on a resource dependency action.

   For example, a file system hosting an Oracle Database home can be named in the dependency list of the associated Oracle Database resource such that issuing a start on the database resource results in mounting the dependent Oracle ACFS hosted database home file system.

   For information, see "Oracle ACFS Individual File System Resource Management" on page B-5.

Oracle ACFS and Dismount or Shutdown Operations

It is important to dismount any active file system configured with an Oracle ADVM volume device file before an Oracle ASM instance is shutdown or a disk group is dismounted. After the file systems are dismounted, all open references to Oracle ASM files are removed and associated disk groups can be dismounted or the instance shut down.

If the Oracle ASM instance or disk group is forcibly shut down or fails while an associated Oracle ACFS is active, the file system is placed into an offline error state. If any file systems are currently mounted on Oracle ADVM volume files, the `SHUTDOWN ABORT` command should not be used to terminate the Oracle ASM instance without first dismounting those file systems. Otherwise, applications encounter I/O errors and Oracle ACFS user data and metadata being written at the time of the termination may not be flushed to storage before the Oracle ASM storage is fenced. If it is not possible to dismount the file system, then you should run two `sync` (1) commands to flush cached file system data and metadata to persistent storage before issuing the `SHUTDOWN ABORT` operation.

Any subsequent attempt to access an offline file system returns an error. Recovering a file system from that state requires dismounting and remounting the Oracle ACFS file system. Dismounting an active file system, even one that is offline, requires stopping all applications using the file system, including any shell references. For example, a previous change directory (`cd`) into a file system directory. The Linux `fuser` or `lsof` commands or Windows `handle` command list information about processes and open files.

For information about shutting down an Oracle ASM instance, see "Shutting Down an Oracle ASM Instance" on page 3-16. For information about dismounting a disk group, see "Mounting and Dismounting Disk Groups" on page 4-44.

Oracle ACFS Security

Oracle ACFS security provides realm-based security for Oracle ACFS file systems, enabling you to create realms to specify security policies for users and groups to determine access on file system objects. This security feature provides a finer-grained access control on top of the access control provided by the operating system. Oracle ACFS security can use the encryption feature to protect the contents of realm-secured files stored in Oracle ACFS file systems.

Oracle ACFS security uses realms, rules, rule sets, and command rules to enforce security policies.

- An Oracle ACFS security realm is a group of files or directories that are secured for access by a user or a group of users. Realms are defined with rule sets which contain groups of rules that apply fine grain access control. Oracle ACFS security realms can also be used as containers to enable encryption.

- Oracle ACFS security rules are Boolean expressions that evaluate to true or false based on a system parameter on which the rule is based.

- Oracle ACFS rule sets are collection of rules. Rule sets evaluate to `TRUE` or `FALSE` based on the evaluation of the rules a rule set contains.

- Oracle ACFS command rules are associations of the file system operation to a rule set. For example, the association of a file system create, delete, or rename operation to a rule set. Command rules are associated with an Oracle ACFS realm.

An existing operating system user must be designated as the first Oracle ACFS security administrator and an existing operating system group must be designated as

the security administrator `admin` group. Security administrators must be members of the designated security group. Additional users can be designated as security administrators. An Oracle ACFS security administrator can manage encryption for an Oracle ACFS file system on a per-realm basis. An Oracle ACFS security administrator is authenticated for security operations with a security realm password, not the operating system password of the user.

The first security administrator is created during the initialization of Oracle ACFS security with the `acfsutil sec init` command which is run by the root user. When the first security administrator is created, the administrator is assigned a password that can be changed by the administrator. The security realm passwords for administrators are stored in a location created during the security initialization process. Each time a security administrator runs an `acfsutil sec` command, the administrator is prompted for the security password.

Auditing and diagnostic data are logged for Oracle ACFS security. The log files include information such as `acfsutil` commands that have been run, the use of security or system administrator privileges, and runtime failures such as realm check authorization failures. Logs are written to the following files:

- `mount_point`/.Security/realm/logs/sec-`host_name`.log

  The directory is created with `acfsutil sec prepare` command and protected by Oracle ACFS security. Refer to "acfsutil sec prepare" on page 13-73.

- `GRID_HOME`/log/`host_name`/acfssec/acfssec.log

  The messages that are logged to this file are for commands that are not associated with a specific file system, such as `acfsutil sec init`. The directory is created during installation and is owned by the root user.

When an active log file grows to a pre-defined maximum size (10 MB), the file is automatically moved to `log_file_name`.bak, the administrator is notified, and logging continues to the regular log file name. When the administrator is notified, the administrator must archive and remove the `log_file_name`.bak file. If an active log file grows to the maximum size and the `log_file_name`.bak file exists, logging stops until the backup file is removed. After the backup log file is removed, logging restarts automatically.

Oracle ACFS security protects the following objects from unauthorized accesses:

- Realm-secured directories and user files

  The directories and files reside on a file system secured by Oracle ACFS security.

- The Oracle ACFS security directory (`mount_point`/.Security) and its contents

  The security directory contains the log files in plain-text format and a security metadata backup file in XML format. The log files generated by Oracle ACFS security can only be accessed by valid Oracle ACFS security administrators.

- Oracle ACFS security objects

  These objects are the security realms, rules, and rule sets used to manage Oracle ACFS security.

Access to files in a security realm of an Oracle ACFS file system must be authorized by both the security realm and the underlying operating system permissions, such as (owner, group, other) permissions on Linux and Access Control Lists (ACLs) on Windows. Each access to a realm-secured file is first checked for security realm authorization. If the access is authorized by the security realm, then access to the files is checked by the underlying operating system access control checks. If both checks pass, access is allowed to the realm-secured file.

Note the following when working with Oracle ACFS security:

- Oracle ACFS security is not supported if the file system has been replicated.

- Oracle ACFS security does not provide any protection for data sent on the network.

- A copy of a realm-protected file is not realm-protected unless the copy is made in a security realm-protected directory.

  Some applications, such as the `vi` editor, recreate a file when the file is modified. The modified file is saved as a temporary file, the original file is removed, and temporary file is copied with the original file name as the destination name. This process creates a new file. If the new file is created in a realm-protected directory, the security policies of the realm also apply to the new file. If the new file is not created in a realm-protected directory, then the new file is not realm-protected. If you are planning to copy a realm-protected file, you should ensure that the parent directory is also security realm protected.

  Security policies also apply to any temporary files created in a realm-protected directory.

To use Oracle ACFS security functionality on Linux, the disk group compatibility attributes for `ASM` and `ADVM` must be set to `11.2.0.2` or higher. To use Oracle ACFS security functionality on Windows, the disk group compatibility attributes for `ASM` and `ADVM` must be set to `11.2.0.3`. For information about disk group compatibility, refer to "Disk Group Compatibility" on page 4-33.

For information about Oracle ACFS security and snapshots, refer to "About Oracle ACFS Snapshots" on page 5-8.

Security information for Oracle ACFS file systems is displayed in the `V$ASM_ACFS_ SECURITY_INFO` view. For information about `V$ASM` views, refer to "Views Containing Oracle ACFS Information" on page 6-6.

To configure security for Oracle ACFS file systems, you can use the `acfsutil sec` command-line functions described in "Securing Oracle ACFS File Systems" on page 13-5 and "Oracle ACFS Command-Line Tools for Security" on page 13-64. Also, you can use ASMCA described in "Managing Security and Encryption for Oracle ACFS File Systems with Oracle ASM Configuration Assistant" on page 11-13.

Oracle ACFS Encryption

Oracle ACFS encryption enables you to encrypt data stored on disk (data-at-rest). The encryption feature protects data in an Oracle ACFS file system in encrypted format to prevent unauthorized use of data in the case of data loss or theft. Both encrypted and non-encrypted files can exist in the same Oracle ACFS file system.

Some of the encryption functionality requires system administrator privileges. This functionality incudes the commands for initiating, setting, and reconfiguring encryption.

System administrators and Oracle ACFS security administrators can initiate encryption operations. Also, unprivileged users can initiate encryption for files they own.

Oracle ACFS encryption provides two type of encryption keys:

- File Encryption Key

  This is a key for a file and is used to encrypt the data in the file.

- Volume Encryption Key

This is a key for a file system and is used to encrypt the file encryption keys.

You must first create the encryption key store, then specify file system-level encryption parameters and identify the directories. No extra steps are required for a user to read encrypted files if the user has the appropriate privileges for accessing the file data.

Oracle ACFS encryption protects data stored on secondary storage against the threat of theft or direct access to the storage medium. Data is never written to secondary storage in plaintext. Even if physical storage is stolen, the data stored cannot be accessed without the encryption keys. The encryption keys are never stored in plaintext. The keys are either obfuscated, or encrypted using a user-supplied password.

An Oracle ACFS security administrator can manage encryption parameters on a per-realm basis. After a file is placed under realm security, file-level encryption operations are not allowed on that file. Even if the realm security allows the file owner or the root user to open the file, file-level encryption operations are blocked. Encryption of realm-protected files is managed entirely by the Oracle ACFS security administrator, who can enable and disable encryption for files at a security realm level.

After a directory has been added to a security realm, all files created in the directory inherit the realm-level encryption parameters, not the directory or file system-level parameters. When a file is removed from its last security realm, the file is encrypted or decrypted to match the file system-level encryption status. The file is not re-encrypted to match file system-level parameters if it has been encrypted with security realm parameters.

A system administrator cannot rekey realm-secured files at the file system or file level. To ensure all realm-secured files are encrypted with the most recent volume encryption key (VEK), you must first remove encryption from all realms, and then re-enable encryption. This action re-encrypts all files with the most recent VEK.

Auditing and diagnostic data are logged for Oracle ACFS encryption. The log files include information such as `acfsutil` commands that have been run, the use of security or system administrator privileges, and runtime failures. Logs are written to the following files:

- *mount_point*/.Security/encryption/logs/encr-*host_name*.log

    The directory is created with `acfsutil encr set` command and protected by Oracle ACFS security if security is enabled. Refer to "acfsutil encr set" on page 13-94.

- *GRID_HOME*/log/*host_name*/acfssec/acfssec.log

    The messages that are logged to this file are for commands that are not associated with a specific file system, such as `acfsutil encr init`. The directory is created during installation and is owned by the root user.

When an active log file grows to a pre-defined maximum size (10 MB), the file is automatically moved to *log_file_name*.bak, the administrator is notified, and logging continues to the regular log file name. When the administrator is notified, the administrator must archive and remove the *log_file_name*.bak file. If an active log file grows to the maximum size and the *log_file_name*.bak file exists, logging stops until the backup file is removed. After the backup log file is removed, logging restarts automatically.

Note the following when working with Oracle ACFS encryption:

- Oracle ACFS encryption is not supported if the file system has been replicated.

- A copy of an encrypted file is not encrypted unless the copy of the file is made in an encrypted directory.

Some applications, such as the `vi` editor, recreate a file when the file is modified. The modified file is saved as a temporary file, the original file is removed, and temporary file is copied with the original file name as the destination name. This process creates a new file. The new file is not encrypted unless it is created in an encrypted directory. If you are planning to copy an encrypted file, you should ensure that the parent directory is also encrypted.

To use Oracle ACFS encryption functionality on Linux, the disk group compatibility attributes for `ASM` and `ADVM` must be set to `11.2.0.2` or higher. The disk group compatibility attributes for `ASM` and `ADVM` must be set to `11.2.0.3` on Linux for the following cases:

- If encryption is configured for the first time on Oracle ASM 11*g* release 2 version 11.2.0.3

- If encryption parameters must be changed or a new volume encryption key must be created following a software upgrade to Oracle ASM 11*g* release 2 version 11.2.0.3. For information, see "acfsutil encr set" on page 13-94 and "acfsutil encr rekey" on page 13-93.

To use Oracle ACFS encryption functionality on Windows, the disk group compatibility attributes for `ASM` and `ADVM` must be set to `11.2.0.3`.

For information about disk group compatibility, refer to "Disk Group Compatibility" on page 4-33.

For information about Oracle ACFS encryption and snapshots, refer to "About Oracle ACFS Snapshots" on page 5-8.

Encryption information for Oracle ACFS file systems is displayed in the `V$ASM_ACFS_ENCRYPTION_INFO` view. For information about `V$ASM` views, refer to "Views Containing Oracle ACFS Information" on page 6-6.

To configure encryption and manage encryptedOracle ACFS file systems, you can use the `acfsutil encr` command-line functions described in "Encrypting Oracle ACFS File Systems" on page 13-10 and "Oracle ACFS Command-Line Tools for Encryption" on page 13-89. Also, you can use Oracle ASM Configuration Assistant with encryption features as described in "Managing Security and Encryption for Oracle ACFS File Systems with Oracle ASM Configuration Assistant" on page 11-13.

Oracle ACFS Replication

Notes:

- Oracle ACFS replication functionality supports only one standby file system for each primary file system.

- Oracle ACFS replication supports eight or fewer nodes mounting the primary file system.

- Oracle ACFS replication is not supported if the file system has been encrypted with Oracle ACFS encryption.

- Oracle ACFS replication is not supported if the file system has been prepared for Oracle ACFS security.

- The sites hosting the primary and standby file systems must be running the same operating system and must have the same machine architecture.

Oracle ACFS replication enables replication of Oracle ACFS file systems across the network to a remote site, providing disaster recovery capability for the file system.

Oracle ACFS replication can only be configured for Oracle RAC systems. The source Oracle ACFS file system of an Oracle ACFS replication is referred to as a primary file system. The target Oracle ACFS file system of an Oracle ACFS replication is referred to as a standby file system.

A site can host both primary and standby file systems. For example, if there are cluster sites A and B, a primary file system hosted at site A can be replicated to a standby file system at site B. Also, a primary file system hosted at site B can be replicated to a standby file system at site A. However, an Oracle ACFS file system cannot be used as a primary and a standby file system.

Oracle ACFS replication captures file system changes written to disk for a primary file system and records the changes in files called replication logs. These logs are transported to the site hosting the associated standby file system where background processes read the logs and apply the changes recorded in the logs to the standby file system. After the changes recorded in a replication log have been successfully applied to the standby file system, the replication log is deleted from the sites hosting the primary and standby file systems.

It is critical that there is enough disk space available on both sites hosting the primary and the standby file systems to contain the replication logs. If the primary file system runs out of space, applications running on the file system may fail because Oracle ACFS cannot create a new replication log to capture the file system changes made by the application. If the standby file system runs out of space, it cannot accept new replication logs from the primary file system and cannot apply those changes to the standby file system. In addition, replication logs accumulate on the primary file system and consume the available disk space.

If the primary file system has less than 2 GB available free disk space, Oracle ACFS attempts to automatically terminate replication on the primary file system. This action prevents further consumption of disk space for replication operations and frees disk space consumed by any replication logs that remain. The auto-terminate process can prevent the primary file system from running out of space in most cases, but it is still possible that the auto-terminate process does not occur quickly enough. Before reaching the 2 GB limit, Oracle ACFS writes warnings about the free space problem in the Oracle Grid Infrastructure home alert log.

You should prevent both the primary file system and the standby file system from running out of space. If either file system runs out of available storage, you should either expand the file system or remove files from the file system to free up space. If the primary file system runs out of space and you decide to free up space by removing files, you should only remove files that are not being replicated because the removal of a file that is replicated is captured in a replication log. Another option is to delete any Oracle ACFS snapshots. For information about resizing an Oracle ACFS file system, refer to "acfsutil size" on page 13-106.

Because replication logs can accumulate when replication is paused, you should resume replication soon after pausing replication. For information on pausing and resuming replication, refer to "acfsutil repl pause" on page 13-59 and "acfsutil repl resume" on page 13-60.

Before using replication on a file system, ensure that you have checked the following:

- There is sufficient network bandwidth to support replication between the primary and standby file systems.

- The configuration of the sites hosting the primary and standby file systems allow the standby file system to keep up with the rate of change on the primary file system.

- The standby file system has sufficient capacity to manage the replication logs that are sent.

- There is sufficient storage capacity to hold excess replication logs that might collect on the primary and the standby file systems when the standby file system cannot process replication logs quickly. For example, this situation can occur during network problems or maintenance on the site hosting the standby file system.

- The primary file system must have a minimum size of 4 GB for each node that is mounting the file system. The standby file system must have a minimum size of 4 GB and should be sized appropriately for the amount of data being replicated and the space necessary for the replication logs sent from the primary file system.

> **See Also:** For information about tuning the network, refer to the documentation at the MAA link on Oracle Technology Network:
>
> http://www.oracle.com/technetwork/database/features/availabi lity/maa-096107.html
>
> Relevant information on tuning the network can be found in the *Data Guard Redo Transport & Network Configuration* paper.

Directories and files in an Oracle ACFS file system can be tagged to select the objects that you want to replicate in a file system. For information on tagging, see "Oracle ACFS Tagging" on page 5-19.

Before replicating an Oracle ACFS file system, a replication configuration must be established that identifies information such as the site hosting the primary file system, the site hosting the standby file system, the file system to be replicated, mount point of the file system, and a list of tags if desired.

To use Oracle ACFS replication functionality on Linux, the disk group compatibility attributes for ASM and ADVM must be set to 11.2.0.2 or higher for the disk groups that contain the primary and standby file systems. To use Oracle ACFS replication functionality on Windows, the disk group compatibility attributes for ASM and ADVM must be set to 11.2.0.3. For information about disk group compatibility, refer to "Disk Group Compatibility" on page 4-33.

To configure replication and manage replicated Oracle ACFS file systems, use the acfsutil repl command-line functions described in "Replicating Oracle ACFS File Systems" on page 13-13 and "Oracle ACFS Command-Line Tools for Replication" on page 13-53.

Oracle ACFS Tagging

Oracle ACFS tagging assigns a common naming attribute to a group of files. Oracle ACFS Replication can use this tag to select files with a unique tag name for replication to a different remote cluster site. The tagging option avoids having to replicate an entire Oracle ACFS file system.

Oracle ACFS implements tagging with Extended Attributes. Some editing tools and backup utilities do not retain the Extended Attributes of the original file by default; you must set a specific switch. The following list describes the necessary requirements and switch settings for some common utilities to ensure Oracle ACFS tag names are preserved on the original file.

- Install the `coreutils` library (version `coreutils-5.97-23.el5_4.1.src.rpm` or `coreutils-5.97-23.el5_4.2.x86_64.rpm` or later) on Linux to install versions of the `cp` command that supports Extended Attribute preservation with the `--preserve=xattr` switch and the `mv` command that supports Extended Attribute preservation without any switches.

- `vim` or `vi` editors require the `set bkc=yes` option in the `.vimrc` (Linux) or `_vimrc` (Windows) file to make a backup copy of a file and overwrite the original. This preserves tag names on the original file.

- `emacs` requires that the `backup-by-copying` option is set to a non-nil value to preserve tag names on the original file name rather than a backup copy. This option must be added to the `.emacs` file.

- The `rsync` file transfer utility requires the `-X` flag option to preserve tag names. In addition, you must set the `-l` and `-X` flags to preserve the tag names assigned to symbolic link files themselves.

- The `tar` backup utility on Linux requires the `--xattrs` flag be set on the command line to preserve tag names on a file. However, `tar` does not retain the tag names assigned to symbolic link files even with the `--xattrs` flag.

  The `tar` backup utility on Windows currently provides no support to retain tag names as no switch exists to save Extended Attributes.

To use Oracle ACFS tagging functionality on Linux, the disk group compatibility attributes for `ASM` and `ADVM` must be set to `11.2.0.2` or higher. To use Oracle ACFS tagging functionality on Windows, the disk group compatibility attributes for `ASM` and `ADVM` must be set to `11.2.0.3`. For information about disk group compatibility, refer to "Disk Group Compatibility" on page 4-33.

To configure tagging and manage tagged Oracle ACFS file systems, use the `acfsutil tag` command-line functions described in "Tagging Oracle ACFS File Systems" on page 13-11 and "Oracle ACFS Command-Line Tools for Tagging" on page 13-50.

Overview of Oracle ASM Dynamic Volume Manager

Oracle ASM Dynamic Volume Manager (Oracle ADVM) provides volume management services and a standard disk device driver interface to clients. File systems and other disk-based applications send I/O requests to Oracle ADVM volume devices as they would to other storage devices on a vendor operating system.

An Oracle ADVM volume device is constructed from an Oracle ASM dynamic volume. One or more Oracle ADVM volume devices may be configured within each Oracle ASM disk group. The Oracle ADVM Driver maps I/O requests against an Oracle ADVM volume device to blocks in a corresponding Oracle ASM dynamic volume and disk set located within an Oracle ASM disk group. An Oracle ADVM volume device exports Oracle ASM volume manager features and ensures that volume mirrors remain consistent in the face of abnormal system shutdowns, Oracle ASM instance failures, or system failures.

Oracle ADVM extends Oracle ASM by providing a disk driver interface to Oracle ASM storage allocated as Oracle ASM volume files. You can use Oracle ADVM to create virtual disks that contain file systems. These file systems contained on Oracle ASM volumes are able to support files beyond Oracle database files, such as executable files, report files, trace files, alert logs, and other application data files. Because Oracle ADVM volumes are actually Oracle ASM files, they require the same administrative privileges as the Oracle ASM files.

Oracle Automatic Storage Management Cluster File System (Oracle ACFS) is layered on Oracle ASM through the Oracle ADVM interface. With the addition of the Oracle ADVM, Oracle ASM becomes a complete storage solution of user data for both database and non-database file needs.

To add a volume to an Oracle ASM disk group, disk group attributes COMPATIBLE.ASM and COMPATIBLE.ADVM must be set to '11.2'. The volume name can be a maximum of 11 alphanumeric characters; dashes are not allowed. The first character must be alphabetic.

> **Note:** Dynamic volumes supersede traditional device partitioning. Each volume is individually named and may be configured for a single file system. Oracle ADVM volumes may be created on demand from Oracle ASM disk group storage and dynamically resized as required. These attributes make Oracle ADVM volumes far more flexible than physical devices and associated partitioning schemes.

The Oracle ADVM functionality includes the following:

- The ALTER DISKGROUP ... ADD | RESIZE |DROP | ENABLE | DISABLE | MODIFY VOLUME SQL statements that manage Oracle ADVM volumes.

    See "Managing Volumes in a Disk Group" on page 4-13.

    > **See Also:** *Oracle Database SQL Language Reference* for information about the ALTER DISKGROUP SQL statement

- ASMCMD command-line tools for managing Oracle ASM volumes.

    See "ASMCMD Volume Management Commands" on page 12-61.

- There are new V$ASM views display information about Oracle ASM volumes.

    See "Views Containing Oracle ACFS Information" on page 6-6.

- Oracle Enterprise Manager support for managing Oracle ASM volumes.

    See Chapter 10, "Administering Oracle ACFS with Oracle Enterprise Manager".

- Oracle ACFS operating system utilities for managing file systems and volume device files.

    See Chapter 13, "Oracle ACFS Command-Line Tools".

6

Using Views to Display Information

This chapter contains information about using dynamic views to display Oracle Automatic Storage Management (Oracle ASM), Oracle Automatic Storage Management Cluster File System (Oracle ACFS), and Oracle ASM Dynamic Volume Manager (Oracle ADVM) information.

- Views Containing Oracle ASM Disk Group Information
- Views Containing Oracle ACFS Information

> **See Also:** *Oracle Database Reference* for information about all of the `V$ASM*` dynamic performance views

Views Containing Oracle ASM Disk Group Information

You can use the views in Table 6–1 to obtain information about Oracle ASM disk groups.

The `V$ASM_ALIAS`, `V$ASM_CLIENT`, `V$ASM_DISK`, `V$ASM_DISK_IOSTAT`, `V$ASM_DISK_STAT`, `V$ASM_DISKGROUP`, `V$ASM_DISKGROUP_STAT`, `V$ASM_FILE`, `V$ASM_TEMPLATE`, `V$ASM_USER`, `V$ASM_USERGROUP`, and `V$ASM_USERGROUP_MEMBER` are accessible from both the Oracle ASM and database instances.

Table 6–1   Oracle ASM dynamic views for disk group information

View	Description
`V$ASM_ALIAS`	Contains one row for every alias present in every disk group mounted by the Oracle ASM instance.
`V$ASM_ATTRIBUTE`	Displays one row for each attribute defined. In addition to attributes specified by `CREATE DISKGROUP` and `ALTER DISKGROUP` statements, the view may show other attributes that are created automatically. Attributes are only displayed for disk groups where `COMPATIBLE.ASM` is set to 11.1 or higher.
`V$ASM_CLIENT`	In an Oracle ASM instance, identifies databases using disk groups managed by the Oracle ASM instance. In an Oracle Database instance, contains information about the Oracle ASM instance if the database has any open Oracle ASM files.
`V$ASM_DISK`	Contains one row for every disk discovered by the Oracle ASM instance, including disks that are not part of any disk group. This view performs disk discovery every time it is queried.

Table 6–1   (Cont.) Oracle ASM dynamic views for disk group information

View	Description
V$ASM_DISK_IOSTAT	Displays information about disk I/O statistics for each Oracle ASM client. In an Oracle Database instance, only the rows for that instance are shown.
V$ASM_DISK_STAT	Contains the same columns as V$ASM_DISK, but to reduce overhead, does not perform a discovery when it is queried. It only returns information about any disks that are part of mounted disk groups in the storage system. To see all disks, use V$ASM_DISK instead.
V$ASM_DISKGROUP	Describes a disk group (number, name, size related info, state, and redundancy type). This view performs disk discovery every time it is queried.
V$ASM_DISKGROUP_STAT	Contains the same columns as V$ASM_DISKGROUP, but to reduce overhead, does not perform a discovery when it is queried. It does not return information about any disks that are part of mounted disk groups in the storage system. To see all disks, use V$ASM_DISKGROUP instead.
V$ASM_FILE	Contains one row for every Oracle ASM file in every disk group mounted by the Oracle ASM instance.
V$ASM_OPERATION	In an Oracle ASM instance, contains one row for every active Oracle ASM long running operation executing in the Oracle ASM instance. In an Oracle Database instance, contains no rows.
V$ASM_TEMPLATE	Contains one row for every template present in every disk group mounted by the Oracle ASM instance.
V$ASM_USER	Contains the effective operating system user names of connected database instances and names of file owners.
V$ASM_USERGROUP	Contains the creator for each Oracle ASM File Access Control group.
V$ASM_USERGROUP_MEMBER	Contains the members for each Oracle ASM File Access Control group.

When querying V$ASM views, note that the value of the disk group number is not a static value. When a disk group is mounted, a disk group number is chosen. This number may change across disk group mounts. A disk group number is not recorded in any persistent structure, but the current value can be viewed in the GROUP_NUMBER column of the V$ASM views.

An example of the use of the V$ASM_ATTRIBUTE and V$ASM_DISKGROUP views is shown in Example 6–1. The COMPATIBLE.ASM value must be set to 11.1 or higher for the disk group to display in the V$ASM_ATTRIBUTE view output. Attributes that are designated as read-only (Y) can only be set during disk group creation.

Example 6–1   Viewing disk group attributes with V$ASM_ATTRIBUTE

```
SQL> SELECT dg.name AS diskgroup, SUBSTR(a.name,1,18) AS name,
     SUBSTR(a.value,1,24) AS value, read_only FROM V$ASM_DISKGROUP dg,
     V$ASM_ATTRIBUTE a WHERE dg.name = 'DATA'
     AND dg.group_number = a.group_number;

DISKGROUP               NAME                    VALUE                 READ_ONLY
----------------------- ----------------------- --------------------- ---------
DATA                    disk_repair_time        3.6h                  N
DATA                    au_size                 4194304               Y
DATA                    sector_size             512                   Y
```

```
DATA                    compatible.asm      11.2.0.0.0          N
DATA                    compatible.rdbms    11.2.0.0.0          N
DATA                    compatible.advm     11.2.0.0.0          N
DATA                    cell.smart_scan_capable  FALSE          N
DATA                    access_control.enabled   TRUE           N
DATA                    access_control.umask     002            N
...
```

You can view the compatibility for a disk group with the V$ASM_DISKGROUP view, as shown in Example 6–2.

Example 6–2   Viewing the compatibility of a disk group with V$ASM_DISKGROUP

```
SQL> SELECT name AS diskgroup, compatibility AS asm_compat,
     database_compatibility AS db_compat FROM V$ASM_DISKGROUP;

DISKGROUP        ASM_COMPAT   DB_COMPAT
---------------- ------------ ----------
DATA             11.2.0.0.0   11.2.0.0.0
FRA              10.1.0.0.0   10.1.0.0.0
```

An example of the use of the V$ASM_DISK and V$ASM_DISKGROUP views is shown in Example 6–3. This example displays the disks associated with a disk group, plus the mount status and state of the disks.

Example 6–3   Viewing disks in disk groups with V$ASM_DISK

```
SQL> SELECT SUBSTR(d.name,1,16) AS asmdisk, d.mount_status, d.state,
     dg.name AS diskgroup FROM V$ASM_DISKGROUP dg, V$ASM_DISK d
     WHERE dg.group_number = d.group_number;

ASMDISK          MOUNT_S STATE    DISKGROUP
---------------- ------- -------- -------------
DATA_0001        CACHED  NORMAL   DATA
DATA_0007        CACHED  NORMAL   DATA
DATA_0000        CACHED  NORMAL   DATA
DATA_0008        CACHED  NORMAL   DATA
DATA_0005        CACHED  NORMAL   DATA
DATA_0002        CACHED  NORMAL   DATA
DATA_0004        CACHED  NORMAL   DATA
DATA_0006        CACHED  NORMAL   DATA
DATA_0003        CACHED  NORMAL   DATA
FRA_0005         CACHED  NORMAL   FRA
FRA_0004         CACHED  NORMAL   FRA
FRA_0000         CACHED  NORMAL   FRA
FRA_0009         CACHED  NORMAL   FRA
FRA_0011         CACHED  NORMAL   FRA
...
```

An example of the use of the V$ASM_CLIENT and V$ASM_DISKGROUP views on an Oracle ASM instance is shown in Example 6–4. This example displays disk groups with information about the connected database client instances.

Example 6–4   Viewing disk group clients with V$ASM_CLIENT

```
SQL> SELECT dg.name AS diskgroup, SUBSTR(c.instance_name,1,12) AS instance,
     SUBSTR(c.db_name,1,12) AS dbname, SUBSTR(c.SOFTWARE_VERSION,1,12) AS software,
     SUBSTR(c.COMPATIBLE_VERSION,1,12) AS compatible
     FROM V$ASM_DISKGROUP dg, V$ASM_CLIENT c
     WHERE dg.group_number = c.group_number;
```

```
DISKGROUP                      INSTANCE      DBNAME   SOFTWARE      COMPATIBLE
------------------------------ ------------ -------- ------------ ------------
DATA                           +ASM          +ASM     11.2.0.0.0   11.2.0.0.0
DATA                           orcl          orcl     11.2.0.0.0   11.2.0.0.0
DATA                           +ASM          asmvol   11.2.0.0.0   11.2.0.0.0
FRA                            orcl          orcl     11.2.0.0.0   11.2.0.0.0
...
```

Viewing Oracle ASM File Access Control Information

You can view information about Oracle ASM File Access Control in the columns of the VASM_USER, VASM_USERGROUP, V$ASM_USERGROUP_MEMBER, and V$ASM_FILE views.

Example 6–5 shows information about Oracle ASM File Access Control users displayed in the V$ASM_USER view.

Example 6–5   Viewing Oracle ASM File Access Control information with V$ASM_USER

```
SQL> SELECT dg.name AS diskgroup, u.group_number, u.user_number, u.os_id, u.os_name
     FROM V$ASM_DISKGROUP dg, V$ASM_USER u
     WHERE dg.group_number = u.group_number AND dg.name = 'DATA';

DISKGROUP       GROUP_NUMBER USER_NUMBER OS_ID OS_NAME
--------------- ------------ ----------- ----- -------
DATA                       1           1 1001  oracle1
DATA                       1           2 1002  oracle2
DATA                       1           3 1003  grid
```

Example 6–6 shows information about Oracle ASM File Access Control user groups displayed in the V$ASM_USERGROUP view.

Example 6–6   Viewing Oracle ASM File Access Control information with V$ASM_USERGROUP

```
SQL>  SELECT dg.name AS diskgroup, ug.group_number, ug.owner_number, u.os_name,
      ug.usergroup_number, ug.name FROM V$ASM_DISKGROUP dg, V$ASM_USER u, V$ASM_USERGROUP ug
      WHERE dg.group_number = ug.group_number AND dg.name = 'DATA'
      AND ug.owner_number = u.user_number;

DISKGROUP          GROUP_NUMBER OWNER_NUMBER OS_NAME         USERGROUP_NUMBER NAME
----------------- ------------ ------------ -------------- ---------------- -----------------
DATA                         1            3 grid                          1 asm_data
```

Example 6–7 shows information about Oracle ASM File Access Control user groups and members displayed in the V$ASM_USERGROUP_MEMBER view.

Example 6–7   Viewing Oracle ASM File Access Control information with V$ASM_USERGROUP_MEMBER

```
SQL> SELECT dg.name AS diskgroup, um.group_number, um.member_number, u.os_name,
     um.usergroup_number, ug.name FROM V$ASM_DISKGROUP dg, V$ASM_USER u, V$ASM_USERGROUP_MEMBER um,
     V$ASM_USERGROUP ug WHERE dg.group_number = um.group_number AND
     dg.group_number = ug.group_number AND dg.group_number = u.group_number AND dg.name = 'DATA'
     AND um.member_number = u.user_number AND um.usergroup_number = ug.usergroup_number;

DISKGROUP       GROUP_NUMBER MEMBER_NUMBER OS_NAME           USERGROUP_NUMBER NAME
--------------- ------------ ------------- ----------------- ---------------- -----------------
DATA                       1             1 oracle1                          1 asm_data
DATA                       1             2 oracle2                          1 asm_data
```

Example 6–8 shows information about Oracle ASM File Access Control file permissions displayed in the V$ASM_FILE view.

Example 6–8   Viewing Oracle ASM File Access Control information with V$ASM_FILE

```
SQL> SELECT dg.name AS diskgroup, a.name, f.permissions, f.user_number, u.os_name,
     f.usergroup_number, ug.NAME FROM V$ASM_DISKGROUP dg, V$ASM_USER u, V$ASM_USERGROUP ug,
     V$ASM_FILE f, V$ASM_ALIAS a WHERE dg.group_number = f.group_number AND
     dg.group_number = u.group_number AND dg.group_number = ug.group_number AND
     dg.name = 'FRA' AND f.usergroup_number = ug.usergroup_number AND f.user_number = u.user_number
     AND f.file_number = a.file_number;

DISKGROUP NAME                    PERMISSIONS USER_NUMBER OS_NAME USERGROUP_NUMBER NAME
--------- --------------------    ----------- ----------- ------- ---------------- -------------
DATA      USERS.259.685366091     rw-r-----             3 grid                   1 asm_fra
DATA      TEMP.264.685366227      rw-r-----             3 grid                   1 asm_fra
...
```

For more information about Oracle ASM File Access Control, see "Managing Oracle ASM File Access Control for Disk Groups" on page 4-40.

Viewing Disk Region Information

Information about Intelligent Data Placement is displayed in the columns of the VASM_DISK, VASM_DISK_IOSTAT, V$ASM_FILE, and V$ASM_TEMPLATE views.

Example 6–9 shows queries for Intelligent Data Placement information in the PRIMARY_REGION and MIRROR_REGION columns of the V$ASM_FILE view.

Example 6–9   Viewing Intelligent Data Placement information with V$ASM_FILE

```
SQL> SELECT dg.name AS diskgroup, f.file_number, f.primary_region, f.mirror_region, f.hot_reads,
     f.hot_writes,  f.cold_reads, f.cold_writes
     FROM V$ASM_DISKGROUP dg, V$ASM_FILE f
     WHERE dg.group_number = f.group_number and dg.name = 'DATA';

DISKGROUP                    FILE_NUMBER PRIM MIRR  HOT_READS HOT_WRITES COLD_READS COLD_WRITES
---------------------------- ----------- ---- ---- ---------- ---------- ---------- -----------
DATA                                 257 COLD COLD          0          0     119770      886575
DATA                                 258 COLD COLD          0          0       1396      222282
DATA                                 259 COLD COLD          0          0       2056         199
DATA                                 260 COLD COLD          0          0      42377     1331016
DATA                                 261 COLD COLD          0          0    4336300     1331027
...
```

Example 6–10 displays Intelligent Data Placement information in the PRIMARY_REGION and MIRROR_REGION columns of the V$ASM_TEMPLATE view.

Example 6–10   Viewing Intelligent Data Placement information with V$ASM_TEMPLATE

```
SQL> SELECT dg.name AS diskgroup, t.name, t.stripe, t.redundancy, t.primary_region, t.mirror_region
     FROM V$ASM_DISKGROUP dg, V$ASM_TEMPLATE t
     WHERE dg.group_number = t.group_number and dg.name = 'DATA' ORDER BY t.name;

DISKGROUP                    NAME                         STRIPE REDUND PRIM MIRR
---------------------------- ---------------------------- ------ ------ ---- ----
DATA                         ARCHIVELOG                   COARSE MIRROR COLD COLD
DATA                         ASMPARAMETERFILE             COARSE MIRROR COLD COLD
DATA                         AUTOBACKUP                   COARSE MIRROR COLD COLD
DATA                         BACKUPSET                    COARSE MIRROR COLD COLD
DATA                         CHANGETRACKING               COARSE MIRROR COLD COLD
DATA                         CONTROLFILE                  FINE   HIGH   COLD COLD
```

```
DATA                            DATAFILE                    COARSE MIRROR COLD COLD
DATA                            DATAGUARDCONFIG             COARSE MIRROR COLD COLD
DATA                            DUMPSET                     COARSE MIRROR COLD COLD
DATA                            FLASHBACK                   COARSE MIRROR COLD COLD
DATA                            FLASHFILE                   COARSE MIRROR COLD COLD
DATA                            OCRFILE                     COARSE MIRROR COLD COLD
DATA                            ONLINELOG                   COARSE MIRROR COLD COLD
DATA                            PARAMETERFILE               COARSE MIRROR COLD COLD
DATA                            TEMPFILE                    COARSE MIRROR COLD COLD
DATA                            XTRANSPORT                  COARSE MIRROR COLD COLD
15 rows selected.
```

For information about setting Intelligent Data Placement, see "Intelligent Data Placement" on page 4-16.

Views Containing Oracle ACFS Information

You can use the views in Table 6–2 to obtain information about Oracle Automatic Storage Management Cluster File System (Oracle ACFS). These views are accessible from the Oracle ASM instance.

Table 6–2   Oracle ASM dynamic views for Oracle ACFS information

View	Description
V$ASM_ACFS_ENCRYPTION_INFO	Contains encryption information for each Oracle ACFS file system.
V$ASM_ACFS_SECURITY_INFO	Contains security realm information for each Oracle ACFS file system.
V$ASM_ACFSSNAPSHOTS	Contains snapshot information for every mounted Oracle ACFS file system.
V$ASM_ACFSVOLUMES	Contains information about mounted Oracle ACFS volumes, correlated with V$ASM_FILESYSTEM.
V$ASM_FILESYSTEM	Contains columns that display information for every mounted Oracle ACFS file system.
V$ASM_VOLUME	Contains information about each Oracle ADVM volume that is a member of an Oracle ASM instance.
V$ASM_VOLUME_STAT	Contains information about statistics for each Oracle ADVM volume.

Example 6–11 shows information displayed from the V$ASM_ACFS_ENCRYPTION_INFO view. The FS_NAME column contains the mount point. The VOL_DEVICE contains the name of the Oracle ADVM device.

Example 6–11   Viewing encryption information in V$ASM_ACFS_ENCRYPTION_INFO

```
SELECT SUBSTR(fs_name,1,32) FILESYSTEM, SUBSTR(vol_device,1,20) DEVICE, set_status, enabled_status,
    algorithm, key_length, last_rekey_time FROM V$ASM_ACFS_ENCRYPTION_INFO;

FILESYSTEM                       DEVICE                SET_STA ENABLED_ ALGORIT KEY_LEN LAST_REKE
-------------------------------- --------------------- ------- -------- ------- ------- ---------
/u01/app/acfsmounts/acfs1        /dev/asm/volume1-228  YES     ENABLED  AES     192
```

See Also: *Oracle Database Reference* for information about the V$ASM_ACFS_ENCRYPTION_INFO view

Example 6–12 shows information displayed from the V$ASM_ACFS_SECURITY_INFO view. The FS_NAME column contains the mount point. The VOL_DEVICE contains the name of the Oracle ADVM device.

Example 6–12   Viewing security information in V$ASM_ACFS_SECURITY_INFO

```
SELECT SUBSTR(fs_name,1,32) FILESYSTEM, SUBSTR(vol_device,1,20) DEVICE, prepared_status,
    enabled_status FROM V$ASM_ACFS_SECURITY_INFO;

FILESYSTEM                       DEVICE                PREPARE ENABLED_
-------------------------------- --------------------- ------- --------
/u01/app/acfsmounts/acfs1        /dev/asm/volume1-228  YES     ENABLED
```

> **See Also:** *Oracle Database Reference* for information about the V$ASM_ACFS_SECURITY_INFO view

Example 6–13 shows information displayed from the V$ASM_ACFSSNAPSHOTS view. The FS_NAME column contains the mount point. The VOL_DEVICE contains the name of the Oracle ADVM device.

Example 6–13   Viewing snapshot information in V$ASM_ACFSSNAPSHOTS

```
SELECT SUBSTR(fs_name,1,32) FILESYSTEM, SUBSTR(vol_device,1,24) DEVICE,
  SUBSTR(snap_name,1,28) SNAPSHOT, create_time TIME FROM V$ASM_ACFSSNAPSHOTS;

FILESYSTEM                       DEVICE                   SNAPSHOT                     TIME
-------------------------------- ------------------------ ---------------------------- ---------
/u01/app/acfsmounts/acfs1        /dev/asm/volume1-228     mysnapshot_0900609a          09-JUL-09
/u01/app/acfsmounts/acfs1        /dev/asm/volume1-228     mysnapshot_0900610a          10-JUL-09
/u01/app/acfsmounts/acfs1        /dev/asm/volume1-228     mysnapshot_0900609c          09-JUL-09
/u01/app/acfsmounts/acfs1        /dev/asm/volume1-228     mysnapshot_0900610b          10-JUL-09
```

> **See Also:** *Oracle Database Reference* for information about the V$ASM_ACFSSNAPSHOTS view

Example 6–14 shows information displayed from the V$ASM_VACFSOLUMES view. The PRIMARY_VOL column contains TRUE if the volume is the primary volume for the file system.

Example 6–14   Viewing volume information with V$ASM_ACFSVOLUMES

```
SELECT fs_name, vol_device, primary_vol, total_mb, free_mb FROM V$ASM_ACFSVOLUMES;

FS_NAME                   VOL_DEVICE            PRIMARY_VOL   TOTAL_MB  FREE_MB
------------------------- --------------------- -----------   --------- ----------
/u01/app/acfsmounts/acfs1 /dev/asm/volume1-228  TRUE          1024000 578626.522
/u01/app/acfsmounts/acfs2 /dev/asm/volume2-375  TRUE          1024000 685761.463
...
```

> **See Also:** *Oracle Database Reference* for information about the V$ASM_ACFSVOLUMES view

Example 6–15 shows information displayed from the V$ASM_FILESYSTEM view.

The STATE column contains the status of the file system, either AVAILABLE or OFFLINE. An offline file system can only be dismounted; other attempts at access result in errors. Offline means that either the Oracle ASM instance is down, the disk group has been

forced dismounted, or less commonly, a metadata I/O failure occurred or serious metadata corruption was detected. With a metadata I/O failure, the file system is also marked as corrupt.

The CORRUPT column indicates whether the file system needs the fsck or acfschkdsk command run on it. See "fsck" on page 13-21 and "acfschkdsk" on page 13-41.

Example 6–15   Viewing volume information with V$ASM_FILESYSTEM

```
SQL> SELECT fs_name, available_time, block_size, state, corrupt FROM V$ASM_FILESYSTEM;

FS_NAME                          AVAILABLE BLOCK_SIZE STATE        CORRUPT
-------------------------------- --------- ---------- ------------ -------
/u01/app/acfsmounts/acfs1        19-JUL-09          4 AVAILABLE    FALSE
/u01/app/acfsmounts/acfs2        19-JUL-09          4 AVAILABLE    FALSE
```

> **See Also:**   *Oracle Database Reference* for information about the V$ASM_FILESYSTEM view

Example 6–16 shows information displayed from the V$ASM_VOLUME view for volumes contained in the DATA disk group.

Example 6–16   Viewing volume information with V$ASM_VOLUME

```
SELECT dg.name AS diskgroup, v.volume_name, v.volume_device, v.mountpath
    FROM V$ASM_DISKGROUP dg, V$ASM_VOLUME v
    WHERE dg.group_number = v.group_number and dg.name = 'DATA';

DISKGROUP        VOLUME_NAME      VOLUME_DEVICE          MOUNTPATH
---------------- ---------------- ---------------------- ----------------------------------
DATA             VOLUME1          /dev/asm/volume1-228   /u01/app/acfsmounts/acfs1
DATA             VOLUME2          /dev/asm/volume2-375   /u01/app/acfsmounts/acfs2
```

> **See Also:**   *Oracle Database Reference* for information about the V$ASM_VOLUME view

Example 6–16 shows information displayed from the V$ASM_VOLUME_STAT view for volumes contained in the DATA disk group. The BYTES_READ column contains the total number of bytes read for the volume. The BYTES_WRITTEN column contains the total number of bytes written for the volume.

Example 6–17   Viewing volume information with V$ASM_VOLUME_STAT

```
SELECT dg.name AS diskgroup, v.volume_name, v.bytes_read, v.bytes_written
    FROM V$ASM_DISKGROUP dg, V$ASM_VOLUME_STAT v
    WHERE dg.group_number = v.group_number and dg.name = 'DATA';

DISKGROUP                      VOLUME_NAME                  BYTES_READ BYTES_WRITTEN
------------------------------ ---------------------------- ---------- -------------
DATA                           VOLUME1                      12370105856     43510272
DATA                           VOLUME2                          2685728     32201504
```

> **See Also:**   *Oracle Database Reference* for information about the V$ASM_VOLUME_STAT view

7

Administering Oracle ASM Files, Directories, and Templates

This chapter describes how to administer files when you use the Oracle Automatic Storage Management (Oracle ASM) storage option.

This chapter contains the following topics:

- What Types of Files Does Oracle ASM Support?
- About Oracle ASM Filenames
- Creating and Referencing Oracle ASM Files in the Database
- Managing Alias Names for Oracle ASM Filenames
- Managing Disk Group Directories
- Accessing Oracle ASM Files with the XML DB Virtual Folder
- Using DBMS_FILE Transfer Utility for Oracle ASM
- Managing Disk Group Templates

What Types of Files Does Oracle ASM Support?

Oracle ASM supports most file types required by the database. Table 7–1 lists file types that are supported directly by Oracle ASM and lists the system default template that provides the attributes for file creation. Some file types shown in the table are related to specific products or features, and are not discussed in this book.

Table 7–1   File types supported by Oracle ASM

File Type	Default Templates
Control files	`CONTROLFILE`
Data files	`DATAFILE`
Redo log files	`ONLINELOG`
Archive log files	`ARCHIVELOG`
Temporary files	`TEMPFILE`
Data file backup pieces	`BACKUPSET`
Data file incremental backup pieces	`BACKUPSET`
Archive log backup piece	`BACKUPSET`
Data file copy	`DATAFILE`

Table 7–1 (Cont.) File types supported by Oracle ASM

File Type	Default Templates
Persistent initialization parameter file (SPFILE)	PARAMETERFILE
Flashback logs	FLASHBACK
Change tracking file	CHANGETRACKING
Data Pump dumpset	DUMPSET
Automatically generated control file backup	AUTOBACKUP
Cross-platform transportable data files	XTRANSPORT
Flash file	FLASHFILE
Oracle ASM Persistent initialization parameter file (SPFILE)	ASMPARAMETERFILE
Oracle ASM Persistent initialization parameter file (SPFILE) backup	ASMPARAMETERFILEBACKUP
Oracle Cluster Registry file	OCRFILE
Oracle ASM Dynamic Volume Manager volumes	n/a

Oracle ASM cannot directly support some administrative file types on disk groups. These include trace files, audit files, alert logs, export files, tar files, and core files.

> **Note:** Oracle Automatic Storage Management Cluster File System (Oracle ACFS) and Oracle ASM Dynamic Volume Manager (Oracle ADVM) extend Oracle ASM support to include database and application executables, database trace files, database alert logs, application reports, BFILEs, and configuration files. Other supported files are video, audio, text, images, engineering drawings, and other general-purpose application file data. See Chapter 5, "Introduction to Oracle ACFS".

> **See Also:** "Managing Disk Group Templates" on page 7-15 for a description of the system default templates

About Oracle ASM Filenames

Every file created in Oracle ASM is provided a filename generated by Oracle Managed Files, known as a fully qualified filename (or system-generated filename). The fully qualified filename represents a complete path name in the Oracle ASM file system. An example of a fully qualified filename is:

```
+data/orcl/controlfile/Current.256.541956473
```

You can use the fully qualified filename to reference (read or retrieve) an Oracle ASM file. You can also use other abbreviated filename formats, such as an alias Oracle ASM filename described in "Alias Oracle ASM Filename Forms" on page 7-5, to reference an Oracle ASM file.

Oracle ASM generates a fully qualified filename upon any request to create a file. A creation request does not or cannot specify a fully qualified filename. Instead, it uses a simpler syntax to specify a file, such as an alias or just a disk group name. Oracle ASM then creates the file, placing it in the correct Oracle ASM path according to file type, and then assigns an appropriate fully qualified filename. If you specify an alias in the

creation request, Oracle ASM also creates the alias so that it references the fully qualified filename.

Oracle ASM file creation requests are either single file creation requests or multiple file creation request.

Notes:

- You can find the generated name in database views displaying Oracle file names, such as `V$DATAFILE` and `V$LOGFILE`. You can use this name, or an abbreviated form of it, if you later must reference an Oracle ASM file in a SQL statement. As with other Oracle Database filenames, Oracle ASM filenames are kept in the control file and the RMAN catalog.

- You can use fully qualified and numeric filenames in single-file creation if you specify the `REUSE` keyword, as described in "Using Oracle ASM Filenames in SQL Statements" on page 7-9.

- Forward slash (/) and backslash (\) are interchangeable in filenames. Filenames are case insensitive, but case retentive.

This section contains the following topics:

- Single File Creation Form
- Multiple File Creation Form
- Fully Qualified File Name Form
- Alias Oracle ASM Filename Forms
- Alias Oracle ASM Filename with Template Form
- Incomplete Oracle ASM Filename Form
- Incomplete Oracle ASM Filename with Template Form

Table 7–2 specifies the valid usage for each filename form. If the form is used for file creation, the table specifies whether the created file is an Oracle Managed Files (OMF).

Table 7–2   Valid usage for Oracle ASM filename forms

Filename Form	Reference to File?	Single-file Creation?	Multiple File Creation?	Created as OMF?
Fully qualified file name	Yes	No	No	No
Alias file name	Yes	Yes	No	No
Alias with template file name	No	Yes	No	No
Incomplete file name	No	Yes	Yes	Yes
Incomplete file name with template	No	Yes	Yes	Yes

Single File Creation Form

A single file creation form is used to create a single file, such as a data file or a control file. The form of the Oracle ASM filename in this type is either an alias, such as `+data/orcl/ctl.f`, or an incomplete file name, such as a disk group name preceded by a plus sign (`+data`). You use the alias or disk group name where a filename is called for in a SQL statement, such as `CREATE TABLESPACE`, `CREATE CONTROLFILE`.

Multiple File Creation Form

A multiple file creation form is used multiple times to create Oracle ASM files, such as when assigning a value to the initialization parameter DB_CREATE_FILE_DEST. You can also run a CREATE TABLESPACE statement (without a filename specification) multiple times. Each time, Oracle ASM creates a different unique data file name.

When using the multiple file creation form, you can use an incomplete filename form, such as a disk group name preceded by a plus sign (+data). In this case, you set DB_CREATE_FILE_DEST to an incomplete filename (for example, +data), and whenever a command is run that must create a database file in DB_CREATE_FILE_DEST, the file is created in the designated disk group and assigned a unique fully qualified name. You can use an incomplete filename in other *_DEST initialization parameters.

Fully Qualified File Name Form

You can use a fully qualified form for referencing existing Oracle ASM files in Oracle ASM operations, except for disk group creation. This form is the Oracle ASM file name that Oracle ASM automatically generates when an Oracle ASM file is created.

A fully qualified file name has the following form:

+diskgroup/dbname/filetype/filetypetag.file.incarnation

Where:

- *+diskgroup* is the disk group name preceded by a plus sign.

  You can think of the plus sign (+) as the root directory of the Oracle ASM file system, similar to the slash (/) on UNIX or Linux computers.

- *dbname* is the DB_UNIQUE_NAME of the database to which the file belongs.

- *filetype* is the Oracle file type and can be one of the file types shown in Table 7–3.

- *filetypetag* is type-specific information about the file and can be one of the tags shown in Table 7–3.

- *file.incarnation* is the file/incarnation pair, used to ensure uniqueness.

An example of a fully qualified Oracle ASM filename is:

+data/orcl/controlfile/Current.256.541956473

Table 7–3   Oracle file types and Oracle ASM file type tags

Oracle ASM File Type	Description	Oracle ASM File Type Tag	Comments
CONTROLFILE	Control files and backup control files	Current	Archive
		Backup	Backup
DATAFILE	Data files and data file copies	*tblspname*	Name of the tablespace into which the file is added
ONLINELOG	Online logs	group_*loggroup#*	
ARCHIVELOG	Archive logs	thread_*thread#*_seq_*sequence#*	
TEMPFILE	Tempfiles	*tsname*	Name of the tablespace into which the file is added

Table 7–3   (Cont.) Oracle file types and Oracle ASM file type tags

Oracle ASM File Type	Description	Oracle ASM File Type Tag	Comments
BACKUPSET	Data file and archive log backup pieces; data file incremental backup pieces	*spfile?_timestamp*	*spfile?* can take one of two values: *s* indicates that the backup set includes the spfile; *n* indicates that the backup set does not include the spfile. *timestamp* is the time that the backup is started.
PARAMETERFILE	Persistent parameter files	spfile	Name of the SPFILE
DATAGUARDCONFIG	Data Guard configuration file	*db-unique-name*	Data Guard uses the value of the DB_UNIQUE_NAME initialization parameter.
FLASHBACK	Flashback logs	log_*flashbacklog#*	
CHANGETRACKING	Block change tracking data	ctf	Used during incremental backups
DUMPSET	Data Pump dumpset	user_*obj#_file#*	Dump set files encode the user name, the job number that created the dump set, and the file number as part of the tag.
XTRANSPORT	Data file convert	*tblspname*	Name of the tablespace where the Oracle ASM metadata is added
AUTOBACKUP	Automatic backup files	*spfile?_timestamp*	*spfile?* can take one of two values: *s* indicates that the backup set includes the spfile; *n* indicates that the backup set does not include the spfile. *timestamp* is the time that the backup is started.
ASMPARAMETERFILE	Oracle ASM persistent parameter files	*spfile*	Name of the Oracle ASM SPFILE
OCRFILE	Oracle Clusterware Registry files	*ocrfile*	Name of the OCR files

Alias Oracle ASM Filename Forms

You can use Alias Oracle ASM filenames both for referencing existing Oracle ASM files and for creating new Oracle ASM files. Alias names start with the disk group name preceded by a plus sign, after which you specify a name string. Alias filenames are implemented using a hierarchical directory structure, with the slash (/) or backslash (\) character separating name components. You can create an alias in any Oracle ASM directory generated by Oracle Managed Files (system-generated) or created by a user. An alias must include the disk group name; you cannot create an alias at the root level (+).

When you create an Oracle ASM file with an alias filename, the file is created with a fully qualified name, and the alias filename is additionally created. You can then access the file with either name.

Alias Oracle ASM filenames are distinguished from fully qualified filenames or numeric filenames because they do not end in a dotted pair of numbers. It is an error to attempt to create an alias that ends in a dotted pair of numbers, such as in the format USERS.259.685366091. Examples of Oracle ASM alias filenames are:

```
+data/orcl/control_file1
+fra/recover/second.dbf
```

Oracle Database references database files by their alias filenames, but only if you create the database files with aliases. If you create database files without aliases and then add aliases later, the database references the files by their fully qualified filenames. The following are examples of how the database uses alias filenames:

- Alias filenames appear in V$ views. For example, if you create a tablespace and use an alias filename for the data file, the V$DATAFILE view shows the alias filename.

- When a control file points to data files and online redo log files, it can use alias filenames.

- The CONTROL_FILES initialization parameter can use the alias filenames of the control files. Database Configuration Assistant (DBCA) creates control files with alias filenames.

Note: Files created using an alias filename are not considered Oracle Managed Files and might require manual deletion in the future if they are no longer needed.

For more information about alias filenames, see "Managing Alias Names for Oracle ASM Filenames" on page 7-9.

Creating a Tablespace in Oracle ASM: Using a Data File with an Alias Name

Example 7–1 creates an undo tablespace with a data file that has an alias name, and with attributes that are set by the user-defined template my_undo_template. This example assumes that a directory has been created in disk group data to contain the alias name and that the user-defined template exists.

Example 7–1   Creating a tablespace using an alias name

```
CREATE UNDO TABLESPACE myundo
     DATAFILE '+data(my_undo_template)/orcl/my_undo_ts' SIZE 200M;
```

Because an alias is used to create the data file, the file is not an Oracle Managed Files (OMF) file and the file is not automatically deleted when the tablespace is dropped. To drop the file manually after the tablespace has been dropped, use the following SQL statement:

```
ALTER DISKGROUP data DROP FILE '+data/orcl/my_undo_ts';
```

Alias Oracle ASM Filename with Template Form

An alias Oracle ASM filename with template is used only for Oracle ASM file creation operations. It has the following format:

```
+diskgroup(templatename)/alias
```

Alias filenames with template behave identically to alias filenames except that a file created with an alias filename with template receives the attributes specified by the named template. The template must belong to the disk group that the file is being created in. See Example 7–1 on page 7-6.

The creation and maintenance of Oracle ASM templates is discussed in "Managing Disk Group Templates" on page 7-15.

An example of an alias Oracle ASM filename with template is:

```
+data(mytemplate)/config1
```

Explicitly specifying a template name, as in the previous example, overrides the system default template for the type of file being created.

> **Note:** Files created using an alias filename with template are not considered Oracle Managed Files and might require manual deletion in the future if they are no longer needed.

Incomplete Oracle ASM Filename Form

Incomplete Oracle ASM filenames are used only for file creation operations and are used for both single and multiple file creation. They consist only of the disk group name. Oracle ASM uses a system default template to determine the Oracle ASM file mirroring and striping attributes. The system template that is used is determined by the file type that is being created. For example, if you are creating a data file for a tablespace, then the DATAFILE template is used.

An example of using an incomplete Oracle ASM filename is setting the DB_CREATE_FILE_DEST initialization parameter to:

```
+data
```

With this setting, every time you create a tablespace, a data file is created in the disk group data, and each data file is assigned a different fully qualified name. See "Creating Oracle ASM Files Using a Default File Location for Disk Group Specification" on page 7-7 for more information.

Incomplete Oracle ASM Filename with Template Form

Incomplete Oracle ASM filenames with templates are used only for file creation operations and are used for both single and multiple file creation. They consist of the disk group name followed by the template name in parentheses. When you explicitly specify a template in a file name, Oracle ASM uses the specified template instead of the default template for that file type to determine attributes for the file.

An example of using an incomplete Oracle ASM filename with template is setting the DB_CREATE_FILE_DEST initialization parameter to:

```
+data(mytemplate)
```

Creating and Referencing Oracle ASM Files in the Database

An Oracle ASM file is an Oracle Managed File unless you created the file using an alias. Any Oracle Managed File is automatically deleted when it is no longer needed.

This section contains the following topics:

- Creating Oracle ASM Files Using a Default File Location for Disk Group Specification
- Using Oracle ASM Filenames in SQL Statements

Creating Oracle ASM Files Using a Default File Location for Disk Group Specification

Using the Oracle Managed Files feature for operating system files, you can specify a directory as the default location for the creation of data files, temporary files, redo log

files, and control files. Using the Oracle Managed Files feature for Oracle ASM, you can specify a disk group, in the form of an incomplete Oracle ASM filename, as the default location for creation of these files, and additional types of files, including archived log files. As for operating system files, the name of the default disk group is stored in an initialization parameter and is used whenever a file specification (for example, DATAFILE clause) is not explicitly specified during file creation.

The following initialization parameters accept the multiple file creation context form of Oracle ASM filenames as a destination:

- DB_CREATE_FILE_DEST

  Specifies the default disk group location in which to create data and temp files.

- DB_CREATE_ONLINE_LOG_DEST_*n*

  Specifies the default disk group location in which to create redo log and control files.

- DB_RECOVERY_FILE_DEST

  Specifies a default disk group for a fast recovery area that contains a copy of the control file and redo log files if this parameter is specified and DB_CREATE_ONLINE_ LOG_DEST_*n* and CONTROL_FILES are not specified.

- CONTROL_FILES

  Specifies a disk group in which to create control files.

The following initialization parameters accept the multiple file creation context form of the Oracle ASM filenames and Oracle ASM directory names as a destination:

- LOG_ARCHIVE_DEST_*n*

  Specifies a default disk group or Oracle ASM directory as destination for archiving redo log files

- LOG_ARCHIVE_DEST

  Optional parameter to use to specify a default disk group or Oracle ASM directory as destination for archiving redo log files. Use when specifying only one destination.

- STANDBY_ARCHIVE_DEST

  Relevant only for a standby database. It specifies a default disk group or Oracle ASM directory that is the location of archive logs arriving from a primary database. Not discussed in this book. This parameter has been deprecated. See *Oracle Data Guard Concepts and Administration*.

The following CREATE TABLESPACE SQL statement illustrates how an Oracle ASM file, in this case a data file, might be created in a default disk group. This example assumes that the DB_CREATE_FILE_DEST initialization parameter has been set to +data.

```
SQL> CREATE TABLESPACE mytblspace;
```

Oracle ASM automatically creates and manages the data file for mytblspace on Oracle ASM disks in the disk group data. File extents are stored using the attributes defined by the default template for a data file.

See Also:

- *Oracle Database Reference* for information about initialization parameters

- *Oracle Database Administrator's Guide* for information about initialization parameter files

Using Oracle ASM Filenames in SQL Statements

You can specify Oracle ASM filenames in the file specification clause of your SQL statements. If you are creating a file for the first time, then use the creation form of an Oracle ASM filename. If the Oracle ASM file exists, then you must use the reference form of the filename, and if you are trying to re-create the file, you must add the REUSE keyword so that the space is reused for the new file. This usage might occur when, for example, trying to re-create a control file, as shown in "Creating Control Files in Oracle ASM" on page A-3.

If a reference form is used with the REUSE keyword and the file does not exist, an error results.

Example 7–2 is an example of specifying an Oracle ASM filename in a SQL statement. In this case, it is used in the file creation form.

Example 7–2   Using an Oracle ASM File name in a SQL statement

```
CREATE TABLESPACE mytblspace DATAFILE '+data' SIZE 200M AUTOEXTEND ON;
```

The tablespace mytblspace is created and comprises one data file of size 200 MB contained in the disk group data. The data file is set to auto-extensible with an unlimited maximum size. You can use an AUTOEXTEND clause to override this default.

For more examples of the use of Oracle ASM filename in SQL statements, see Appendix A, "Creating Database Files in Oracle ASM Using SQL*Plus".

Managing Alias Names for Oracle ASM Filenames

Alias names, or aliases, are intended to provide a more user-friendly means of referring to Oracle ASM files, rather than using the system-generated filenames.

You can create an alias for a file when you create it in the database, or you can add an alias to an existing file using the ADD ALIAS clause of the ALTER DISKGROUP statement. You can create an alias in any system-generated or user-created Oracle ASM directory. An alias must include the disk group name; you cannot create an alias at the root level (+).

For more information about creating aliases with ASMCMD, refer to "mkalias" on page 12-27.

This section contains the following sections:

- Adding an Alias Name for an Oracle ASM Filename

- Renaming an Alias Name for an Oracle ASM Filename

- Dropping an Alias Name for an Oracle ASM Filename

- Dropping Files and Associated Aliases from a Disk Group

Adding an Alias Name for an Oracle ASM Filename

Use the `ADD ALIAS` clause of the `ALTER DISKGROUP` statement to create an alias name for an Oracle ASM filename. The alias name must consist of the full directory path and the alias itself.

Example 7–3 adds a new alias name for a system-generated file name.

Example 7–3   Adding an alias name for an Oracle ASM filename

```
ALTER DISKGROUP data ADD ALIAS '+data/orcl/second.dbf'
    FOR '+data/orcl/datafile/mytable.342.123456789';
```

Renaming an Alias Name for an Oracle ASM Filename

Use the `RENAME ALIAS` clause of the `ALTER DISKGROUP` statement to rename an alias for an Oracle ASM filename. The old and the new alias names must consist of the full directory paths of the alias names.

Example 7–4 renames an alias.

Example 7–4   Renaming an alias name for an Oracle ASM filename

```
ALTER DISKGROUP data RENAME ALIAS '+data/orcl/datafile.dbf'
    TO '+data/payroll/compensation.dbf';
```

Renaming an alias does not update the database reference to the alias.

Dropping an Alias Name for an Oracle ASM Filename

Use the `DROP ALIAS` clause of the `ALTER DISKGROUP` statement to drop an alias for an Oracle ASM filename. The alias name must consist of the full directory path and the alias itself. The underlying file to which the alias refers is unchanged.

Example 7–5 drops an alias.

Example 7–5   Dropping an alias name for an Oracle ASM filename

```
ALTER DISKGROUP data DROP ALIAS '+data/payroll/compensation.dbf';
```

Example 7–6 fails because it attempts to drop a system-generated filename and this syntax is not allowed.

Example 7–6   Dropping an alias name for an Oracle ASM filename

```
ALTER DISKGROUP data
    DROP ALIAS '+data/orcl/datafile/mytable.342.123456789';
```

Dropping Files and Associated Aliases from a Disk Group

You can delete Oracle ASM files and their associated aliases from a disk group using the `DROP FILE` clause of the `ALTER DISKGROUP` statement. You must use a fully qualified filename, or an alias name when specifying the file to delete. The underlying file on the file system is not dropped when the alias is dropped.

Some reasons why you may want to delete files include:

- A file created using aliases is not an Oracle Managed File. Consequently, it is not automatically deleted.

- A point-in-time-recovery of a database might restore the database to a time before a tablespace was created. The restore does not delete the tablespace, but there is no reference to the tablespace or its data file in the restored database. You could manually delete the data file.

In Example 7–7, the alias name for the file is used to delete both the file and the alias from a disk group.

Example 7–7   Dropping a file and alias from a disk group using the alias name

```
ALTER DISKGROUP data DROP FILE '+data/payroll/compensation.dbf';
```

In Example 7–8, the Oracle Managed Files (system-generated) file name is used to drop the file and any associated alias.

Example 7–8   Dropping a file and alias from a disk group using the system-generated name

```
ALTER DISKGROUP data
     DROP FILE '+data/orcl/datafile/mytable.342.123456789';
```

Managing Disk Group Directories

Oracle ASM disk groups contain a system-generated hierarchical directory structure for storing Oracle ASM files. The system-generated filename that Oracle ASM assigns to each file represents a path in this directory hierarchy. The following is an example of a system-generated filename:

```
+data/orcl/CONTROLFILE/Current.256.541956473
```

The plus sign represents the root of the Oracle ASM file system. The data directory is the parent directory for all files in the data disk group. The orcl directory is the parent directory for all files in the orcl database, and the CONTROLFILE directory contains all control files for the orcl database.

You can create your own directories within this hierarchy to store aliases that you create. Thus, in addition to having user-friendly alias names for Oracle ASM files, you can have user-friendly paths to those names.

This section describes how to use the ALTER DISKGROUP statement to create a directory structure for aliases. It also describes how you can rename a directory or drop a directory. This section contains the following topics:

- Creating a Directory
- Renaming a Directory
- Dropping a Directory

Creating a Directory

Use the ADD DIRECTORY clause of the ALTER DISKGROUP statement to create a hierarchical directory structure for alias names for Oracle ASM files. Use the slash (/) or backslash (\) character to separate components of the directory path. The directory path must start with the disk group name, preceded by a plus sign (+), followed by any subdirectory names of your choice.

The parent directory must exist before attempting to create a subdirectory or alias in that directory. A directory must be created below the disk group level.

Example 7–9 creates a hierarchical directory for disk group `data`, which can contain, for example, the alias name `+data/orcl/control_file1`.

Example 7–9   Creating a new directory

```
ALTER DISKGROUP data ADD DIRECTORY '+data/orcl';
```

Assuming no subdirectory exists under the directory `+data/orcl`, the SQL statement fails in Example 7–10.

Example 7–10   Creating a new subdirectory

```
ALTER DISKGROUP data
    ADD DIRECTORY '+data/orcl/first_dir/second_dir';
```

Renaming a Directory

The `RENAME DIRECTORY` clause of the `ALTER DISKGROUP` statement enables you to rename a directory. System-created directories (those containing system-generated names) cannot be renamed. The root level (disk group name) cannot be renamed.

Example 7–11 renames a directory.

Example 7–11   Renaming a directory

```
ALTER DISKGROUP data RENAME DIRECTORY '+data/mydir'
    TO '+data/yourdir';
```

Dropping a Directory

You can delete a directory using the `DROP DIRECTORY` clause of the `ALTER DISKGROUP` statement. You cannot drop a system-created directory. You cannot drop a directory containing alias names unless you also specify the `FORCE` clause.

Example 7–12 deletes a directory along with its contents.

Example 7–12   Dropping a directory

```
ALTER DISKGROUP data DROP DIRECTORY '+data/yourdir' FORCE;
```

Accessing Oracle ASM Files with the XML DB Virtual Folder

Oracle ASM files and directories can be accessed through a virtual folder in the XML DB repository. The repository path to the virtual folder is `/sys/asm`. The folder is virtual because its contents do not actually reside in the repository; they exist as normal Oracle ASM files and directories. For example, `/sys/asm` provides a means to access and manipulate the Oracle ASM files and directories with programmatic APIs such as the `DBMS_XDB` package and with XML DB protocols such as FTP and HTTP/WebDAV. You can also use the ASMCMD `cp` command to copy files. See "cp" on page 12-21.

A typical use for this capability might be to view `/sys/asm` as a Web Folder in a graphical user interface (with the WebDAV protocol), and then copy a Data Pump dumpset from an Oracle ASM disk group to an operating system file system by dragging and dropping.

You must log in as a user other than `SYS` and you must have been granted the `DBA` role to access `/sys/asm` with XML DB protocols.

> **Note:** For security reasons, FTP is disabled, by default. This is because the IETF FTP protocol specification requires that passwords be transmitted in clear text. Disabling is done by configuring the FTP server port as zero (0). To enable FTP, set the `ftp-port` parameter to the FTP port to use, such as `2100`.

See Also:

- *Oracle XML DB Developer's Guide* for information about Oracle XML DB, including additional ways to configure port numbers for the XML DB protocol servers

- *Oracle Database PL/SQL Packages and Types Reference* for information about the `DBMS_XDB` package

Inside /sys/asm

The Oracle ASM virtual folder is created by default during XML DB installation. If the database is not configured to use Oracle ASM, the folder is empty and no operations are permitted on it.

The Oracle ASM virtual folder contains folders and subfolders that follow the hierarchy defined by the structure of an Oracle ASM fully qualified file name. Figure 7–1 illustrates an example of this hierarchy, which for simplicity, excludes directories created for aliases.

The folder `/sys/asm` contains one subfolder for every mounted disk group, and each disk group folder contains one subfolder for each database that uses the disk group. In addition, a disk group folder might contain files and folders corresponding to aliases created by the administrator. Continuing the hierarchy, the database folders contain file type folders, which contain the Oracle ASM files.

Figure 7–1   Hierarchical Structure of Oracle ASM Folders in an XML DB Installation
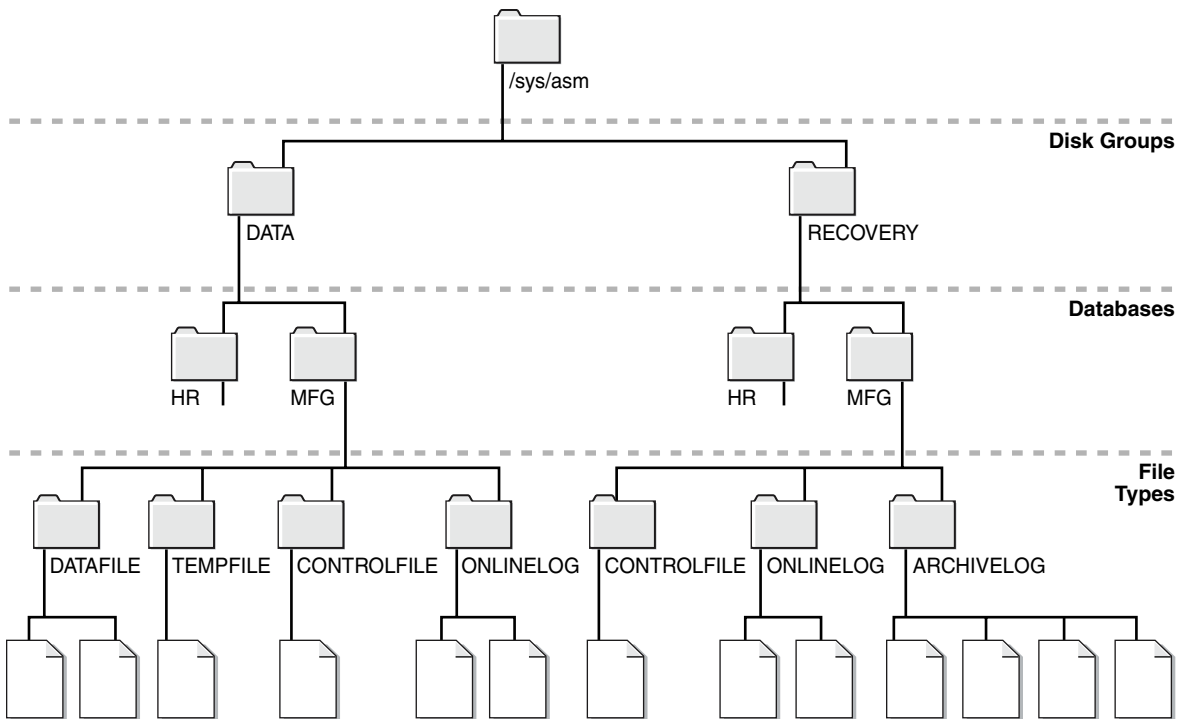
The following are usage restrictions on /sys/asm:

- You cannot create hard links to existing Oracle ASM files or directories with APIs such as DBMS_XDB.LINK.

- You cannot rename (move) an Oracle ASM file to another disk group or to a directory outside Oracle ASM.

You can use the directory /sys/asm for storing the names of disk groups. You cannot store other files in this directory. Within the disk group directories under /sys/asm, such as /sys/asm/DATA, you can only store database files in these sub-directories. Oracle ASM rejects attempts to store non-database files in these directories.

Using DBMS_FILE Transfer Utility for Oracle ASM

The DBMS_FILE_TRANSFER package provides procedures to copy Oracle ASM files within a database or to transfer binary files between databases that use Oracle ASM. The DBMS_FILE_TRANSFER package has the following procedures:

- COPY_FILE—Reads a file from a source directory and creates a copy of the file in a destination directory. The source and destination directories can both be in a local file system or in an Oracle ASM disk group. You can also use this procedure to copy between a local file system and an Oracle ASM disk group; the copy operation is valid in either direction.

- GET_FILE—Contacts a remote database to read a remote file and then creates a copy of the file in the local file system or Oracle ASM disk group.

- PUT_FILE—Reads a local file or Oracle ASM disk group and contacts a remote database to create a copy of the file in the remote file system.

> **See Also:** *Oracle Database PL/SQL Packages and Types Reference* for more information about the `DBMS_FILE_TRANSFER` package

Managing Disk Group Templates

This section describes how to manage disk group templates under the following topics:

- [Template Attributes](#)
- [Adding Templates to a Disk Group](#)
- [Modifying a Disk Group Template](#)
- [Dropping Templates from a Disk Group](#)
- [Creating Tablespaces in Oracle ASM: Specifying Attributes with Templates](#)

Templates are used to set attributes of files created in an Oracle ASM disk group. When a file is created, redundancy and striping attributes are set for that file based on an explicitly named template or the system template that is the default template for the file type.

When a disk group is created, Oracle ASM creates a set of default templates for that disk group. The set consists of one template for each file type (data file, control file, redo log file, and so on) that is supported by Oracle ASM. For example, a template named `ONLINELOG` provides the default file redundancy and striping attributes for all redo log files written to Oracle ASM disks. Default template settings depend on the disk group type. The default template for data files for a normal redundancy disk group sets two-way mirroring, while the corresponding default template in a high redundancy disk group sets three-way mirroring. You can modify these default templates.

For example, default redundancy for the online redo log files (`ONLINELOG` template) for a normal redundancy disk group is `MIRROR`. In Example 4–1 on page 4-6, this setting means that when one copy of a redo log file extent is written to a disk in failure group `controller1`, a mirrored copy of the file extent is written to a disk in failure group `controller2`. To support the default mirroring of a normal redundancy disk group, at least two failure groups must be defined.

Table 7–6 on page 7-16 lists the default templates and the attributes that are associated to matching files. As the table shows, the initial redundancy value of each default template depends on the type of disk group that the template belongs to.

Using clauses of the `ALTER DISKGROUP` statement, you can add new templates to a disk group, modify existing ones, or drop templates. The reason to add templates is to create the right combination of attributes to meet unique requirements. You can then reference a template name when creating a file, thereby assigning desired attributes based on an individual file rather than on the file type.

The `V$ASM_TEMPLATE` view lists all of the templates known to the Oracle ASM instance. For an example of the information displayed in the `V$ASM_TEMPLATE` view, see Example 6–10 on page 6-5.

Template Attributes

Table 7–4 shows the permitted striping attribute values and allocation unit size chunks. These values correspond to the `STRIPE` column of `V$ASM_TEMPLATE`.

Table 7–4   Permitted values for Oracle ASM template striping attribute

Striping Attribute Value	Description
FINE	Striping in 128 KB chunks.
COARSE	Striping in 1 MB chunks.

Table 7–5 shows the permitted redundancy values for Oracle ASM templates. These values correspond to the REDUND column of V$ASM_TEMPLATE.

Table 7–5   Permitted values for Oracle ASM template redundancy attribute

Redundancy Attribute Value	Resulting Mirroring in Normal Redundancy Disk Group	Resulting Mirroring in High Redundancy Disk Group	Resulting Mirroring in External Redundancy Disk Group
MIRROR	Two-way mirroring	Three-way mirroring	(Not allowed)
HIGH	Three-way mirroring	Three-way mirroring	(Not allowed)
UNPROTECTED	No mirroring	(Not allowed)	No mirroring

Table 7–6 shows the initial attribute settings for the default templates. The type of mirroring associated with the **Mirroring** column for normal, high, and external redundancy disk groups is specified in Table 7–5. For example, the **Mirror** column for high redundancy disk groups displays MIRROR in Table 7–6. In Table 7–5, the MIRROR value for high redundancy disk groups corresponds to Three-way mirroring.

Table 7–6   Oracle ASM system default templates attribute settings

Template Name	File Striping	Mirroring, Normal Redundancy Disk Group	Mirroring, High Redundancy Disk Group	Mirroring, External Redundancy Disk Group	Primary Region	Mirror Region
CONTROLFILE	FINE	HIGH	MIRROR	UNPROTECTED	COLD	MIRRORCOLD
DATAFILE	COARSE	MIRROR	MIRROR	UNPROTECTED	COLD	MIRRORCOLD
ONLINELOG	COARSE	MIRROR	MIRROR	UNPROTECTED	COLD	MIRRORCOLD
ARCHIVELOG	COARSE	MIRROR	MIRROR	UNPROTECTED	COLD	MIRRORCOLD
TEMPFILE	COARSE	MIRROR	MIRROR	UNPROTECTED	COLD	MIRRORCOLD
BACKUPSET	COARSE	MIRROR	MIRROR	UNPROTECTED	COLD	MIRRORCOLD
PARAMETERFILE	COARSE	MIRROR	MIRROR	UNPROTECTED	COLD	MIRRORCOLD
DATAGUARDCONFIG	COARSE	MIRROR	MIRROR	UNPROTECTED	COLD	MIRRORCOLD
FLASHBACK	COARSE	MIRROR	MIRROR	UNPROTECTED	COLD	MIRRORCOLD
CHANGETRACKING	COARSE	MIRROR	MIRROR	UNPROTECTED	COLD	MIRRORCOLD
DUMPSET	COARSE	MIRROR	MIRROR	UNPROTECTED	COLD	MIRRORCOLD
XTRANSPORT	COARSE	MIRROR	MIRROR	UNPROTECTED	COLD	MIRRORCOLD
AUTOBACKUP	COARSE	MIRROR	MIRROR	UNPROTECTED	COLD	MIRRORCOLD
ASMPARAMETERFILE	COARSE	MIRROR	MIRROR	UNPROTECTED	COLD	MIRRORCOLD
OCRFILE	COARSE	MIRROR	MIRROR	UNPROTECTED	COLD	MIRRORCOLD

Table 7–7 shows the permitted primary and mirror region values for Oracle ASM templates. These values correspond to the PRIMARY_REGION and MIRROR_REGION columns of V$ASM_TEMPLATE. For more information about disk regions, see "Intelligent Data Placement" on page 4-16.

Table 7–7   Permitted values for Oracle ASM template primary and mirror region attribute

Primary and Mirror Region Attribute Value	Description
COLD, MIRRORCOLD	Use the inner most tracks (closest to spindle) on the disk drive.
HOT, MIRRORHOT	Use the outermost tracks which have greater speed and higher bandwidth.

Adding Templates to a Disk Group

To add a template to a disk group, use the ADD TEMPLATE clause of the ALTER DISKGROUP statement.

If the name of your new template is not listed in Table 7–6, then it is not used as a default template for database file types. To use the template, you must reference its name when creating a file.

When adding a template to a disk group, the attributes are optional. If no redundancy attribute is specified, then the value defaults to MIRROR for a normal redundancy disk group, HIGH for a high redundancy disk group, and UNPROTECTED for an external redundancy disk group. If you do not specify a striping attribute, then the value defaults to COARSE. If you do not specify a value for the primary or mirror region attributes, then the value defaults to COLD and MIRRORCOLD respectively.

Example 7–13 creates a new template named reliable for the normal redundancy disk group data.

Example 7–13   Adding a template with high redundancy to a disk group

```
ALTER DISKGROUP data ADD TEMPLATE reliable ATTRIBUTES (HIGH FINE);
```

Example 7–14 creates a new template named unreliable that specifies UNPROTECTED (no mirroring) for files.

Example 7–14   Adding a template with external redundancy to a disk group

```
ALTER DISKGROUP data ADD TEMPLATE unreliable ATTRIBUTES (UNPROTECTED);
```

> **Note:**   Oracle discourages using unprotected files unless you have implemented hardware mirroring. The previous example is presented only to further illustrate how the attributes for templates are set.

> **See Also:**   *Oracle Database SQL Language Reference* for more information about the ALTER DISKGROUP...ADD TEMPLATE command

Modifying a Disk Group Template

The MODIFY TEMPLATE clause of the ALTER DISKGROUP statement enables you to modify the attribute specifications of an existing system default or user-defined disk group template. Only specified template attributes are changed. Unspecified properties retain

their current value. When you modify an existing template, only new files created by the template reflect the attribute changes. Existing files maintain their attributes.

Example 7–15 changes the striping attribute specification of the reliable template for disk group data.

Example 7–15   Modifying a disk group template

```
ALTER DISKGROUP data MODIFY TEMPLATE reliable
    ATTRIBUTES (COARSE);
```

Dropping Templates from a Disk Group

Use the DROP TEMPLATE clause of the ALTER DISKGROUP statement to drop one or more templates from a disk group. You can only drop templates that are user-defined; you cannot drop system default templates.

Example 7–16 drops the previously created template unreliable from data:

Example 7–16   Dropping a template from a disk group

```
ALTER DISKGROUP data DROP TEMPLATE unreliable;
```

Creating Tablespaces in Oracle ASM: Specifying Attributes with Templates

Use the SQL ALTER SYSTEM and CREATE TABLESPACE statements to create a tablespace that uses a user-defined template to specify the attributes of the data file.

Example 7–17 assumes that the template (mytemplate) has been defined.

Example 7–17   Using a user-defined template to specify attributes

```
ALTER SYSTEM SET DB_CREATE_FILE_DEST = '+data(mytemplate)';

CREATE TABLESPACE mytblspace;
```

8

Performing Oracle ASM Data Migration with RMAN

This chapter describes how to migrate data into and out of Oracle Automatic Storage Management (Oracle ASM) storage with Recovery Manager (RMAN).

This chapter includes the following topics:

- Overview of Oracle ASM Data Migration
- Preparing to Migrate the Database to Oracle ASM Using RMAN
- Migrating the Database to Oracle ASM Using RMAN
- Migrating a Database from Oracle ASM to Alternative Storage
- Moving Data Files Between Oracle ASM Disk Groups Using RMAN

> **See Also:**   *Oracle Database Backup and Recovery User's Guide* for complete information about using RMAN

Overview of Oracle ASM Data Migration

This section explains the basic concepts and tasks involved in migrating data to and from Oracle ASM.

This section includes the following topics:

- Purpose of Oracle ASM Data Migration
- Basic Concepts of Oracle ASM Data Migration
- Basics Steps of Data Migration to Oracle ASM Using RMAN

Purpose of Oracle ASM Data Migration

Alternatives to Oracle ASM storage include file systems, raw disks, and SAN configurations. Oracle ASM includes numerous benefits over these storage alternatives, including performance optimization, redundancy protection, and load balancing. You do not need a third-party Logical Volume Manager because Oracle ASM manages disks for you. Oracle Real Application Clusters (Oracle RAC) databases benefit from Oracle ASM because it provides ready-made shared storage.

If a database currently uses a storage system other than Oracle ASM, then you can migrate all or part of the database into Oracle ASM, thereby simplifying database administration. You can also migrate a fast recovery area to Oracle ASM.

Native operating system commands such as Linux `cp` or Windows `COPY` cannot write or read files in Oracle ASM storage. Because RMAN can read and write Oracle ASM

files, you can use RMAN to copy data files into and out of Oracle ASM storage or between Oracle ASM disk groups. This technique is useful if you must store backups on user-managed disks.

Basic Concepts of Oracle ASM Data Migration

You can migrate data to Oracle ASM with RMAN even if you are not using RMAN as your primary backup tool. The migration requires one RMAN database backup.

If you have sufficient disk space to hold the entire database both in Oracle ASM and alternative storage systems, then you can move a database directly into Oracle ASM. If you do not have sufficient storage, then you can back the database up to tape, create an Oracle ASM disk group that uses old disk space, and restore the database from tape to Oracle ASM.

After you set the location of the new recovery area, existing backups remain in the old recovery area and count against the total disk quota of the recovery area. The backups are deleted from the old recovery area when space is needed. These backups are usable by RMAN. It is not necessary to move legacy backups to the new Oracle ASM recovery area unless you need disk space. To free space consumed by files in the old recovery area, you can back them up to tape or migrate them to the Oracle ASM recovery area.

> **Note:** A foreign archived redo log is a log received by a logical standby database for a LogMiner session. Foreign archived redo logs cannot be migrated. Unlike normal archived logs, foreign archived logs have a different internal database identifier (DBID). For this reason, they cannot be backed up or restored on a logical standby database.

Migrating a database from Oracle ASM to an alternative storage system is similar to migration from an alternative storage system to Oracle ASM. The primary change is to modify each step to refer to file locations in the alternative storage system.

For information about migrating the database to Oracle ASM with Enterprise Manager, see Chapter 9, "Administering Oracle ASM with Oracle Enterprise Manager".

Basics Steps of Data Migration to Oracle ASM Using RMAN

This section discusses the process of migrating the entire database and fast recovery area from alternative storage to Oracle ASM using RMAN.

The fast recovery area is an optional disk location that you can use to store recovery-related files such as control file and online redo log copies, archived redo log files, flashback logs, and RMAN backups. Oracle Database and RMAN manage the files in the fast recovery area automatically. You can specify the disk quota, which is the user-specified maximum size of the fast recovery area. When the disk quota is reached, Oracle automatically deletes files that are no longer needed.

Flashback logs are Oracle-generated logs used to perform flashback database operations. The database can only write flashback logs to the fast recovery area. Flashback logs are written sequentially and are not archived. They cannot be backed up to disk.

To migrate the entire database and fast recovery area from alternative storage to Oracle ASM, perform the following steps:

1. Back up the database and server parameter file, and disable Oracle Flashback Database.

   The Oracle Flashback Database option returns the entire database to a prior consistent System Change Number (SCN) with the FLASHBACK DATABASE command in RMAN or SQL. A database flashback is different from traditional media recovery because it does not involve the restore of physical files, instead restoring your current data files to past states using saved images of changed data blocks. This feature uses flashback logs and archived redo logs.

   This step is described in "Preparing to Migrate the Database to Oracle ASM Using RMAN" on page 8-3.

2. Restore files to Oracle ASM, recover the database, and optionally migrate the fast recovery area to Oracle ASM.

   This step is described in "Migrating the Database to Oracle ASM Using RMAN" on page 8-5.

To migrate files from Oracle ASM to alternative storage, see "Migrating a Database from Oracle ASM to Alternative Storage" on page 8-9.

Preparing to Migrate the Database to Oracle ASM Using RMAN

This section explains how to prepare the database for migration. This section makes the following assumptions:

- You want to migrate the database to two Oracle ASM disk groups: +DATA for the database and +FRA for the fast recovery area.

- The database to be migrated to Oracle ASM storage is named mydb.

To prepare the database for Oracle ASM migration:

1. If the COMPATIBLE initialization parameter setting for the database is less than 11.0.0, then make any read-only transportable tablespaces read/write.

   Read-only transportable tablespaces cannot be migrated because RMAN cannot back them up.

2. If the database is a physical standby database, and if managed recovery is started, then stop managed recovery.

   A physical standby database is a copy of a production database that you can use for disaster protection.

   For example, connect SQL*Plus to the database with SYSDBA privileges, and run the following statement to stop managed recovery:

   ```
   SQL> ALTER DATABASE RECOVER MANAGED STANDBY DATABASE CANCEL;
   ```

   Keep this terminal window open.

3. Copy the server parameter file or initialization parameter file to a temporary location.

   The following example uses an operating system utility to copy the server parameter file:

   ```
   $ cp spfileMYDB.ora orig_spfileMYDB.ora
   ```

4. In a new terminal window, start RMAN session and connect as TARGET to the database to be migrated. Optionally, connect to a recovery catalog.

5. Back up the data files to the Oracle ASM disk group.

The following example uses a `RUN` command to make a level 0 incremental backup and allocates four channels to increase the backup speed. A level 0 incremental backup is an RMAN incremental backup that backs up all data blocks in the data files being backed up. An incremental backup at level 0 is identical in content to a full backup, but unlike a full backup the level 0 backup is considered a part of the incremental backup strategy.

An incremental backup is a RMAN backup in which only modified blocks are backed up. Incremental backups are classified by **level**. A level 0 incremental backup performs the same function as a full backup in that they both back up all blocks that have ever been used. The difference is that a full backup does not affect blocks backed up by subsequent incremental backups, whereas an incremental backup affects blocks backed up by subsequent incremental backups.

A full backup is a non-incremental RMAN backup. Full does not refer to how much of the database is backed up, but to the fact that the backup is not incremental. Consequently, you can make a full backup of one data file.

Increase or decrease this number accordingly. The format clause specifies `+DATA`, which is the name of the Oracle ASM disk group to be used for storing the database.

```
RUN
{
  ALLOCATE CHANNEL dev1 DEVICE TYPE DISK;
  ALLOCATE CHANNEL dev2 DEVICE TYPE DISK;
  ALLOCATE CHANNEL dev3 DEVICE TYPE DISK;
  ALLOCATE CHANNEL dev4 DEVICE TYPE DISK;
  BACKUP AS COPY
    INCREMENTAL LEVEL 0
    DATABASE
    FORMAT '+DATA'
    TAG 'ORA_ASM_MIGRATION';
}
```

6. If block change tracking is enabled for the database, then optionally make a level 1 incremental backup that you can use later to recover the database copy.

Block change tracking is a database option that causes Oracle to track data file blocks affected by each database update. The tracking information is stored in a block change tracking file. When block change tracking is enabled, RMAN uses the record of changed blocks from the change tracking file to improve incremental backup performance by only reading those blocks known to have changed, instead of reading data files in their entirety.

The following example makes an incremental level 1 copy of the level 0 backup created in the previous step:

```
RUN
{
  ALLOCATE CHANNEL dev1 DEVICE TYPE DISK;
  ALLOCATE CHANNEL dev2 DEVICE TYPE DISK;
  ALLOCATE CHANNEL dev3 DEVICE TYPE DISK;
  ALLOCATE CHANNEL dev4 DEVICE TYPE DISK;
  BACKUP INCREMENTAL LEVEL 1
    FOR RECOVER OF COPY WITH TAG 'ORA_ASM_MIGRATION'
    DATABASE;
}
```

7. If the database is in `ARCHIVELOG` mode, and if the database is open, then archive the online logs.

The following example uses the `SQL` command to archive the current redo logs:

```
RMAN> SQL "ALTER SYSTEM ARCHIVE LOG CURRENT";
```

8. If the database instance is currently using a server parameter file, then back it up.

The following example backs up the server parameter file:

```
RMAN> BACKUP AS BACKUPSET SPFILE;
```

9. If block change tracking is enabled, then disable it.

The following command disables block change tracking:

```
RMAN> SQL "ALTER DATABASE DISABLE BLOCK CHANGE TRACKING";
```

10. If Flashback Database is enabled, then disable it and drop any guaranteed restore points.

> **Note:** If you are not migrating the fast recovery area, then skip this step.

Disabling Oracle Flashback Database is necessary because you cannot migrate flashback logs to Oracle ASM. The following command disables Flashback Database:

```
RMAN> SQL "ALTER DATABASE FLASHBACK OFF";
```

The following command drops the guaranteed restore point named `Q106`:

```
RMAN> SQL "DROP RESTORE POINT Q106";
```

11. Shut down the database consistently.

The following command shuts down the database:

```
RMAN> SHUTDOWN IMMEDIATE;
```

Migrating the Database to Oracle ASM Using RMAN

The following procedure is intended to minimize database downtime. The steps differ slightly depending on whether you are migrating a primary or standby database. The procedure makes the same assumptions described in "Preparing to Migrate the Database to Oracle ASM Using RMAN" on page 8-3. If you are not migrating the recovery area to Oracle ASM, then you must modify some steps, which are noted.

> **Note:** The following procedure switches between SQL*Plus and RMAN, so keep a terminal window open for each utility.

To migrate the database to Oracle ASM:

1. Follow the steps in "Preparing to Migrate the Database to Oracle ASM Using RMAN" on page 8-3.

2. Restore or create a server parameter file in Oracle ASM storage.

The steps depend on whether the database is using a server parameter file:

- If the database is using a server parameter file, then restore it to the Oracle ASM disk group with the following commands, where *sid* is the SID of the instance:

```
RMAN> STARTUP MOUNT;
RMAN> RESTORE SPFILE TO '+DATA/spfilesid.ora';
RMAN> SHUTDOWN IMMEDIATE;
```

- If the database is not using a server parameter file, then create one in Oracle ASM. Run the `CREATE SPFILE` command in SQL*Plus as follows, where *sid* is the SID of the database (the command spans two lines):

```
SQL> CREATE SPFILE='+DATA/spfilesid.ora' FROM PFILE='?/dbs/initsid.ora';
```

Afterward, delete `spfilesid.ora` and `initsid.ora` from the `?/dbs` directory and create a new `initsid.ora` with the following line of content:

```
SPFILE='+DATA/spfilesid.ora'
```

3. Set Oracle Managed Files initialization parameters to Oracle ASM locations.

> **Note:** If you are not migrating the fast recovery area, then do not change the `DB_RECOVERY_FILE_DEST` and `DB_RECOVERY_FILE_DEST_SIZE` initialization parameter settings. However, you must set `DB_CREATE_ONLINE_LOG_DEST_n` parameter to an Oracle ASM location for migration of the online redo logs.

Set the `DB_CREATE_FILE_DEST` and optional `DB_CREATE_ONLINE_LOG_DEST_n` initialization parameters to Oracle ASM disk groups. If the database uses a recovery area, then change the recovery area location to the Oracle ASM disk group. Also, change the recovery area size.

Run commands in SQL*Plus as shown in the following example. The example assumes that the size of the fast recovery area is 100 GB and specifies the disk group `+FRA` for the fast recovery area.

```
SQL> STARTUP FORCE NOMOUNT;
SQL> ALTER SYSTEM SET DB_CREATE_FILE_DEST='+DATA' SID='*';
SQL> ALTER SYSTEM SET DB_RECOVERY_FILE_DEST_SIZE=100G SID='*';
SQL> ALTER SYSTEM SET DB_RECOVERY_FILE_DEST='+FRA' SID='*';
```

4. Set the `CONTROL_FILES` initialization parameter to Oracle ASM locations.

If you are migrating the fast recovery area, then enter the following commands in SQL*Plus to restart the database instance and set the control file locations to disk groups `+DATA` and `+FRA`:

```
SQL> STARTUP FORCE NOMOUNT;
SQL> ALTER SYSTEM SET CONTROL_FILES='+DATA','+FRA' SCOPE=SPFILE SID='*';
```

If you are not migrating the fast recovery area, then enter the following commands in SQL*Plus to restart the database instance and set the control file locations to disk group `+DATA`:

```
SQL> STARTUP FORCE NOMOUNT;
SQL> ALTER SYSTEM SET CONTROL_FILES='+DATA','+DATA' SCOPE=SPFILE SID='*';
```

5. Migrate the control file to Oracle ASM and mount the control file.

Switch to the RMAN terminal to restore the control file. In the following example, *original_cf_name* is a control file name in the initialization parameter file before migration:

```
RMAN> STARTUP FORCE NOMOUNT;
RMAN> RESTORE CONTROLFILE FROM 'original_cf_name';
RMAN> ALTER DATABASE MOUNT;
```

6. Migrate the data files to Oracle ASM.

Use RMAN to switch to the database copy that you created in step 5 "Back up the data files to the Oracle ASM disk group" in "Preparing to Migrate the Database to Oracle ASM Using RMAN" on page 8-3. The switch renames all the data files to files on Oracle ASM disk groups. Afterward, recover the database. If incremental backups were taken, then RMAN applies them during recovery. For example, enter the following commands at the RMAN prompt:

```
SWITCH DATABASE TO COPY;
RUN
{
  ALLOCATE CHANNEL dev1 DEVICE TYPE DISK;
  ALLOCATE CHANNEL dev2 DEVICE TYPE DISK;
  ALLOCATE CHANNEL dev3 DEVICE TYPE DISK;
  ALLOCATE CHANNEL dev4 DEVICE TYPE DISK;
  RECOVER DATABASE;
}
```

7. If the database uses block change tracking or Flashback Database, then enable these features.

> **Note:** If you are not migrating the recovery area, then you do not enable Flashback Database unless you had disabled it previously.

For example, enter the following statements in SQL*Plus:

```
SQL> ALTER DATABASE ENABLE BLOCK CHANGE TRACKING USING FILE '+DATA';
SQL> ALTER DATABASE FLASHBACK ON;
```

8. Place the database in its normal operation mode.

The normal operational mode depends on whether the database is a primary or standby database:

- If the database is a primary database, then open it as follows:

  ```
  SQL> ALTER DATABASE OPEN;
  ```

- If the database is a standby database, then resume managed recovery mode as follows:

  ```
  SQL> ALTER DATABASE RECOVER MANAGED STANDBY DATABASE;
  ```

9. Drop the tempfiles and re-create them in Oracle ASM.

Use SQL*Plus to re-create the tempfiles. In the following example, the name of the tempfile in the original storage is *tempfile_name*. The name of the temporary tablespace is *temp_tbs_name*.

```
SQL> ALTER DATABASE TEMPFILE 'tempfile_name' DROP;
SQL> ALTER TABLESPACE temp_tbs_name ADD TEMPFILE;
```

10. Migrate the online redo log files.

If this is a primary database, then add new log group members in Oracle ASM and drop the old members. You can use the following PL/SQL script to migrate the online redo log groups into an Oracle ASM disk group. The PL/SQL script assumes that the Oracle Managed Files initialization parameters specified in step 3 "Set Oracle Managed Files initialization parameters to Oracle ASM locations" in "Migrating the Database to Oracle ASM Using RMAN" on page 8-5 are set.

Example 8–1   Migrating the online redo logs

```
SET SERVEROUTPUT ON;
DECLARE
   CURSOR rlc IS
       SELECT GROUP# GRP, THREAD# THR, BYTES, 'NO' SRL
       FROM   V$LOG
       UNION
       SELECT GROUP# GRP, THREAD# THR, BYTES, 'YES' SRL
       FROM   V$STANDBY_LOG
       ORDER BY 1;
   stmt     VARCHAR2(2048);
BEGIN
   FOR rlcRec IN rlc LOOP
      IF (rlcRec.srl = 'YES') THEN
         stmt := 'ALTER DATABASE ADD STANDBY LOGFILE THREAD ' ||
                 rlcRec.thr || ' SIZE ' || rlcRec.bytes;
         EXECUTE IMMEDIATE stmt;
         stmt := 'ALTER DATABASE DROP STANDBY LOGFILE GROUP ' || rlcRec.grp;
         EXECUTE IMMEDIATE stmt;
      ELSE
         stmt := 'ALTER DATABASE ADD LOGFILE THREAD ' ||
                 rlcRec.thr || ' SIZE ' ||  rlcRec.bytes;
         EXECUTE IMMEDIATE stmt;
         BEGIN
            stmt := 'ALTER DATABASE DROP LOGFILE GROUP ' || rlcRec.grp;
            DBMS_OUTPUT.PUT_LINE(stmt);
            EXECUTE IMMEDIATE stmt;
         EXCEPTION
            WHEN OTHERS THEN
               EXECUTE IMMEDIATE 'ALTER SYSTEM SWITCH LOGFILE';
               EXECUTE IMMEDIATE 'ALTER SYSTEM CHECKPOINT GLOBAL';
               EXECUTE IMMEDIATE stmt;
         END;
      END IF;
   END LOOP;
END;
/
```

11. Optionally, migrate backups and copies in the old fast recovery area to Oracle ASM as follows:

a. If foreign archived logs exists in the recovery area, then you cannot migrate them to Oracle ASM. Run the following command at the RMAN prompt:

```
RMAN> DELETE REMOTE ARCHIVELOG ALL;
```

b. Back up archived redo log files, backup sets, and data file copies to Oracle ASM. For example, run the following command at the RMAN prompt:

```
RUN
{
```

```
ALLOCATE CHANNEL dev1 DEVICE TYPE DISK;
ALLOCATE CHANNEL dev2 DEVICE TYPE DISK;
ALLOCATE CHANNEL dev3 DEVICE TYPE DISK;
ALLOCATE CHANNEL dev4 DEVICE TYPE DISK;

BACKUP AS COPY ARCHIVELOG ALL DELETE INPUT;
BACKUP BACKUPSET ALL DELETE INPUT;
BACKUP AS COPY DATAFILECOPY ALL DELETE INPUT;
}
```

Migrating a Database from Oracle ASM to Alternative Storage

Migrating a database from Oracle ASM to an alternative storage system is essentially the reverse of the migration to Oracle ASM. Modify the steps in "Preparing to Migrate the Database to Oracle ASM Using RMAN" on page 8-3 and "Migrating the Database to Oracle ASM Using RMAN" on page 8-5 as follows:

- If the procedure specifies Oracle Managed Files locations, then alter the procedure to use locations in alternative storage.

- If the FORMAT clause of the BACKUP command specifies an Oracle ASM location, then change the backup format to an alternative storage location.

- If a filename used in a SQL statement is an Oracle ASM location, then change it to a filename in the alternative storage location.

Moving Data Files Between Oracle ASM Disk Groups Using RMAN

You may want to move an active data file in an ARCHIVELOG mode database from one Oracle ASM disk group to another. In this case, you use BACKUP AS COPY to copy the data file to the new disk group and SET NEWNAME and SWITCH commands to rename the data file in the control file.

For this scenario, assume that you are using disk groups DATA and USERDATA. You want to move data file +DATA/orcl/datafile/users.261.689589837 to disk group USERDATA.

Ensure that ARCHIVELOG mode is enabled for the database before beginning the procedure to move data files.

To move a data file from one Oracle ASM disk group to another disk group, perform the following steps.

1. Start RMAN and connect to the target database.

    For example:

    ```
    $ rman
    RMAN> CONNECT TARGET SYS@orcl
    target database Password: XXXXXXXXX
    connected to target database: ORCL (DBID=1217369048)
    ```

2. Generate a report that shows the names of the data files.

    Run the following REPORT command after connecting RMAN to the target database. Note the data file name of the file to be moved.

    For example:

    ```
    RMAN> REPORT SCHEMA;

    Report of database schema for database with db_unique_name ORCL
    ```

```
List of Permanent Datafiles
===========================
File Size(MB) Tablespace         RB segs Datafile Name
---- -------- -------------- ------- ----------------------
1    740      SYSTEM         ***     +DATA/orcl/datafile/system.258.689589737
2    570      SYSAUX         ***     +DATA/orcl/datafile/sysaux.259.689589785
3    55       UNDOTBS1       ***     +DATA/orcl/datafile/undotbs1.260.689589831
4    5        USERS          ***     +DATA/orcl/datafile/users.261.689589837

List of Temporary Files
=======================
File Size(MB) Tablespace         Maxsize(MB) Tempfile Name
---- -------- -------------- ----------- --------------------
1    20       TEMP           32767       +DATA/orcl/tempfile/temp.262.689589851
```

3. Back up the data file to the new Oracle ASM disk group.

Run the BACKUP AS COPY command to back up the data file on DATA to USERDATA.

For example:

```
RMAN> BACKUP AS COPY
        DATAFILE "+DATA/orcl/datafile/users.261.689589837"
        FORMAT   "+USERDATA";

Starting backup at 16-JUN-09
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=51 device type=DISK
channel ORA_DISK_1: starting datafile copy
input datafile file number=00004 name=+DATA/orcl/datafile/users.261.689589837
output file name=+USERDATA/orcl/datafile/users.256.689682663
  tag=TAG20090616T103101 RECID=13 STAMP=689682663
channel ORA_DISK_1: datafile copy complete, elapsed time: 00:00:01
Finished backup at 16-JUN-09
```

You could also specify the data file by the data file number and data file type.

For example:

```
BACKUP AS COPY
  DATAFILE 4
  FORMAT   "+USERDATA";
```

4. Offline the data file that you intend to move to a new disk group.

Run the following SQL command in the RMAN client. Note that you use two single quotation marks around the name of the data file, not double quotation marks.

For example:

```
RMAN> SQL "ALTER DATABASE DATAFILE
        ''+DATA/orcl/datafile/users.261.689589837'' OFFLINE";

sql statement: ALTER DATABASE DATAFILE
     ''+DATA/orcl/datafile/users.261.689589837''  OFFLINE
```

5. Point the control file to the newly created copy of the data file.

Run the SWITCH...TO COPY command in the RMAN client. The TO COPY option of SWITCH switches the data file to the most recent copy of the data file.

For example:

```
RMAN> SWITCH DATAFILE "+DATA/orcl/datafile/users.261.689589837" TO COPY;

datafile 4 switched to datafile copy
    "+USERDATA/orcl/datafile/users.256.689682663"
```

The output of this command displays the new name of the data file.

6. Recover the renamed data file.

Run the RECOVER command in the RMAN client.

For example:

```
RMAN> RECOVER DATAFILE "+USERDATA/orcl/datafile/users.256.689682663";

Starting recover at 16-JUN-09
using channel ORA_DISK_1
starting media recovery
media recovery complete, elapsed time: 00:00:01
Finished recover at 16-JUN-09
```

7. Bring the data file online.

Run the SQL command in the RMAN client. Note that you use two single quotation marks around the name of the data file, not double quotation marks.

For example:

```
RMAN> SQL "ALTER DATABASE DATAFILE
    ''+USERDATA/orcl/datafile/users.256.689682663'' ONLINE";

sql statement: ALTER DATABASE DATAFILE
    ''+USERDATA/orcl/datafile/users.256.689682663'' ONLINE
```

8. Delete the data file copy from the original Oracle ASM disk group.

In this scenario, +DATA/orcl/datafile/users.261.689589837 is the original data file in DATA. Because you issued SET NEWNAME and SWITCH commands for this data file, the original file is now recorded in the RMAN repository as a data file copy. Run a DELETE command in the RMAN client to remove this file.

For example:

```
RMAN> DELETE DATAFILECOPY "+DATA/orcl/datafile/users.261.689589837";

released channel: ORA_DISK_1
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=51 device type=DISK
List of Datafile Copies
=======================
Key     File S Completion Time Ckp SCN    Ckp Time
------- ---- - --------------- ---------- ---------------
14      4    A 16-JUN-09       864471     16-JUN-09
        Name: +DATA/orcl/datafile/users.261.689589837
        Tag: TAG20090615T084217

Do you really want to delete the above objects (enter YES or NO)? y
deleted datafile copy
datafile copy file name=+DATA/orcl/datafile/users.261.689589837 RECID=14
STAMP=689683255
Deleted 1 objects
```

9

Administering Oracle ASM with Oracle Enterprise Manager

This chapter describes how to administer Oracle Automatic Storage Management (Oracle ASM) by using Oracle Enterprise Manager to provision and manage data file storage. This chapter contains the following topics:

- Oracle Automatic Storage Management Home Page
- Configuring Oracle ASM Initialization Parameters with Oracle Enterprise Manager
- Managing Oracle ASM Users with Oracle Enterprise Manager
- Managing Disk Groups with Oracle Enterprise Manager
- Managing Oracle ASM File Access Control with Oracle Enterprise Manager
- Managing Directories, Files, and Aliases with Oracle Enterprise Manager
- Managing Disk Group Templates with Oracle Enterprise Manager
- Monitoring Oracle ASM Performance with Oracle Enterprise Manager
- Backing Up Oracle ASM Files with Oracle Enterprise Manager
- Performing Bad Block Recovery with Oracle Enterprise Manager
- Migrating to Oracle ASM with Oracle Enterprise Manager
- Oracle ASM Support Workbench
- Oracle by Example Series

Oracle Automatic Storage Management Home Page

All Oracle ASM administration tasks begin with the Oracle Automatic Storage Management Home page in Oracle Enterprise Manager, shown in Figure 9–1. The Oracle Automatic Storage Management Home page displays:

- The status of the Oracle ASM instance.
- A chart that shows the used and free space of each disk group and disk group internal usage.
- A list of databases that are serviced by the Oracle ASM instance.
- A list of Oracle Automatic Storage Management Cluster File System (Oracle ACFS) file systems that are serviced by the Oracle ASM instance.
- A list of other non-Oracle ACFS volumes.

- A list of alerts for the Oracle ASM instance and the host computer.

- Links to the Oracle ASM Performance, Disk Groups, Configuration, Users, and Oracle ACFS pages.

Figure 9–1   Oracle Automatic Storage Management Home Page
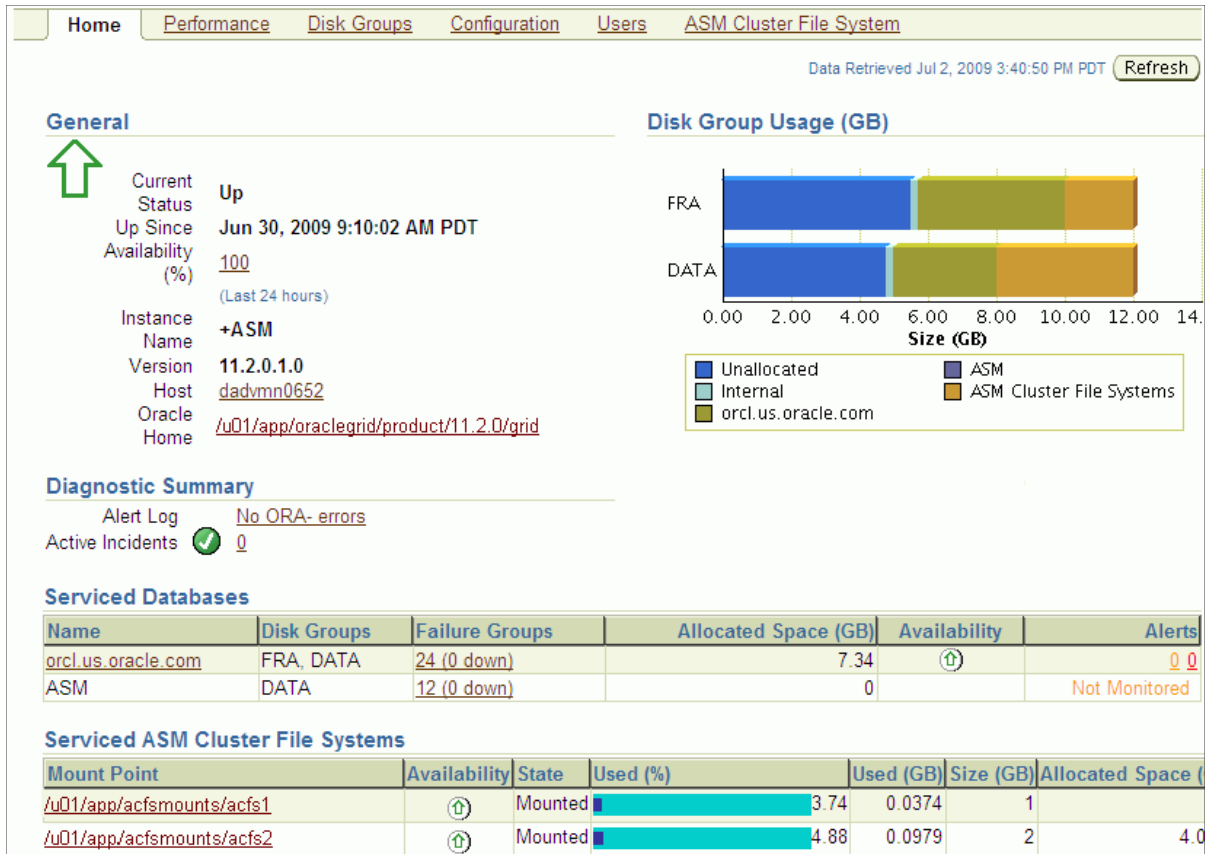
Accessing the Oracle ASM Home Page in Single-Instance Oracle Databases

To access the Oracle Automatic Storage Management Home page on a single-instance system:

1. Access the Database Home page by logging in as any database user.

    See Also: *Oracle Database 2 Day DBA* for more information

2. Under the heading labeled **General** on the left side of the Database Instance Home page, click the link next to the last entry in the General column labeled **Oracle ASM**.

3. If prompted for Oracle ASM login credentials, then enter the user SYS, provide the SYS password that was set for the Oracle ASM instance upon installation, and connect as SYSASM. Oracle Enterprise Manager displays the Oracle Automatic Storage Management Home page similar to the page in Figure 9–1.

    See Also: "Authentication for Accessing Oracle ASM Instances" on page 3-25 for more information about authentication

Accessing the Oracle ASM Home Page in Oracle RAC Databases

To access the Oracle Automatic Storage Management Home page in an Oracle Real Application Clusters (Oracle RAC) environment:

1. Log in to Oracle Enterprise Manager Database Control on any node that is running Oracle Management Service (OMS).

   OMS is automatically started on the node on which Database Configuration Assistant (DBCA) was originally run to create the cluster database. Depending on your configuration, OMS might also be running on other nodes.

   > **See Also:** *Oracle Database 2 Day + Real Application Clusters Guide* for more information

2. On the Cluster Database page, under the Instances heading, click a link to access an Oracle ASM instance.

Configuring Oracle ASM Initialization Parameters with Oracle Enterprise Manager

You can configure Oracle ASM with the Oracle Enterprise Manager Configuration Parameters page similar to the screen shown in Figure 9–2.

Figure 9–2 Oracle Automatic Storage Management Configuration Parameters Page

To configure the Oracle ASM instance:

1. Access the Oracle Automatic Storage Management Home page.

   See "Oracle Automatic Storage Management Home Page" on page 9-1 for instructions.

2. Click the **Configuration** link at the top of the page to view the Configuration Parameters page.

3. Update the configuration options on the page as follows:

   - Disk Discovery Path (`ASM_DISKSTRING` initialization parameter)

     Enter a string value. See "ASM_DISKSTRING" on page 3-7.

   - Auto Mount Disk Groups (`ASM_DISKGROUPS` initialization parameter)

Enter a string value. See "ASM_DISKGROUPS" on page 3-7.

- Rebalance Power (ASM_POWER_LIMIT initialization parameter)

  Enter an integer value. See "ASM_POWER_LIMIT" on page 3-8.

- Preferred Read Failure Groups (Only in Oracle RAC environments) (ASM_PREFERRED_READ_FAILURE_GROUPS initialization parameter)

  Enter a comma-delimited string of values in which each string represents a failure group. This parameter is generally used only for clustered Oracle ASM instances. See "ASM_PREFERRED_READ_FAILURE_GROUPS" on page 3-9.

4. Click **Apply** to save your changes or click **Revert** to discard any changes made on the page.

Managing Oracle ASM Users with Oracle Enterprise Manager

You can manage Oracle ASM users with the Oracle Enterprise Manager Users page, shown in Figure 9–3. Access this page by clicking the Users tab that is near the top center of the Oracle Automatic Storage Management Home page.

Figure 9–3   Oracle Automatic Storage Management Users Page
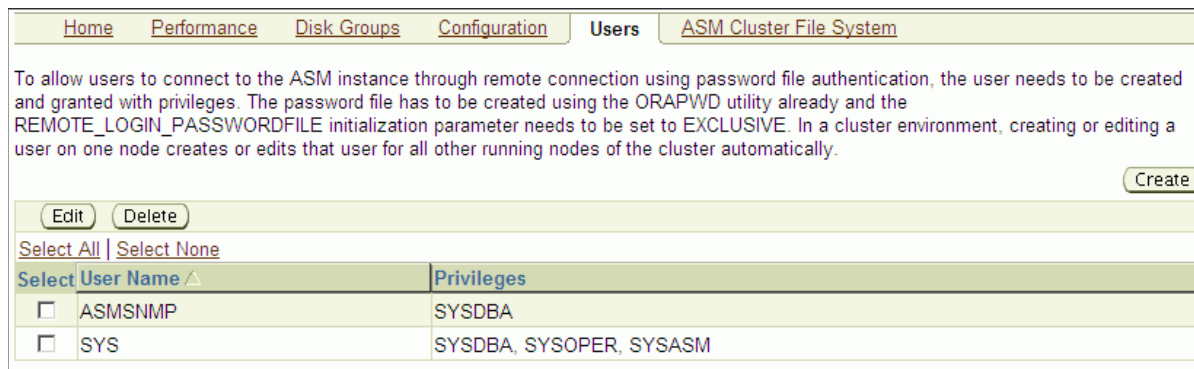
To manage Oracle ASM users:

1. Access the Oracle Automatic Storage Management Home page.

   See "Oracle Automatic Storage Management Home Page" on page 9-1 for instructions.

2. Click the **Users** link at the top to view the Users property page.

   Note that ASMSNMP is a less privileged user that is primarily used by Oracle Enterprise Manager to monitor Oracle ASM instances. This account is granted the SYSDBA privilege. ASMSNMP is created by Oracle ASM Configuration Assistant (ASMCA) when an Oracle ASM instance is created. For information about creating an Oracle ASM instance with ASMCA, see "Creating Oracle ASM Instances with Oracle ASM Configuration Assistant" on page 11-2.

   See Also:

   - *Oracle Grid Infrastructure Installation Guide* for information about installing Oracle ASM

   - *Oracle Database Installation Guide* for information about reviewing database accounts and passwords

3. Click **Create** to add a new Oracle ASM user. When you add users, Oracle Enterprise Manager displays a Create User page in which you must enter user login credentials that include the user name, password, and password confirmation. You can also grant privileges to the new user by selecting privileges in the Available Privileges column and clicking the right arrow buttons to move privileges to the Granted Privileges column, or move the privilege by double clicking it. The available privileges include SYSDBA, SYSOPER, and SYSASM. Click **OK** to create the user, **Cancel** to cancel the procedure, or **Show SQL** to view the SQL that Oracle Enterprise Manager uses to create the user.

4. To edit a user's properties, select the box in the Select column of the user to edit, then click **Edit**. On the Edit User page, modify the password if needed. You can also alter the privileges that are assigned to the user by selecting the privilege and using the right and left arrow buttons to move the privilege from the Granted Privileges column to the Available Privileges column and vice versa. You can also move the privilege by double clicking it. Click **OK** to edit the user properties, **Revert** to cancel the procedure, or **Show SQL** to view the SQL that Oracle Enterprise Manager uses to edit the user's properties.

5. To delete an Oracle ASM user, select the box in the Select column for the user to delete, then click **Delete**. On the confirmation dialog, click **Yes** to delete the user or **No** to stop the user deletion procedure.

> **See Also:** for more information about user authentication

Managing Disk Groups with Oracle Enterprise Manager

This section describes how to manage disk groups under the following topics:

- Creating Disk Groups
- Adding Disks to Disk Groups
- Dropping Disks from Disk Groups
- Bringing Disks Online and Offline
- Dropping Disk Groups
- Monitoring Disk Group Usage
- Mounting and Dismounting Disk Groups
- Administering Advanced Disk Group Properties

Creating Disk Groups

You can create additional Oracle ASM disk groups to:

- Have disk groups with different redundancy levels (normal, high, or external), depending on availability requirements and storage system capabilities.
- Separate different classes of storage, such as SCSI drives and SATA drives, into different disk groups. Disks in a disk group should have similar size and performance characteristics.
- Store the fast recovery area in a separate disk group from the database.

To create a disk group:

1. Access the Oracle Automatic Storage Management Home page.

See "Oracle Automatic Storage Management Home Page" on page 9-1 for instructions.

2. Click the **Disk Groups** link at the top of the page to display the Disk Groups property page.

3. If you are not logged in, then the Log In page appears and you should log in as the SYS user, connecting as SYSASM. Provide the SYS password for the Oracle ASM instance. This password was set up when the Oracle ASM instance was created.

4. Click **Create** and Oracle Enterprise Manager displays a Create Disk Group page similar to the page in Figure 9–4. The Create Disk Group page displays a list of Oracle ASM disks that are available to be added to a disk group. This includes disks with a header status of CANDIDATE, PROVISIONED, or FORMER, and so on.

Figure 9–4 Oracle Automatic Storage Management Create Disk Group Page

5. To display not only disks that are available to be added to a disk group, but all Oracle ASM disks, including disks that belong to a disk group (header status = MEMBER), select **All Disks** from the Select Member Disks list.

The page is re-displayed with the complete list of Oracle ASM disks that were discovered by the Oracle ASM instance.

6. Enter the disk name, select a redundancy type (high, normal, or external), and optionally, you can enter a failure group name.

7. Select disks as follows:

   ■ Select the box to the left of each disk to be included in the new disk group.

   ■ Optionally, you can enter a disk name, the size of the disk for Oracle ASM to use, and the failure group name.

   ■ To force the inclusion of a disk in the disk group, select the **Force Usage** box for that disk.

> **Caution:** Selecting the Force Usage box causes the disk to be added to the new disk group even if the disk belongs to another disk group and has valid database data. The existing data is then lost. You must be certain that you are selecting a disk that can legitimately be added to the disk group. See "Creating Disk Groups" on page 4-4 for a discussion of the FORCE option.

8. Specify a **Disk Compatibility** value for the disk group to enable your system to use the latest Oracle ASM features. See "Configuring Disk Group Compatibility Attributes" on page 9-15.

> **Note:** You cannot reverse setting a disk compatibility value.

For information about disk group compatibility attributes, see "Disk Group Compatibility" on page 4-33.

9. Click **OK** to create the disk group or click **Cancel** to stop the disk group creation operation. You can also click **Show SQL** to show the SQL statements that the disk group creation operation uses.

> **See Also:** See "Creating Disk Groups" on page 4-4 for more information about creating disk groups and the V$ASM_DISK view in *Oracle Database Reference* for information about the various header statuses of Oracle ASM disks

Adding Disks to Disk Groups

You can add disks to a disk group to increase the storage space in a disk group. If you add multiple disks in a single operation, then Oracle ASM rebalances the disk group to evenly distribute the data on all disks, including the newly added disks.

You can control the power of a rebalance operation with the setting of the Rebalance Power. The higher the number, the faster the rebalance operation completes. Lower numbers cause rebalancing to take longer, but consume fewer processing and I/O resources. This leaves these resources available for the database. The default value of 1 minimizes disruption to the database.

A value of 0 blocks the rebalance operation. Manual or automatic rebalance can then occur later. You might want to postpone rebalancing until there are fewer demands on the database, or when you have completed all of your disk group changes.

For information about controlling rebalance operations, refer to "Manually Rebalancing Disk Groups" on page 4-18 and "Tuning Rebalance Operations" on page 4-19. For information about rebalance power settings and the ASM_POWER_LIMIT initialization parameter, refer to "ASM_POWER_LIMIT" on page 3-8.

To add one or more disks to a disk group:

1. Access the Oracle Automatic Storage Management Home page.

   For instructions, refer to "Oracle Automatic Storage Management Home Page" on page 9-1

2. Click the **Disk Groups** link to display the Disk Groups property page.

3. If the Oracle ASM Login page appears, then log in as the SYS user, connecting as SYSASM. Provide the SYS password for the Oracle ASM instance. This password was set up when the Oracle ASM instance was created.

4. Click a link in the **Name** column to select the disk group to which you want to add disks. The Disk Group page similar to the page in Figure 9–7 on page 9-10 appears, displaying a list of disks that are in the disk group.

5. Click **Add** and Oracle Enterprise Manager displays the Add Disks page. This page displays a list of Oracle ASM disks that are available to be added to the disk group. This includes disks with the header status of CANDIDATE, PROVISIONED, or FORMER, and so on.

Figure 9–5   Oracle Automatic Storage Management Add Disks Page

6. To display not only disks that are available to be added to a disk group, but all of the Oracle ASM disks, including disks that belong to a disk group (header status = MEMBER), select **All Disks** from the Select Member Disks drop-down list on the right hand side of the page. Oracle Enterprise Manager re-displays the page with the complete list of Oracle ASM disks that were discovered by the Oracle ASM instance.

7. Optionally change the rebalance power by selecting a value from the Rebalance Power list.

8. Select disks as follows:

- Check the box to the left of each disk to be added to the disk group.

- To force the inclusion of a disk in the disk group, select the **Force Reuse** box in the right-most column.

Caution: The Force Reuse box causes the disk to be added to the disk group even if the disk belongs to another disk group and has valid database data. The existing data is then lost. You must be certain that you are selecting a disk that can legitimately be added to the disk group. Refer to "Adding Disks to a Disk Group" on page 4-14 for an example of the FORCE option.

- Optionally enter an Oracle ASM disk name for each disk. Oracle ASM provides a name if you do not.

9. Click **OK** to add the selected disks.

> **See Also:** "Altering Disk Groups" on page 4-12 for more information about modifying disk group properties

Bringing Disks Online and Offline

Perform the following procedures to bring a disk online or offline:

1. Access the Oracle Automatic Storage Management Home page.

   See "Oracle Automatic Storage Management Home Page" on page 9-1 for instructions.

2. Click the **Disk Groups** link at the top of the page to view the Disk Groups page. Oracle Enterprise Manager displays a page similar to the one shown in Figure 9–6.

Figure 9–6 Oracle Automatic Storage Management Disk Groups List Page

3. Click the name of a disk in the Name column and Oracle Enterprise Manager displays the General page for the disk group to modify. The General page that you see should be similar to the one shown in Figure 9–7.

Figure 9–7   Oracle Automatic Storage Management Disk Group General Page
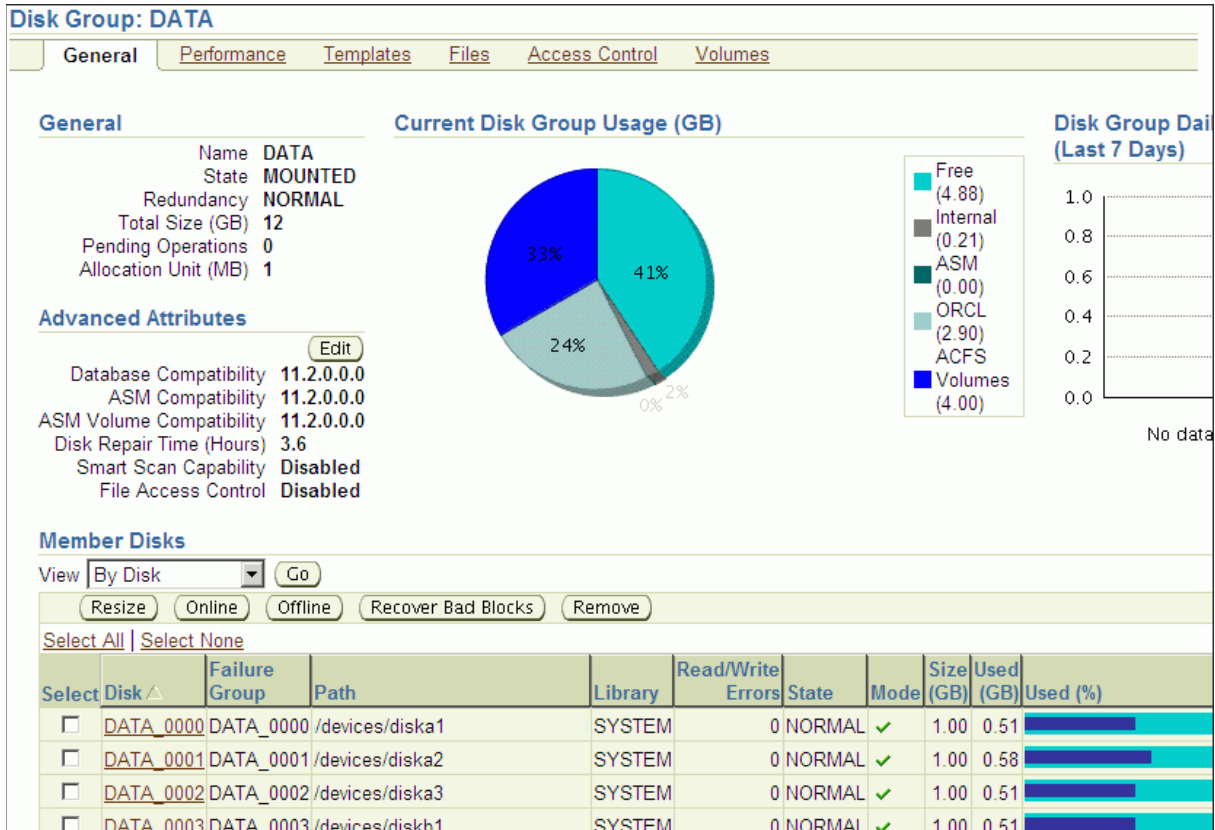
4. To bring a disk online, select the disk to bring online and click **Online**. If you bring a disk online, then Oracle ASM performs the following:

    ■   Oracle ASM performs the online operation on the selected disks. You can select one or more or disks, or all of the disks to bring online. If you select all of the disks, then Oracle uses the `ALL` keyword for this operation. Otherwise, only the selected disks are affected by the online operation.

    ■   During the online process, Oracle Enterprise Manager first places a disk into online mode to accommodate write operations. Then Oracle Enterprise Manager resynchronizes stale data on the disk with the current data until the disk is fully online for read operations; this is the Oracle ASM fast mirror resync feature.

    ■   You can also choose the By Failure Group view to perform this operation on a selected disk group. In this case, Oracle uses the SQL for `disks in failure group`.

    ■   You can also select a disk that is online, but Oracle Enterprise Manager ignores that selection and only brings disks online that were previously offline.

    ■   Click **Yes** on the Oracle Enterprise Manager Confirmation dialog to begin bringing the disk online or click **No** to cancel the operation. You can also click **Show SQL** to review the SQL that Oracle Enterprise Manager uses for the online operation.

5. To take a disk offline, select the disk to take offline and click **Offline**. If you bring a disk offline, then Oracle ASM performs the following:

- Depending on how much data is stale, the operation may require additional time to complete. You can monitor the operation's progress using the Pending Operations link from the Disk Group Home page.

- The value that you set for Rebalance Power using `ASM_POWER_LIMIT` must be a whole number. The text on the page describes how the value of Rebalance Power affects the fast mirror resync operation. If the `ASM_POWER_LIMIT` is set to `0`, then Oracle ASM does not perform automatic rebalance. For information about rebalance power settings and the `ASM_POWER_LIMIT` initialization parameter, refer to "ASM_POWER_LIMIT" on page 3-8.

- By default, the Rebalance Power value uses the value of the `ASM_POWER_LIMIT` initialization parameter.

- Oracle Enterprise Manager displays an Offline Confirmation page and the text on this page explains that after Oracle Enterprise Manager takes the selected disks offline, the Oracle ASM instance waits for the time interval that is specified by the value that you have set for the `DISK_REPAIR_TIME` initialization parameter before Oracle ASM drops the disks. During the disk repair time period, you can repair the disks and then place them back online. For each disk that you successfully place online, the pending drop operation is canceled and a resynchronization operation begins.

- If you have set `DISK_REPAIR_TIME` to a large value, for example, greater than one day, then this increases the possibility of data loss. This is because the files in the disk group are protected with a lower degree of redundancy during this period.

  Click **Yes** on the Oracle Enterprise Manager Conformation dialog to begin the offline processing or click **No** to stop the operation. You can also click **Show SQL** to see the SQL that Oracle Enterprise Manager uses for the offline operation.

Dropping Disks from Disk Groups

When you drop a disk from a disk group, Oracle ASM rebalances the disk group by moving all of the file extents from the dropped disk to other disks in the disk group. Oracle ASM then releases the disk and you can then add the disk to another disk group or use it for other purposes. You can drop one or more disks in a single operation. You can also optionally set rebalance power for the drop operation. The following is a list of possible reasons for dropping a disk:

- A disk is starting to fail and you must replace it

- To upgrade a disk

- To reallocate the disk to a different disk group or reallocate the disk to a different storage system

---

Note: Dropping disks from a disk group only logically deletes the disks from the disk group. It does not delete the disk contents. However, the contents are lost when you add the disk to a new disk group.

To drop one or more disks from a disk group:

1. Access the Oracle Automatic Storage Management Home page.

See "Oracle Automatic Storage Management Home Page" on page 9-1 for instructions.

2. Click the **Disk Groups** link to display the Disk Groups property page.

3. If the Oracle Automatic Storage Management Login page appears, log in as the SYS user, connecting as SYSASM. Provide the SYS password for the Oracle ASM instance. This password was set up when the Oracle ASM instance was created.

4. Click a link in the **Name** column to select the disk group that contains the disks to be dropped.

    The Disk Group page appears.

5. Under the Member Disks heading, in the Select column, select the boxes for the disks that you plan to drop, and then click **Remove**.

    A confirmation page appears.

6. To change the rebalance power, or to set the FORCE option for dropping disks, do the following:

    a. Click **Show Advanced Options**.

    b. Optionally select the **Force** option or select a rebalance power in the Rebalance Power list.

    > **Note:** You might need the FORCE option if Oracle ASM cannot read from or write to the disk. See "Dropping Disks from Disk Groups" on page 4-15 for a discussion of the FORCE option.

7. Click **Yes** to drop the disks.

    The Disk Group page returns, and displays a state of DROPPING for each disk being dropped.

8. Refresh the page until the dropped disks no longer appear in the disk group.

    > **Caution:** You cannot reuse or disconnect the dropped disks until the drop and rebalance operations are complete. Operations are complete when the dropped disks no longer appear in the disk group.See "Dropping Disks from Disk Groups" on page 4-15 for more information, and for other caveats for dropping disks.

Dropping Disk Groups

When you drop a disk group, you delete the disk group optionally with all of its files. You cannot drop a disk group if any of its database files are open. Oracle ASM deletes all of the disk group contents only if you specify the **including contents** option. After dropping a disk group, you can add its member disks to other disk groups or use them for other purposes.

One reason to drop a disk group is to change the redundancy level (normal, high, or external). Because you cannot change the redundancy of a disk group, you must drop the disk group and then re-create it with the proper redundancy. In this case, you must back up or move disk group data before you drop the disk group.

To drop a disk group:

1. Access the Oracle ASM Home page.

> **See Also:** "Oracle Automatic Storage Management Home Page" on page 9-1 for instructions about how to access the Oracle Automatic Storage Management Home page

2. Click the **Disk Group** tab to display the Disk Groups page.

3. If the Oracle ASM Login page appears, log in as the `SYS` user, connecting as `SYSASM`. Provide the `SYS` password for the Oracle ASM instance. This password was set up when the Oracle ASM instance was created.

4. Click the box in the **Select** column to select a disk group that you plan to drop.

5. Click **Delete** and the confirmation page appears.

6. To delete the disk group even if it still contains files, click **Show Advanced Options** and ensure that the **Including Contents** option is selected. If the disk group contains files and the **Including Contents** option is not selected, then you cannot drop the disk group.

7. To delete a disk group that belongs to a disk group that you cannot mount, use the `FORCE` option in the Advanced Options confirmation dialog. The `FORCE` option applies to the `DISMOUNTED` disk group only. This option enables you to delete a disk header and mark it as `FORMER`.

> **Caution:** Use extreme care when using the `FORCE` option because the Oracle ASM instance does not verify whether the disk group is used by any other Oracle ASM instance before Oracle ASM deletes the disk group.

8. On the confirmation page, click **Yes**.

> **See Also:** "Dropping Disk Groups" on page 4-46 for more information about dropping disks

Monitoring Disk Group Usage

Perform the following procedures to monitor disk group usage:

To view Oracle ASM disk group usage and free space:

1. Access the Oracle Automatic Storage Management Home page.

> **See Also:** "Oracle Automatic Storage Management Home Page" on page 9-1 for instructions about how to access the home page

2. Click the **Disk Groups** link to view the Disk Groups property page.

3. If the Oracle ASM Login page appears, then log in as the `SYS` user, connecting as `SYSASM`. Provide the `SYS` password that was set up for Oracle ASM when the Oracle ASM instance was created.

The Disk Group list page, similar to the page shown in Figure 9–6, displays all of the disk groups and their space usage information.

The Usable Free column displays the space in megabytes that is available in the disk group. The values in this column consider the redundancy level of the disk group, and exclude the space that the disk group reserves for restoring full redundancy for all files in the disk group after a disk failure.

> **See Also:** "Managing Capacity in Disk Groups" on page 4-22 and "Views Containing Oracle ASM Disk Group Information" on page 6-1

Mounting and Dismounting Disk Groups

Use the FORCE option to mount a disk group when there are missing disks or missing failure groups. The FORCE option corrects configuration errors, such as incorrect values for ASM_DISKSTRING, without incurring unnecessary rebalance operations.

Disk groups mounted with the FORCE option have one or more disks offline if the disks were not available at the time of the mount. You must take corrective action to restore those devices before the time set with the DISK_REPAIR_TIME value expires. Failing to restore and put those disks back online within the disk repair time frame results in Oracle ASM automatically removing the disks from the disk group. This would then require a rebalance operation to restore redundancy for all of the files in the disk group. Perform the following procedures to mount a disk with the FORCE option:

1. From the Oracle ASM home page click the **Disk Groups** tab.

2. Select the disk group that has a disk that you must repair and click **Dismount**. Click **OK** on the Dismount confirmation dialog. The State column of the Disk Group home page should show the Status as DISMOUNTED.

3. Perform any needed repairs to the disk in the disk group.

4. Click the box in the Select column next to the disk that you plan to force mount, then click **Mount**.

5. On the Disk Mount Confirmation dialog, click the **Show Advanced Options** link.

6. On the Mount Force Confirmation dialog as shown inFigure 9–8, select the **Force** option and click **Yes**. You can also click **No** to cancel the mount force operation and **Show SQL** to review the SQL that Oracle Enterprise Manager uses to perform the mount force operation.

Figure 9–8   Mount Force Confirmation Dialog

Administering Advanced Disk Group Properties

This section explains how to administer advanced disk group properties.

Perform the following steps to change the values for disk group attributes that can be altered after the disk group has been created.

1. On the Automatic Storage Disk Group List page, click a disk group name and Oracle Enterprise Manager displays the Disk Group Home page.

2. Click **Edit** under the Advanced Attributes section.

The topics contained in this section are:

- Configuring Disk Group Compatibility Attributes

- Configuring Disk Repair Time

- Configuring Smart Scan Compatibility

- Configuring File Access Control

Figure 9–9 shows the advanced attributes that can be altered for an existing disk group. This screenshot only shows the top portion of the **Edit Advanced Attributes for Disk Group** Page.

Figure 9–9   Oracle Automatic Storage Management Edit Advanced Attributes for Disk Group Page

Configuring Disk Group Compatibility Attributes

Specify a **Disk Compatibility** value for the disk group to enable your system to use the latest Oracle ASM features.

> **Note:** You cannot reverse setting a disk compatibility value.

- **ASM Compatibility**

  Specify the value that is the minimum software version required for an Oracle ASM instance to mount this disk group.

- **Database Compatibility**

Specify a value that must be less than or equal to the Oracle ASM compatibility. This is the minimum software version that is required for a database instance to use the files in the disk group. If the database compatibility value that you use is greater than the Oracle ASM compatibility, then Oracle Enterprise Manager displays an error.

> **Note:** You cannot modify the database compatibility for an Oracle ASM cluster that is in a rolling migration state.

- **ASM Volume**

  Specify a value that is 11.2 or greater to use the Oracle ASM Dynamic Volume Manager (Oracle ADVM) functionality. The ASM disk group compatibility must be set to 11.2 or greater to enable Oracle ADVM. Also, the Oracle ADVM volume drivers must be loaded in the supported environment.

For information about disk group compatibility attributes, see "Disk Group Compatibility" on page 4-33.

Configuring Disk Repair Time

Specify a time period that is long enough for you to repair a disk in the disk group.

For information about disk group repair time, see "Oracle ASM Fast Mirror Resync" on page 4-28.

Configuring Smart Scan Compatibility

Specify to disable or enable smart scan compatibility for the disk group.

This attribute is only applicable to Oracle Exadata storage.

> **See Also:** *Oracle Exadata documentation* for information about the `CELL.SMART_SCAN_CAPABLE` attribute

Configuring File Access Control

Specify to disable or enable Oracle ASM File Access Control. If Oracle ASM File Access Control is enabled, then specify the default permissions for newly created files in the disk group.

For information about Oracle ASM File Access Control, see "Managing Oracle ASM File Access Control for Disk Groups" on page 4-40.

Managing Oracle ASM File Access Control with Oracle Enterprise Manager

You can manage Oracle ASM File Access Control with Oracle Enterprise Manager Integration for Oracle ASM File Access Control, shown in Figure 9–10.

Figure 9–10   Oracle Automatic Storage Management Disk Group Access Control Home Page

Disk Group: DATA

General   Performance   Templates   Files   **Access Control**   Volumes

Users

In order to have access to files on a disk group, operating system users need to be granted access privileges to the disk group. Once granted access privileges, users can be added as members of groups. The operating system user who owns the oracle binary for the database instance is automatically added and cannot be removed.

Current Users   **grid, oracle1, oracle2**

(Add)   (Remove)

Groups

A group contains a list of operating system users. File access privileges can be granted to the group. Members of the group inherit the file access privileges granted to the group.

(Create)

(Delete)

Select All | Select None

Select	Name △	Owner	Members	Edit
☐	db_users	grid	oracle1,oracle2	✎

To manage Oracle ASM File Access Control:

1. Access the Oracle Automatic Storage Management Home page.

    See "Oracle Automatic Storage Management Home Page" on page 9-1 for instructions.

2. Click the **Disk Groups** link to display the Disk Groups property page.

3. Click a link in the **Name** column to select a disk group.

4. Click the **Access Control** tab to display the **Access Control** page.

    On this page, you can add or remove users from a user group and create or delete user groups.

Figure 9–11 shows the **Create Group** page for Oracle ASM File Access Control.

Figure 9–11   Oracle Automatic Storage Management Create Group Page
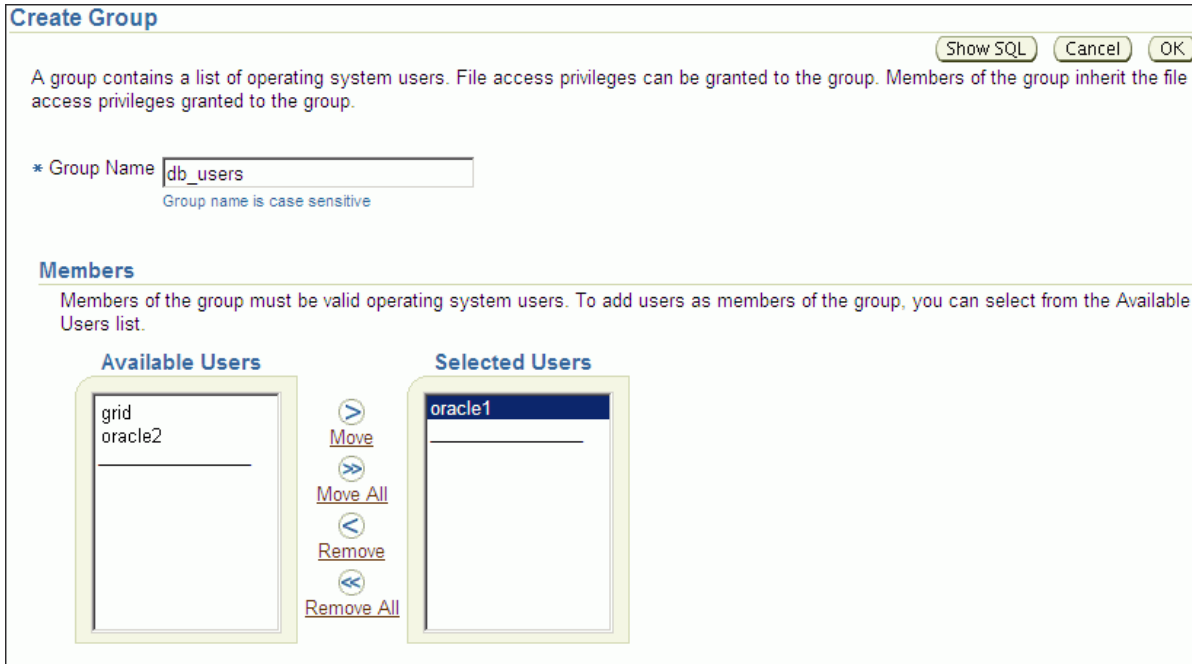
Figure 9–12 shows the **Edit Group** page for Oracle ASM File Access Control.

Figure 9–12   Oracle Automatic Storage Management Edit Group Page
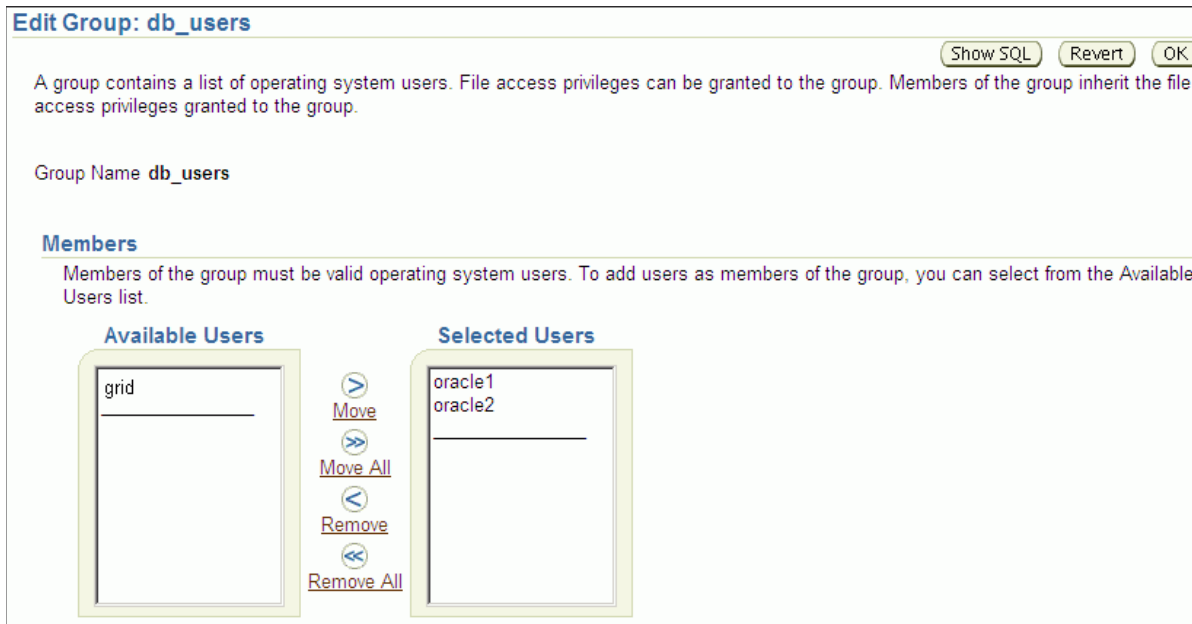
For more information about Oracle ASM File Access Control, see "Managing Oracle ASM File Access Control for Disk Groups" on page 4-40.

Managing Directories, Files, and Aliases with Oracle Enterprise Manager

You can manage directories, files, and aliases with the **Files** tab of the Oracle Enterprise Manager Disk Group page, shown in .

Figure 9–13   Oracle Automatic Storage Management Files Tab

To manage Oracle ASM files:

1. Access the Oracle Automatic Storage Management Home page.

    See for instructions.

2. Click the **Disk Groups** link to display the Disk Groups property page.

3. Click a link in the **Name** column to select a disk group.

4. Click the **Files** tab to display the **Files** page.

5. Click the **+** to expand a database folder to view template folders.

6. Click the **+** to expand a templates folder to view individual files.

7. You can select a file with the **Select** box, then click **Rename** to rename the file, click **Edit** to change the properties of a file, or click **Delete** to remove a file.

    You can click the name of a file to display the properties page.

REGISTRY.253.690887395: Properties

OK

General

        Name **REGISTRY.253.690887395**
        Type **ASMPARAMETERFILE**
  Redundancy **MIRROR**
Block Size (Bytes) **512**
      Blocks **3**
Logical Size (KB) **1.5**
     Striped **COARSE**
Creation Date **Jun 30, 2009 9:09:55 AM PDT**
Modification Date **Jun 30, 2009 9:00:00 AM PDT**

Regions

  Primary **COLD**
   Mirror **COLD**

Permissions

  Owner **Read write**
  Group **Read write**
  Other **Read write**

Ownership

  Owner **None**
  Group **None**

For more information about managing Oracle ASM files, directories, and aliases, see Chapter 7, "Administering Oracle ASM Files, Directories, and Templates".

Managing Disk Group Templates with Oracle Enterprise Manager

You can manage disk group templates with the Templates tab of the Disk Group page, shown in Figure 9–14.

Figure 9–14   Oracle Automatic Storage Management Templates Page

Disk Group: DATA

General   Performance   **Templates**   Files   Access Control   Volumes

Create

Edit | Delete

Select	Name	Redundancy	Striped	Internal	Primary Region	Mirror Region
⦿	PARAMETERFILE	Mirror	Coarse	Yes	COLD	COLD
○	ASMPARAMETERFILE	Mirror	Coarse	Yes	COLD	COLD
○	ASM_STALE	High	Coarse	Yes	COLD	COLD
○	OCRBACKUP	Mirror	Coarse	Yes	COLD	COLD
○	OCRFILE	Mirror	Coarse	Yes	COLD	COLD
○	DATAGUARDCONFIG	Mirror	Coarse	Yes	COLD	COLD
○	FLASHBACK	Mirror	Coarse	Yes	COLD	COLD
○	CHANGETRACKING	Mirror	Coarse	Yes	COLD	COLD
○	XTRANSPORT	Mirror	Coarse	Yes	COLD	COLD
○	AUTOBACKUP	Mirror	Coarse	Yes	COLD	COLD
○	BACKUPSET	Mirror	Coarse	Yes	COLD	COLD
○	TEMPFILE	Mirror	Coarse	Yes	COLD	COLD
○	DATAFILE	Mirror	Coarse	Yes	COLD	COLD
○	ONLINELOG	Mirror	Coarse	Yes	COLD	COLD
○	ARCHIVELOG	Mirror	Coarse	Yes	COLD	COLD
○	FLASHFILE	Mirror	Coarse	Yes	COLD	COLD
○	CONTROLFILE	High	Fine	Yes	COLD	COLD
○	DUMPSET	Mirror	Coarse	Yes	COLD	COLD
○	ASMPARAMETERBAKFILE	Mirror	Coarse	Yes	COLD	COLD

Edit | Delete

To manage Oracle ASM templates:

1. Access the Oracle Automatic Storage Management Home page.

   See "Oracle Automatic Storage Management Home Page" on page 9-1 for instructions.

2. Click the **Disk Groups** link to display the Disk Groups property page.

3. Click a link in the **Name** column to select a disk group.

4. Click the **Templates** tab to display the Templates page.

5. You can click **Create** to add a new template, click **Edit** to change a page, or click **Delete** to remove a template.

   If you click **Create**, then the Create Template page displays and you can enter a name in the Template Name field. You can also specify a redundancy level, striped granularity, and extent regions (hot or cold).

For more information about disk group templates, see "Managing Disk Group Templates" on page 7-15. For information about hot and cold disk regions, see "Intelligent Data Placement" on page 4-16.

Monitoring Oracle ASM Performance with Oracle Enterprise Manager

This section describes how to perform the following performance-related Oracle ASM administrative tasks:

- Monitoring General Oracle ASM Performance
- Checking Disk Group Integrity

Monitoring General Oracle ASM Performance

You can monitor Oracle ASM performance with the Oracle Enterprise Manager Performance page, shown in Figure 9–15.

Figure 9–15   Oracle Automatic Storage Management Performance Page
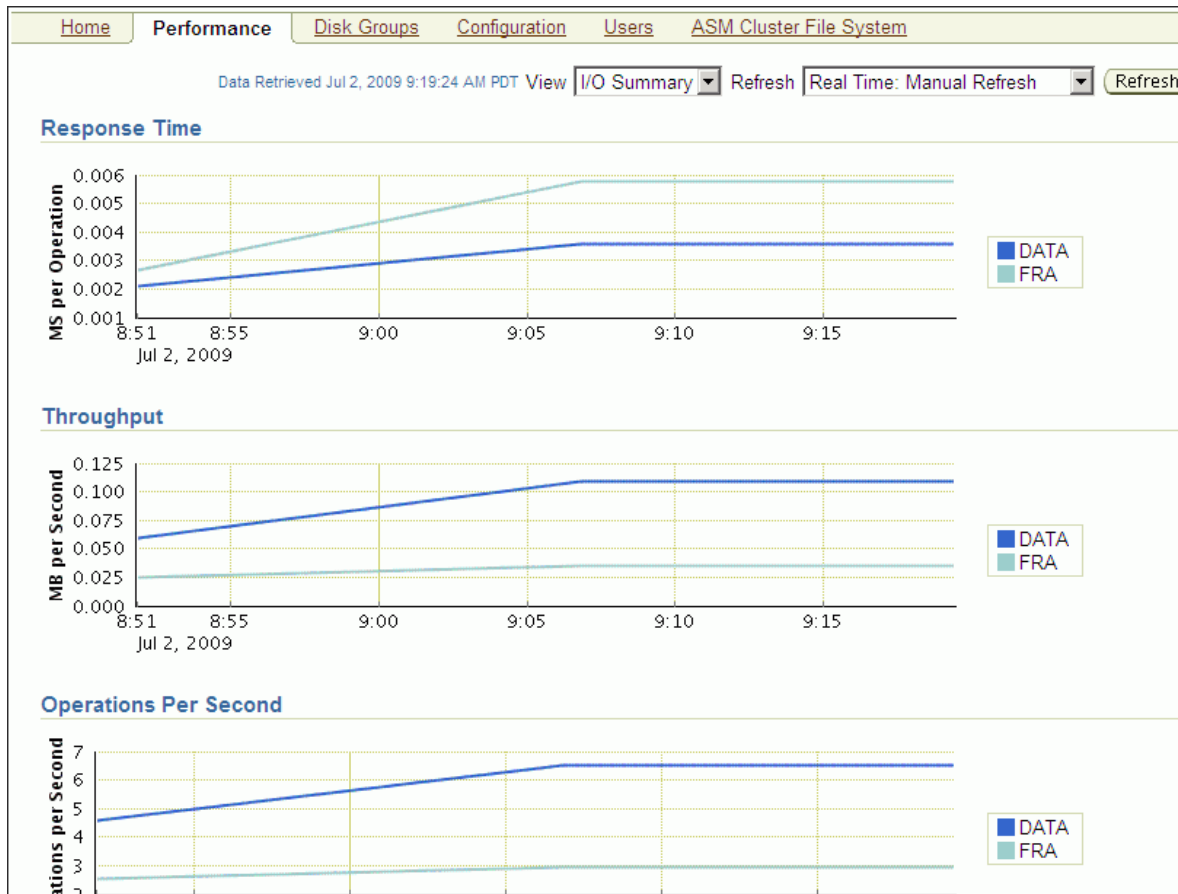
To monitor Oracle ASM performance:

1. Access the Oracle Automatic Storage Management Home page.

   See Also: "Oracle Automatic Storage Management Home Page" on page 9-1 for more information about accessing the home page

2. Click the **Performance** link to view the Performance property page.

3. Determine the view and refresh rate of the performance graphs with the **View** and **Refresh** options. Click **Refresh** to manually refresh the graphs. The refresh option has the following two intervals:

   - Manual refresh: manual, 15 second interval, 30 second interval

   - Historical data: Last 24 hours, last 7 days, last 31 days

     See Also: "Performance and Scalability Considerations for Disk Groups" on page 4-32

Checking Disk Group Integrity

When you check the integrity of Oracle ASM disk groups, Oracle Enterprise Manager performs consistency checks on all of the disk group metadata. If Oracle ASM detects errors, then Oracle ASM stores information about errors in the alert log. To check disk groups, perform the following steps:

1. From the Disk Groups Administration page click **Check** and Oracle ASM displays a confirmation dialog for the check disk group operation.

Figure 9–16   Oracle Automatic Storage Management Check Diskgroup Confirmation Dialog

2. Select **Check Without Repair** to record information about disk group inconsistencies in the alert log. In this operation, Oracle ASM does not resolve the errors. Select **Check And Repair** to attempt to repair errors that Oracle Enterprise Manager identifies during the disk group consistency check.

3. Click **Yes** to start the disk group check operation or **No** to cancel it. You can also select **Show SQL** to review the SQL statements that the disk group check operation uses. Click **OK** to run the check operation as a command.

4. To schedule a job, click **Schedule Job**.

5. Optionally, you can modify or reenter the job name and description. Also provide host credentials and enter the scheduling options by selecting **Immediately** or **Later** and modifying the date and time as needed.

6. Click **OK** to schedule the job or **Cancel** to terminate the job creation process.

Backing Up Oracle ASM Files with Oracle Enterprise Manager

Oracle recommends that you use RMAN to back up files managed by Oracle ASM. The procedures for using RMAN are documented in *Oracle Database 2 Day DBA*.

You can also back up your Oracle ASM managed files by performing a files related operation using the following procedures:

1. Create an alias.

2. Create a director.

3. Delete the alias and directory.

4. Rename the alias and directory.

For information about migrating data with RMAN, see Chapter 8, "Performing Oracle ASM Data Migration with RMAN".

> **See Also:**   For additional information about RMAN, refer to the *Oracle Database Backup and Recovery User's Guide*.

Performing Bad Block Recovery with Oracle Enterprise Manager

The procedures that you use in Oracle Enterprise Manager to perform bad block recovery correspond to the steps that you could manually perform with the remap

command using the Oracle ASM Command utility (ASMCMD). Use the following procedures to perform bad block recovery:

1. Go to the Oracle Automatic Storage Management home page.

2. Click the **Disk Groups** link at the top of the page.

3. If you are not logged in, then the Log In page appears and you should log in as the SYS user, connecting as SYSASM. Provide the SYS password for the Oracle ASM instance. This password was set up when the Oracle ASM instance was created.

4. Click a disk group name and Oracle Enterprise Manager displays the disk group home page for that disk group.

5. Click a box in the Select column to the left of a disk in which to recover bad blocks, then click **Recover Bad Blocks**.

> **Note:** You can only recover bad blocks for disk groups that you have configured with normal or high redundancy; you cannot use this feature for disk groups that you have configured with external redundancy.

Migrating to Oracle ASM with Oracle Enterprise Manager

This section explains how to use Oracle Enterprise Manager to migrate your database storage to use Oracle ASM.

Perform the following steps to migrate to Oracle ASM:

1. At the top of the Oracle Enterprise Manager Database Instance Home Page, click the link labeled **Server**. Oracle Enterprise Manager displays Database Server Page.

2. On the top left-hand side of the Database Server Page under the Storage heading, click the link labeled **Migrate to Oracle ASM**.

3. If prompted for Oracle ASM login credentials, then enter the user SYS, provide the SYS password that was set for the Oracle ASM instance during installation, and connect as SYSDBA. Oracle Enterprise Manager displays the Migrate Database To Oracle ASM: Migration Options Page.

   For more information about authentication, see "Authentication for Accessing Oracle ASM Instances" on page 3-25.

4. On the **Migrate Database To ASM: Migration Options Page**, which is the first step in a four-step process, verify the information about the files that you plan to migrate to Oracle ASM. If the target database is in ARCHIVELOG mode, then you could perform online or offline migration by selecting the corresponding option. Otherwise, using offline migration is the default. If you perform an online migration, then you can suspend the migration by selecting the check box next to **Suspend Migration Job Before Switching Database To ASM**.

   Additionally, you can enter an e-mail address to which Oracle Enterprise Manager sends a message to notify you about the suspension. By default on this page, the Recovery-related Files option is selected. Deselect this option if you do not want to use it. Also verify the entry for the initialization parameter file. Lastly, the **Parallelize File Copy Operations** option is selected by default. Deselect this if you do not want Oracle Enterprise Manager to perform copy operations in parallel. Click **Next** when you complete your selections on this page and Oracle Enterprise Manager displays the **Migrate Database To ASM**: **Disk Group Options** page.

5. On the Disk Group Options page under Database Area, verify the location where Oracle ASM should create data files, temporary files, redo log files, and control files. Also verify the fast recovery area where Oracle ASM should create the recovery-related files such as archived redo log files, backup files, and so on. Also under fast recovery area, the option for Setup fast recovery area without Migrating Recovery-related Files is selected by default. If needed, change the selection to migrate the files. Click **Next** when you complete your selections on the Disk Group Options page.

6. If you chose to use a fast recovery area and Oracle Enterprise Manager displays a warning that the available space on the disk group is less than the specified fast recovery area size for the migration, then you can either continue by clicking **Yes**, or click **No** to stop the operation and modify your storage.

7. When you continue, Oracle Enterprise Manager displays the Schedule page on which you can specify a job name and a date and time for the migration job to begin. Click **Next** when you have configured your desired settings.

8. Oracle Enterprise Manager displays the Review page on which you can review all of your selections before beginning your migration operation. To expand the tablespace details and review them, click the plus (+) sign next to Tablespaces and Files To Be Migrated. Click **Back** to modify any of your selections or **Submit Job** to begin the migration. Click **Cancel** to cancel the operation.

> **Caution:** If your migration job fails, then do not start your database until you complete a recovery operation. Otherwise, you may corrupt your database.

9. After you submit your job, Oracle Enterprise Manager should display a confirmation dialog indicating that the migration job was successful. Click **View Status** on the confirmation dialog to examine the progress of your migration job.

After the migration job has completed, the Oracle ASM links on the Oracle Enterprise Manager page may not display immediately. You can wait for the links to display after a subsequent refresh, or restart the Oracle Enterprise Manager agent as follows:

```
$ emctl stop dbconsole
$ emctl start dbconsole
```

Oracle ASM Support Workbench

Oracle Enterprise Manager provides Oracle ASM Support Workbench to monitor Oracle ASM alerts and incidents.

To access Support Workbench for Oracle ASM:

1. Click the **Software and Support** tab on the database home page.

2. Click **Support Workbench** under the **Support** section on the **Software and Support** page.

3. Click **Support Workbench**(*ASM_instance_name*) under the **Related Links** section on the **Support Workbench** page.

Figure 9–17 shows the Oracle ASM Support Workbench page with the **Problems** tab displayed. You can view information about current and past problems in this tab. On this page, you have options to view additional details and create a package that you can send to Oracle Support Services.

Figure 9–17   Oracle Automatic Storage Management Support Workbench Problems

To create a package, select an incident and click **Package** on the Support Workbench Problems page. Support Workbench guides you through the packaging process.

Figure 9–18 shows step one the Oracle ASM Support Workbench **Create New Package** wizard.

Figure 9–18   Oracle Automatic Storage Management Support Workbench Create Package

Figure 9–19 shows the Oracle ASM Support Workbench page with the **Packages** tab displayed. On this tab, you can view and edit information about incident packages created for problems.

Figure 9–19   Oracle Automatic Storage Management Support Workbench Check Packages

You can view additional details for a problem by selecting a problem on the **Problems** tab, and then clicking on **View**. The problem details page displays as shown in Figure 9–20.

Figure 9–20   Oracle Automatic Storage Management Support Workbench Problem Details

On the Problem Details page, you can access additional diagnostic tools. You can also use quick packaging to send an incident package to Oracle Support Services.

See Also:

- *Oracle Database Administrator's Guide* for information about viewing problems with Oracle Enterprise Manager Support Workbench

- *Oracle Database 2 Day DBA* for information about accessing the Oracle Enterprise Manager Support Workbench home page and viewing problems using Oracle Enterprise Manager Support Workbench

Oracle by Example Series

Oracle by Example (OBE) has a series on the *Oracle Database 2 Day DBA* book. This OBE steps you through Oracle Database tasks and includes annotated screen shots.

To view the Oracle Database OBE, point your browser to the following location:

http://www.oracle.com/webfolder/technetwork/tutorials/obe/db/11g/r2/2day_dba/asm/asm.htm

10

Administering Oracle ACFS with Oracle Enterprise Manager

This chapter describes how to administer Oracle Automatic Storage Management Cluster File System (Oracle ACFS) using Oracle Enterprise Manager.

This chapter contains the following topics:

- Managing Oracle ACFS with Oracle Enterprise Manager
- Managing Oracle ACFS Snapshots with Oracle Enterprise Manager

All Oracle ASM administration tasks begin with the Oracle Automatic Storage Management Home page in Oracle Enterprise Manager. For information about the Oracle ASM home page and how to access that page, see "Oracle Automatic Storage Management Home Page" on page 9-1.

For information about using command-line tools to administer Oracle ACFS file systems, see Chapter 13, "Oracle ACFS Command-Line Tools". For information about the basic steps for creating Oracle ACFS file systems, see "Basic Steps to Manage Oracle ACFS Systems" on page 13-1.

Managing Oracle ACFS with Oracle Enterprise Manager

This section discusses how to manage Oracle ACFS volumes and file systems with Oracle Enterprise Manager.

This section contains the following topics:

- Creating Oracle ACFS Volumes and File Systems
- Viewing and Modifying Oracle ACFS Volumes and File Systems

Creating Oracle ACFS Volumes and File Systems

To create an Oracle ACFS volume, perform the following steps.

1. Access the **Disk Groups** page from the Oracle ASM home page.

2. Click the **Volumes** link in the **General** tab of the **Disk Group** page. See Figure 10–6 on page 10-5.

3. Click **Create** in the **Volumes** tab of the **Disk Group** page.

   The **Create ASM Volume** page displays, shown in Figure 10–1.

   Enter the volume name (maximum 11 characters), disk group name that contains the volume, and the initial size of the volume.

You can also select the redundancy setting for the volume and the region settings for primary and mirror extents. For information about redundancy settings, refer to "Mirroring, Redundancy, and Failure Group Options" on page 4-24. For information about hot and cold disk regions, refer to "Intelligent Data Placement" on page 4-16. For information about creating a volume with the ASMCMD `volcreate` command, refer to "volcreate" on page 12-62.

Figure 10–1 shows the **Oracle ASM Create ASM Volume** page.

Figure 10–1   Oracle Automatic Storage Management Create ASM Volume Page

To create an Oracle ACFS file system on a volume in a disk group, perform the following steps.

1. Access the **Volumes** tab of the **Disk Group** page. See Figure 10–6 on page 10-5.

2. Select an existing volume in the disk group and click **Create ASM Cluster File System**.

   Figure 10–2 shows the **Create ASM Cluster File System** page.

   The compatibility parameters COMPATIBLE.ASM and COMPATIBLE.ADVM must be set to 11.2 or higher for the disk group. See "Disk Group Compatibility Attributes" on page 4-34.

3. On the **Create ASM Cluster File System** page, enter the information to create a file system. You can optionally register and mount the file system.

   You must enter the volume name. You can enter an optional volume label.

   You can optionally choose to register and mount the file system when it is created. Select a mount point from available directories.

   Enter the host credentials needed to run the command when prompted. To register or mount a file system, you need root or administrator privileges. For information about the registry process, see "About the Oracle ACFS Mount Registry" on page 5-8.

After a directory has been selected, click **Show Command** to have Oracle Enterprise Manager generate the commands that you can run at an operating system prompt. Examples of commands on a Linux system are:

```
/sbin/mkfs -t acfs -b 4k /dev/asm/volume1-361
/sbin/acfsutil registry -f -a /dev/asm/volume1-361 /oracle/acfsmounts/acfs2
/bin/mount -t acfs -rw /dev/asm/volume1-361 /oracle/acfsmounts/acfs2
```

You can also generate the commands to register or mount an Oracle ACFS file system on the **ASM Cluster File System** tab. See Figure 10–3 on page 10-3.

Figure 10–2 Oracle Automatic Storage Management Create Oracle ACFS Page
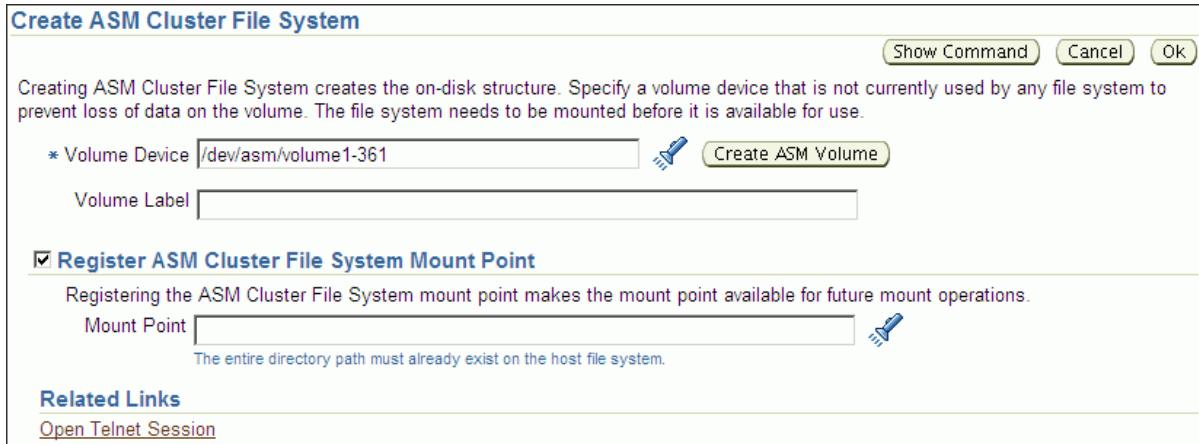
Viewing and Modifying Oracle ACFS Volumes and File Systems

To view and modify information about Oracle ACFS file systems, click the **ASM Cluster File System** link in the Oracle ASM home page.

Figure 10–3 shows the **ASM Cluster File System** tab in Oracle Enterprise Manager. This tab lists all of the Oracle ACFS associated with the Oracle ASM instance.

On this page, you can choose to mount, dismount, delete, create snapshot, view content, register, and deregister a selected file system. In addition, you can create a file system, mount all file systems, or dismount all file systems.

For each Oracle ACFS, the columns provide information for Mount Point, Availability, State, Snapshots, Used (%), Used (GB), Size (GB), Allocated Space, Redundancy, Volume, and Disk Group. Redundancy, Volume, and Disk Group columns are on the far right of the page and are not shown in the illustration. The mount point, snapshots, volume, and disk group are provided as links to further information.

Figure 10–3 Oracle Automatic Storage Management Oracle ACFS Page

To view information about a specific volume, click the volume name in the **Volume** column on the **ASM Cluster File System** page.

Figure 10–4 shows the **General** tab of the ASM **Volumes** page.

Figure 10–4   Oracle Automatic Storage Management General Tab of the Volumes Page
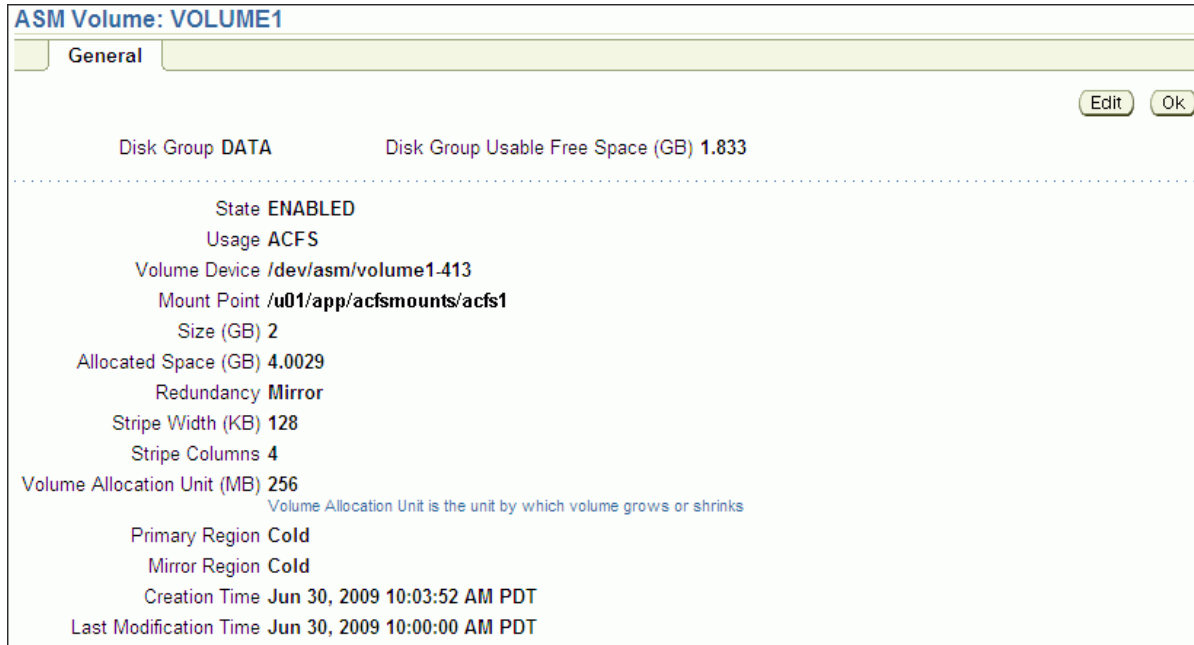
To view information about a file system, click the link in the **Mount Point** column on the **ASM Cluster File System** page.

Figure 10–5 shows the **General** tab of the **ASM Cluster File System** page.

Figure 10–5   Oracle Automatic Storage Management Oracle ACFS General Page
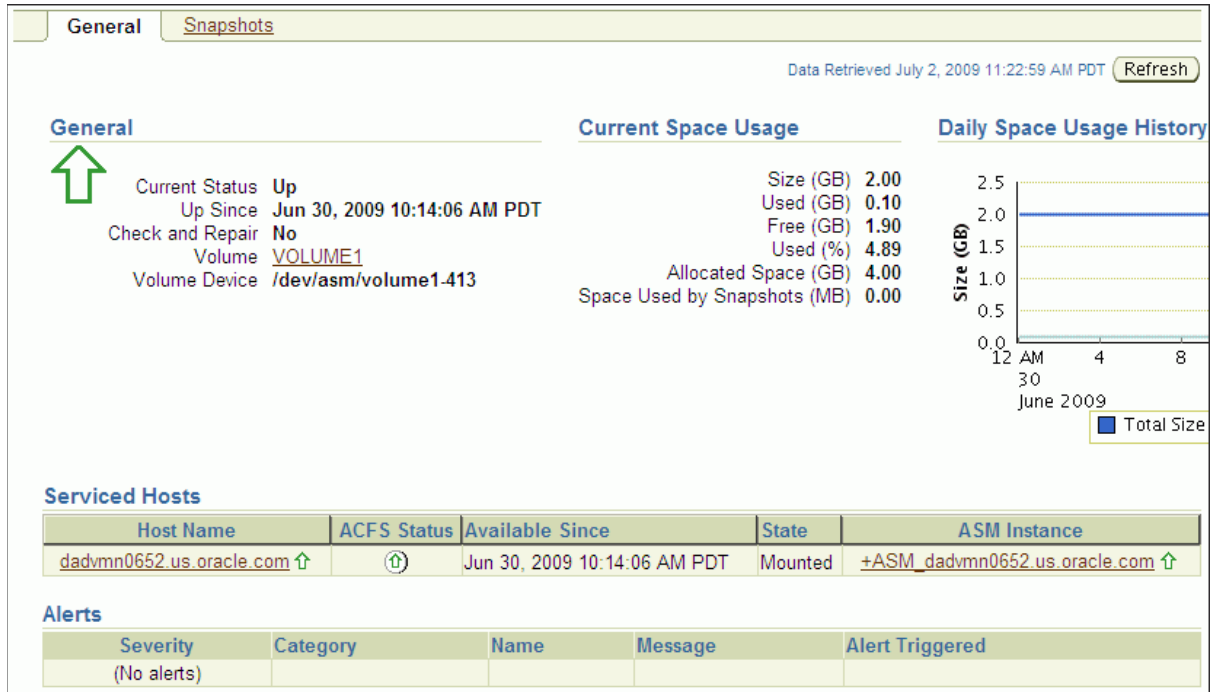
To view information about a disk group for an Oracle ACFS, click the disk group name in the **Disk Group** column on the **ASM Cluster File System** page. The **General** tab of the **Disk Group** page displays as shown in Figure 9–7 on page 9-10.

To view information about volumes in a disk group, click the **Volumes** tab at the top of the **Disk Group** page.

Figure 10–6 shows the Oracle ACFS Volume page.

Figure 10–6   Oracle Automatic Storage Management Oracle ACFS Volume Page
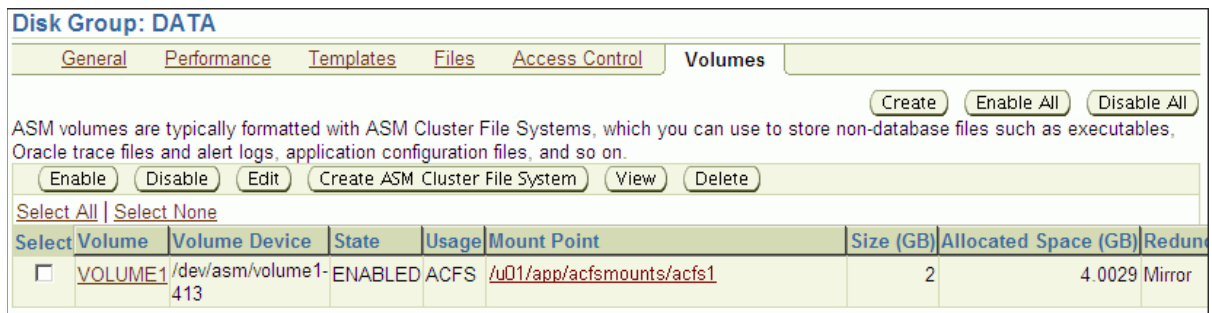
Managing Oracle ACFS Snapshots with Oracle Enterprise Manager

To view and modify information about existing snapshots or create snapshots for a file system, perform the following steps.

1. Click the **ASM Cluster File System** tab on the Oracle ASM home page

2. Click a mount point link in the **Mount Point** column as shown in Figure 10–3 on page 10-3

3. Click the **Snapshots** tab, shown in Figure 10–7.

Optionally, you can click a number link for existing snapshots in the **Snapshots** column on the **ASM Cluster File System** page to display the **Snapshots** page.

On the **Snapshots** page, you can create snapshots or search for and display specific snapshots. To search for a snapshot, enter a name with optional wildcard characters in the search field then click **Search**.

Figure 10–7 Oracle Automatic Storage Management Oracle ACFS Snapshots Tab

To create a snapshot, perform the following steps.

1. Click **Create** in the **Snapshots** page.

   Figure 10–8 shows the **Create Snapshot** page.

2. Complete the information on the page.

   Accept the default snapshot name, or provide a name. Optionally, you can choose to delete the oldest snapshot.

3. When you have completed the screen, you can click **OK** to run the command, or click **Show Command** to view the generated command.

   For example, the following is a generated command for creating a snapshot:

   ```
   /sbin/acfsutil snap create "snapshot_20090702_142135" /oracle/acfsmounts/acfs1
   ```

   To run the generated command, you need the appropriate privileges on the host computer. Oracle Enterprise Manager prompts you to enter host credentials if they have not been set up.

You can also open a Telnet session from the **Create Snapshot** page to manually enter the generated operating system command.

Figure 10–8   Oracle Automatic Storage Management Create Snapshot Page

Create Snapshot on ASM Cluster File System: /u01/app/acfsmounts/acfs1

[Show Command] [Cancel] [Ok]

Number of Snapshots **3**   Last Snapshot Name snapshot_20090702_141449
Total Free Space (GB) **1.9021**   Last Snapshot Creation Time **Jul 2, 2009 2:14:52 PM PDT**

The Create Snapshot operation creates a point-in-time copy of the ASM Cluster File System. Snapshots are always mounted and available. The snapshots are created in the .ACFS/snaps directory of the file system with the snapshot name as the directory name.

* Snapshot Name snapshot_20090702_142110
☐ Delete the oldest snapshot (61 remaining before the maximum limit is reached)
                    Oldest Snapshot Name snapshot_20090702_140518
        Oldest Snapshot Creation Time **Jul 2, 2009 2:05:57 PM PDT**

Related Links
Open Telnet Session

To drill down in a snapshot directory, click the snapshot name in the **Snapshots** page to display the **Search and List Directory** page.

Figure 10–9 shows the **Search and List Directory** page for a specific snapshot directory. The directory contains the snapshots of the file system

Figure 10–9   Oracle Automatic Storage Management Search and List Directory Page

Search and List: Directory

[Close]

Host   **dadvmn0652**
Host Username   **grid**   [Change]
                    / > u01 > app >
Current Directory   acfsmounts > acfs1 > .ACFS > snaps >
                    snapshot_20090702_142110 >

Search

Search in Directory /u01/app/acfsmounts/acfs1/.ACFS/snaps/snapshot_20090702_142
                    Directory path only, no wildcard.

File or Directory Name
                    This search is case sensitive. Use * or % as wildcard.
                    [Go]

Name	Type △	Owner	Group	Size(KB)	Last Modified
.	Directory	grid	asmadmin	4	Jul 2, 2009 2:24:21 PM
..	Directory	root	root	4	Jul 2, 2009 2:24:21 PM
.ACFS	Directory	root	root	4	Jun 30, 2009 10:14:06 AM
lost+found	File	root	root	0	Dec 31, 1969 4:00:00 PM
mytestfile	File	grid	asmadmin	0	Jul 2, 2009 2:10:01 PM

11

Oracle ASM Configuration Assistant

Oracle ASM Configuration Assistant (ASMCA) supports installing and configuring Oracle ASM instances, disk groups, volumes, and Oracle Automatic Storage Management Cluster File System (Oracle ACFS). In addition, you can use the ASMCA command-line interface as a non-GUI utility.

This chapter discusses the following topics:

- Oracle ASM Configuration Assistant GUI Tool
- Oracle ASM Configuration Assistant Command-Line Interface

Oracle ASM Configuration Assistant GUI Tool

This section contains the following topics:

- Starting Oracle ASM Configuration Assistant
- Creating Oracle ASM Instances with Oracle ASM Configuration Assistant
- Upgrading an Oracle ASM Instance with Oracle ASM Configuration Assistant
- Managing Disk Groups with Oracle ASM Configuration Assistant
- Managing Oracle ADVM Volumes with Oracle ASM Configuration Assistant
- Managing Oracle ACFS File Systems with Oracle ASM Configuration Assistant
- Managing Security and Encryption for Oracle ACFS File Systems with Oracle ASM Configuration Assistant
- Creating an Oracle ACFS File System for a Database Home

Starting Oracle ASM Configuration Assistant

To start ASMCA, enter `asmca` at a command prompt inside the Oracle Grid Infrastructure home. For example:

```
$ asmca
```

The GUI tool displays and attempts to connect to the Oracle ASM instance identified with the Oracle system identifier (SID) set to `+ASM`.

- If ASMCA detects an Oracle ASM instance that is not running, then ASMCA prompts whether you want to start the instance.

- If ASMCA detects an Oracle ASM instance from a previous release, then ASMCA prompts whether you want to upgrade the instance. Oracle recommends that you respond `No` this prompt.

An Oracle ASM instance should be upgraded with Oracle Universal Installer (OUI). OUI automatically defaults to upgrade mode when it detects an Oracle ASM instance at a previous release level. See "Upgrading an Oracle ASM Instance in an Oracle Restart Configuration with Oracle Universal Installer" on page 3-18.

To continue the upgrade process with ASMCA, see "Upgrading an Oracle ASM Instance with Oracle ASM Configuration Assistant" on page 11-3. If you need help with upgrading an Oracle ASM instance, contact Oracle Support Services.

To start the command-line ASMCA, see "Oracle ASM Configuration Assistant Command-Line Interface" on page 11-16.

Creating Oracle ASM Instances with Oracle ASM Configuration Assistant

Oracle ASM Configuration Assistant enables you to create an Oracle ASM instance if an Oracle ASM instance is not found on the host, as shown in Figure 11–1.

On the **Create ASM** page, you enter the SYS password for the administrator of the Oracle ASM instance and the password for the ASMSNMP user that can monitor this Oracle ASM instance. Note that ASMSNMP is a less privileged user that is primarily used by Oracle Enterprise Manager to monitor Oracle ASM instances.

> **See Also:**
>
> - *Oracle Grid Infrastructure Installation Guide* for information about installing Oracle ASM
>
> - *Oracle Database Installation Guide* for information about reviewing database accounts and passwords

You can select the listener with which to register this Oracle ASM instance. Click **ASM Parameters** to update the initialization parameters for the Oracle ASM instance.

Figure 11–1 Oracle ASM Configuration Assistant Create ASM Page

When you can click **ASM Parameters** in the **Create ASM** page, you can update initialization parameters for an Oracle ASM instance on the Oracle ASM Configuration Assistant page, as shown in Figure 11–2.

For information about Oracle ASM instance initialization parameters, see "Configuring Initialization Parameters for Oracle ASM Instances" on page 3-2.

Figure 11–2   Oracle ASM Configuration Assistant Create ASM Page
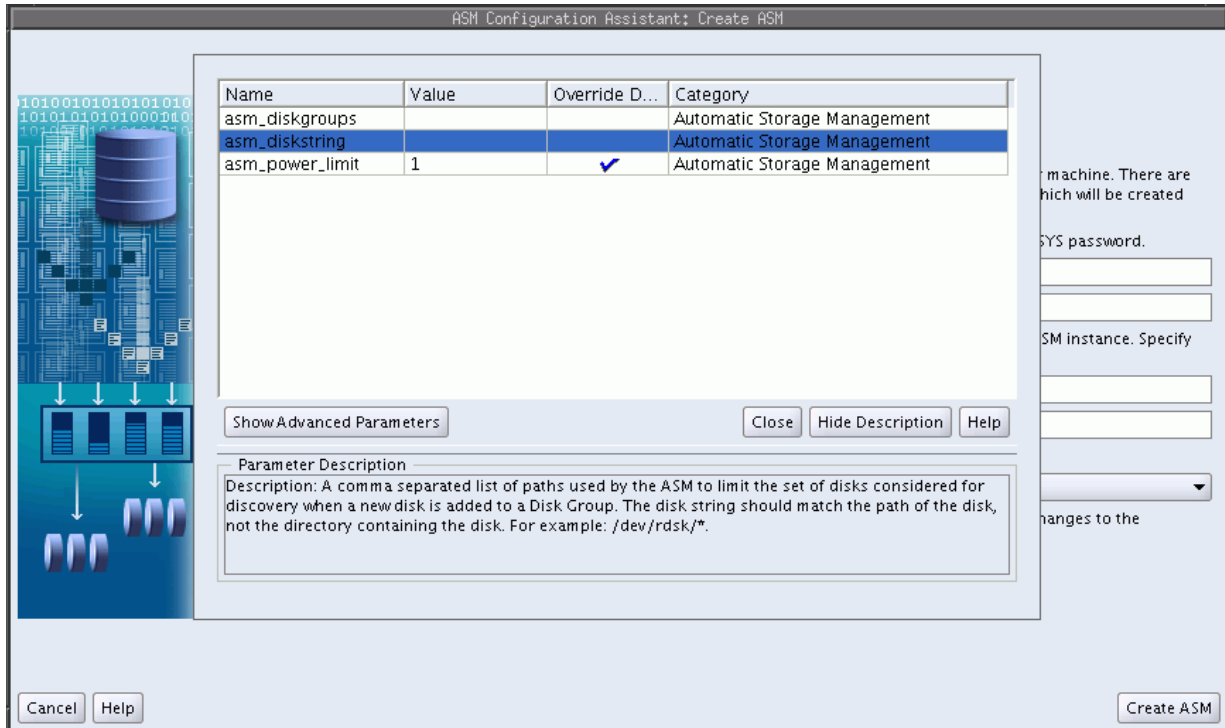
Upgrading an Oracle ASM Instance with Oracle ASM Configuration Assistant

> **Note:**
> - For information about performing an Oracle ASM Rolling Upgrade from 11.1.0.6 to 11.2.0.1, refer to Oracle Database Readme.
> - For information about upgrading Oracle ASM from version 11.2.0.1 to 11.2.0.2, see "Out of Place Upgrades" on page 3-24.

Oracle ASM Configuration Assistant also enables you to upgrade an existing Oracle ASM instance to the current software level. However, the recommended practice is to upgrade an Oracle ASM instance with Oracle Universal Installer (OUI). OUI automatically defaults to upgrade mode when it detects an Oracle ASM instance at a previous release level. See "Upgrading an Oracle ASM Instance in an Oracle Restart Configuration with Oracle Universal Installer" on page 3-18.

If Oracle Clusterware has been upgraded, then there is no need to upgrade the Oracle ASM instance in that Oracle Grid Infrastructure home. Oracle ASM is usually upgraded as part of the Oracle Clusterware upgrade.

See Also:

- *Oracle Grid Infrastructure Installation Guide* for information about installing and upgrading Oracle Grid Infrastructure

- Articles at My Oracle Support (https://support.oracle.com) for information about upgrading Oracle ASM

The following procedure describes how to upgrade an Oracle ASM instance from 11*g* Release 1 (11.1) to 11*g* Release 2 (11.2). In this scenario:

- The Oracle ASM and Oracle Database instances 11*g* Release 1 (11.1) exist in separate homes.

- The Oracle Grid Infrastructure software 11*g* Release 2 (11.2) is installed in a separate home.

- The Oracle ASM instance 11*g* Release 2 (11.2) is set up as an Oracle Restart (single-instance) configuration.

1. Shut down the Oracle Enterprise Manager agent, Oracle Database instance, and the listener in the older database home.

   - Run `emctl stop dbconsole` to stop the Oracle Enterprise Manager agent.

   - Connect to the database instance with SQL*Plus as a privileged user and run the `SHUTDOWN` command.

   - Run `lsnrctl` and enter the `STOP` command to stop the listener.

     See Also:

     - Oracle Enterprise Manager manuals and online help for information about starting and stopping the Oracle Enterprise Manager agent

     - *Oracle Database Administrator's Guide* for more information about starting up and shutting down Oracle instances

     - *Oracle Database Net Services Administrator's Guide* for information about configuring a listener

2. Run Oracle Universal Installer (OUI) and select the **Install Oracle Grid Infrastructure Software Only** option.

3. Complete the screens in the OUI installer and run the scripts as prompted by the OUI installer.

   For example, on Linux you must run the `root.sh` script as the `root` user.

   ```
   # GRID_HOME/grid/perl/bin/root.sh
   ```

   When you run the `root.sh` script on Linux, note the instructions in the output of that script. You must run different scripts as `root` to configure either a standalone (single-instance) or cluster configuration after OUI has completed.

   See Also: *Oracle Grid Infrastructure Installation Guide* for information about installing and upgrading Oracle Grid Infrastructure

4. After the OUI has completed, perform the procedures that are described in the output of the `root.sh` script that you ran during the installation.

For example, for a standalone (single-instance) configuration on Linux, run the following script as `root` to upgrade the necessary services, including Oracle Cluster Synchronization Services (CSS).

```
# GRID_HOME/grid/perl/bin/perl -I/GRID_HOME/perl/lib -I/GRID_HOME/crs/install
 GRID_HOME/crs/install/roothas.pl
```

For an upgrade, you must update the node list by running OUI as the installation user in the Oracle Grid Infrastructure home.

```
$ GRID_HOME/oui/bin/runInstaller -updateNodeList -silent -local CRS=false
  ORACLE_HOME=ASM_11G_R1_HOME
```

```
$ GRID_HOME/oui/bin/runInstaller -updateNodeList -silent -local CRS=true
  ORACLE_HOME=GRID_HOME
```

5. Ensure that the current Oracle Cluster Synchronization Services (CSS) are running in the home where the Oracle ASM instance is located. For example:

    On Linux run the `crsctl check css` command. If the CSS daemon is not running, run `localconfig add` as the root user.

    On Windows, you can use the `crsctl` and `localconfig` commands or the Windows Services GUI tool. To use Windows Services GUI tool, double-click the Services icon in the Windows Control Panel and locate the `OracleCSService` service. The status of `OracleCSService` should be `Started` and its startup type should be `Automatic`.

6. Run the Network Configuration Assistant (NETCA) to configure the listener.

    Start Network Configuration Assistant (NETCA) in the Oracle Grid Infrastructure home with `netca`. Follow the prompts in the wizard to add and configure the listener in the Oracle Grid Infrastructure home.

7. Run ASMCA from the Oracle Grid Infrastructure home to complete the upgrade.

    For information about starting ASMCA, see "Starting Oracle ASM Configuration Assistant" on page 11-1. If the Oracle ASM instance has been shut down during the Oracle Grid Infrastructure configuration, start that instance when ASMCA prompts you to start that instance.

    When ASMCA detects an earlier version of Oracle ASM, the assistant automatically determines the location of that instance and the location of the Oracle Grid Infrastructure home, as shown in Figure 11–3.

    Enter a password for the `ASMSNMP` user. Note that `ASMSNMP` is a less privileged user that is primarily used by Oracle Enterprise Manager to monitor Oracle ASM instances.

    When you have completed the page, select **Upgrade ASM** to complete the upgrade. After the process has completed, you can exit from ASMCA.

    When upgrading an Oracle ASM instance to the current software level, ASMCA moves the Oracle ASM instance pre-11g Release 2 (11.2) to the Oracle Grid Infrastructure home and updates the appropriate environmental variables.

8. Ensure that the Oracle Database instance and Oracle Enterprise Manager agent are running in the Oracle Database 11*g* Release 1 (11.1) home.

    Start the following services if necessary.

    -   Connect to the Oracle Database instance with SQL*Plus as a privileged user and run the `STARTUP` command.

■ Restart the Oracle Enterprise Manager agent with `emctl start dbconsole`.

> **Note:** The procedure described in this section upgrades the Oracle
> ASM instance only. Oracle Database, and Oracle Enterprise Manager,
> will not have the latest features. To upgrade Oracle Database, see
> *Oracle Database Upgrade Guide*.

Figure 11–3   Oracle ASM Configuration Assistant Upgrade ASM Page
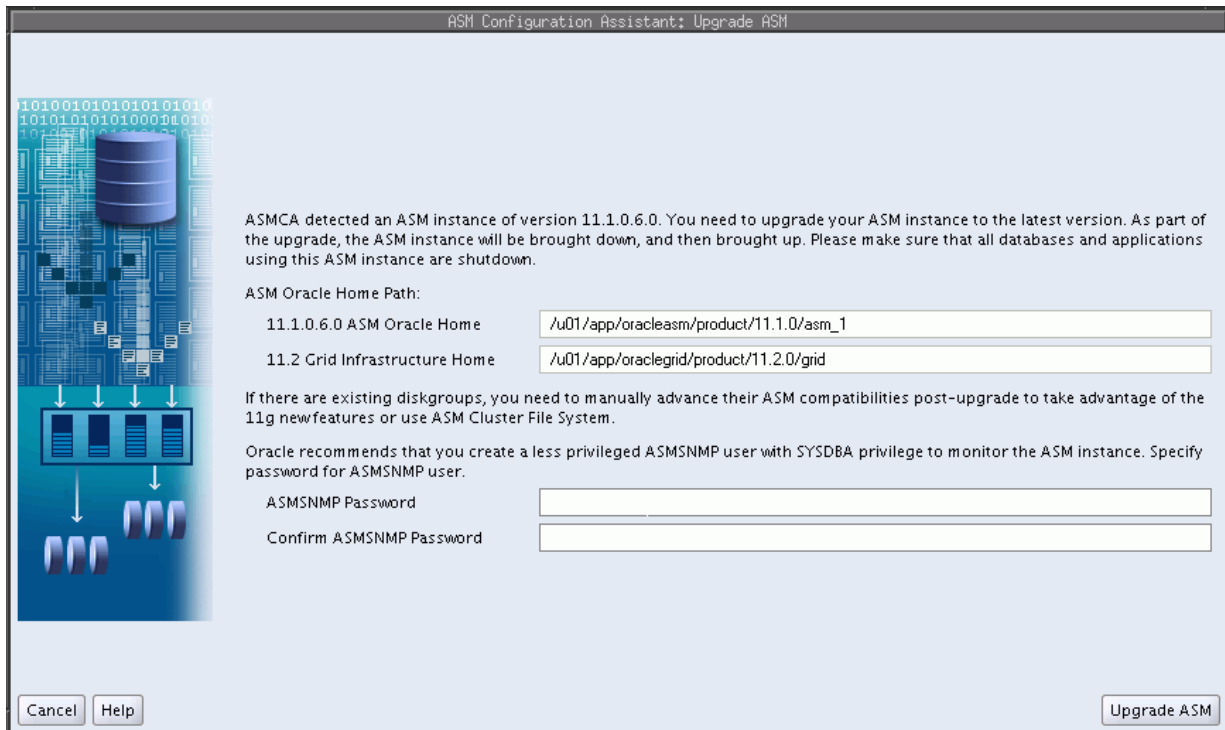
After upgrading an Oracle ASM instance, you can copy or move an Oracle ASM
instance. After copying or moving the SPFILE, you must restart the instance with the
SPFILE in the new location to use that SPFILE. For information, see "Backing Up,
Copying, and Moving an Oracle ASM Initialization Parameter File" on page 3-4.

> **See Also:**
>
> ■ *Oracle Database Upgrade Guide* for information about upgrading
>   Oracle Database
>
> ■ *Oracle Grid Infrastructure Installation Guide* for information about
>   installing Oracle Grid Infrastructure and performing a rolling
>   upgrade of Oracle ASM

Managing Disk Groups with Oracle ASM Configuration Assistant

Oracle ASM Configuration Assistant enables you to configure or create Oracle ASM
disk groups with the **Configure ASM Disk Groups** tab, as shown in Figure 11–4. The
disk group tab displays selected details about the disk group, such as name, size, free
space, usable space, redundancy level, and state of the disk group.

See Also: *Oracle Database Reference* for descriptions of disk group information displayed in the V$ASM_DISKGROUP view

You can right click a selected disk group in the screen to display the configuration options menu. The menu options enable you to:

- Add disks to the disk group
- Edit the disk group attributes
- Manage templates for the disk group
- Create an Oracle ACFS file system on the disk group for a database home
- Dismount and mount the disk group
- Drop the disk group

Figure 11–4   Oracle ASM Configuration Assistant Configure Disk Groups Page
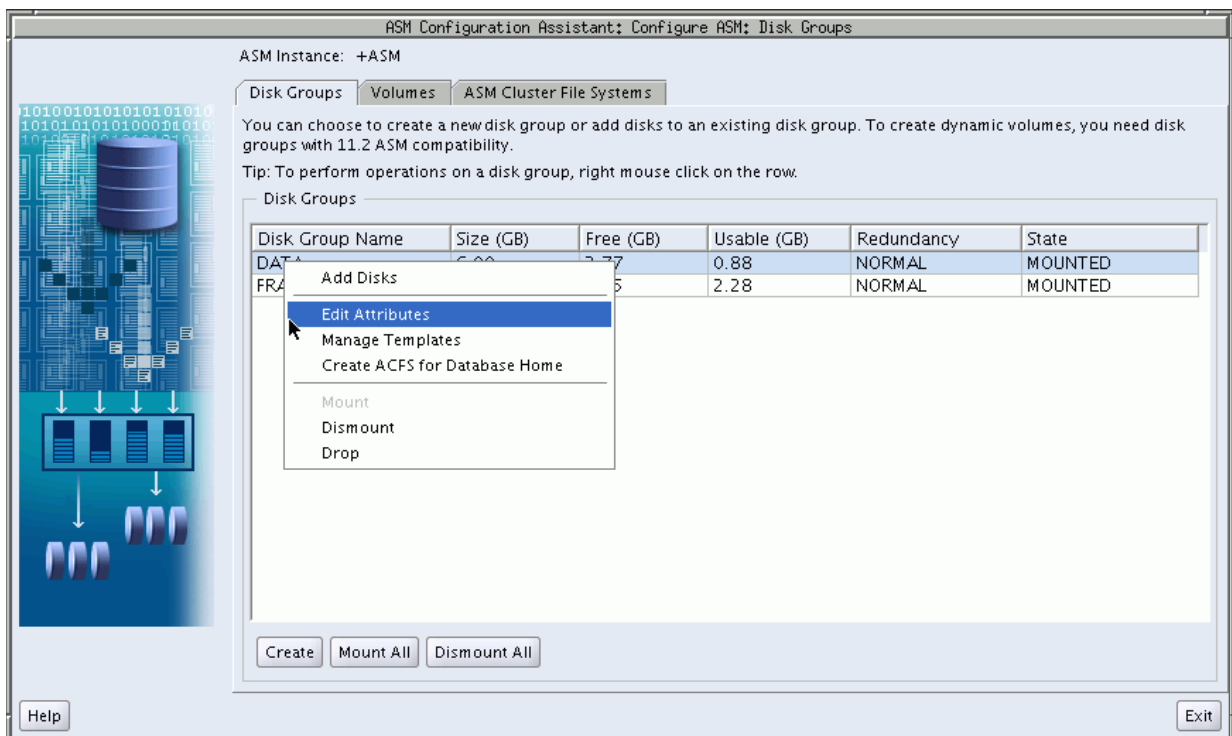
There are additional options for creating a disk group, mounting all disk groups, and dismounting all disk groups.

Oracle ASM Configuration Assistant enables you to create an Oracle ASM disk group by clicking **Create** on the Oracle ASM Configuration Assistant disk group page. The page for creating a disk group is shown in Figure 11–5.

Figure 11–5   Oracle ASM Configuration Assistant Create Disk Groups Page

On this page, you can enter a name for the disk group, select the redundancy level, change the disk discovery path to locate disks in a different directory, identify the disks to be in the disk group, enter a name for a failure group, and specify if the failure group is a quorum type. A quorum failure group is a special type of failure group and disks in these failure groups do not contain user data and a quorum failure group is not considered when determining redundancy requirements with respect to user storage. However, a quorum failure group counts when mounting a disk group. For information about quorum failure groups, see "Oracle Cluster Registry and Voting Files in Oracle ASM Disk Groups" on page 4-11.

The default disk group compatibility settings are 11.2 for Oracle ASM compatibility, 10.1 for database compatibility, and no value for Oracle ADVM compatibility. For information about disk group compatibility attributes, see "Disk Group Compatibility" on page 4-33. For information about disk discovery, see "Oracle ASM Disk Discovery" on page 4-20 and "ASM_DISKSTRING" on page 3-7.

You can choose to show advanced options when creating a disk group, as shown in Figure 11–6. The advanced options page enables you to set additional options, such as disk group compatibility settings and allocation unit (AU) size. When you are finished with the create disk group specifications, click **OK**.

> **Note:**   The disk group compatibility settings can only be advanced. You cannot revert to a lower compatibility setting after you advance the disk group compatibility settings. See "Overview of Disk Group Compatibility" on page 4-34

Figure 11–6   Oracle ASM Configuration Assistant Create Disk Groups Page - Advanced Options

For more information about managing disk groups, see Chapter 4, "Administering Oracle ASM Disk Groups".

Managing Oracle ADVM Volumes with Oracle ASM Configuration Assistant

Oracle ASM Configuration Assistant enables you to create or configure an Oracle ADVM volume, as shown in Figure 11–7.

Right click a selected volume in the Oracle ASM Configuration Assistant screen to display the configuration options menu. The menu options include:

■   Enable a disabled volume

■   Disable an enabled volume

■   Resize a volume that has not been mounted on an Oracle ACFS file system

■   Delete a volume

There are options for creating a volume, enabling all volumes, and disabling all volumes.

Figure 11–7 Oracle ASM Configuration Assistant Configure ASM Volumes Page
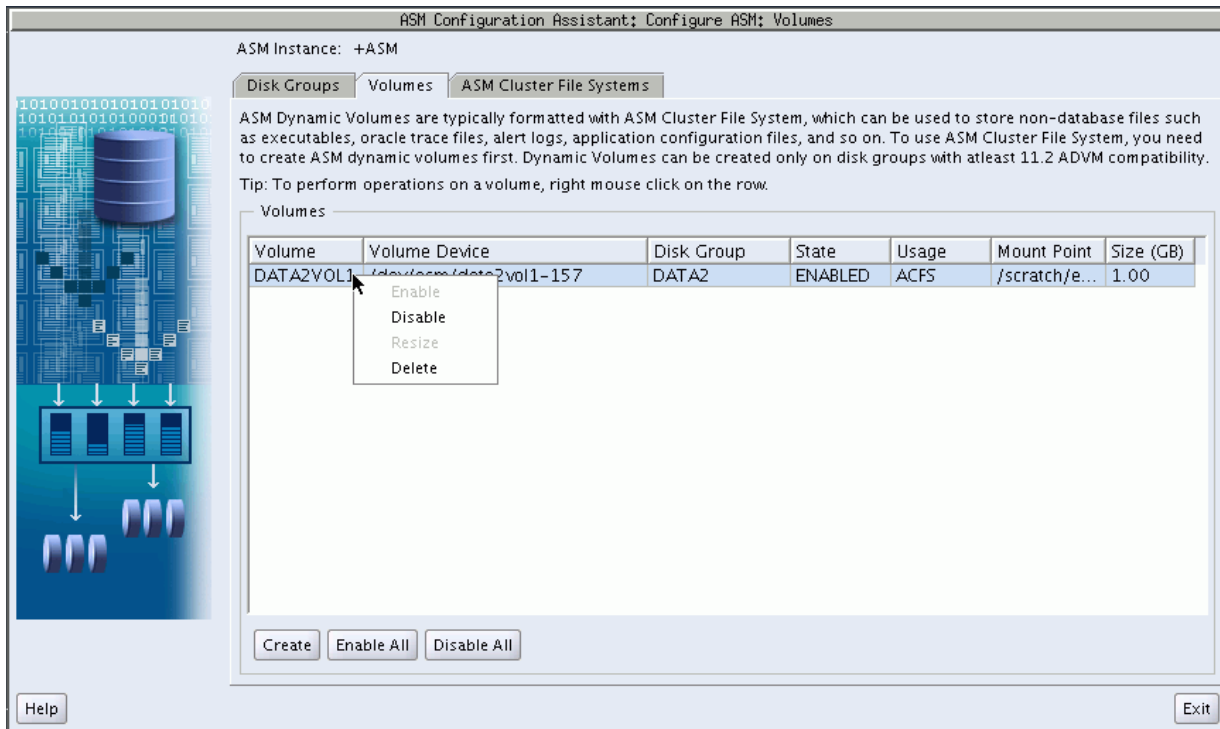
Oracle ASM Configuration Assistant enables you to create Oracle ADVM volumes, as shown in Figure 11–7.

Click **Create** to create a volume. You must provide a unique volume name for the existing Oracle ASM disk group that you select from the list. The volume name can be a maximum of 11 alphanumeric characters; dashes are not allowed. The first character must be alphabetic.

Specify the size for the volume. You can optionally choose to display advanced options for creating a volume which enables you to specify the redundancy level and striping characteristics.

If the number of stripe columns is set to 1, then striping is disabled and the stripe width is equal to the volume extent size. The volume extent size is 8 MB if the disk group allocation unit (AU) is less than or equal to 8 MB. If the AU size is greater than 8 MB, then the Oracle ADVM volume extent size is equivalent to the disk group AU size.

The compatibility parameters `COMPATIBLE.ASM` and `COMPATIBLE.ADVM` must be set to 11.2 or higher for the disk group to contain an Oracle ADVM volume. To use Oracle ACFS encryption, replication, security, or tagging, the disk group on which the volume is created for the file system must have compatibility attributes for `ASM` and `ADVM` set to `11.2.0.2`. See "Disk Group Compatibility Attributes" on page 4-34.

Before creating an Oracle ADVM volume on AIX, ensure that the necessary user authorizations have been created. For information, refer to "Oracle ACFS Command-line Tools for the AIX Environment" on page 13-34.

Figure 11–8   Oracle ASM Configuration Assistant Create an ASM Volume Page
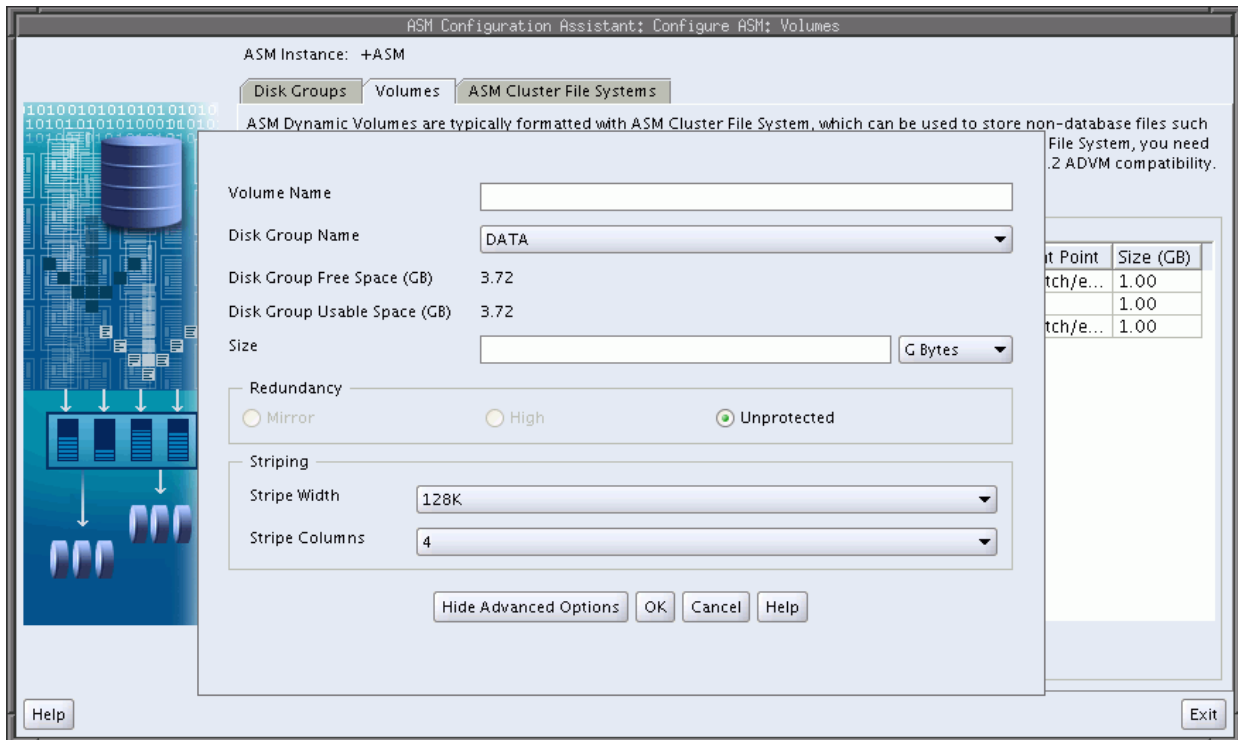
For information about creating a volume with the ASMCMD `volcreate` command, refer to "volcreate" on page 12-62. For an overview of Oracle ACFS, see Chapter 5, "Introduction to Oracle ACFS".

Managing Oracle ACFS File Systems with Oracle ASM Configuration Assistant

Oracle ASM Configuration Assistant enables you to create or configure an Oracle ACFS file system, as shown in Figure 11–9.

To configure an existing file system, right click a selected file system in the Oracle ASM Configuration Assistant screen to display the configuration options menu.

The menu options include:

- Show mount command for the file system
- Show dismount command for the file system
- Register mount point for the file system
- Deregister mount point for the file system
- Resize the file system
- Delete the file system

Some commands require root privileges, such as mounting a file system. Oracle ASM Configuration Assistant generates the command for you to run manually as root or as a privileged user.

There are buttons for Create, Show Mount All, and Show Dismount All commands.

Figure 11–9   Oracle ASM Configuration Assistant Configure Oracle ACFS Page
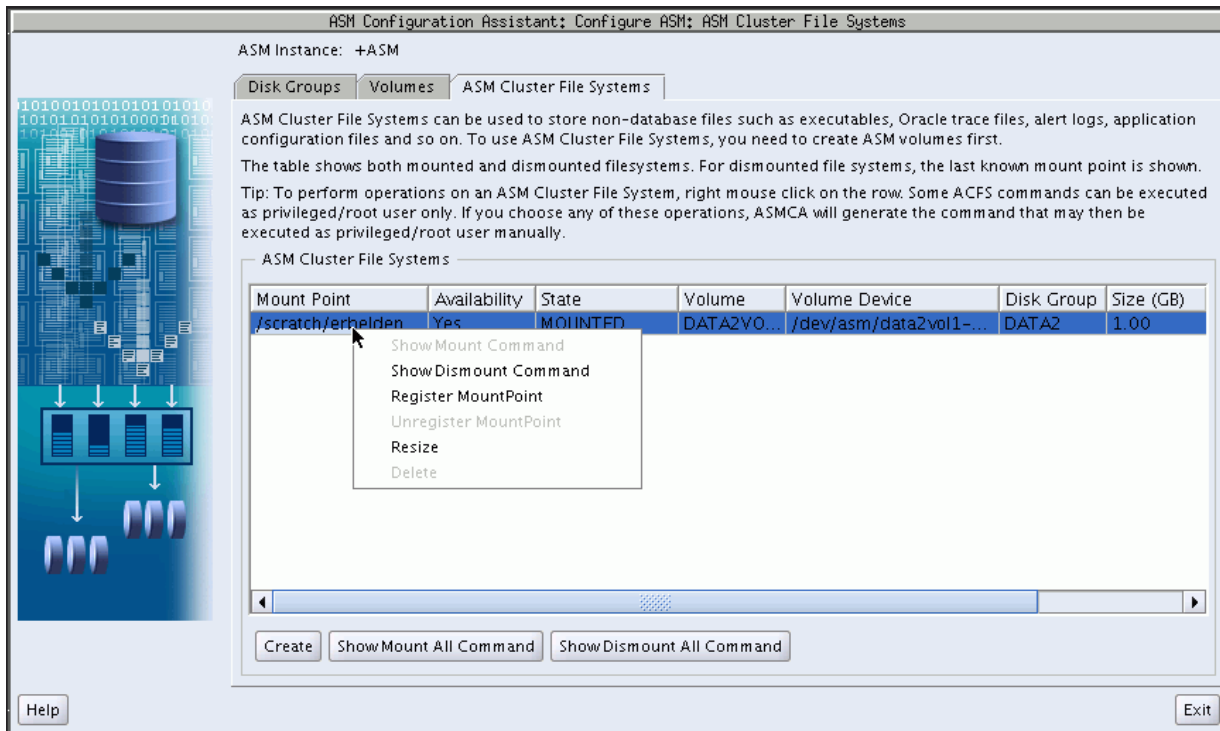
Oracle ASM Configuration Assistant enables you to create an Oracle ACFS file system, as shown in Figure 11–9.

Select **Create**, then select an existing Oracle ADVM volume from the list and determine the mount point field, show in Figure 11–10. The mount point must be an existing directory. The file system must be mounted to make it available. The mount command must be manually run as root or the Windows `Administrator` at an operating system prompt. For information about mounting an Oracle ACFS file system, see "mount" on page 13-24 for Linux or "acfsmountvol" on page 13-45 for Windows.

You can also choose to create an Oracle ACFS file system for an Oracle Database. For more information about creating a file system for a database home, see "Creating an Oracle ACFS File System for a Database Home" on page 11-15.

Figure 11–10   Oracle ASM Configuration Assistant Create Oracle ACFS Page
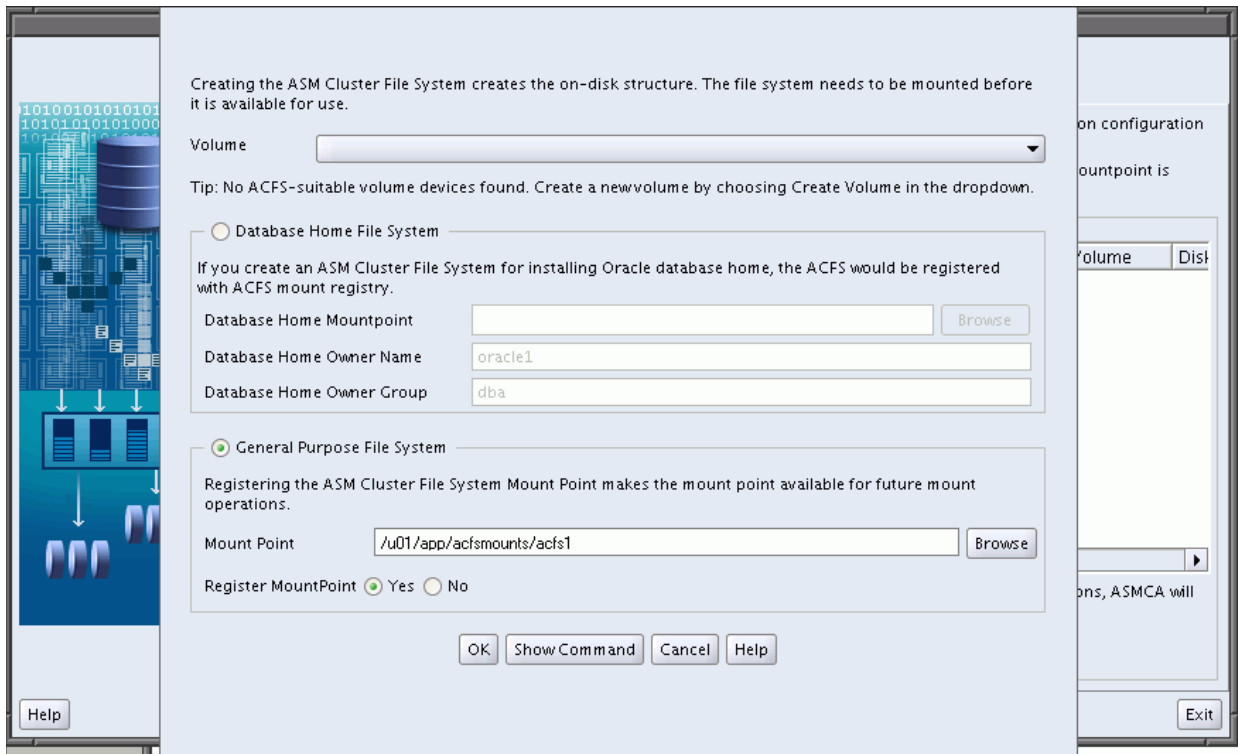
For information about Oracle ACFS, see Chapter 5, "Introduction to Oracle ACFS". For a summary of the basic steps for creating an Oracle ACFS file system, see "Basic Steps to Manage Oracle ACFS Systems" on page 13-1.

Managing Security and Encryption for Oracle ACFS File Systems with Oracle ASM Configuration Assistant

Oracle ASM Configuration Assistant enables you to configure security and encryption for an Oracle ACFS file system, as shown in Figure 11–11.

Figure 11–11   Oracle ASM Configuration Assistant Configure Oracle ACFS Security and Encryption Page
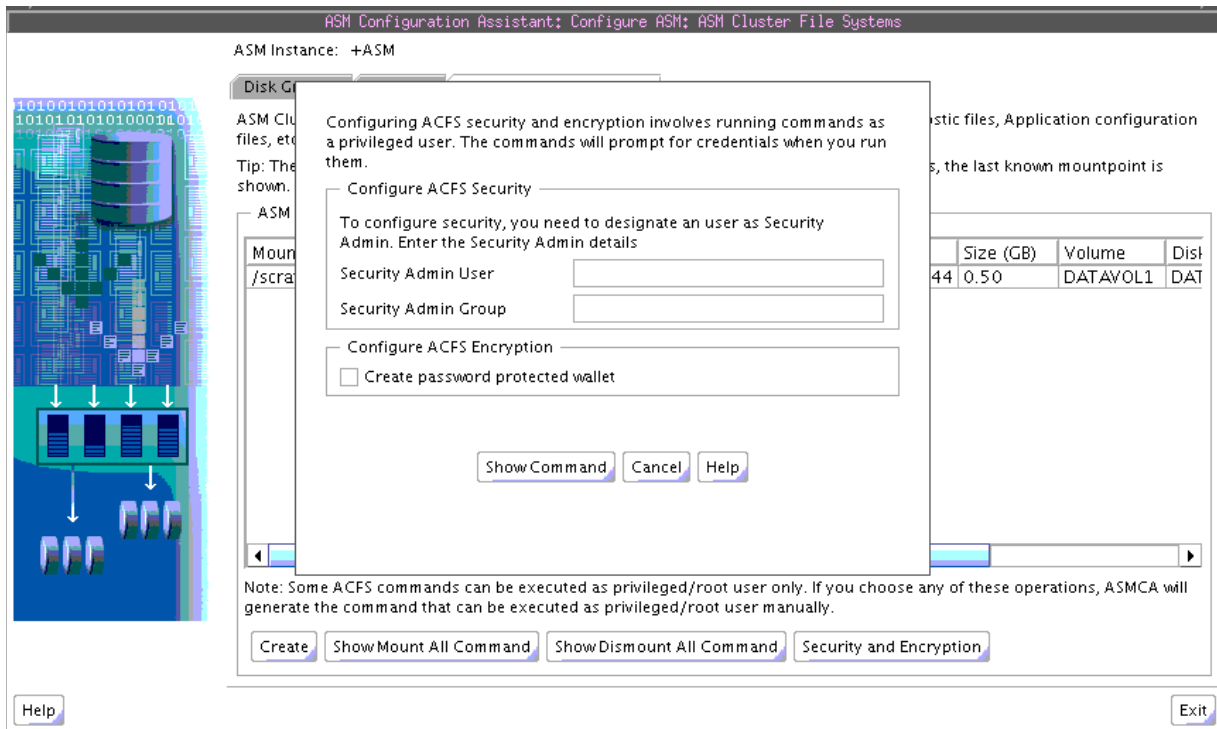
You must initialize the security system as the first step in configuring security for an Oracle ACFS file system. You must also initialize the encryption system as the first step in encrypting an Oracle ACFS file system. You do not have to use both Oracle ACFS security and encryption on the same file system. If you decide to use both security and encryption, then encryption must be initialized and set before enabling encryption on a security realm.

After you complete the entry fields in the dialog shown in Figure 11–11, click Show Command to display the commands you must run as a root or Administrator user at an operating system prompt. For example:

```
# /sbin/acfsutil sec init -u grid -g asmadmin
# /sbin/acfsutil encr init
```

For information about the `acfsutil sec init` and `acfsutil encr init` commands, refer to "acfsutil sec init" on page 13-71 and "acfsutil encr init" on page 13-90.

After security has been initialized, you can use the menu options to manage security and encryption for an Oracle ACFS file system, as shown in Figure 11–12. These menu options include:

- Enable Encryption
- Set Encryption
- Disable Encryption
- Prepare and Enable Security
- Disable Security
- Enable Security

Figure 11–12   Oracle ASM Configuration Assistant Oracle ACFS Security Menu Options
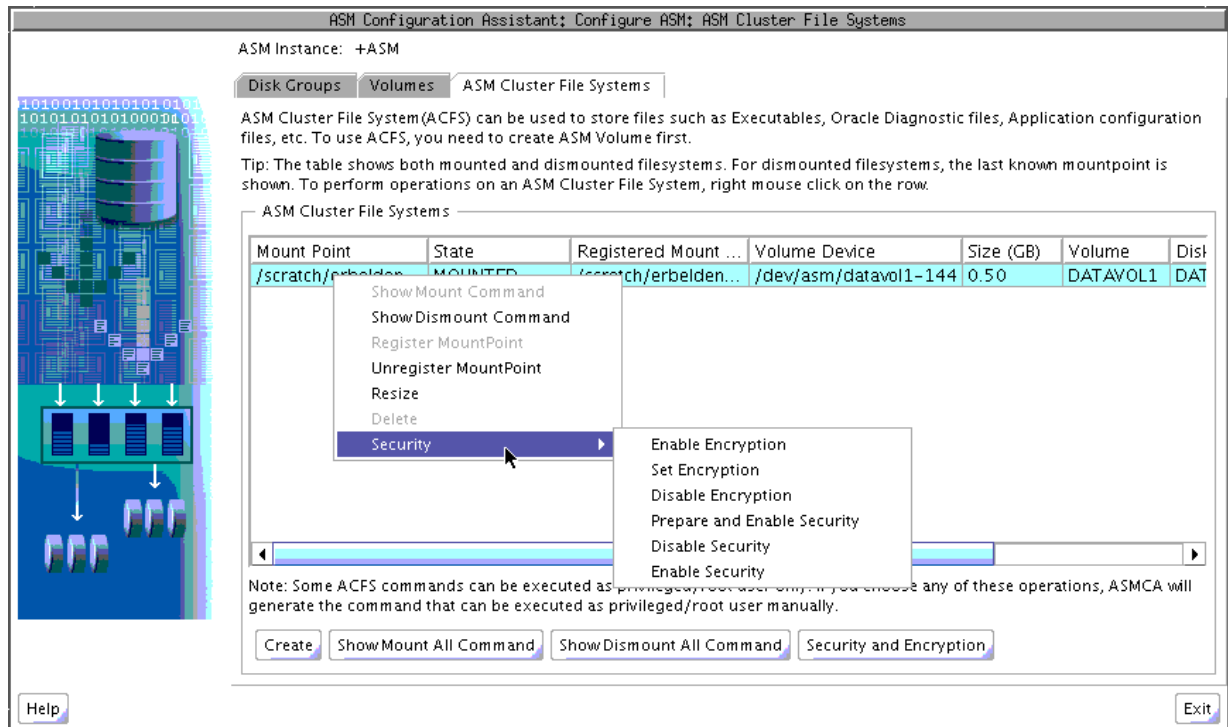
For information about Oracle ACFS security, see "Oracle ACFS Security" on page 5-13. For a summary of the basic steps for setting up security on an Oracle ACFS file system, see "Securing Oracle ACFS File Systems" on page 13-5.

For information about Oracle ACFS encryption, see "Oracle ACFS Encryption" on page 5-15. For a summary of the basic steps for setting up encryption on an Oracle ACFS file system, see "Encrypting Oracle ACFS File Systems" on page 13-10.

For information about the `acfsutil encr` and `acfsutil sec` commands, see "Oracle ACFS Command-Line Utilities for Multiple Environments" on page 13-95.

Creating an Oracle ACFS File System for a Database Home

To create an Oracle ACFS file system for a database home, you can select the **Create ACFS for Database Home** from the disk group configuration options menu, as shown in Figure 11–4 on page 11-7. The **Create ACFS Hosted Database Home** dialog displays as shown in Figure 11–13.

In this dialog, you must enter:

- Database Home Volume Name

  This is the name of the Oracle ADVM volume you want to create. For information about Oracle ADVM volumes, see "About Oracle ACFS" on page 5-4.

- Database Home Mount Point

  This is the mount point for the file system where you want to install the database home. The file system that contains the database home should not be located under the Oracle Grid Infrastructure base (`ORACLE_BASE` for grid) directory. For information about mount points and database homes, see "About Oracle ACFS and Oracle Database Homes" on page 5-5.

- Database Home Size in gigabytes (GB)

The default is 6 GB and the minimum size allowed.

- Database Home Owner Name

  This is the operating system name of the user that installs the database and owns the software in the database home.

- Database Home Owner Group

  This is the operating system group of the owner of the database home.

The mount point must be an existing directory. The file system must be mounted to make it available. The mount command must be manually run as root or the Windows Administrator at an operating system prompt. For information about mounting an Oracle ACFS file system, see "mount" on page 13-24 for Linux or "acfsmountvol" on page 13-45 for Windows.

For information on the mount registry, see "About the Oracle ACFS Mount Registry" on page 5-8.

Figure 11–13 Oracle ASM Configuration Assistant Create Oracle ACFS Hosted Database Home Page
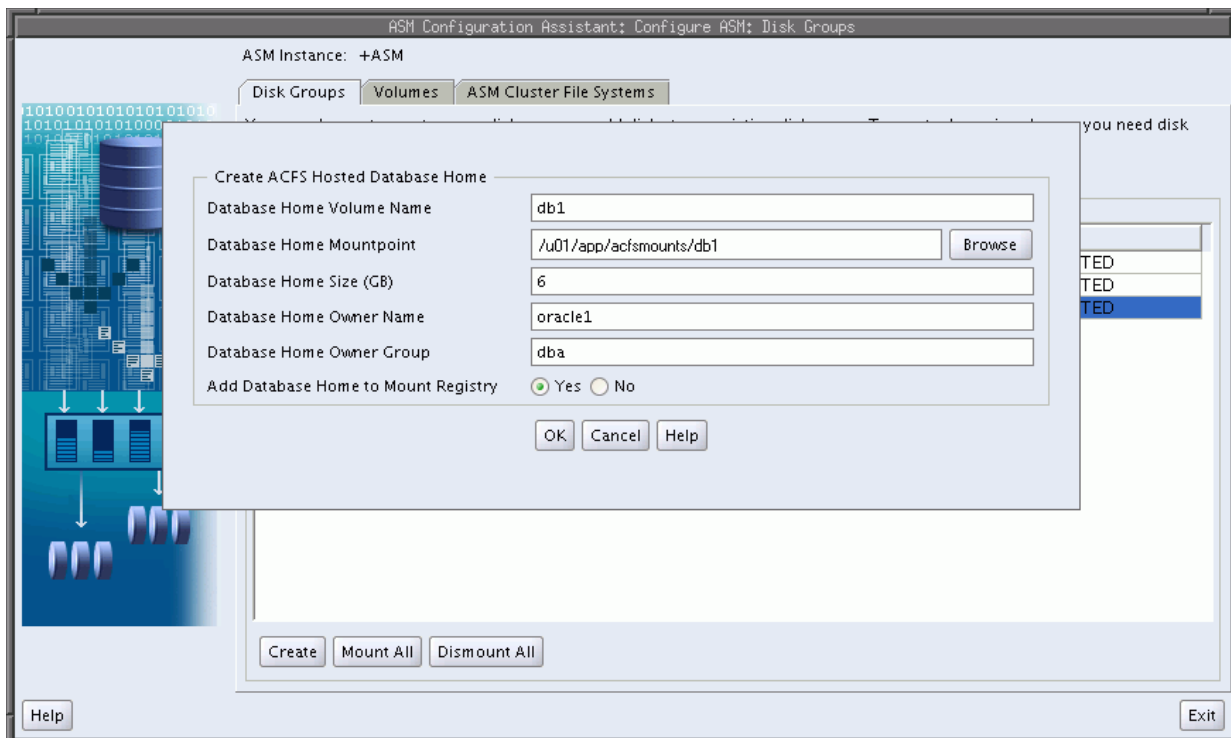
Oracle ASM Configuration Assistant Command-Line Interface

The ASMCA command-line interface provides non-GUI support for configuring Oracle ASM disk groups, volumes, and Oracle ACFS.

- Running ASMCA Command-Line
- ASMCA Commands

Running ASMCA Command-Line

The syntax for running the ASMCA command-line tool is:

```
asmca -silent command_name [options]
```

Table 11–1 contains the options available with the `asmca` command-line tool.

Table 11–1   Options for the ASMCA command-line tool

Option	Description
`-silent`	Specifies to run `asmca` in command-line mode.

`command_name`	Specifies the command to run. The command can be any of the following:
	
	`-configureASM`
	`-upgradeASM`
	`-configureParameter`
	`-deleteASM`
	`-createDiskGroup`
	`-addDisk`
	`-createVolume`
	`-createACFS`
	
	See "ASMCA Commands" on page 11-17.

`options`	Specifies the options to run for a command. The options vary by command. See the individual commands described in "ASMCA Commands" on page 11-17 for the available options. The following list describes the options that are common for multiple commands. ■　`-help` 　　Displays help for running ASMCA in silent mode. ■　`-asmsnmpPassword` *asmsnmp_password* 　　The Oracle ASM monitor user (`ASMSNMP`) password. This is optional and only required when creating and migrating an Oracle ASM instance. ■　`-sysAsmPassword` *sysasm_password* 　　The `SYSASM` password for Oracle ASM instance. This option is required.

The following is an example of the `-help` option.

```
$ asmca -help
asmca [-silent] [-sysAsmPassword <SYS user password>] [-asmsnmpPassword <ASMSNMP password>]
{<command> <options>}
Please refer to the manual for details.
You can enter one of the following commands:
...
```

ASMCA Commands

This section describes the commands that can be run with ASMCA command-line.

- Configure an Oracle ASM Instance

- Upgrade an Oracle ASM Instance

- Configure Parameters for an Oracle ASM Instance

- Delete Oracle ASM Instances

- Create a Disk Group

- Add a Disk to a Disk Group

- Create a Volume

- Create an Oracle ACFS File System

Configure an Oracle ASM Instance

`-configureASM` configures or creates an Oracle ASM instance.

Syntax

```
asmca -silent
    -configureASM
      [-sysAsmPassword sysasm_password]
      [-asmsnmpPassword asm_monitor_password]
      [{-param paramname=paramvalue,...}]
      [{-diskString discovery_path
            -diskGroupName diskgroup
            -diskList disk_list
            [-redundancy{HIGH | NORMAL | EXTERNAL}]}]
```

Table 11–2 contains the options available with the `-configureASM` command.

Table 11–2   Options for the -configureASM command

Option	Description
-param paramname=paramvalue,...	Specifies a list of initialization parameters with values for the Oracle ASM instance.
-diskString discovery_path	Specifies the Oracle ASM discovery string for locating disks.
-diskGroupName diskgroup	Specifies the name of the Oracle ASM disk group to create.
-diskList disk_list	Specifies a comma-delimited list of disk names to be added to the disk group.
-redundancy{HIGH	NORMAL	EXTERNAL}	Specifies the redundancy setting for the disk group.

For additional options that are common to multiple commands, see Table 11–1 on page 11-17.

Examples

To configure an Oracle ASM instance:

Example 11–1   Using asmca -silent -configureASM

```
asmca -silent
        -configureASM
          -diskString '/devices/disk*'
          -sysAsmPassword my_sysasm_password
          -asmsnmpPassword my_asmsnmp_passwd
```

Upgrade an Oracle ASM Instance

`-upgradeASM` upgrades an Oracle ASM instance from a previous release to the current software release.

Upgrading an Oracle ASM instance with ASMCA only updates the Oracle ASM instance software and does not configure Oracle Clusterware or the listener.

An Oracle ASM instance should be upgraded with Oracle Universal Installer (OUI). OUI automatically defaults to upgrade mode when it detects an Oracle ASM instance at a previous release level. See "Upgrading an Oracle ASM Instance in an Oracle Restart Configuration with Oracle Universal Installer" on page 3-18.

Syntax
```
asmca -silent
    -upgradeASM
      -local -lastnode
      [ -asmsnmpPassword asmsnmp_password ]
```

The options available with the -upgradeASM command are common to multiple commands. For a description of those options, see Table 11–1 on page 11-17.

Examples
To upgrade an Oracle ASM instance:

Example 11–2   Using asmca -silent -upgradeASM
```
asmca -silent
      -upgradeASM
        -asmsnmpPassword my_asmsnmp_passwd
```

Configure Parameters for an Oracle ASM Instance

-configureParameter modifies the parameters of an Oracle ASM instance.

Syntax
```
asmca -silent
      -configureParameter
        [ { -param paramname=paramvalue , ... } ]
        [-sysAsmPassword sysasm_password ]
```

Table 11–3 contains the options available with the -configureParameter command.

Table 11–3   Options for the -configureParameter command

Option	Description
-param paramname=paramvalue , ...	Specifies a list of initialization parameters with values for the Oracle ASM instance.

For additional options that are common to multiple commands, see Table 11–1 on page 11-17.

Examples
To configure parameters for an Oracle ASM instance:

Example 11–3   Using asmca -silent -configureParameter
```
asmca -silent
      -configureParameter
          (-param
           asm_power_limit=3)
        -sysAsmPassword my_sysasm_password
```

Delete Oracle ASM Instances

-deleteASM deletes all the Oracle ASM instances that are present. The operation stops and an error is raised if there is a database using the Oracle ASM instance.

Syntax

```
asmca -silent
    -deleteASM
      [-sysAsmPassword sysasm_password ]
      [ -force ]
      [ -dropDiskGroups ]
```

Table 11–4 contains the options available with the -deleteASM command.

Table 11–4   Options for the -deleteASM command

Option	Description
-force	Forces the deletion of an Oracle ASM instance.
-dropDiskGroups	Drops the disk groups on this instance.

For additional options that are common to multiple commands, see Table 11–1 on page 11-17.

Examples

To delete an Oracle ASM instance:

Example 11–4   Using asmca -silent -deleteASM

```
asmca -silent
        -deleteASM
          -sysAsmPassword my_sysasm_password
          -force
```

Create a Disk Group

-createDiskGroup creates disk groups in an Oracle ASM instance.

Syntax

```
asmca -silent
    -createDiskGroup
      [ -diskString discovery_path ]
      { -diskGroupName diskgroup
         { -disk disk_path [ -diskName disk] |
           -diskList disk_list
              [ -diskSize disk_size_MB]
              [ -failuregroup failure_group]
              [ -force|-noforce]
              [ -quorum|-noquorum] }
      [ -redundancy { HIGH | NORMAL | EXTERNAL } ] } ]
      [ -au_size value]
      [ -compatible.asm value ]
      [ -compatible.rdbms value ]
      [ -compatible.advm value ] }
      [ -sysAsmPassword sysasm_password ]
```

Table 11–5 contains the options available with the -createDiskGroup command.

Table 11–5   Options for the -createDiskGroup command

Option	Description
-diskString *discovery_path*	Specifies the Oracle ASM discovery string for locating disks.
-diskGroupName *diskgroup*	Specifies the name of the Oracle ASM disk group to create.
-disk *disk_path*	Specifies the full path of the disk device.
-diskName *disk*	Specifies a user-supplied disk name.
-diskList *disk_list*	Specifies a comma-delimited list of disk names to be added to the disk group.
-diskSize *disk_size_MB*	Specifies the size of the disk to use for Oracle ASM storage.
-failuregroup *failure_group*	Specifies the failure group name.
-force | -noforce	Specifies where to force the use of the disk.
-quorum | -noquorum	Specifies a quorum or standard failure group.
-redundancy { HIGH | NORMAL | EXTERNAL }	Specifies the redundancy setting for the Oracle ASM disk group.
-au_size	Specifies the allocation unit size (1,2,4,8,16,32,64)
-compatible.asm	Specifies the attribute setting.
-compatible.rdbms	Specifies the attribute setting.
-compatible.advm	Specifies the attribute setting.

For additional options that are common to multiple commands, see Table 11–1 on page 11-17.

The default disk group compatibility settings are 11.2 for Oracle ASM compatibility, 10.1 for database compatibility, and no value for Oracle ADVM compatibility. For information about disk group compatibility attributes, see "Disk Group Compatibility" on page 4-33.

Examples

To create disk groups in an Oracle ASM instance:

Example 11–5   Using asmca -silent -createDiskGroup

```
$ asmca -silent -createDiskGroup
      -diskGroupName mynewdg
          -disk '/devices/diske*'
          -disk '/devices/diskk*'
      -redundancy NORMAL
      -au_size 64
      -compatible.asm '11.2.0.0.0'
      -compatible.rdbms '11.2.0.0.0'
      -compatible.advm '11.2.0.0.0'

DiskGroup mynewdg created successfully.
```

Add a Disk to a Disk Group

-addDisk add disks to an existing disk group in an Oracle ASM instance.

Syntax

```
asmca -silent
      -addDisk
        [ -diskString discovery_path ]
          {-diskGroupName diskgroup
            {-disk disk_path [-diskName disk] |
             -diskList disk_list
                [-diskSize disk_size_MB]
                [-failuregroup failure_group]
                [-force|-noforce]
                [-quorum|-noquorum] }
        [-sysAsmPassword sysasm_password ]
```

Table 11–6 contains the options available with the -addDisk command.

Table 11–6   Options for the -addDisk command

Option	Description
-diskString discovery_path	Specifies the Oracle ASM discovery string for locating disks.
-diskGroupName diskgroup	Specifies the name of the Oracle ASM disk group to update.
-diskList disk_list	Specifies a comma-delimited list of disk names to be added to the disk group.
-diskSize disk_size_MB	Specifies the size of the disk to use for Oracle ASM storage.
-failuregroup failure_group	Specifies the failure group name.
-force	-noforce	Specifies where to force the use of the disk.
-quorum	-noquorum	Specifies a quorum or standard failure group.

For additional options that are common to multiple commands, see Table 11–1 on page 11-17.

Examples

To add disks to an existing disk group:

Example 11–6   Using asmca -silent -addDisk

```
asmca -silent
      -addDisk
        -diskGroupName mynewdg
            -disk '/devices/diskl1'
            -disk '/devices/diskl2'

Disks added successfully to diskgroup mynewdg.
```

Create a Volume

-createVolume creates Oracle ADVM volumes.

Syntax

```
asmca -silent
    -createVolume
      { -volumeName volume_name
        -volumeDiskGroup diskgroup
        -volumeSizeGB size_GB
      [ -volumeRedundancy { INHERIT | HIGH | NORMAL | EXTERNAL } ] ... }
      [-sysAsmPassword sysasm_password ]
```

Table 11–7 contains the options available with the -createVolume command.

Table 11–7   Options for the -createVolume command

Option	Description
-volumeName volume_name	Specifies the name of the volume to create.
-volumeDiskGroup diskgroup	Specifies the name of the disk group where you want to create the volume.
-volumeSizeGB size_GB	Specifies the size of the volume in Gigabytes.
-volumeRedundancy { INHERIT	HIGH	NORMAL	EXTERNAL }	Specifies the redundancy setting for the volume.

For additional options that are common to multiple commands, see Table 11–1 on page 11-17.

Examples

To create an Oracle ADVM volume:

Example 11–7   Using asmca -silent -createVolume

```
$ asmca -silent
      -createVolume
          -volumeName volume1
          -volumeDiskGroup mynewdg
          -volumeSizeGB 1

Volume volume1 created successfully.
```

Create an Oracle ACFS File System

-createACFS creates an Oracle Automatic Storage Management Cluster File System (Oracle ACFS).

This command does not mount the Oracle ACFS file system. For information about mounting an Oracle ACFS file system, see "mount" on page 13-24 for Linux environments and "acfsmountvol" on page 13-45 for Windows environments.

Syntax

```
asmca -silent
    -createACFS
      { -acfsVolumeDevice volume_device }
      [-sysAsmPassword sysasm_password ]
```

Table 11–8 contains the options available with the -createACFS command.

Table 11–8   Options for the -createACFS command

Option	Description
`-acfsVolumeDevice` *volume_device*	Specifies the name of the Oracle ADVM volume device.

For additional options that are common to multiple commands, see Table 11–1 on page 11-17.

Examples

To create an Oracle ACFS file system:

Example 11–8   Using asmca -silent -createACFS

```
$ asmca -silent
        -createACFS
            -acfsVolumeDevice /dev/asm/volume1-457


ACFS creation on /dev/asm/volume1-457 completed successfully.
The ACFS should be mounted for use.
```

12

Oracle ASM Command-Line Utility

This chapter describes the Oracle Automatic Storage Management (Oracle ASM) Command-Line Utility (ASMCMD). This chapter contains the following topics:

- About ASMCMD
- ASMCMD Instance Management Commands
- ASMCMD File Management Commands
- ASMCMD Disk Group Management Commands
- ASMCMD Template Management Commands
- ASMCMD File Access Control Commands
- ASMCMD Volume Management Commands

About ASMCMD

ASMCMD is a command-line utility that you can use to manage Oracle ASM instances, disk groups, file access control for disk groups, files and directories within disk groups, templates for disk groups, and volumes.

You can run the ASMCMD utility in either interactive or noninteractive mode.

This section contains the following topics:

- Types of ASMCMD Commands
- About Oracle ASM Files, Filenames, Directories, and Aliases
- Preparing to Run ASMCMD
- Running ASMCMD in Interactive Mode
- Running ASMCMD in Noninteractive Mode
- Getting Help

Types of ASMCMD Commands

The types of ASMCMD commands are listed in Table 12–1.

Table 12–1   Types of ASMCMD commands

Category	Commands
ASMCMD Instance Management Commands	dsget, dsset, lsct, lsop, lspwusr, orapwusr, shutdown, spbackup, spcopy, spget, spmove, spset, startup
ASMCMD File Management Commands	cd, cp, du, find, ls, lsof, mkalias, pwd, rm, rmalias
ASMCMD Disk Group Management Commands	chdg, chkdg, dropdg, iostat, lsattr, lsdg, lsdsk, lsod, md_backup, md_restore, mkdg, mount, offline, online, rebal, remap, setattr, umount
ASMCMD Template Management Commands	chtmpl, lstmpl, mktmpl, rmtmpl
ASMCMD File Access Control Commands	chgrp, chmod, chown, groups, grpmod, lsgrp, lsusr, mkgrp, mkusr, passwd, rmgrp, rmusr
ASMCMD Volume Management Commands	volcreate, voldelete, voldisable, volenable, volinfo, volresize, volset, volstat

About Oracle ASM Files, Filenames, Directories, and Aliases

ASMCMD works with Oracle ASM files, directories, and aliases. Before using ASMCMD, you should understand how these common computing concepts apply to the Oracle ASM environment.

This section contains the following topics about some key definitions:

- System-Generated Filename or Fully Qualified Filename
- Directory
- Alias
- Absolute Path and Relative Path
- Wildcard Characters

System-Generated Filename or Fully Qualified Filename

Every file created in Oracle ASM gets a system-generated filename, otherwise known as a fully qualified filename. This is similar to a complete path name in a local file system.

Oracle ASM generates filenames according to the following scheme:

+diskGroupName/databaseName/fileType/fileTypeTag.fileNumber.incarnation

An example of a fully qualified filename is the following:

+data/orcl/CONTROLFILE/Current.256.541956473

In the previous fully qualified filename, data is the disk group name, orcl is the database name, CONTROLFILE is the file type, and so on.

Only the slash (/) is supported by ASMCMD. Filenames are not case sensitive, but are case retentive. If you type a path name as lowercase, ASMCMD retains the lowercase.

For more information about Oracle ASM filenames, refer to "Fully Qualified File Name Form" on page 7-4.

Directory

As in other file systems, an Oracle ASM directory is a container for files, and an Oracle ASM directory can be part of a tree structure of other directories. The fully qualified filename represents a hierarchy of directories in which the plus sign (+) represents the

root directory. In each disk group, Oracle ASM automatically creates a directory hierarchy that corresponds to the structure of the fully qualified filenames in the disk group. The directories in this hierarchy are known as system-generated directories.

ASMCMD enables you to move up and down in this directory hierarchy with the `cd` (change directory) command. The ASMCMD `ls` (list directory) command lists the contents of the current directory, while the `pwd` command prints the name of the current directory.

When you start ASMCMD, the current directory is set to root (+). For an Oracle ASM instance with two disk groups, for example, `data` and `fra`, entering an `ls` command with the root directory as the current directory produces the following output:

```
ASMCMD> ls
data/
fra/
```

The following example demonstrates navigating the Oracle ASM directory tree (refer to the fully qualified filename shown previously):

```
ASMCMD> cd +data/orcl/CONTROLFILE
ASMCMD> ls
Current.256.541956473
Current.257.541956475
```

You can create your own directories as subdirectories of the system-generated directories using the ASMCMD `mkdir` command. The directories that you create can have subdirectories, and you can navigate the hierarchy of both system-generated directories and user-created directories with the `cd` command.

The following example creates the directory `mydir` under `orcl` in the disk group `data`:

```
ASMCMD> mkdir +data/orcl/mydir
```

> **Note:** The directory `orcl` is a system-generated directory. The contents of `data` represent the contents of disk group `data`.

If you start ASMCMD with the `-p` flag, then ASMCMD shows the current directory as part of its prompt. See "Including the Current Directory in the ASMCMD Prompt" on page 12-8.

```
ASMCMD [+] > cd data/orcl
ASMCMD [+data/orcl] >
```

ASMCMD retains the case of the directory that you entered.

Alias

Aliases are filenames that are references or pointers to system-generated filenames. However, aliases are user-friendly names. Aliases are similar to symbolic links in UNIX or Linux computers. You can create aliases to simplify Oracle ASM filename administration. You can create aliases with the `mkalias` ASMCMD command or a SQL `ALTER DISKGROUP` command.

An alias has at a minimum the disk group name as part of its complete path. You can create aliases at the disk group level or in any system-generated or user-created subdirectory. The following are examples of aliases:

```
+data/ctl1.f
```

```
+data/orcl/ctl1.f
+data/mydir/ctl1.f
```

If you run the ASMCMD `ls` (list directory) with the `-l` flag, each alias is listed with the system-generated file to which the alias refers.

```
ctl1.f => +data/orcl/CONTROLFILE/Current.256.541956473
```

For more information about aliases, refer to "Alias Oracle ASM Filename Forms" on page 7-5.

Absolute Path and Relative Path

When you run an ASMCMD command that accepts a filename or directory name as an argument, you can use the name as either an absolute path or a relative path.

An absolute path refers to the full path of a file or directory. An absolute path begins with a plus sign (+) followed by a disk group name, followed by subsequent directories in the directory tree. The absolute path includes directories until the file or directory of interest is reached. A complete system-generated filename, otherwise known as the fully qualified filename, is an example of an absolute path to a file.

Using an absolute path enables the command to access the file or directory regardless of where the current directory is set. The following `rm` command uses an absolute path for the filename:

```
ASMCMD [+] > rm +data/orcl/datafile/users.259.555341963
```

The following `cd` command uses an absolute path to the directory.

```
ASMCMD [+data/mydir] > cd +data/orcl/CONTROLFILE
```

A relative path includes only the part of the filename or directory name that is not part of the current directory. That is, the path to the file or directory is relative to the current directory.

In the following example, the `rm` command operates on the file `undotbs1.272.557429239`, which in this case is a relative path. ASMCMD appends the current directory to the command argument to obtain the absolute path to the file. In this example this is `+data/orcl/DATAFILE/undotbs1.272.557429239`.

```
ASMCMD [+] > cd +data
ASMCMD [+data] > cd orcl/DATAFILE
ASMCMD [+data/orcl/DATAFILE] > ls
EXAMPLE.269.555342243
SYSAUX.257.555341961
SYSTEM.256.555341961
UNDOTBS1.258.555341963
UNDOTBS1.272.557429239
USERS.259.555341963
ASMCMD [+data/orcl/DATAFILE] > rm undotbs1.272.557429239
```

Paths to directories can also be relative. You can go up or down the hierarchy of the current directory tree branch by providing a directory argument to the `cd` command whose path is relative to the current directory.

In addition, you can use the pseudo-directories "." and ".." rather than a directory name. The "." pseudo-directory is the current directory. The ".." pseudo-directory is the parent directory of the current directory.

The following example demonstrates how to use relative directory paths and pseudo-directories:

```
ASMCMD [+data/orcl] > cd DATAFILE
ASMCMD [+data/orcl/DATAFILE] >cd ..
ASMCMD [+data/orcl] >
```

Wildcard Characters

The wildcard characters `*` and `%` match zero or more characters anywhere within an absolute or relative path, which saves typing of the full directory or file name. The two wildcard characters behave identically. There are various ASMCMD commands that accept wildcards. These include `cd`, `du`, `find`, `ls`, `lsattr`, `lsdg`, `lsdsk`, `lsgrp`, `lsusr`, and `rm`.

If a wildcard pattern matches only one directory when using wildcard characters with `cd`, then `cd` changes the directory to that destination. If the wildcard pattern matches multiple directories, then ASMCMD does not change the directory but instead returns an error.

Example 12–1 illustrates the use of wildcards.

Example 12–1   Using wildcards with ASMCMD commands

```
ASMCMD [+] > cd +data/orcl/*FILE
ASMCMD-08005: +data/orcl/*FILE: ambiguous

ASMCMD [+] > cd +data/orcl/C*
ASMCMD [+data/orcl/CONTROLFILE] >

ASMCMD [+] > ls +fra/orcl/A%
2009_07_13/
2009_07_14/

ASMCMD [+] >  ls +fra/orcl/ARCHIVELOG/2009%

+fra/orcl/ARCHIVELOG/2009_07_13/:
thread_1_seq_3.260.692103543
thread_1_seq_4.261.692108897
thread_1_seq_5.262.692125993
thread_1_seq_6.263.692140729
thread_1_seq_7.264.692143333

+fra/orcl/ARCHIVELOG/2009_07_14/:
thread_1_seq_8.271.692158265
thread_1_seq_9.272.692174597

ASMCMD [+] > ls data/orcl/*

+data/orcl/CONTROLFILE/:
Current.260.692103157

+data/orcl/DATAFILE/:
EXAMPLE.265.692103187
SYSAUX.257.692103045
SYSTEM.256.692103045
UNDOTBS1.258.692103045
USERS.259.692103045

+data/orcl/ONLINELOG/:
group_1.261.692103161
group_2.262.692103165
group_3.263.692103169
```

```
+data/orcl/PARAMETERFILE/:
spfile.266.692103315

+data/orcl/TEMPFILE/:
TEMP.264.692103181
spfileorcl.ora
```

Preparing to Run ASMCMD

You can run the ASMCMD utility in either interactive or noninteractive mode.

Before running ASMCMD, review the items in the following list.

- Log in to the host which contains the Oracle ASM instance that you plan to administer.

   You must log in as a user that has SYSASM or SYSDBA privileges through operating system authentication. The SYSASM privilege is the required connection to administer the Oracle ASM instance. See "Authentication for Accessing Oracle ASM Instances" on page 3-25.

   > **See Also:**   Refer to the *Oracle Database Administrator's Guide* for information about operating system authentication

- To connect to the Oracle ASM instance, run ASMCMD that is located in the `bin` subdirectory of the Oracle Grid Infrastructure home (Oracle ASM home).

   Connect as SYSASM, the default connection, to administer an Oracle ASM instance.

   Ensure that the `ORACLE_HOME` and `ORACLE_SID` environment variables to refer to the Oracle ASM instance. Depending on your operating system, you might have to set other environment variables to properly connect to the Oracle ASM instance.

   Ensure that the `bin` subdirectory of your Oracle Grid Infrastructure home is in your `PATH` environment variable.

   > **See Also:**   Refer to the *Oracle Database Administrator's Guide* for more information about setting environment variables

   The default value of the Oracle ASM SID for a single-instance database is `+ASM`. In Oracle Real Application Clusters environments, the default value of the Oracle ASM SID on any node is `+ASM`*node#*.

- To use most of the ASMCMD commands, ensure that the Oracle ASM instance is started and the Oracle ASM disk groups are mounted.

   If the Oracle ASM instance is not running, ASMCMD runs only those commands that do not require an Oracle ASM instance. The commands include `startup`, `shutdown`, `lsdsk`, `help`, and `exit`. If you attempt to run other ASMCMD commands, an error message displays.

- You can connect to the database instance as SYSDBA by running ASMCMD that is located in the `bin` directory of the Oracle Database home.

   Ensure that the `ORACLE_HOME` and `ORACLE_SID` environment variables to refer to the database instance. Depending on your operating system, you might have to set other environment variables to properly connect to the database instance.

You must include the `--privilege` option to connect as SYSDBA. See "Specifying the Type of Connection" on page 12-8.

With this connection, there is a limited set of operations that can be run. For more information, see "The SYSDBA Privilege for Managing Oracle ASM Components" on page 3-29.

When administering disk groups, Oracle recommends that you run ASMCMD from the database home of the database instance that is the owner of the files in the disk group.

Running ASMCMD in Interactive Mode

The interactive mode of the ASMCMD utility provides a shell-like environment where you are prompted to enter ASMCMD commands. The syntax for starting ASMCMD in interactive mode is:

```
asmcmd [-V]
asmcmd [-v {errors|warnings|normal|info|debug}]
       [ --privilege connection_type ] [-p]
```

Table 12–2 summarizes the ASMCMD options.

Table 12–2   ASMCMD interactive options

Option	Description
-V	Displays the version of ASMCMD and then exits.
-v *display_level*	Displays additional information with some commands to help users diagnose problems and writes to a message file when logging is required. normal is the default if the display level is not specified.
--privilege *connection_ type*	Specifies the privilege to connect as when accessing an Oracle ASM or database instance. Can be either SYSASM or SYSDBA. The default is SYSASM and is used when administering the Oracle ASM instance.
-p	Displays the current directory in the prompt.

ASMCMD can be started with multiple options. For example:

```
asmcmd -p -v info
```

For the majority of the examples in this chapter, ASMCMD was started with the `-p` option.

Running ASMCMD without Any Options

To run ASMCMD in interactive mode without any options:

1. Enter the following at the operating system command prompt:

   ```
   asmcmd
   ```

   Oracle displays an ASMCMD command prompt as follows:

   ```
   ASMCMD>
   ```

2. Enter an ASMCMD command and press `Enter`. The command runs and displays its output, if any, and then ASMCMD prompts for the next command.

3. Continue entering ASMCMD commands until you have completed the tasks.

4. Enter the `exit` or `quit` command to exit ASMCMD.

Specifying the Type of Connection

You can specify the `--privilege` option to choose the type of connection, either `SYSASM` or `SYSDBA`. The default value is `SYSASM` and is used when administering an Oracle ASM instance. Connect as `SYSDBA` when connecting to the database instance.

For example:

```
$ asmcmd --privilege sysasm
```

Specifying the Verbose Mode

You can specify the `-v` option with the `asmcmd` command to display additional information with some commands to help users diagnose problems as shown in the following example:

```
$ asmcmd -v debug
```

When there is a message that must be logged during `asmcmd` operations, the message is written to the `alert.log` file or the `trace.trc` file.

The `alert.log` file is created or updated when `asmcmd` is started with the `-v errors`, `-v warning`, or `-v normal` display level. The `trace.trc` file is created or updated when `asmcmd` is started with the `-v info` or `-v debug` display level.

By default, the `alert.log` is file created in the `$ORACLE_HOME/log/diag/asmcmd/user_username/hostname/alert/` directory. By default, the `trace.trc` file is created in the `$ORACLE_HOME/log/diag/asmcmd/user_username/hostname/trace/` directory.

Under certain circumstances, `$ORACLE_BASE` and `$ORACLE_HOME` can be set to override the default locations of the `alert.log` and `trace.trc` files.

Including the Current Directory in the ASMCMD Prompt

You can specify the `-p` option with the `asmcmd` command to include the current directory in the ASMCMD prompt as shown in the following example:

```
$ asmcmd -p
ASMCMD [+] > cd data
ASMCMD [+data] >
```

Displaying the ASMCMD Version Number

You can specify the `-V` option when starting `asmcmd` to display the `asmcmd` version number. After displaying the version number, `asmcmd` immediately exits.

For example:

```
$ asmcmd -V
asmcmd version 11.2.0.3.0
$
```

Running ASMCMD in Noninteractive Mode

In noninteractive mode, you run a single ASMCMD command by including the command and command options on the command line when invoking ASMCMD. ASMCMD runs the command, generates output if any, and then exits. The noninteractive mode is especially useful for running scripts.

To run ASMCMD in noninteractive mode where `command` is any valid ASMCMD command and `options` is a list of command options, at the command prompt enter the following:

```
asmcmd command options
```

In noninteractive mode, ASMCMD returns the codes listed in Table 12–3.

Table 12–3   ASMCMD return codes

Type	Return Code	Description
General	0	No issues for the command.
General	1	Internal error.
General	255 or -1	External user error as returned by a command.
lsdsk	0	Results include member disks only.
lsdsk	1	Results include candidate disks only.
lsdsk	2	Results include both member and candidate disks.

Example 12–2 shows how to run ASMCMD in the noninteractive mode. The first example runs the `ls` command to list the disk groups for the Oracle ASM instance. The second example redirects the output of the `lsod` command to the `my_lsod_test` file. The third example runs `lsdsk` using a pattern with a wildcard character to list specific disks in the `data` disk group.

Example 12–2   Running ASMCMD commands in noninteractive mode

```
$ asmcmd ls -l
State    Type    Rebal  Name
MOUNTED  NORMAL  N      DATA/
MOUNTED  NORMAL  N      FRA/

$ asmcmd lsod --suppressheader -G data > my_lsod_test

$ asmcmd lsdsk -G data '/devices/diska*'
Path
/devices/diska1
/devices/diska2
/devices/diska3
```

Example 12–3 raises an error with an incorrect disk group name and the return code equal (255) is displayed.

Example 12–3   Displaying a return code when running ASMCMD in noninteractive mode

```
$ asmcmd ls -l dat
ASMCMD-08001: diskgroup 'dat' does not exist or is not mounted

$ echo $?
255
```

Example 12–4 shows an example of ASMCMD commands run in noninteractive mode inside a script.

Example 12–4   Running ASMCMD commands in a script

```
#!/bin/sh
for ((i = 1; i <=3; i++))
```

```
 do
    asmcmd lsdsk -G data '/devices/diska'$i
done

$ ./asmcmd_test_script
Path
/devices/diska1
Path
/devices/diska2
Path
/devices/diska3
```

Getting Help

Type `help` at the ASMCMD prompt or as a command in noninteractive mode to view general information about ASMCMD and a list of available ASMCMD commands.

You can type `help` *command* to display help text for a specific command, including usage information about how to run the command with its options.

The following is an example of the use of the `help` command.

Example 12–5   Displaying ASMCMD help text

```
ASMCMD [+] > help startup
        startup [--nomount] [--restrict] [--pfile <pfile.ora>]

        Start the ASM instance.

        [--nomount]           specifies the nomount option.
        [--restrict]          start the instance in restricted mode.
        [--pfile <pfile.ora>] specifies the location of the pfile.
```

ASMCMD Instance Management Commands

This section describes the ASMCMD Oracle ASM instance management commands.

Some commands in this section affect the Grid Plug and Play (GPnP) profile, which is a resource in a clustered configuration. In an Oracle Restart configuration, the profile is actually located in a resource attribute, not the Grid Plug and Play (GPnP) profile. However, the functionality of the commands is the same for both configurations.

Table 12–4 provides a summary of the Oracle ASM instance management commands.

Table 12–4   Summary of ASMCMD instance management commands

Command	Description
dsget	Retrieves the discovery diskstring value that is used by the Oracle ASM instance and its clients.
dsset	Sets the disk discovery diskstring value that is used by the Oracle ASM instance and its clients.
lsct	Lists information about current Oracle ASM clients.
lsop	Lists the current operations on a disk group or Oracle ASM instance.
lspwusr	Lists the users from an Oracle ASM password file.
orapwusr	Adds, drops, or changes an Oracle ASM password user.
shutdown	Shuts down an Oracle ASM instance.

Table 12–4   (Cont.) Summary of ASMCMD instance management commands

Command	Description
spbackup	Backs up an Oracle ASM SPFILE.
spcopy	Copies an Oracle ASM SPFILE.
spget	Retrieves the location of the Oracle ASM SPFILE.
spmove	Moves an Oracle ASM SPFILE.
spset	Sets the location of the Oracle ASM SPFILE.
startup	Starts up an Oracle ASM instance.

dsget

Purpose

Retrieves the discovery diskstring value that is used by the Oracle ASM instance and its clients.

Syntax and Description

```
dsget [ --normal | --parameter | --profile [-f] ]
```

The syntax options for the dsget command are described in Table 12–5.

Table 12–5   Options for the dsget command

Option	Description
--normal	Retrieves the discovery string from the Grid Plug and Play (GPnP) profile and the one that is set in the Oracle ASM instance. It returns one row each for the profile and parameter setting. This is the default setting.
--parameter	Retrieves the ASM_DISKSTRING parameter setting of the Oracle ASM instance.
--profile [-f]	Retrieves the discovery string from the GPnP profile. If -f is specified with --profile, dsget retrieves the discovery string from the local GPnP profile.

Example

The following example uses dsget to retrieve the current discovery diskstring value from the GPnP profile and the ASM_DISKSTRING parameter.

Example 12–6   Using the ASMCMD dsget command

```
ASMCMD [+] > dsget
profile: /devices/disk*
parameter: /devices/disk*
```

dsset

Purpose

Sets the discovery diskstring value that is used by the Oracle ASM instance and its clients.

The specified diskstring must be valid for existing mounted disk groups. The updated value takes effect immediately.

Syntax and Description

```
dsset [ --normal | --parameter | --profile [-f] ] diskstring
```

The syntax options for the dsset command are described in Table 12–6.

Table 12–6   Options for the dsset command

Option	Description
--normal	Sets the discovery string in the Grid Plug and Play (GPnP) profile and in the Oracle ASM instance. The update occurs after the Oracle ASM instance has successfully validated that the specified discovery string has discovered all the necessary disk groups and voting files. This command fails if the instance is not using a server parameter file (SPFILE).
	This is the default setting.
--parameter	Specifies that the diskstring is updated in memory after validating that the discovery diskstring discovers all the current mounted disk groups and voting files. The diskstring is not persistently recorded in either the SPFILE or the GPnP profile.
--profile [-f]	Specifies the discovery diskstring that is pushed to the GPnP profile without any validation by the Oracle ASM instance, ensuring that the instance can discover all the required disk groups. The update is guaranteed to be propagated to all the nodes that are part of the cluster.
	If -f is specified with --profile, the specified diskstring is pushed to the local GPnP profile without any synchronization with other nodes in the cluster. This command option updates only the local profile file. This option should only be used for recovery. The command fails if the Oracle Clusterware stack is running.
diskstring	Specifies the value for the discovery diskstring.

For information about disk discovery and the discovery diskstring, see "Oracle ASM Disk Discovery" on page 4-20.

Example

The following example uses dsset to set the current value of the discovery diskstring in the GPnP profile.

Example 12–7   Using the ASMCMD dsset command

```
ASMCMD [+] > dsset /devices/disk*
```

lsct

Purpose

Lists information about current Oracle ASM clients from the V$ASM_CLIENT view. A client, such as Oracle Database or Oracle ASM Dynamic Volume Manager (Oracle ADVM), uses disk groups that are managed by the Oracle ASM instance to which ASMCMD is currently connected.

Syntax and Description

```
lsct [--suppressheader] [-g] [diskgroup]
```

 Table 12–7 lists the options for the lsct command.

Table 12–7   Options for the lsct command

Option	Description
(none)	Displays information about current clients from the V$ASM_CLIENT view.
-g	Selects from the GV$ASM_CLIENT view. GV$ASM_CLIENT.INST_ID is included in the output.
--suppressheader	Suppresses column headings.
diskgroup	Specifies the disk group.

If *diskgroup* is specified, then only information about that disk group is listed.

Example

The following example displays information about the clients that are accessing the data disk group.

Example 12–8   Using the ASMCMD lsct command

```
ASMCMD [+] > lsct data
DB_Name  Status     Software_Version Compatible_version  Instance_Name  Disk_Group
+ASM     CONNECTED       11.2.0.2.0         11.2.0.0.0  +ASM           DATA
asmvol   CONNECTED       11.2.0.2.0         11.2.0.0.0  +ASM           DATA
orcl     CONNECTED       11.2.0.2.0         11.2.0.0.0  orcl           DATA
```

lsop

Purpose

Lists the current operations on a disk group in an Oracle ASM instance.

Syntax and Description

```
lsop
```

lsop displays information from the V$ASM_OPERATION view.

Example

The following are examples of the lsop command. The examples list operations on the disk groups of the current Oracle ASM instance.

Example 12–9   Using the ASMCMD lsop command

```
ASMCMD [+] > lsop
Group_Name  Dsk_Num  State  Power
DATA        REBAL    WAIT   2

ASMCMD [+] > lsop
Group_Name  Dsk_Num  State  Power
FRA         REBAL    REAP   3
```

lspwusr

Purpose

List the users from the local Oracle ASM password file.

Syntax and Description

lspwusr [--suppressheader]

Table 12–8 lists the options for the lspwusr command.

Table 12–8   Options for the lspwusr command

Option	Description
--suppressheader	Suppresses column headers from the output.

Examples

The following is an example of the lspwusr example. The example lists the current users in the local Oracle ASM password file.

Example 12–10   Using the ASMCMD lspwusr command

```
ASMCMD [+] > lspwusr
Username sysdba sysoper sysasm
     SYS    TRUE    TRUE    TRUE
 ASMSNMP    TRUE   FALSE   FALSE
```

orapwusr

Purpose

Add, drop, or modify an Oracle ASM password file user.

Syntax and Description

orapwusr { { {--add | --modify [--password] }
 [--privilege {sysasm|sysdba|sysoper}] } | --delete } user

Table 12–9 lists the options for the orapwusr command.

Table 12–9   Options for the orapwusr command

Option	Description
--add	Adds a user to the password file. Also prompts for a password.
--modify	Changes a user in the password file.
--password	Prompts for and then changes the password of a user.
--privilege *role*	Sets the role for the user. The options are sysasm, sysdba, and sysoper.
--delete	Drops a user from the password file.
user	Name of the user to add, drop, or modify.

orapwusr attempts to update passwords on all nodes in a cluster. The command requires the SYSASM privilege to run. A user logged in as SYSDBA cannot change its password using this command.

Examples

The following is an example of the orapwusr command. This example adds the hruser to the Oracle ASM password file with the role of the user set to SYSDBA.

Example 12–11   Using the ASMCMD orapwusr command

```
ASMCMD [+] > orapwusr --add --privilege sysdba hruser
```

shutdown

Purpose
Shuts down an Oracle ASM instance.

Syntax and Description
```
shutdown [--normal | --abort | --immediate ]
```

Table 12–10 lists the options for the shutdown command.

Table 12–10   Options for the shutdown command

Option	Description
--normal	Shut down normal.
--abort	Shut down aborting all existing operations.
--immediate	Shut down immediately.

The default action is a normal shutdown if an option is not specified.

Oracle strongly recommends that you shut down all database instances that use the Oracle ASM instance and dismount all file systems mounted on Oracle ASM Dynamic Volume Manager (Oracle ADVM) volumes before attempting to shut down the Oracle ASM instance with the abort (--abort) option.

For more information about shutting down an Oracle ASM instance, see "Shutting Down an Oracle ASM Instance" on page 3-16.

Example
The following are examples of the shutdown command. The first example performs a shut down of the Oracle ASM instance with normal action. The second example performs a shut down with immediate action. The third example performs a shut down that aborts all existing operations.

Example 12–12   Using the ASMCMD shutdown command

```
ASMCMD [+] > shutdown --normal

ASMCMD [+] > shutdown --immediate

ASMCMD [+] > shutdown --abort
```

spbackup

Purpose
Backs ups an Oracle ASM SPFILE to a backup file.

Syntax and Description
```
spbackup source destination
```

Table 12–11 lists the options for the spbackup command.

Table 12–11   Options for the spbackup command

Option	Description
source	Specifies the source file name.
destination	Specifies the destination file.

spbackup should be used when you want to make single or multiple backups of an SPFILE in the same or a different disk group without creating an SPFILE in the target disk group.

Note the following about the use of spbackup:

- spbackup can back up an Oracle ASM SPFILE from a disk group to a disk group or to an operating system file.

- spbackup can back up an Oracle ASM SPFILE from an operating system file to a disk group.

- spbackup can back up an Oracle ASM SPFILE when the SPFILE is being used by an open Oracle ASM instance.

- spbackup can make multiple backups of an Oracle ASM SPFILE in the same disk group.

spbackup does not affect the GPnP profile. The backup file that is created is not a special file type and is not identified as an SPFILE. This backup file cannot be copied with spcopy. To copy this backup file to and from a disk group, use the ASMCMD cp command.

To make a copy of a backup file in a disk group that is identified as an SPFILE file:

1. Use the ASMCMD cp command to copy the backup file from the disk group to an operating system file. See "cp" on page 12-21.

2. Use the ASMCMD spcopy command to copy the operating system file to a disk group. See "spcopy" on page 12-16.

Example

The following are examples of the spbackup command. The first example backs up the SPFILE in the data disk group. The second example backs up the SPFILE from the data disk group to the fra disk group.

Example 12–13   Using the ASMCMD spbackup command

```
ASMCMD> spbackup  +DATA/asm/asmparameterfile/registry.253.721810181
                  +DATA/spfileBackASM.bak

ASMCMD> spbackup  +DATA/asm/asmparameterfile/registry.253.721810181
                  +FRA/spfileBackASM.bak
```

spcopy

Purpose

Copies an Oracle ASM SPFILE from the source location to an SPFILE in the destination location.

Syntax and Description

spcopy [-u] *source destination*

Table 12–12 lists the options for the spcopy command.

Table 12–12   Options for the spcopy command

Option	Description
-u	Updates the Grid Plug and Play (GPnP) profile.
source	Specifies the source file name.
destination	Specifies the destination.

Note the following about the use of spcopy:

- spcopy can copy an Oracle ASM SPFILE from a disk group to a different disk group or to an operating system file.

- spcopy can copy an Oracle ASM SPFILE from an operating system file to a disk group.

- spcopy can copy an Oracle ASM SPFILE when the SPFILE is being used by an open Oracle ASM instance.

- spcopy cannot make multiple copies of an Oracle ASM SPFILE in the same disk group. You can use spbackup for that purpose.

To update the GPnP profile, include the -u option with spcopy. You can also use spset to update the GPnP profile if spcopy is run without the -u option. See "spset" on page 12-19. For information about copying and moving an Oracle ASM instance initialization parameter file after upgrading, see "Backing Up, Copying, and Moving an Oracle ASM Initialization Parameter File" on page 3-4.

After copying the SPFILE and updating the GPnP profile, you must restart the instance with the SPFILE in the new location to use that SPFILE. When the Oracle ASM instance is running with the SPFILE in the new location, you can remove the source SPFILE.

To copy an Oracle ASM SPFILE into a disk group using spcopy, the COMPATIBLE.ASM attribute must be set to 11.2 or greater in the target disk group.

> **See Also:**   The CREATE SPFILE SQL statement in the *Oracle Database SQL Language Reference* for information about creating a server parameter file

Example

The following are examples of the spcopy command. The first example copies the Oracle ASM SPFILE from the data disk group to the fra disk group. The second example copies the Oracle ASM SPFILE from the data disk group to an operating system location. The third example copies an Oracle ASM SPFILE from an operating system location to the data disk group and updates the GPnP profile with the -u option.

Example 12–14   Using the ASMCMD spcopy command

```
ASMCMD> spcopy  +DATA/asm/asmparameterfile/registry.253.721810181
                +FRA/spfileCopyASM.ora

ASMCMD> spcopy +DATA/asm/asmparameterfile/registry.253.721810181
                $ORACLE_HOME/dbs/spfileCopyASM.ora

ASMCMD> spcopy -u /u01/oracle/product/11.2.0/grid/dbs/spfileTestASM.ora
```

```
+DATA/ASM/spfileCopyASM.ora
```

spget

Purpose

Retrieves the location of the Oracle ASM SPFILE from the Grid Plug and Play (GPnP) profile.

Syntax and Description

```
spget
```

The location retrieved by spget is the location in the GPnP profile, but not always the location of the SPFILE currently used. For example, the location could have been recently updated by spset or spcopy with the -u option on an Oracle ASM instance that has not been restarted. After the next restart of the Oracle ASM, this location points to the Oracle ASM SPFILE currently being used.

Example

The following is an example of the spget command that retrieves and displays the location of the SPFILE from the GPnP profile.

Example 12–15   Using the ASMCMD spget command

```
ASMCMD [+] > spget
+DATA/asm/asmparameterfile/registry.253.691575633
```

spmove

Purpose

Moves an Oracle ASM SPFILE from source to destination and automatically updates the GPnP profile.

Syntax and Description

```
spmove source destination
```

 Table 12–13 lists the options for the spmove command.

Table 12–13   Options for the spmove command

Option	Description
source	Specifies the source file.
destination	Specifies the destination file.

Note the following about the use of spmove:

- spmove can move an Oracle ASM SPFILE when the open instance is using a PFILE or a different SPFILE. After moving the SPFILE, you must restart the instance with the SPFILE in the new location to use that SPFILE.

- spmove cannot move an Oracle ASM SPFILE when the SPFILE is being used by an open Oracle ASM instance.

For information about copying and moving an Oracle ASM instance initialization parameter file after upgrading, see "Backing Up, Copying, and Moving an Oracle ASM Initialization Parameter File" on page 3-4.

To use spmove to move an Oracle ASM SPFILE into a disk group, the disk group attribute COMPATIBLE.ASM must be set to 11.2 or greater.

Example

The following are examples of the spmove command. The first example moves an Oracle ASM SPFILE from the data disk group to an operating system location. The second example moves an SPFILE from an operating system location to the data disk group.

Example 12–16   Using the ASMCMD spmove command

```
ASMCMD> spmove +DATA/spfileASM.ora
               /u01/oracle/product/11.2.0/grid/dbs/spfileMoveASM.ora

ASMCMD> spmove /u01/oracle/product/11.2.0/grid/dbs/spfile+ASM.ora
               +DATA/ASM/spfileMoveASM.ora
```

spset

Purpose

Sets the location of the Oracle ASM SPFILE in the Grid Plug and Play (GPnP) profile.

Syntax and Description

spset *location*

Table 12–14 lists the options for the spset command.

Table 12–14   Options for the spset command

Option	Description
location	Specifies the location of the Oracle ASM SPFILE. The location is the full path to the SPFILE.

Example

The following is an example of the spset command that sets the location of the Oracle ASM SPFILE command in the data disk group.

Example 12–17   Using the ASMCMD spset command

```
ASMCMD> spset +DATA/asm/asmparameterfile/asmspfile.ora
```

startup

Purpose

Starts up an Oracle ASM instance.

Syntax and Description

startup [--nomount] [--restrict] [--pfile *pfile*]

Table 12–15 lists the options for the startup command.

Table 12–15   Options for the startup command

Option	Description
--nomount	Specifies no mount operation.
--restrict	Specifies restricted mode.
--pfile *pfile*	Oracle ASM initialization parameter file.

The default action is a startup that mounts disk groups and enables Oracle ASM Dynamic Volume Manager (Oracle ADVM) volumes. For information about disk groups that are mounted at startup time, see "About Mounting Disk Groups at Startup" on page 3-16.

For more information about starting up an Oracle ASM instance, see "Starting Up an Oracle ASM Instance" on page 3-13.

Example

The following is an example of the startup command that starts the Oracle ASM instance without mounting disk groups and uses the asm_init.ora initialization parameter file.

Example 12–18   Using the ASMCMD startup command

```
ASMCMD> startup --nomount --pfile asm_init.ora
```

ASMCMD File Management Commands

This section describes the ASMCMD file management commands.

Table 12–16 provides a summary of the file management commands.

Table 12–16   Summary of ASMCMD file management commands

Command	Description
cd	Changes the current directory to the specified directory.
cp	Enables you to copy files between disk groups, and between a disk group and the operating system.
du	Displays the total disk space occupied by files in the specified Oracle ASM directory and all of its subdirectories, recursively.
find	Lists the paths of all occurrences of the specified name (with wildcards) under the specified directory.
ls	Lists the contents of an Oracle ASM directory, the attributes of the specified file, or the names and attributes of all disk groups.
lsof	Lists the open files.
mkalias	Creates an alias for system-generated filenames.
mkdir	Creates Oracle ASM directories.
pwd	Displays the path of the current Oracle ASM directory.
rm	Deletes the specified Oracle ASM files or directories.
rmalias	Deletes the specified alias, retaining the file that the alias points to.

cd

Purpose
Changes the current directory to the specified directory.

Syntax and Description
cd [*dir*]

Table 12–17 lists the options for the cd command.

Table 12–17   Options for the cd command

Option	Description
dir	Name of the directory.

dir can be specified as either an absolute path or a relative path, including the . and .. pseudo-directories. *dir* can contain wildcard characters. See "Wildcard Characters" on page 12-5.

Examples
The following are examples of the cd command changing into various directories.

Example 12–19   Using the ASMCMD cd command

```
ASMCMD [+data/hr] > cd +data/orcl

ASMCMD [+data/orcl] > cd DATAFILE

ASMCMD [+data/orcl/DATAFILE] >cd ..
```

cp

Purpose
Enables you to copy files between Oracle ASM disk groups and between a disk group and the operating system.

Syntax and Description
cp *src_file* [*rem_connect_str*:]*tgt_file*
cp [*rem_connect_str*:]*src_file* *tgt_file*

Table 12–18 lists the syntax options for the cp command.

Table 12–18   Options for the cp command

Option	Description
rem_connect_str	Specifies the connection string for a remote instance.
src_file	Name of the source file to copy.
tgt_file	A user alias for the created target file name or an alias directory name.

cp cannot copy files between two remote instances. The local Oracle ASM instance must be either the source or the target of the operation.

You can use the `cp` command to:

- Copy files from a disk group to the operating system
- Copy files from a disk group to a disk group
- Copy files from the operating system to a disk group

Some file types cannot be the source or destination of the `cp` command. These file types include OCR and OCR backup file types. To back up, copy, or move an Oracle ASM SPFILE, use the `spbackup`, `spcopy`, or `spmove` commands.

rem_connect_str is not required for a local instance copy, which is the default case. For a remote instance copy, you must specify the connect string and Oracle ASM prompts for a password in a non-echoing prompt. The *rem_connect_str* is in the form:

user@host[*.port_number*]*.SID*

user, *host*, and *SID* are required in the *rem_connect_str* parameter. The default port number is `1521`.

The connection privilege (`SYSASM` or `SYSDBA`) is determined by the use of the `--privilege` option when starting ASMCMD. For more information, refer to "Specifying the Type of Connection" on page 12-8.

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

src_file must be either a fully qualified file name or an Oracle ASM alias.

The `cp` command performs a bit-wise copy. There is no data transformation.

Examples
The following are examples of the `cp` command. The first example shows a copy of a file in the `data` disk group to a file on the operating system. The second example shows a copy of a file on the operating system to the `data` disk group. The third example shows how to copy a file in the `data` disk group to a remote server. If the password is not included in the command line, you are prompted to enter the password.

Example 12–20   Using the ASMCMD cp command

```
ASMCMD [+] > cp +data/orcl/datafile/EXAMPLE.265.691577295 /mybackups/example.bak
copying +data/orcl/datafile/EXAMPLE.265.691577295 -> /mybackups/example.bak

ASMCMD [+] >  cp /mybackups/examples.bak  +data/orcl/datafile/myexamples.bak
copying /mybackups/examples.bak -> +data/orcl/datafile/myexamples.bak

ASMCMD [+] > cp +data/orcl/datafile/EXAMPLE.265.691577295
sys@myserver.+ASM:/scratch/backup/myexamples.bak
Enter Password:
```

du

Purpose
Displays the total space used for files in the specified directory and in the entire directory tree under the directory.

Syntax and Description

du [--suppressheader] [*dir*]

Table 12–19 lists the syntax options for the du command.

Table 12–19   Options for the du command

Option	Description
dir	Name of the directory.
--suppressheader	Suppresses column headings from the output.

If you do not specify *dir*, then information about the current directory is displayed. *dir* can contain wildcard characters. See "Wildcard Characters" on page 12-5.

The following two values are displayed, both in units of megabytes.

- Used_MB - This value does not include mirroring.

- Mirror_used_MB - This value includes mirroring.

For example, if a normal redundancy disk group contains 100 MB of data and each file in the disk group is 2-way mirrored, then Used_MB is 100 MB and Mirror_used_MB is roughly 200 MB.

Example

The following is an example of the du command. The example shows disk space used in the orcl directory in the data disk group, including all of the directories under the orcl directory.

Example 12–21   Using the ASMCMD du command

```
ASMCMD [+] > du data/orcl
Used_MB      Mirror_used_MB
   1756                3519
```

find

Purpose

Displays the absolute paths of all occurrences of the specified name pattern (with wildcards) in a specified directory and its subdirectories.

Syntax and Description

find [--type *type*] *dir pattern*

Table 12–20 lists the syntax options for the find command.

Table 12–20   Options for the find command

Option	Description
--type *type*	Type of target to find.
dir	Directory name where you want to start searching.
pattern	Name of a target or a wildcard pattern.

This command searches the specified directory and all subdirectories under it in the directory tree for the supplied *pattern*. The value used for *pattern* can be a directory

name or a filename, and can include wildcard characters. See "Wildcard Characters" on page 12-5.

The ASMCMD `find` command is case insensitive.

In the output of the command, directory names are suffixed with the slash character (/) to distinguish them from filenames.

Use the `--type` flag to find all the files of a particular type (specified as *type*). For example, you can search for control files by specifying *type* as `CONTROLFILE`. Valid values for *type* are listed in Table 7–1, " File types supported by Oracle ASM" on page 7-1. These are type values from the `type` column of the `V$ASM_FILE` view.

Examples

The following are examples of the `find` command. The first example searches the `data` disk group for files that begin with `undo`. The second example returns the absolute path of all the control files (`--type CONTROLFILE`) in the `+data/orcl` directory.

Example 12–22   Using the ASMCMD find command

```
ASMCMD [+] > find +data undo*
+data/ORCL/DATAFILE/UNDOTBS1.258.691577151

ASMCMD [+] >  find --type CONTROLFILE +data/orcl *
+data/orcl/CONTROLFILE/Current.260.691577263
```

ls

Purpose
Lists the contents of an Oracle ASM directory, the attributes of the specified file, or the names and attributes of all disk groups.

Syntax and Description

```
ls [--suppressheader] [-lsdtLg][--absolutepath][--reverse]
   [--permission][pattern]
```

Table 12–21 lists the syntax options for the `ls` command.

Table 12–21   Options for the ls command

Option	Description
(none)	Displays only filenames and directory names.
-l	Displays extended file information, including striping and redundancy information and whether the file was system-generated (indicated by `Y` under the `SYS` column) or user-created (as with an alias, indicated by `N` under the `SYS` column). When used in the "`ls -l +`" command, displays directory information.
	Not all possible file attributes or disk group attributes are included. To view the complete set of column values for a file or a disk group, query the `V$ASM_FILE` and `V$ASM_DISKGROUP` views.
-s	Displays file space information.
-d	If the value for the *pattern* argument is a directory, then `ls` displays information about that directory, rather than the directory contents. Typically used with another flag, such as the `-l` flag.

Table 12–21   (Cont.) Options for the ls command

Option	Description
--reverse	Reverses the sort order of the listing.
-t	Sorts the listing by timestamp (latest first) instead of by name.
-L	If the value for the *pattern* argument is an alias, then ASMCMD displays information about the file that it references. Typically used with another flag, such as the -l flag.
--absolutepath	For each listed file, displays the absolute path of the alias that references it, if any.
-g	GV$ASM_DISKGROUP.INST_ID is included in the output.
--suppressheader	Suppresses column headings.
--permission	Shows the permissions of a file (V$ASM_FILE.permission, V$ASM_FILE.owner, V$ASM_FILE.usergroup, V$ASM_ALIAS.name).
pattern	Name of a file, directory, or a pattern.

Command options enable you to modify and customize the output of the command. Table 12–21 lists the options and their descriptions. For disk group information, this command queries the V$ASM_DISKGROUP_STAT view by default. If you specify all of the options, then the command shows a union of their attributes, with duplicates removed. If you enter ls +, the top level directory structure is displayed.

pattern can be a file name, directory name, or a pattern including wildcard characters. See "Wildcard Characters" on page 12-5.

- If *pattern* is a directory name, then ls lists the contents of the directory and depending on flag settings, ls also lists information about each directory member. Directories are listed with a trailing slash (/) to distinguish them from files.

- If the value that you enter for *pattern* is a file name, then ls lists the file and depending on the flag settings, ls also lists information about the file. The file must be located in the current directory if the file name is specified with a relative path.

Examples

The following are examples of the ls command that display various information about directories and the contents of the directories.

Example 12–23   Using the ASMCMD ls command

```
ASMCMD [+] > ls +data/orcl/datafile
EXAMPLE.265.691577295
SYSAUX.257.691577149
SYSTEM.256.691577149
UNDOTBS1.258.691577151
USERS.259.691577151

ASMCMD [+] > ls -lt +data/orcl/datafile
Type      Redund  Striped  Time            Sys  Name
DATAFILE  MIRROR  COARSE   JUL 13 08:00:00  Y    EXAMPLE.265.691577295
DATAFILE  MIRROR  COARSE   JUL 13 05:00:00  Y    SYSAUX.257.691577149
DATAFILE  MIRROR  COARSE   JUL 13 02:00:00  Y    USERS.259.691577151
DATAFILE  MIRROR  COARSE   JUL 13 02:00:00  Y    UNDOTBS1.258.691577151
DATAFILE  MIRROR  COARSE   JUL 13 02:00:00  Y    SYSTEM.256.691577149

ASMCMD [+] > ls -l +data/orcl/datafile/sy*
```

```
Type      Redund  Striped  Time              Sys  Name
DATAFILE  MIRROR  COARSE   JUL 13 05:00:00   Y    SYSAUX.257.691577149
DATAFILE  MIRROR  COARSE   JUL 13 02:00:00   Y    SYSTEM.256.691577149

ASMCMD [+] > ls -s  +data/orcl/datafile
Block_Size  Blocks      Bytes         Space  Name
      8192   12801  104865792     214958080  EXAMPLE.265.691577295
      8192   88321  723525632    1452277760  SYSAUX.257.691577149
      8192   88321  723525632    1452277760  SYSTEM.256.691577149
      8192    7681   62922752     131072000  UNDOTBS1.258.691577151
      8192     641    5251072      12582912  USERS.259.691577151

ASMCMD [+] > ls --permission  +data/orcl/datafile
User  Group  Permission  Name
               rw-rw-rw-  EXAMPLE.265.691577295
               rw-rw-rw-  SYSAUX.257.691577149
               rw-rw-rw-  SYSTEM.256.691577149
               rw-rw-rw-  UNDOTBS1.258.691577151
               rw-rw-rw-  USERS.259.691577151
```

lsof

Purpose
Lists the open files of the local clients.

Syntax and Description
lsof [--suppressheader] {-G *diskgroup*|--dbname *db*| -C *instance*}

Table 12–22 lists the syntax options for the lsof command.

Table 12–22   Options for the lsof command

Option	Description
--suppressheader	Suppresses column headings.
-G *diskgroup*	List files only from the specified disk group.
--dbname *db*	List files only from the specified database.
-C *instance*	List files only from the specified instance.

Example
The following are examples of the lsof command. The first example lists the open files
for the data disk group. The second example lists the open files for the Oracle ASM
instance.

Example 12–24   Using the ASMCMD lsof command

```
ASMCMD [+] > lsof -G data
DB_Name  Instance_Name  Path
orcl     orcl           +data/orcl/controlfile/current.260.691577263
orcl     orcl           +data/orcl/datafile/example.265.691577295
orcl     orcl           +data/orcl/datafile/sysaux.257.691577149
orcl     orcl           +data/orcl/datafile/system.256.691577149
orcl     orcl           +data/orcl/datafile/undotbs1.258.691577151
orcl     orcl           +data/orcl/datafile/users.259.691577151
orcl     orcl           +data/orcl/onlinelog/group_1.261.691577267
orcl     orcl           +data/orcl/onlinelog/group_2.262.691577271
```

```
orcl    orcl          +data/orcl/onlinelog/group_3.263.691577275
orcl    orcl          +data/orcl/tempfile/temp.264.691577287

ASMCMD [+] > lsof -C +ASM
DB_Name  Instance_Name  Path
asmvol   +ASM           +data/VOLUME1.271.679226013
asmvol   +ASM           +data/VOLUME2.272.679227351
```

mkalias

Purpose

Creates an alias for the specified system-generated filename.

Syntax and Description

mkalias *file alias*

Table 12–23 lists the syntax options for the mkalias command.

Table 12–23   Options for the mkalias command

Option	Description
file	System-generated file name.
alias	Alias for the file name.

alias must be in the same disk group as the system-generated file. Only one alias is permitted for each Oracle ASM file.

Example

The following example creates the sysaux.f alias for the fully qualified filename +data/orcl/DATAFILE/SYSAUX.257.721811945. Following the mkalias command, ls --absolutepath is run to check the results.

Example 12–25   Using the ASMCMD mkalias command

```
ASMCMD [+data/orcl/datafile] > mkalias SYSAUX.257.721811945 sysaux.f

ASMCMD [+data/orcl/datafile] > ls --absolutepath
none => EXAMPLE.265.721812093
+DATA/ORCL/DATAFILE/sysaux.f => SYSAUX.257.721811945
none => SYSTEM.256.721811945
none => UNDOTBS1.258.721811945
none => USERS.259.721811947
sysaux.f
```

mkdir

Purpose

Creates Oracle ASM directories under the current directory.

Syntax and Description

mkdir *dir* [*dir...*]

Table 12–24 lists the syntax options for the mkdir command.

Table 12–24   Options for the mkdir command

Option	Description
dir	Directory name to create.

The directory from which mkdir is run can be a system-created or user-created directory. You cannot create a directory at the root (+) level.

Example

The following is an example of the mkdir command. The example creates the directories subdir1 and subdir2 at the disk group level in the disk group data.

Example 12–26   Using the ASMCMD mkdir command

```
ASMCMD [+data] > mkdir subdir1 subdir2

ASMCMD [+data] > ls
ASM/
ORCL/
subdir1/
subdir2/
```

pwd

Purpose

Displays the absolute path of the current directory.

Syntax and Description

pwd

Example

The following is an example of the pwd command. The example displays the current directory.

Example 12–27   Using the ASMCMD pwd command

```
ASMCMD [+data/orcl/datafile] > pwd
+data/orcl/datafile
```

rm

Purpose

Deletes the specified Oracle ASM files and directories.

Syntax and Description

rm [-f│-r] *pattern* [*pattern*...]

Table 12–25 lists the syntax options for the rm command.

Table 12–25   Options for the rm command

Option	Description
-r	Recursively deletes files and subdirectories.

Table 12–25   (Cont.) Options for the rm command

Option	Description
-f	Forces the deletion of files and subdirectories.
pattern	Name of a file, directory, or wildcard pattern.

If *pattern* is a file or alias, then the `rm` command can delete the file or alias only if it is not currently in use. If *pattern* is a directory, then the `rm` command can delete it only if it is empty (unless the `-r` flag is used) and it is not a system-generated directory. If *pattern* is an alias, then the `rm` command deletes both the alias and the file to which the alias refers. To delete only an alias and retain the file that the alias references, use the `rmalias` command.

> **Note:**   When you delete all of the files in a system-created directory, the directory is removed. If the parent directories are empty, all of the parent directories are also removed.

pattern can contain wildcard characters. See "Wildcard Characters" on page 12-5.

If you use a wildcard, the `rm` command deletes all of the matches except nonempty directories, unless you use the `-r` flag. To recursively delete, use the `-r` flag. With `-r` option you can delete a nonempty directory, including all files and directories in it and in the entire directory tree underneath it. If you use the `-r` flag or a wildcard character, then the `rm` command prompts you to confirm the deletion before proceeding, unless you specify the `-f` flag.

If a wildcard character matches an alias or a system-generated file that has an alias, then both the alias and the system-generated file that it references are deleted. When using the `-r` flag to delete an alias that matches a wildcard pattern, either the alias or the system-generated file that has an alias must be present in the directory in which you run the `rm` command.

For example, if you have a user alias, `+data/dir1/file.alias` that points to `+data/orcl/DATAFILE/System.256.146589651`, then running the `rm -r +data/dir1` command removes the `+data/dir1/file.alias` and `+data/orcl/DATAFILE/System.256.146589651`.

Example

The following are examples of the `rm` command. The first example deletes the `myexamples.bak` file. The second example removes the `subdir2` directory and its contents.

Example 12–28   Using the ASMCMD rm command

```
ASMCMD [+data/orcl/datafile] > rm myexamples.bak

ASMCMD [+data] > rm -r subdir2
You may delete multiple files and/or directories.
Are you sure? (y/n) y
```

rmalias

Purpose
Removes the specified aliases, retaining the files that the aliases reference.

Syntax and Description
```
rmalias [-r] alias [alias...]
```

Table 12–26 lists the syntax options for the rmalias command.

Table 12–26   Options for the rmalias command

Option	Description
-r	Recursively removes aliases.
alias	Alias for the file name or directory.

The -r flag enables you to remove all of the aliases in the current directory and in the entire directory tree beneath the current directory. If any user-created directories become empty after deleting aliases, they are also deleted. Files and directories created by the system are not deleted.

Example
The following is an example of the rmalias command. The example deletes the alias sysaux.f, retaining the data file that it references.

Example 12–29   Using the ASMCMD rmalias command
```
ASMCMD [+data/orcl/datafile] > rmalias sysaux.f
```

ASMCMD Disk Group Management Commands

This section describes the ASMCMD disk group management commands.

Table 12–27 provides a summary of the disk group management commands.

Table 12–27   Summary of ASMCMD Disk group management commands

Command	Description
chdg	Changes a disk group (add, drop, or rebalance).
chkdg	Checks or repairs a disk group.
dropdg	Drops a disk group.
iostat	Displays I/O statistics for disks.
lsattr	Lists the attributes of a disk group.
lsdg	Lists disk groups and their information.
lsdsk	Lists disks Oracle ASM disks.
lsod	Lists open devices.
md_backup	Creates a backup of the metadata of mounted disk groups.
md_restore	Restores disk groups from a backup of the metadata.
mkdg	Creates a disk group.
mount	Mounts a disk group.

Table 12–27 (Cont.) Summary of ASMCMD Disk group management commands

Command	Description
offline	Offlines a disk or a failure group.
online	Onlines a disk or a failure group.
rebal	Rebalances a disk group.
remap	Relocates data in a range of physical blocks on a disk.
setattr	Sets attributes in a disk group.
umount	Dismounts a disk group.

chdg

Purpose

Changes a disk group (adds disks, drops disks, resizes disks, or rebalances a disk group) based on an XML configuration file.

Syntax and Description

chdg { *config_file*.xml | '*contents_of_xml_file*' }

Table 12–28 lists the syntax options for the chdg command.

Table 12–28 Options for the chdg command

Option	Description
config_file	Name of the XML file that contains the changes for the disk group. chdg searches for the XML file in the directory where ASMCMD was started unless a path is specified. For examples of the valid tags and XML configuration file, see Example 12–30 and Example 12–31.
contents_of_xml_file	The XML script enclosed in single quotations.

chdg modifies a disk group based on an XML configuration file. The modification includes adding or deleting disks from an existing disk group, and the setting rebalance power level. The power level can be set to the same values as the ASM_POWER_LIMIT initialization parameter. For information about the initialization parameter, see "ASM_POWER_LIMIT" on page 3-8.

When adding disks to a disk group, the diskstring must be specified in a format similar to the ASM_DISKSTRING initialization parameter. For information about the initialization parameter, see "ASM_DISKSTRING" on page 3-7.

The failure groups are optional parameters. The default causes every disk to belong to a its own failure group. For information about failure groups, see"Oracle ASM Failure Groups" on page 4-26.

Dropping disks from a disk group can be performed through this operation. An individual disk can be referenced by its Oracle ASM disk name. A set of disks that belong to a failure group can be specified by the failure group name. For information about dropping disks, see"Dropping Disks from Disk Groups" on page 4-15.

You can resize a disk inside a disk group with chdg. The resize operation fails if there is not enough space for storing data after the resize. For information about resizing disks, see"Resizing Disks in Disk Groups" on page 4-17.

Example 12–30 shows the basic structure and the valid tags with their respective attributes for the chdg XML configuration file.

Example 12–30   Tags for the chdg XML configuration template

```
<chdg> update disk clause (add/delete disks/failure groups)
     name         disk group to change
     power        power to perform rebalance

<add>  items to add are placed here
</add>

<drop> items to drop are placed here
</drop>

<fg>  failure group
     name         failure group name
</fg>

<dsk> disk
     name         disk name
     string       disk path
     size         size of the disk to add
     force        true specifies to use the force option
</dsk>

</chdg>
```

For information about creating a disk group with ASMCMD mkdg, see "mkdg" on page 12-43. For information about altering disk groups, see "Altering Disk Groups" on page 4-12.

Example

The following is an example of an XML configuration file for chdg. This XML file alters the disk group named data. The failure group fg1 is dropped and the disk data_0001 is also dropped. The /dev/disk5 disk is added to failure group fg2. The rebalance power level is set to 3.

Example 12–31   chdg sample XML configuration file

```
<chdg name="data" power="3">
  <drop>
    <fg name="fg1"></fg>
    <dsk name="data_0001"/>
  </drop>
  <add>
    <fg name="fg2">
      <dsk string="/dev/disk5"/>
    </fg>
  </add>
</chdg>
```

The following are examples of the chdg command with the configuration file or configuration information on the command line.

Example 12–32   Using the ASMCMD chdg command

```
ASMCMD [+] > chdg data_config.xml
```

```
ASMCMD [+] > chdg '<chdg name="data" power="3">
        <drop><fg name="fg1"></fg><dsk name="data_0001"/></drop>
        <add><fg name="fg2"><dsk string="/dev/disk5"/></fg></add></chdg>'
```

chkdg

Purpose

Checks or repairs the metadata of a disk group.

Syntax and Description

chkdg [--repair] *diskgroup*

Table 12–29 lists the syntax options for the chkdg command.

Table 12–29   Options for the chkdg command

Option	Description
--repair	Repairs the disk group.
diskgroup	Name of disk group to check or repair.

chkdg checks the metadata of a disk group for errors and optionally repairs the errors.

Example

The following is an example of the chkdg command used to check and repair the data disk group.

Example 12–33   Using the ASMCMD chkdg command

```
ASMCMD [+] > chkdg --repair data
```

dropdg

Purpose

Drops a disk group.

Syntax and Description

dropdg [-r -f] [-r] *diskgroup*

Table 12–30 lists the syntax options for the dropdg command.

Table 12–30   Options for the dropdg command

Option	Description
-f	Force the operation. Only applicable if the disk group cannot be mounted.
-r	Recursive, include contents.
diskgroup	Name of disk group to drop.

dropdg drops an existing disk group. The disk group should not be mounted on multiple nodes.

Example

These are examples of the use of dropdg. The first example forces the drop of the disk group data, including any data in the disk group. The second example drops the disk group fra, including any data in the disk group.

Example 12–34   Using the ASMCMD dropdg command

```
ASMCMD [+] > dropdg -r -f data

ASMCMD [+] > dropdg -r fra
```

iostat

Purpose

Displays I/O statistics for Oracle ASM disks in mounted disk groups.

Syntax and Description

```
iostat [--suppressheader] [-et] [--io] [--region]
       [-G diskgroup] [interval]
```

iostat lists disk group statistics using the V$ASM_DISK_IOSTAT view.

Table 12–31 lists the syntax options for the iostat command.

Table 12–31   Options for the iostat command

Option	Description
-e	Displays error statistics (Read_Err, Write_Err).
-G diskgroup	Displays statistics for the disk group name.
--suppressheader	Suppresses column headings.
--io	Displays information in number of I/Os, instead of bytes.
-t	Displays time statistics (Read_Time, Write_Time).
--region	Displays information for cold and hot disk regions (Cold_Reads, Cold_Writes, Hot_Reads, Hot_Writes).
interval	Refreshes the statistics display based on the interval value (seconds). Use Ctrl-C to stop the interval display.

Table 12–32 shows the statistics for a disk group. To view the complete set of statistics for a disk group, use the V$ASM_DISK_IOSTAT view.

Table 12–32   Attribute descriptions for iostat command output

Attribute Name	Description
Group_Name	Name of the disk group.
Dsk_Name	Name of the disk.
Reads	Number of bytes read from the disk. If the --io option is entered, then the value is displayed as number of I/Os.
Writes	Number of bytes written to the disk. If the --io option is entered, then the value is displayed as number of I/Os.
Cold_Reads	Number of bytes read from the cold disk region. If the --io option is entered, then the value is displayed as number of I/Os.

Table 12–32   (Cont.) Attribute descriptions for iostat command output

Attribute Name	Description
Cold_Writes	Number of bytes written from the cold disk region. If the `--io` option is entered, then the value is displayed as number of I/Os.
Hot_Reads	Number of bytes read from the hot disk region. If the `--io` option is entered, then the value is displayed as number of I/Os.
Hot_Writes	Number of bytes written to the hot disk region. If the `--io` option is entered, then the value is displayed as number of I/Os.
Read_Err	Number of failed I/O read requests for the disk.
Write_Err	Number of failed I/O write requests for the disk.
Read_Time	I/O time (in hundredths of a second) for read requests for the disk if the `TIMED_STATISTICS` initialization parameter is set to `TRUE` (0 if set to `FALSE`).
Write_Time	I/O time (in hundredths of a second) for write requests for the disk if the `TIMED_STATISTICS` initialization parameter is set to `TRUE` (0 if set to `FALSE`).

If a refresh interval is not specified, the number displayed represents the total number of bytes or I/Os. If a refresh interval is specified, then the value displayed (bytes or I/Os) is the difference between the previous and current values, not the total value.

Example

The following are examples of the `iostat` command. The first example displays disk I/O statistics for the `data` disk group in total number of bytes. The second example displays disk I/O statistics for the `data` disk group in total number of I/O operations.

Example 12–35   Using the ASMCMD iostat command

```
ASMCMD [+] > iostat -G data
Group_Name  Dsk_Name   Reads       Writes
DATA        DATA_0000  180488192   473707520
DATA        DATA_0001  1089585152  469538816
DATA        DATA_0002  191648256   489570304
DATA        DATA_0003  175724032   424845824
DATA        DATA_0004  183421952   781429248
DATA        DATA_0005  1102540800  855269888
DATA        DATA_0006  171290624   447662592
DATA        DATA_0007  172281856   361337344
DATA        DATA_0008  173225472   390840320
DATA        DATA_0009  288497152   838680576
DATA        DATA_0010  196657152   375764480
DATA        DATA_0011  436420096   356003840

ASMCMD [+] > iostat --io -G data
Group_Name  Dsk_Name   Reads  Writes
DATA        DATA_0000  2801   34918
DATA        DATA_0001  58301  35700
DATA        DATA_0002  3320   36345
DATA        DATA_0003  2816   10629
DATA        DATA_0004  2883   34850
DATA        DATA_0005  59306  38097
DATA        DATA_0006  2151   10129
DATA        DATA_0007  2686   10376
DATA        DATA_0008  2105   8955
DATA        DATA_0009  9121   36713
```

```
DATA          DATA_0010  3557   8596
DATA          DATA_0011  17458  9269
```

lsattr

Purpose
Lists the attributes of a disk group.

Syntax and Description
```
lsattr [--suppressheader][-G diskgroup] [-lm] [pattern]
```

Table 12–33 lists the syntax options for the lsattr command.

Table 12–33   Options for the lsattr command

Option	Description
-G diskgroup	Disk group name.
--suppressheader	Suppresses column headings.
-l	Display names with values.
-m	Displays additional information, such as the RO and Sys columns.
pattern	Display the attributes that contain pattern expression.

Information about disk group attributes is retrieved from the V$ASM_ATTRIBUTE view. For information about disk group attributes, see "Disk Group Attributes" on page 4-2.

The RO (read-only) column identifies those attributes that can only be set when a disk group is created. The Sys column identifies those attributes that are system-created.

To display information about the disk group template attributes, see "lstmpl" on page 12-51.

To set disk group attributes, see "setattr" on page 12-49.

Example
The following are examples of the lsattr command. The first displays information about all attributes for the data disk group. The second example displays only those attributes with names containing the string compat for the fra disk group. Note the use of both the % and * wildcard characters on Linux.

Example 12–36   Using the ASMCMD lsattr command

```
ASMCMD [+] > lsattr -G data -l
Name                     Value
access_control.enabled   FALSE
access_control.umask     066
au_size                  1048576
cell.smart_scan_capable  FALSE
compatible.advm          11.2.0.0.0
compatible.asm           11.2.0.0.0
compatible.rdbms         11.2.0.0.0
disk_repair_time         3.6h
sector_size              512

ASMCMD [+] > lsattr -G fra -l %compat*
```

```
Name             Value
compatible.asm   11.2.0.0.0
compatible.rdbms 10.1.0.0.0
```

lsdg

Purpose

Lists mounted disk groups and their information. lsdg queries V$ASM_DISKGROUP_STAT by default. If the --discovery flag is specified, the V$ASM_DISKGROUP is queried instead. The output also includes notification of any current rebalance operation for a disk group. If a disk group is specified, then lsdg returns only information about that disk group.

Syntax and Description

lsdg [--suppressheader] [-g] [--discovery] [*pattern*]

Table 12–34 lists the options for the lsdg command.

Table 12–34 Options for the lsdg command

Option	Description
(none)	Displays the disk group attributes listed in Table 12–35.
--discovery	Selects from V$ASM_DISKGROUP, or from GV$ASM_DISKGROUP if the -g flag is also specified. This option is always enabled if the Oracle ASM instance is version 10.1 or earlier. This flag is disregarded if lsdg is running in non-connected mode.
-g	Selects from GV$ASM_DISKGROUP_STAT, or from GV$ASM_DISKGROUP if the --discovery flag is also specified. GV$ASM_DISKGROUP.INST_ID is included in the output. The REBAL column of the GV$ASM_OPERATION view is also included in the output.
--suppressheader	Suppresses column headings.
pattern	Returns only information about the specified disk group or disk groups that match the supplied pattern. See "Wildcard Characters" on page 12-5.

Table 12–35 shows the attributes for each disk group. To view the complete set of attributes for a disk group, use the V$ASM_DISKGROUP_STAT or V$ASM_DISKGROUP view.

> **See Also:** *Oracle Database Reference* for descriptions of disk group information displayed in the V$ASM_DISKGROUP view

Table 12–35 Attribute descriptions for lsdg command output

Attribute Name	Description
State	State of the disk group. Values include BROKEN, CONNECTED, DISMOUNTED, MOUNTED, QUIESCING, and UNKNOWN.
Type	Disk group redundancy (NORMAL, HIGH, EXTERNAL).
Rebal	Y if a rebalance operation is in progress.
Sector	Sector size in bytes.
Block	Block size in bytes.
AU	Allocation unit size in bytes.

Table 12–35   (Cont.) Attribute descriptions for lsdg command output

Attribute Name	Description
Total_MB	Size of the disk group in megabytes.
Free_MB	Free space in the disk group in megabytes, without regard to redundancy. From the V$ASM_DISKGROUP view.
Req_mir_free_MB	Amount of space that must be available in the disk group to restore full redundancy after the most severe failure that can be tolerated by the disk group. This is the REQUIRED_MIRROR_FREE_MB column from the V$ASM_DISKGROUP view.
Usable_file_MB	Amount of free space, adjusted for mirroring, that is available for new files. From the V$ASM_DISKGROUP view.
Offline_disks	Number of offline disks in the disk group. Offline disks are eventually dropped.
Voting_files	Specifies whether the disk group contains voting files (Y or N).
Name	Disk group name.

Example
The following example lists the attributes of the data disk group.

Example 12–37   Using the ASMCMD lsdg command

```
ASMCMD [+] > lsdg data
State     Type    Rebal  Sector  Block        AU  Total_MB  Free_MB  Req_mir_free_MB  Usable_file_MB
MOUNTED   NORMAL  N         512   4096  4194304     12288     8835             1117            3859

(continued)
Offline_disks  Voting_files  Name
            0             N  DATA
```

lsdsk

Purpose
Lists Oracle ASM disks.

Syntax and Description
```
lsdsk [--suppressheader] [-kptgMI] [-G diskgroup ]
      [ --member|--candidate]
      [--discovery][--statistics][pattern]
```

Table 12–36 lists the options for the lsdsk command.

Table 12–36   Options for the lsdsk command

Option	Description
(none)	Displays the PATH column of the V$ASM_DISK_STAT view.
-k	Displays the TOTAL_MB, FREE_MB, OS_MB,NAME, FAILGROUP, LIBRARY, LABEL, UDID, PRODUCT, REDUNDANCY, and PATH columns of the V$ASM_DISK view.
--statistics	Displays the READS, WRITES, READ_ERRS, WRITE_ERRS, READ_TIME, WRITE_TIME, BYTES_READ, BYTES_WRITTEN, and the PATH columns of the V$ASM_DISK view.

Table 12–36   (Cont.) Options for the lsdsk command

Option	Description
-p	Displays the GROUP_NUMBER, DISK_NUMBER, INCARNATION, MOUNT_STATUS, HEADER_STATUS, MODE_STATUS, STATE, and the PATH columns of the V$ASM_DISK view.
-t	Displays the CREATE_DATE, MOUNT_DATE, REPAIR_TIMER, and the PATH columns of the V$ASM_DISK view.
-g	Selects from GV$ASM_DISK_STAT, or from GV$ASM_DISK if the --discovery flag is also specified. GV$ASM_DISK.INST_ID is included in the output.
--discovery	Selects from V$ASM_DISK, or from GV$ASM_DISK if the -g flag is also specified. This option is always enabled if the Oracle ASM instance is version 10.1 or earlier. This flag is disregarded if lsdsk is running in non-connected mode.
--suppressheader	Suppresses column headings.
-I	Scans disk headers for information rather than extracting the information from an Oracle ASM instance. This option forces non-connected mode.
-G	Restricts results to only those disks that belong to the group specified by *diskgroup*.
-M	Displays the disks that are visible to some but not all active instances. These are disks that, if included in a disk group, cause the mount of that disk group to fail on the instances where the disks are not visible.
--candidate	Restricts results to only disks having membership status equal to CANDIDATE.
--member	Restricts results to only disks having membership status equal to MEMBER.
pattern	Returns only information about the specified disks that match the supplied pattern.

The lsdsk command can run in connected or non-connected mode. The connected mode is always attempted first. The -I option forces non-connected mode.

- In connected mode, lsdsk uses the V$ASM_DISK_STAT and V$ASM_DISK dynamic views to retrieve disk information. The V$ASM_DISK_STAT view is used by default.

- In non-connected mode, lsdsk scans disk headers to retrieve disk information. Some information is not available in this mode and some options are not valid combinations with this mode.

> **Note:** The non-connected mode is not supported on Windows.

pattern restricts the output to only disks that match the pattern specified. Wild-card characters and slashes (/ or \) can be part of the pattern. *pattern* should be specified as the last option for the command. For information about wildcards, see "Wildcard Characters" on page 12-5.

The -k, -p, -t, and --statistics options modify how much information is displayed for each disk. If any combination of the options are specified, then the output shows the union of the attributes associated with each flag.

Example

The following are examples of the `lsdsk` command. The first and second examples list information about disks in the `data` disk group. The third example lists information about candidate disks.

Example 12–38   Using the ASMCMD lsdsk command

```
ASMCMD [+] > lsdsk -t -G data
Create_Date  Mount_Date  Repair_Timer  Path
13-JUL-09    13-JUL-09   0             /devices/diska1
13-JUL-09    13-JUL-09   0             /devices/diska2
13-JUL-09    13-JUL-09   0             /devices/diska3
13-JUL-09    13-JUL-09   0             /devices/diskb1
13-JUL-09    13-JUL-09   0             /devices/diskb2
13-JUL-09    13-JUL-09   0             /devices/diskb3
13-JUL-09    13-JUL-09   0             /devices/diskc1
13-JUL-09    13-JUL-09   0             /devices/diskc2
...


ASMCMD [+] > lsdsk -p -G data /devices/diska*
Group_Num  Disk_Num     Incarn  Mount_Stat  Header_Stat  Mode_Stat  State   Path
        1         0  2105454210  CACHED      MEMBER       ONLINE     NORMAL  /devices/diska1
        1         1  2105454199  CACHED      MEMBER       ONLINE     NORMAL  /devices/diska2
        1         2  2105454205  CACHED      MEMBER       ONLINE     NORMAL  /devices/diska3


ASMCMD [+] > lsdsk --candidate -p
Group_Num  Disk_Num     Incarn  Mount_Stat  Header_Stat  Mode_Stat  State   Path
        0         5  2105454171  CLOSED      CANDIDATE    ONLINE     NORMAL  /devices/diske1
        0        25  2105454191  CLOSED      CANDIDATE    ONLINE     NORMAL  /devices/diske2
        0        18  2105454184  CLOSED      CANDIDATE    ONLINE     NORMAL  /devices/diske3
        0        31  2105454197  CLOSED      CANDIDATE    ONLINE     NORMAL  /devices/diskk1
        0        21  2105454187  CLOSED      CANDIDATE    ONLINE     NORMAL  /devices/diskk2
        0        26  2105454192  CLOSED      CANDIDATE    ONLINE     NORMAL  /devices/diskk3
        0        14  2105454180  CLOSED      CANDIDATE    ONLINE     NORMAL  /devices/diskl1
...
```

lsod

Purpose

Lists the open Oracle ASM disks.

Syntax and Description

`lsod [--suppressheader] [-G diskgroup] [--process process] [pattern]`

Table 12–37 lists the syntax options for the `lsod` command.

Table 12–37   Options for the lsod command

Option	Description
`--suppressheader`	Suppresses column header information from the output.
`-G diskgroup`	Specifies the disk group that contains the open disks.
`--process process`	Specifies a pattern to filter the list of processes.
`pattern`	Specifies a pattern to filter the list of disks.

The rebalance operation (RBAL) opens a disk both globally and locally so the same disk may be listed twice in the output for the RBAL process.

Example

The following are examples of the lsod command. The first example lists the open devices associated with the data disk group and the LGWR process. The second example lists the open devices associated with the LGWR process for disks that match the diska pattern.

Example 12–39   Using the ASMCMD lsod command

```
ASMCMD [+] > lsod -G data --process *LGWR*
Instance Process               OSPID Path
1       oracle@dadvmn0652 (LGWR) 26593 /devices/diska1
1       oracle@dadvmn0652 (LGWR) 26593 /devices/diska2
1       oracle@dadvmn0652 (LGWR) 26593 /devices/diska3
1       oracle@dadvmn0652 (LGWR) 26593 /devices/diskb1
1       oracle@dadvmn0652 (LGWR) 26593 /devices/diskb2
1       oracle@dadvmn0652 (LGWR) 26593 /devices/diskb3
1       oracle@dadvmn0652 (LGWR) 26593 /devices/diskd1

ASMCMD [+] > lsod --process *LGWR* *diska*
Instance Process               OSPID Path
1       oracle@dadvmn0652 (LGWR) 26593 /devices/diska1
1       oracle@dadvmn0652 (LGWR) 26593 /devices/diska2
1       oracle@dadvmn0652 (LGWR) 26593 /devices/diska3
```

For another example of the lsod command, see

md_backup

Purpose

The md_backup command creates a backup file containing metadata for one or more disk groups.

Syntax and Description

```
md_backup backup_file
     [-G 'diskgroup[,diskgroup,...]']
```

Table 12–38 describes the options for the md_backup command.

Table 12–38   Options for the md_backup command

Option	Description
backup_file	Specifies the backup file in which you want to store the metadata.
-G diskgroup	Specifies the disk group name of the disk group that must be backed up

By default all the mounted disk groups are included in the backup file, which is saved in the current working directory if a path is not specified with the file name.

Oracle Automatic Storage Management Cluster File System (Oracle ACFS) volume and file system information is not backed up.

Example

The first example shows the use of the backup command when run without the disk group option. This example backs up all the mounted disk groups and creates the backup image in the `/scratch/backup/alldgs20100422` file. The second example creates a backup of the `data` disk group. The metadata backup that this example creates is saved in the `/scratch/backup/data20100422` file.

Example 12–40   Using the ASMCMD md_backup command

```
ASMCMD [+] > md_backup /scratch/backup/alldgs20100422
Disk group metadata to be backed up: DATA
Disk group metadata to be backed up: FRA
Current alias directory path: ORCL/ONLINELOG
Current alias directory path: ORCL/PARAMETERFILE
Current alias directory path: ORCL
Current alias directory path: ASM
Current alias directory path: ORCL/DATAFILE
Current alias directory path: ORCL/CONTROLFILE
Current alias directory path: ASM/ASMPARAMETERFILE
Current alias directory path: ORCL/TEMPFILE
Current alias directory path: ORCL/ARCHIVELOG/2010_04_20
Current alias directory path: ORCL
Current alias directory path: ORCL/BACKUPSET/2010_04_21
Current alias directory path: ORCL/ARCHIVELOG/2010_04_19
Current alias directory path: ORCL/BACKUPSET/2010_04_22
Current alias directory path: ORCL/ONLINELOG
Current alias directory path: ORCL/BACKUPSET/2010_04_20
Current alias directory path: ORCL/ARCHIVELOG
Current alias directory path: ORCL/BACKUPSET
Current alias directory path: ORCL/ARCHIVELOG/2010_04_22
Current alias directory path: ORCL/DATAFILE
Current alias directory path: ORCL/CONTROLFILE
Current alias directory path: ORCL/ARCHIVELOG/2010_04_21

ASMCMD [+] > md_backup /scratch/backup/data20100422 -G data
Disk group metadata to be backed up: DATA
Current alias directory path: ORCL/ONLINELOG
Current alias directory path: ASM
Current alias directory path: ORCL/CONTROLFILE
Current alias directory path: ASM/ASMPARAMETERFILE
Current alias directory path: ORCL/PARAMETERFILE
Current alias directory path: ORCL
Current alias directory path: ORCL/DATAFILE
Current alias directory path: ORCL/TEMPFILE
```

md_restore

Purpose

The `md_restore` command restores disk groups from a metadata backup file.

Syntax and Description

```
md_restore backup_file [--silent]
     [--full|--nodg|--newdg -o 'old_diskgroup:new_diskgroup [,...]']
     [-S sql_script_file] [-G 'diskgroup [,diskgroup...]']
```

Table 12–39 describes the options for the `md_restore` command.

Table 12–39   Options for the md_restore command

Option	Description
backup_file	Reads the metadata information from *backup_file*.
--silent	Ignore errors. Typically, if md_restore encounters an error, it stops. Specifying this flag ignores any errors.
--full	Specifies to create a disk group and restore metadata.
--nodg	Specifies to restore metadata only.
--newdg -o *old_diskgroup*:*new_diskgroup*]	Specifies to create a disk group with a different name when restoring metadata. The -o option is required with --newdg.
-S *sql_script_file*	Write SQL commands to the specified SQL script file instead of executing the commands.
-G *diskgroup*	Select the disk groups to be restored. If no disk groups are defined, then all disk groups are restored.

Example

The first example restores the disk group data from the backup script and creates a copy. The second example takes an existing disk group data and restores its metadata. The third example restores disk group data completely but the new disk group that is created is named data2. The fourth example restores from the backup file after applying the overrides defined in the override.sql script file.

Example 12–41   Using the ASMCMD md_restore command

```
ASMCMD [+] > md_restore --full –G data --silent /scratch/backup/alldgs20100422

ASMCMD [+] > md_restore --nodg –G data --silent /scratch/backup/alldgs20100422

ASMCMD [+] > md_restore --newdg -o 'data:data2' --silent
/scratch/backup/data20100422

ASMCMD [+] > md_restore -S override.sql --silent /scratch/backup/data20100422
```

mkdg

Purpose

Creates a disk group based on an XML configuration file.

Syntax and Description

mkdg { *config_file*.xml | '*contents_of_xml_file*' }

Table 12–40 lists the syntax options for the mkdg command.

Table 12–40   Options for the mkdg command

Option	Description
config_file	Name of the XML file that contains the configuration for the new disk group. mkdg searches for the XML file in the directory where ASMCMD was started unless a path is specified.
	For examples of the valid tags and XML configuration file, see Example 12–42 and Example 12–43.

Table 12–40   (Cont.) Options for the mkdg command

Option	Description
contents_of_xml_file	The XML script enclosed in single quotations.

mkdg creates a new disk group with an XML configuration file that specifies the name of the disk group, redundancy, attributes, and paths of the disks that form the disk group. Redundancy is an optional parameter; the default is normal redundancy. For some types of redundancy, disks are required to be gathered into failure groups. In the case that failure groups are not specified for a disk group, each disk in the disk group belongs to its own failure group.

Note that mkdg only mounts a disk group on the local node.

It is possible to set some disk group attribute values during disk group creation. Some attributes, such as AU_SIZE and SECTOR_SIZE, can be set only during disk group creation. For more information about disk groups attributes, refer to "Disk Group Attributes" on page 4-2.

The default disk group compatibility settings are 10.1 for Oracle ASM compatibility, 10.1 for database compatibility, and no value for Oracle ADVM compatibility. For information about disk group compatibility attributes, see "Disk Group Compatibility" on page 4-33.

Example 12–42 shows the basic structure and the valid tags with their respective attributes for the mkdg XML configuration file.

Example 12–42   Tags for mkdg XML configuration file

```
<dg>  disk group
      name        disk group name
      redundancy  normal, external, high

<fg>  failure group
      name        failure group name
</fg>

<dsk> disk
      name        disk name
      string      disk path
      size        size of the disk to add
      force       true specifies to use the force option
</dsk>

<a>   attribute
      name        attribute name
      value       attribute value
</a>

</dg>
```

For information about altering a disk group with ASMCMD chdg, see "chdg" on page 12-31. For information about creating a disk group, see "Creating Disk Groups" on page 4-4.

Example

The following is an example of an XML configuration file for mkdg. The configuration file creates a disk group named data with normal redundancy. Two failure groups, fg1

and `fg2`, are created, each with two disks identified by associated disk strings. The disk group compatibility attributes are all set to `11.2`.

Example 12–43   mkdg sample XML configuration file

```
<dg name="data" redundancy="normal">
  <fg name="fg1">
    <dsk string="/dev/disk1"/>
    <dsk string="/dev/disk2"/>
  </fg>
  <fg name="fg2">
    <dsk string="/dev/disk3"/>
    <dsk string="/dev/disk4"/>
  </fg>
  <a name="compatible.asm" value="11.2"/>
  <a name="compatible.rdbms" value="11.2"/>
  <a name="compatible.advm" value="11.2"/>
</dg>
```

The following are examples of the `mkdg` command. The first example runs `mkdg` with an XML configuration file in the directory where ASMCMD was started. The second example runs `mkdg` using information on the command line.

Example 12–44   Using the ASMCMD mkdg command

```
ASMCMD [+] > mkdg data_config.xml

ASMCMD [+] > mkdg '<dg name="data"><dsk string="/dev/disk*"/></dg>'
```

mount

Purpose
Mounts a disk group.

Syntax and Description
mount [--restrict] { [-a] | [-f] *diskgroup*[*diskgroup* ...] }

Table 12–41 lists the syntax options for the `mount` command.

Table 12–41   Options for the mount command

Option	Description
diskgroup	Name of the disk group.
-a	Mounts all disk groups.
--restrict	Mounts in restricted mode.
-f	Forces the mount operation.

This operation mounts one or more disk groups. A disk group can be mounted with or without force or restricted options. For more information about mounting disk groups, see "Mounting and Dismounting Disk Groups" on page 4-44.

Example
The following are examples of the `mount` command showing the use of the force, restrict, and all options.

Example 12–45   Using the ASMCMD mount command

```
ASMCMD [+] > mount -f data

ASMCMD [+] > mount --restrict data

ASMCMD [+] > mount -a
```

offline

Purpose

Offline disks or failure groups that belong to a disk group.

Syntax and Description

```
offline -G diskgroup
        { -F failgroup |-D disk}
        [-t {minutes | hours}]
```

Table 12–42 lists the syntax options for the offline command.

Table 12–42   Options for the offline command

Option	Description
-G diskgroup	Disk group name.
-F failgroup	Failure group name.
-D disk	Specifies a single disk name.
-t minutes | hours	Specifies the time before the specified disk is dropped as *n*m or *n*h, where m specifies minutes and h specifies hours. For example, 120m or 2h. The default unit is hours.

When a failure group is specified, this implies all the disks that belong to it should be offlined.

Example

The following are examples of the offline command. The first example offlines the failgroup1 failure group of the data disk group. The second example offlines the data_0001 disk of the data disk group with a time of 1.5 hours before the disk is dropped.

Example 12–46   Using the ASMCMD offline command

```
ASMCMD [+] > offline -G data -F failgroup1

ASMCMD [+] > offline -G data -D data_0001 -t 1.5h
```

online

Purpose

Online all disks, a single disk, or a failure group that belongs to a disk group.

Syntax and Description

```
online -G diskgroup { -a | -F failgroup |-D disk} [-w]
```

Table 12–43 lists the syntax options for the `online` command.

Table 12–43   Options for the online command

Option	Description
-a	Online all offline disks in the disk group.
-G *diskgroup*	Disk group name.
-F *failgroup*	Failure group name.
-D *disk*	Disk name.
-w	Wait option. Causes ASMCMD to wait for the disk group to be rebalanced before returning control to the user. The default is not waiting.

When a failure group is specified, this implies all the disks that belong to it should be onlined.

Example

The following are examples of the `online` command. The first example onlines all disks in the `failgroup1` failure group of the `data` disk group with the `wait` option enabled. The second example onlines the `data_0001` disk in the `data` disk group.

Example 12–47   Using the ASMCMD online command

```
ASMCMD [+] > online -G data -F failgroup1 -w

ASMCMD [+] > online -G data -D data_0001
```

rebal

Purpose

Rebalances a disk group.

Syntax and Description

rebal [--power *power*] [-w] *diskgroup*

Table 12–44 lists the syntax options for the `rebal` command.

Table 12–44   Options for the rebal command

Option	Description
diskgroup	Disk group name.
--power *power*	Power setting.
-w	Wait option. Causes ASMCMD to wait for the disk group to be rebalanced before returning control to the user. The default is not waiting.

The power level can be set to the same values as the `ASM_POWER_LIMIT` initialization parameter. A value of `0` disables rebalancing. If the rebalance power is not specified, the value defaults to the setting of the `ASM_POWER_LIMIT` initialization parameter. For information about the power level, see "ASM_POWER_LIMIT" on page 3-8 and "Tuning Rebalance Operations" on page 4-19.

You can determine if a rebalance operation is occurring with the ASMCMD `lsop` command. See "lsop" on page 12-13. For more information about rebalancing a disk group, see "Manually Rebalancing Disk Groups" on page 4-18.

Example

The following is an example of the `rebal` command that rebalances the `fra` disk group with a power level set to `4`.

Example 12–48   Using the ASMCMD rebal command

```
ASMCMD [+] > rebal --power 4 fra

ASMCMD [+] > lsop
Group_Name  Dsk_Num  State  Power
FRA         REBAL    RUN    4
```

remap

Purpose

Marks a range of blocks as unusable on the disk and relocates any data allocated in that range.

Syntax and Description

remap *diskgroup disk block_range*

Table 12–45 lists the syntax options for the `remap` command.

Table 12–45   Options for the remap command

Option	Description
diskgroup	Disk group name in which a disk must have data relocated.
disk	Name of the disk that must have data relocated. The name must match the `NAME` column in the `V$ASM_DISK` view.
block_range	Range of physical blocks to relocate in the format *start_range_number-end_range_number*.

The `remap` command only relocates blocks. It does not correct or repair blocks that contain corrupted contents. The command uses a physical block size based on the `SECTOR_SIZE` disk group attribute.

Examples

The first example remaps blocks `5000` through `5999` for disk `DATA_0001` in disk group `DATA`. The second example remaps blocks `6230` through `6339` for disk `FRA_0002` in disk group `FRA`

Example 12–49   Using the ASMCMD remap command

```
ASMCMD [+] > remap DATA DATA_0001 5000-5999

ASMCMD [+] > remap FRA FRA_0002 6230-6339
```

setattr

Purpose
Sets the attributes for an Oracle ASM disk group.

Syntax and Description
```
setattr -G diskgroup attribute_name attribute_value
```

Table 12–46 lists the syntax options for the setattr command.

Table 12–46   Options for the setattr command

Option	Description
-G diskgroup	Disk group name.
attribute_name	Name of the attribute.
attribute_value	Value of the attribute.

The COMPATIBLE.ASM attribute must be advanced before advancing other disk group compatibility attributes and its value must be greater than or equal to the value of other disk group compatibility attributes.

For information about disk group attributes, see "Disk Group Attributes" on page 4-2.

Example
The following are examples of the setattr command. The first example sets the disk group attribute COMPATIBLE.ASM to 11.2 for the data disk group. The second example sets the disk group attribute COMPATIBLE.RDBMS to 11.2 for the data disk group.

Example 12–50   Using the ASMCMD setattr command

```
ASMCMD [+] > setattr -G data compatible.asm 11.2.0.0.0

ASMCMD [+] > setattr -G data compatible.rdbms 11.2.0.0.0
```

umount

Purpose
Dismounts a disk group.

Syntax and Description
```
umount { -a | [-f] diskgroup }
```

Table 12–47 lists the syntax options for the umount command.

Table 12–47   Options for the umount command

Option	Description
diskgroup	Name of the disk group.
-a	Dismounts all mounted disk groups. These disk groups are listed in the output of the V$ASM_DISKGROUP view.
-f	Forces the dismount operation.

Example

The following are examples of the `umount` command. The first example dismounts all disk groups mounted on the Oracle ASM instance. The second example forces the dismount of the `data` disk group.

Example 12–51   Using the ASMCMD umount command

```
ASMCMD [+] > umount -a

ASMCMD [+] > umount -f data
```

ASMCMD Template Management Commands

This section describes the ASMCMD template management commands.

Table 12–48 provides a summary of the template management commands.

Table 12–48   Summary of ASMCMD template management commands

Command	Description
chtmpl	Changes the attributes of a template.
lstmpl	Lists disk group templates.
mktmpl	Adds a template to a disk group.
rmtmpl	Removes a template from a disk group.

chtmpl

Purpose

Changes the attributes of a template.

Syntax and Description

```
chtmpl -G diskgroup { [--striping {coarse|fine}]
    [--redundancy {high|mirror|unprotected} ] [--primary {hot|cold}]
    [--secondary {hot|cold}]} template
```

Table 12–49 lists the syntax options for the `chtmpl` command.

Table 12–49   Options for the chtmpl command

Option	Description
-G diskgroup	Name of the disk group.
template	Name of the template to change.
--striping {coarse | fine}	Striping specification, either `coarse` or `fine`.
--redundancy { high | mirror | unprotected}	Redundancy specification, either `high`, `mirror`, or `unprotected`.
--primary { hot | cold }	Intelligent Data Placement specification for primary extents, either `hot` or `cold` region.
--secondary { hot | cold }	Intelligent Data Placement specification for secondary extents, either `hot` or `cold` region.

At least one of these options is required: `--striping`, `--redundancy`, `--primary`, or `--secondary`.

For information about updating templates, see "Modifying a Disk Group Template" on page 7-17.

Example

The following is an example of the `chtmpl` command that updates the `mytemplate` template of the `data` disk group. The redundancy attribute is set to high and the striping attribute is set to fine.

Example 12–52   Using the ASMCMD chtmpl command

```
ASMCMD [+] > chtmpl -G data --redundancy high --striping fine mytemplate
```

lstmpl

Purpose

Lists all templates for all disk groups or the templates for a specified disk group.

Syntax and Description

`lstmpl [--suppressheader] [-l] [-G diskgroup] [pattern]`

Table 12–50 lists the syntax options for the `lstmpl` command.

Table 12–50   Options for the lstmpl command

Option	Description
-G *diskgroup*	Specifies disk group name.
--suppressheader	Suppresses column headings.
-l	Displays all details.
pattern	Displays the templates that match pattern expression.

For information about template attributes, see "Template Attributes" on page 7-15.

Example

The following is an example of the `lstmpl` command. The example lists all details of the templates in the `data` disk group.

Example 12–53   Using the ASMCMD lstmpl command

```
ASMCMD [+] > lstmpl -l -G data

Group_Name  Group_Num  Name             Stripe  Sys  Redund  PriReg  MirrReg
DATA        1          ARCHIVELOG       COARSE  Y    MIRROR  COLD    COLD
DATA        1          ASMPARAMETERFILE COARSE  Y    MIRROR  COLD    COLD
DATA        1          AUTOBACKUP       COARSE  Y    MIRROR  COLD    COLD
DATA        1          BACKUPSET        COARSE  Y    MIRROR  COLD    COLD
DATA        1          CHANGETRACKING   COARSE  Y    MIRROR  COLD    COLD
DATA        1          CONTROLFILE      FINE    Y    HIGH    COLD    COLD
DATA        1          DATAFILE         COARSE  Y    MIRROR  COLD    COLD
DATA        1          DATAGUARDCONFIG  COARSE  Y    MIRROR  COLD    COLD
DATA        1          DUMPSET          COARSE  Y    MIRROR  COLD    COLD
DATA        1          FLASHBACK        COARSE  Y    MIRROR  COLD    COLD
DATA        1          FLASHFILE        COARSE  Y    MIRROR  COLD    COLD
DATA        1          MYTEMPLATE       FINE    N    HIGH    COLD    COLD
DATA        1          OCRFILE          COARSE  Y    MIRROR  COLD    COLD
```

```
DATA      1         ONLINELOG        COARSE  Y    MIRROR  COLD    COLD
DATA      1         PARAMETERFILE    COARSE  Y    MIRROR  COLD    COLD
DATA      1         TEMPFILE         COARSE  Y    MIRROR  COLD    COLD
DATA      1         XTRANSPORT       COARSE  Y    MIRROR  COLD    COLD
```

mktmpl

Purpose
Adds a template to a disk group.

Syntax and Description

```
mktmpl -G diskgroup [--striping {coarse|fine}]
     [--redundancy {high|mirror|unprotected}] [--primary {hot|cold}]
     [--secondary {hot|cold}] template
```

Table 12–51 lists the syntax options for the `mktmpl` command.

Table 12–51   Options for the mktmpl command

Option	Description
-G *diskgroup*	Name of the disk group.
template	Name of the template to create.
--striping {coarse | fine}	Striping specification, either `coarse` or `fine`.
--redundancy { high | mirror | unprotected}	Redundancy specification, either `high`, `mirror`, or `unprotected`.
--primary { hot | cold }	Intelligent Data Placement specification for primary extents, either `hot` or `cold` region.
--secondary { hot | cold }	Intelligent Data Placement specification for secondary extents, either `hot` or `cold` region.

For information about adding templates, see "Adding Templates to a Disk Group" on page 7-17.

Example
The following is an example of the `mktmpl` command that adds the `mytemplate` template to the `data` disk group. The new template has the redundancy set to mirror and the striping set to coarse.

Example 12–54   Using the ASMCMD mktmpl command

```
ASMCMD [+] > mktmpl -G data --redundancy mirror --striping coarse mytemplate
```

rmtmpl

Purpose
Removes a template from a disk group.

Syntax and Description

```
rmtmpl -G diskgroup template
```

Table 12–52 lists the syntax options for the `rmtmpl` command.

Table 12–52   Options for the rmtmpl command

Option	Description
`-G diskgroup`	Name of the disk group.
`template`	Name of the template to delete.

For information about template attributes, see "Template Attributes" on page 7-15.

Example
The following is an example of the `rmtmpl` command that removes the `mytemplate` template from the `data` disk group.

Example 12–55   Using the ASMCMD rmtmpl command

```
ASMCMD [+] > rmtmpl -G data mytemplate
```

ASMCMD File Access Control Commands

This section describes the ASMCMD Oracle ASM File Access Control commands. For information about Oracle ASM File Access Control, see "Managing Oracle ASM File Access Control for Disk Groups" on page 4-40.

Table 12–53 provides a summary of ASMCMD Oracle ASM File Access Control commands.

Table 12–53   Summary of ASMCMD file access control commands

Command	Description
chgrp	Changes the user group of a file or list of files.
chmod	Changes permissions of a file or list of files.
chown	Changes the owner of a file or list of files.
groups	Lists the user groups to which a user belongs.
grpmod	Adds or removes users from an existing user group.
lsgrp	Lists user groups.
lsusr	Lists users in a disk group.
mkgrp	Creates a new user group.
mkusr	Adds a user to disk group.
passwd	Changes the password of a user.
rmgrp	Deletes a user group.
rmusr	Deletes a user from a disk group.

chgrp

Purpose
Changes the user group of a file or list of files.

Syntax and Description
`chgrp usergroup file [file ...]`

Table 12–54 lists the syntax options for the chgrp command.

Table 12–54   Options for the chgrp command

Option	Description
usergroup	Name of the user group.
file	Name of a file.

Only the file owner or the Oracle ASM administrator can use this command. If the user is the file owner, then the user must also be either the owner or a member of the group for this command to succeed.

This command accepts a file name or multiple file names separated by spaces.

Examples
The following are examples of the chgrp command that change the Oracle ASM user group of the specified files.

Example 12–56   Using the ASMCMD chgrp command

```
ASMCMD [+] > chgrp asm_data +data/orcl/controlfile/Current.260.684924747

ASMCMD [+fra/orcl/archivelog/flashback] > chgrp asm_fra log_7.264.684968167 log_8.265.684972027
```

chmod

Purpose
Changes permissions of a file or list of files.

Syntax and Description
chmod *mode file* [*file* ...]

mode can be one of the following forms:

- { ugo | ug | uo | go | u | g | o | a } {+|-} {r|w |rw}

  a specifies permissions for all users, u specifies permissions for the owner of the file, g specifies the group permissions, and o specifies permissions for other users.

- { 0|4|6} {0|4|6} {0|4|6}

  The first digit specifies owner permissions, the second digit specifies group permissions, and the third digit specifies other permissions.

Table 12–55 lists the syntax options for the chmod command.

Table 12–55   Options for the chmod command

Option	Description
6	Read write permissions
4	Read only permissions
0	No permissions
u	Owner permissions, used with r or w
g	Group permissions, used with r or w

Table 12–55   (Cont.) Options for the chmod command

Option	Description
o	Other user permissions, used with r or w
a	All user permissions, used with r or w
+	Add a permission, used with r or w
-	Removes a permission, used with r or w
r	Read permission
w	Write permission
file	Name of a file

This command accepts a file name or multiple file names separated by spaces. The specified files must be closed.

You can only set file permissions to read-write, read-only, and no permissions. You cannot set file permissions to write-only.

To view the permissions on a file, use the ASMCMD ls command with the --permission option. See "ls" on page 12-24.

Examples

The following are examples of the chmod command that change the permissions of the specified files.

Example 12–57   Using the ASMCMD chmod command

```
ASMCMD [+fra/orcl/archivelog/flashback] > chmod ug+rw log_7.264.684968167 log_8.265.684972027

ASMCMD [+fra/orcl/archivelog/flashback] > chmod 640 log_7.264.684968167 log_8.265.684972027

ASMCMD [+] > ls --permission +fra/orcl/archivelog/flashback
User      Group      Permission  Name
grid     asm_fra      rw-r-----  log_7.264.684968167
grid     asm_fra      rw-r-----  log_8.265.684972027
```

chown

Purpose

Changes the owner of a file or list of files.

Syntax and Description

chown *user*[:*usergroup*] *file*[*file*...]

Table 12–56 lists the syntax options for the chown command.

Table 12–56   Options for the chown command

Option	Description
user	The name of the user that becomes the new owner.
usergroup	Name of the user group to which the user belongs.
file	Name of a file.

user typically refers to the user that owns the database instance home. Oracle ASM File Access Control uses the operating system (OS) name to identify a database.

This command accepts a file name or multiple file names separated by spaces. The specified files must be closed.

Only the Oracle ASM administrator can use this command.

Examples

The following are examples of the chown command that change the owner of the specified files to the oracle1 operating system user.

Example 12–58   Using the ASMCMD chown command

```
ASMCMD [+fra/orcl/archivelog/flashback] > chown oracle1 log_7.264.684968167 log_8.265.684972027

ASMCMD [+fra/orcl/archivelog/flashback] > chown oracle1:asm_fra log_9.264.687650269
```

groups

Purpose

Lists all the user groups to which the specified user belongs.

Syntax and Description

groups *diskgroup user*

Table 12–57 lists the syntax options for the groups command.

Table 12–57   Options for the groups command

Option	Description
diskgroup	Name of the disk group to which the user belongs.
user	Name of the user.

Examples

The following is an example of the groups command that displays the user groups of the data disk group to which the oracle1 user belongs.

Example 12–59   Using the ASMCMD groups command

```
ASMCMD [+] > groups data oracle1
asm_data
```

grpmod

Purpose

Adds or removes operating system (OS) users to and from an existing Oracle ASM user group.

Syntax and Description

grpmod { --add | --delete } *diskgroup usergroup user* [*user*...]

Table 12–58 lists the syntax options for the grpmod command.

Table 12–58   Options for the grpmod command

Option	Description
--add	Specifies to add users to the user group.
--delete	Specifies to delete users from the user group.
diskgroup	Name of the disk group to which the user group belongs.
usergroup	Name of the user group.
user	Name of the user to add or remove from the user group.

Only the owner of the user group can use this command. The command requires the SYSASM privilege to run.

This command accepts an operating system user name or multiple user names separated by spaces. The operating system users are typically owners of a database instance home.

Examples

The following are examples of the grpmod command. The first example adds the oracle1 and oracle2 users to the asm_fra user group of the fra disk group. The second example removes the oracle2 user from the asm_data user group of the data disk group.

Example 12–60   Using the ASMCMD grpmod command

```
ASMCMD [+] > grpmod --add fra asm_fra oracle1 oracle2

ASMCMD [+] > grpmod --delete data asm_data oracle2
```

lsgrp

Purpose

Lists all Oracle ASM user groups or only groups that match a specified pattern.

Syntax and Description

```
lsgrp [--suppressheader][-a] [ -G diskgroup ] [ pattern ]
```

Table 12–59 lists the syntax options for the lsgrp command.

Table 12–59   Options for the lsgrp command

Option	Description
--suppressheader	Suppresses column headings.
-a	Lists all columns.
-G *diskgroup*	Limits the results to the specified disk group name.
pattern	Displays the user groups that match the pattern expression.

Examples

The following are examples of the lsgrp command. The first example displays a subset of information about the user groups whose name matches the asm% pattern. The second example displays all information about all the user groups.

Example 12–61   Using the ASMCMD lsgrp command

```
ASMCMD [+] > lsgrp asm%
DG_Name  Grp_Name  Owner
FRA      asm_fra   grid
DATA     asm_data  grid

ASMCMD [+] > lsgrp -a
DG_Name  Grp_Name  Owner    Members
FRA      asm_fra   grid     oracle1
DATA     asm_data  grid     oracle1 oracle2
```

lsusr

Purpose
Lists Oracle ASM users in a disk group.

Syntax and Description
```
lsusr [--suppressheader][-a] [-G diskgroup ] [ pattern ]
```

Table 12–60 lists the syntax options for the lsusr command.

Table 12–60   Options for the lsusr command

Option	Description
-a	List all users and the disk groups to which the users belong.
--suppressheader	Suppresses column headings.
-G diskgroup	Limits the results to the specified disk group name.
pattern	Displays the users that match the pattern expression.

Examples
The following is an example of the lsusr command. The example lists users in the data disk group and also shows the operating system Id assigned to the user.

Example 12–62   Using the ASMCMD lsusr command

```
ASMCMD [+] > lsusr -G data
User_Num OS_ID OS_Name
3       1001  grid
1       1021  oracle1
2       1022  oracle2
```

mkgrp

Purpose
Creates a new Oracle ASM user group.

Syntax and Description
```
mkgrp diskgroup usergroup [user] [user...]
```

Table 12–61 lists the syntax options for the mkgrp command.

Table 12–61   Options for the mkgrp command

Option	Description
diskgroup	Name of the disk group to which the user group is added.
usergroup	Name of the user group to add. 30 is the maximum number of characters.
user	Name of the database user to add to the user group.

You can optionally specify a list of users to be included as members of the new user group.

Examples

The following is an example of the `mkgrp` command. This example creates the `asm_data` user group in the `data` disk group and adds the `oracle1` and `oracle2` users to the user group.

Example 12–63   Using the ASMCMD mkgrp command

```
ASMCMD [+] > mkgrp data asm_data oracle1 oracle2
```

mkusr

Purpose

Adds an operating system (OS) user to a disk group.

Syntax and Description

mkusr *diskgroup user*

Table 12–62 lists the syntax options for the `mkusr` command.

Table 12–62   Options for the mkusr command

Option	Description
diskgroup	Specifies the name of the disk group to which the user is to be added.
user	Name of the user to add.

The user to be added must be a valid operating system user. Only a user authenticated as `SYSASM` can run this command.

Examples

The following are examples of the `mkusr` command. The first example adds the `oracle1` user to the `data` disk group. The second example adds the `oracle2` user to the `fra` disk group.

Example 12–64   Using the ASMCMD mkusr command

```
ASMCMD [+] > mkusr data oracle1

ASMCMD [+] > mkusr fra oracle2
```

passwd

Purpose
Changes the password of a user.

Syntax and Description
passwd *user*

Table 12–63 lists the syntax options for the passwd command.

Table 12–63   Options for the passwd command

Option	Description
user	Name of the user.

An error is raised if the user does not exist in the Oracle ASM password file. The user is first prompted for the current password, then the new password. The command requires the SYSASM privilege to run.

Examples
The following is an example of the passwd command that changes the password of the oracle2 user.

Example 12–65   Using the ASMCMD passwd command

```
ASMCMD [+] > passwd oracle2
Enter old password (optional):
Enter new password: ******
```

rmgrp

Purpose
Removes a user group from a disk group.

Syntax and Description
rmgrp *diskgroup usergroup*

Table 12–64 lists the syntax options for the rmgrp command.

Table 12–64   Options for the rmgrp command

Option	Description
diskgroup	Name of the disk group to which the user group belongs.
usergroup	Name of the user group to delete.

Removing a group might leave some files without a valid user group. To ensure that those files have a valid group, explicitly update those files to a valid user group. See "chgrp" on page 12-53.

The command must be run by the owner of the user group and also requires the SYSASM privilege to run.

Examples

The following is an example of the `rmgrp` command that removes the `asm_data` user group from the `data` disk group.

Example 12–66   Using the ASMCMD rmgrp command

```
ASMCMD [+] > rmgrp data asm_data
```

rmusr

Purpose

Deletes an operating system (OS) user from a disk group.

Syntax and Description

```
rmusr [-r] diskgroup user
```

Table 12–65 lists the syntax options for the `rmusr` command.

Table 12–65   Options for the rmusr command

Option	Description
-r	Removes all files in the disk group that the user owns at the same time that the user is removed.
diskgroup	Specifies the name of the disk group from which the user is to be deleted.
user	Name of the user to delete.

The command requires the SYSASM privilege to run.

Examples

The following is an example of the `rmusr` command that removes the `oracle2` user from the `data` disk group.

Example 12–67   Using the ASMCMD rmusr command

```
ASMCMD [+] > rmusr data oracle2
```

ASMCMD Volume Management Commands

This section describes the ASMCMD volume management commands.

For information about Oracle ASM Dynamic Volume Manager (Oracle ADVM), refer to "Overview of Oracle ASM Dynamic Volume Manager" on page 5-20. For information about using SQL statements to manage volumes, refer to "Managing Volumes in a Disk Group" on page 4-13.

Table 12–66 provides a summary of the Oracle ADVM volume management commands. To successfully run these commands, the local Oracle ASM instance must be running and the disk group required by this command must have been created and mounted in the Oracle ASM instance.

Table 12–66   Summary of ASMCMD volume management commands

Command	Description
volcreate	Creates an Oracle ADVM volume in the specified disk group.
voldelete	Deletes an Oracle ADVM volume.
voldisable	Disables Oracle ADVM volumes in mounted disk groups.
volenable	Enables Oracle ADVM volumes in mounted disk groups.
volinfo	Displays information about Oracle ADVM volumes.
volresize	Resizes an Oracle ADVM volume.
volset	Sets attributes of an Oracle ADVM volume in mounted disk groups.
volstat	Reports volume I/O statistics.

volcreate

Purpose

Creates an Oracle ADVM volume in the specified disk group.

Syntax and Description

```
volcreate -G diskgroup -s size
    [ --column number ] [ --width stripe_width ]
    [--redundancy {high|mirror|unprotected} ]
    [--primary {hot|cold}] [--secondary {hot|cold}] volume
```

Table 12–67 describes the options for the `volcreate` command.

Table 12–67   Options for the volcreate command

Option	Description
`-G diskgroup`	Name of the disk group containing the volume.
`-s size`	Size of the volume to be created in units of `K`, `M`, `G`, or `T`. The unit designation must be appended to the number specified. No space is allowed. For example: `200M` or `20G`
`--column number`	Number of columns in a stripe set. Values range from `1` to `8`. The default value is `4`.
`--width stripe`	Stripe width of a volume. The value can range from 4 KB to 1 MB, at power-of-two intervals, with a default of 128 KB.
`--redundancy {high|mirror|unprotected}`	Redundancy of the Oracle ADVM volume which can be specified for normal redundancy disk groups. The range of values is as follows: `unprotected` for non-mirrored redundancy, `mirror` for double-mirrored redundancy, or `high` for triple-mirrored redundancy. If redundancy is not specified, the setting defaults to the redundancy level of the disk group.
`--primary {hot|cold}`	Intelligent Data Placement specification for primary extents, either `hot` or `cold` region.
`--secondary {hot|cold}`	Intelligent Data Placement specification for secondary extents, either `hot` or `cold` region.

Table 12–67 (Cont.) Options for the volcreate command

Option	Description
volume	Name of the volume. Only alphanumeric characters are allowed; hyphens are not allowed. The first character must be alphabetic.

When creating an Oracle ADVM volume, a volume device name is created with a unique Oracle ADVM persistent disk group number that is concatenated to the end of the volume name. The unique number can be one to three digits.

On Linux, the volume device name is in the format *volume_name-nnn*, such as volume1-123. On Windows the volume device name is in the format asm-*volume_name-nnn*, such as asm-volume1-123. For information on mounting the volume device file, see "Creating an Oracle ACFS File System" on page 13-2.

On Linux platforms, the volume name must be less than or equal to eleven alphanumeric characters, starting with an alphabetic character. On AIX platforms, the volume name must be less than or equal to twenty three alphanumeric characters, starting with an alphabetic character. On Windows and Solaris platforms, the volume name must be less than or equal to thirty alphanumeric characters, starting with an alphabetic character.

You can determine the volume device name with the volinfo command, described in "volinfo" on page 12-66.

If the --column option is set to 1, then striping is disabled and the stripe width is equal to the volume extent size. The volume extent size is 8 MB if the disk group allocation unit (AU) is less than or equal to 8 MB. If the AU size is greater than 8 MB, then the Oracle ADVM volume extent size is equivalent to the disk group AU size.

A successful volume creation automatically enables the volume device.

The volume device file functions as any other disk or logical volume to mount file systems or for applications to use directly.

For information about redundancy settings, see "Mirroring, Redundancy, and Failure Group Options" on page 4-24. For information about hot and cold disk regions, see "Intelligent Data Placement" on page 4-16.

Before creating an Oracle ADVM volume on AIX, ensure that the necessary user authorizations have been created. For information, refer to "Oracle ACFS Command-line Tools for the AIX Environment" on page 13-34.

Examples

The following is an example of the volcreate command that creates volume1 in the data disk group with the size set to 10 gigabytes.

Example 12–68 Using the ASMCMD volcreate command

```
ASMCMD [+] >  volcreate -G data -s 10G --width 64K --column 8 volume1

ASMCMD [+] > volinfo -G data volume1
Diskgroup Name: DATA

        Volume Name: VOLUME1
        Volume Device: /dev/asm/volume1-123
        State: ENABLED
        Size (MB): 10240
        Resize Unit (MB): 64
```

```
Redundancy: MIRROR
Stripe Columns: 8
Stripe Width (K): 64
Usage:
Mountpath:
```

voldelete

Purpose

Deletes an Oracle ADVM volume.

Syntax and Description

```
voldelete -G diskgroup volume
```

Table 12–68 describes the options for the `voldelete` command.

Table 12–68   Options for the voldelete command

Option	Description
-G *diskgroup*	Name of the disk group containing the volume.
volume	Name of the volume.

To successfully run this command, the local Oracle ASM instance must be running and the disk group required by this command must be mounted in the Oracle ASM instance. Before deleting a volume, you must ensure that there are no active file systems associated with the volume.

Examples

The following is an example of the `voldelete` command that deletes `volume1` from the `data` disk group.

Example 12–69   Using the ASMCMD voldelete command

```
ASMCMD [+] > voldelete -G data volume1
```

voldisable

Purpose

Disables Oracle ADVM volumes in mounted disk groups and removes the volume device on the local node.

Syntax and Description

```
voldisable { -a | -G diskgroup -a | -G diskgroup volume }
```

Table 12–69 describes the options for the `voldisable` command.

Table 12–69   Options for the voldisable command

Option	Description
-a	When used without a disk group name, specifies all volumes within all disk groups.
	When used with a disk group name (-G *diskgroup* -a), specifies all volumes within that disk group.
-G *diskgroup*	Name of the disk group containing the volume.
volume	Name of the volume.

You can disable volumes before shutting down an Oracle ASM instance or dismounting a disk group to verify that the operations can be accomplished normally without including a force option due to open volume files. Disabling a volume also prevents any subsequent opens on the volume or device file because it no longer exists.

Before disabling a volume, you must ensure that there are no active file systems associated with the volume. You must first dismount the Oracle ACFS file system before disabling the volume. See "Dismounting an Oracle ACFS File System" on page 13-19.

You can delete a volume without first disabling the volume.

Examples

The following is an example of the voldisable command that disables volume1 in the data disk group.

Example 12–70   Using the ASMCMD voldisable command

```
ASMCMD [+] > voldisable -G data volume1
```

volenable

Purpose

Enables Oracle ADVM volumes in mounted disk groups.

Syntax and Description

```
volenable { -a | -G diskgroup -a | -G diskgroup volume }
```

Table 12–70 describes the options for the volenable command.

Table 12–70   Options for the volenable command

Option	Description
-a	When used without a disk group name, specifies all volumes within all disk groups.
	When used with a disk group name (-G *diskgroup* -a), specifies all volumes within that disk group.
-G *diskgroup*	Name of the disk group containing the volume.
volume	Name of the volume.

A volume is enabled when it is created.

Examples

The following is an example of the volenable command that enables volume1 in the data disk group.

Example 12–71   Using the ASMCMD volenable command

```
ASMCMD [+] > volenable -G data volume1
```

volinfo

Purpose

Displays information about Oracle ADVM volumes.

Syntax and Description

```
volinfo { -a | -G diskgroup -a | -G diskgroup volume }
volinfo { --show_diskgroup | --show_volume } volumedevice
```

Table 12–71 describes the options for the volinfo command.

Table 12–71   Options for the volinfo command

Option	Description
-a	When used without a disk group name, specifies all volumes within all disk groups.
	When used with a disk group name (-G diskgroup -a), specifies all volumes within that disk group.
-G diskgroup	Name of the disk group containing the volume.
volume	Name of the volume.
--show_diskgroup	Returns only the disk group name. A volume device name is required.
--show_volume	Returns only the volume name. A volume device name is required.
volumedevice	Name of the volume device.

Examples

The first example displays information about the volume1 volume in the data disk group. This example was produced in a Linux environment. The second example displays information about the volume1 volume in the data disk group and was produced in a Windows environment.

The mount path field displays the last mount path for the volume.

Example 12–72   Using the ASMCMD volinfo command

```
ASMCMD [+] > volinfo -G data volume1
Diskgroup Name: DATA
        Volume Name: VOLUME1
        Volume Device: /dev/asm/volume1-123
        State: ENABLED
        Size (MB): 10240
        Resize Unit (MB): 64
        Redundancy: MIRROR
        Stripe Columns: 8
        Stripe Width (K): 64
```

```
          Usage: ACFS
          Mountpath: /u01/app/acfsmounts/acfs1

ASMCMD [+] > volinfo -G data -a
Diskgroup Name: DATA
          Volume Name: VOLUME1
          Volume Device: \\.\asm-volume1-311
          State: ENABLED
          Size (MB): 1024
          Resize Unit (MB): 32
          Redundancy: MIRROR
          Stripe Columns: 4
          Stripe Width (K): 128
          Usage: ACFS
          Mountpath: C:\oracle\acfsmounts\acfs1
```

volresize

Purpose

Resizes an Oracle ADVM volume.

Syntax and Description

```
volresize -G diskgroup -s size [ -f ] volume
```

Table 12–72 describes the options for the volresize command.

Table 12–72   Options for the volresize command

Option	Description
-G *diskgroup*	Name of the disk group containing the volume.
-f	Forces the shrinking of a volume that is not an Oracle ACFS volume and suppresses any warning message.
volume	Name of the volume.
-s *size*	New size of the volume in units of K, M, G, or T.

If the volume is mounted on a non-Oracle ACFS file system, then dismount the file system first before resizing. If the new size is smaller than current, you are warned of possible data corruption. Unless the -f (force) option is specified, you are prompted whether to continue with the operation.

If there is an Oracle ACFS file system on the volume, then you cannot resize the volume with the volresize command. You must use the acfsutil size command, which also resizes the volume and file system. For information, see "acfsutil size" on page 13-106.

Examples

The following is an example of the volresize command that resizes volume1 in the data disk group to 20 gigabytes.

Example 12–73   Using the ASMCMD volresize command

```
ASMCMD [+] > volresize -G data -s 20G volume1
```

volset

Purpose

Sets attributes of an Oracle ADVM volume in mounted disk groups.

Syntax and Description

```
volset -G diskgroup [ --usagestring string]
        [--mountpath mount_path ]
        [--primary {hot|cold}] [--secondary {hot|cold}] volume
```

Table 12–73 describes the options for the volset command.

Table 12–73   Options for the volset command

Option	Description
-G diskgroup	Name of the disk group containing the volume.
--usagestring string	Optional usage string to tag a volume which can be up to 30 characters. This string is set to ACFS when the volume is attached to an Oracle ACFS file system and should not be changed.
--mountpath mount_path	Optional string to tag a volume with its mount path string which can be up to 1024 characters. This string is set when the file system is mounted and should not be changed.
--primary {hot|cold}	Intelligent Data Placement specification for primary extents, either hot or cold region.
--secondary {hot|cold}	Intelligent Data Placement specification for secondary extents, either hot or cold region.
volume	Name of the volume.

When running the mkfs command to create a file system, the usage field is set to ACFS and mountpath field is reset to an empty string if it has been set. The usage field should remain at ACFS.

When running the mount command to mount a file system, the mountpath field is set to the mount path value to identify the mount point for the file system. After the value is set by the mount command, the mountpath field should not be updated.

Examples

The following is an example of a volset command that sets the usage string for a volume that is not associated with a file system.

Example 12–74   Using the ASMCMD volset command

```
ASMCMD [+] > volset -G data --usagestring 'no file system created' volume1
```

volstat

Purpose

Reports I/O statistics for Oracle ADVM volumes.

Syntax and Description

```
volstat [-G diskgroup] [volume]
```

Table 12–74 describes the options for the `volstat` command.

Table 12–74 Options for the volstat command

Option	Description
-G *diskgroup*	Name of the mounted disk group containing the volume.
volume	Name of the volume.

The following apply when using the `volstat` command.

- If the disk group is not specified and the volume name is specified, all mounted disk groups are searched for the specified volume name.

- If the disk group name is specified and the volume name is omitted, all volumes are displayed for the named disk group.

- If both the disk group name and the volume name are omitted, all volumes on all disk groups are displayed.

Examples

The following is an example of the `volstat` command that displays information about volumes in the `data` disk group.

Example 12–75 Using the ASMCMD volstat command

```
ASMCMD [+] > volstat -G data
DISKGROUP NUMBER / NAME:  1 / DATA
---------------------------------------
  VOLUME_NAME
    READS           BYTES_READ      READ_TIME       READ_ERRS
    WRITES          BYTES_WRITTEN   WRITE_TIME      WRITE_ERRS
  -----------------------------------------------------------
  VOLUME1
    10085           2290573312      22923           0
    1382            5309440         1482            0
```

13

Oracle ACFS Command-Line Tools

Command-line tools for managing and implementing Oracle Automatic Storage Management Cluster File System (Oracle ACFS), include the following:

- Basic Steps to Manage Oracle ACFS Systems
- Oracle ACFS Command-line Tools for Linux and UNIX Environments
- Oracle ACFS Command-line Tools for the Solaris Environment
- Oracle ACFS Command-line Tools for the AIX Environment
- Oracle ACFS Command-line Tools for Windows Environments
- Oracle ACFS Command-Line Tools for Tagging
- Oracle ACFS Command-Line Tools for Replication
- Oracle ACFS Command-Line Tools for Security
- Oracle ACFS Command-Line Tools for Encryption
- Oracle ACFS Command-Line Utilities for Multiple Environments

---

Note: Oracle ACFS encryption, replication, security, and tagging are available starting with Oracle Automatic Storage Management (Oracle ASM) 11g Release 2 (11.2.0.2) on Linux and Oracle ASM 11g Release 2 (11.2.0.3) on Windows.

Basic Steps to Manage Oracle ACFS Systems

This section provides an outline of the basic steps when managing Oracle ACFS file systems using command-line utilities.

The examples in this section show operating system commands that are run in a Linux environment system. ASMCMD commands manage the volumes, but you can also use SQL*PLus, Oracle ASM Configuration Assistant (ASMCA), and Oracle Enterprise Manager to manage volumes.

This section contains these topics:

- Creating an Oracle ACFS File System
- Accessing an Oracle ACFS File System on a Different Node in the Cluster
- Managing Oracle ACFS Snapshots
- Securing Oracle ACFS File Systems
- Encrypting Oracle ACFS File Systems

- Tagging Oracle ACFS File Systems
- Replicating Oracle ACFS File Systems
- Deregistering, Dismounting, and Disabling Volumes and Oracle ACFS File Systems
- Removing an Oracle ACFS File System and a Volume

Creating an Oracle ACFS File System

To create and verify a file system, perform the following steps:

1. Create an Oracle ASM volume in a mounted disk group with the ASMCMD `volcreate` command.

   The compatibility parameters `COMPATIBLE.ASM` and `COMPATIBLE.ADVM` must be set to 11.2 or higher for the disk group to contain an Oracle ADVM volume. To use Oracle ACFS encryption, replication, security, or tagging, the disk group on which the volume is created for the file system must have compatibility attributes for `ASM` and `ADVM` set to `11.2.0.2`. See "Disk Group Compatibility Attributes" on page 4-34.

   Start ASMCMD connected to the Oracle ASM instance. You must be a user in the OSASM operating system group. See "About Privileges for Oracle ASM" on page 3-26.

   When configuring Oracle ADVM volume devices within a disk group, Oracle recommends assigning the Oracle Grid Infrastructure user and Oracle ASM administrator roles to users who have root privileges.

   To create a volume:

   ```
   ASMCMD [+] > volcreate -G data -s 10G volume1
   ```

   On Linux platforms, the volume name must be less than or equal to eleven alphanumeric characters, starting with an alphabetic character. On AIX platforms, the volume name must be less than or equal to twenty three alphanumeric characters, starting with an alphabetic character. On Windows and Solaris platforms, the volume name must be less than or equal to thirty alphanumeric characters, starting with an alphabetic character.

   When creating an Oracle ASM volume, a volume device name is created that includes a unique Oracle ADVM persistent disk group number. The volume device file functions in the same manner as any other disk or logical volume to mount file systems or for applications to use directly.

   For information about the `volcreate` command, see "volcreate" on page 12-62.

2. Determine the device name of the volume that was created.

   You can determine the volume device name with the ASMCMD `volinfo` command or from the `VOLUME_DEVICE` column in the `V$ASM_VOLUME` view.

   For example:

   ```
   ASMCMD [+] > volinfo -G data volume1
   Diskgroup Name: DATA

           Volume Name: VOLUME1
           Volume Device: /dev/asm/volume1-123
           State: ENABLED
       ...
   ```

```
SQL> SELECT volume_name, volume_device FROM V$ASM_VOLUME
     WHERE volume_name ='VOLUME1';

VOLUME_NAME        VOLUME_DEVICE
----------------   --------------------------------------
VOLUME1            /dev/asm/volume1-123
```

For information about the volinfo command, see "volinfo" on page 12-66.

> **See Also:**   *Oracle Database Reference* for information about the V$ASM_
> VOLUME view

3. Create a file system with the Oracle ACFS mkfs command.

   Create a file system using an existing volume device.

   For example:

   ```
   $ /sbin/mkfs -t acfs /dev/asm/volume1-123

   mkfs.acfs: version              = 11.2.0.1.0.0
   mkfs.acfs: on-disk version      = 39.0
   mkfs.acfs: volume              = /dev/asm/volume1-123
   mkfs.acfs: volume size         = 10737418240
   mkfs.acfs: Format complete.
   ```

   See "mkfs" on page 13-23 (Linux or UNIX) or "acfsformat" on page 13-44
   (Windows). The root privilege is not required. The ownership of the volume
   device file dictates who can run this command.

4. Optionally register the file system with the acfsutil registry command.

   For example:

   ```
   $ /sbin/acfsutil registry -a /dev/asm/volume1-123 /u01/app/acfsmounts/myacfs

   acfsutil registry: mount point /u01/app/acfsmounts/myacfs successfully added
     to Oracle Registry
   ```

   See "acfsutil registry" on page 13-104. The root or asmadmin privileges are required
   to modify the registry. The Windows Administrator privilege is equivalent to the
   root privilege on Linux.

   Registering a file system is optional. After registering an Oracle ACFS file system
   in the cluster mount registry, the file system is mounted automatically on each
   cluster member listed in the registry entry during the next registry check action.
   This automatic process runs every 30 seconds and eliminates the requirement to
   manually mount the file system on each member of the cluster.

   Registering an Oracle ACFS file system also causes the file system to be mounted
   automatically whenever Oracle Clusterware or the system is restarted.

   > **Note:**   In an Oracle Grid Infrastructure Clusterware configuration,
   > you can run srvctl add filesystem to automount a file system; this
   > method is required when an Oracle Database home is installed on an
   > Oracle ACFS file system. However, that file system should not be
   > added to the registry. For information about Server Control Utility
   > (SRVCTL), see *Oracle Real Application Clusters Administration and
   > Deployment Guide*.

For more information, see "About the Oracle ACFS Mount Registry" on page 5-8.

> **Note:** A file system is not automatically mounted for an Oracle Restart configuration, which is a single-instance (non-clustered) environment.

5. Mount the file system with the Oracle ACFS `mount` command. You can mount a file system before or after registering the file system. If the file system has been registered, you can wait for the file system to be mounted automatically.

   For example:

   ```
   # /bin/mount -t acfs /dev/asm/volume1-123 /u01/app/acfsmounts/myacfs
   ```

   See "mount" on page 13-24 (Linux or UNIX) or "acfsmountvol" on page 13-45 (Windows). The `root` privilege is required to run the `mount` command and the Windows `Administrator` privilege is required to run the `acfsmountvol` command.

   After the file system has been mounted, ensure that the permissions are set to allow access to the file system for the appropriate users. For example:

   ```
   # chown -R oracle:dba /u01/app/acfsmounts/myacfs
   ```

6. Create a test file in the file system.

   The user that creates the test file should be a user that is intended to access the file system. This test ensures that the appropriate user can write to the file system.

   For example:

   ```
   $ echo "Oracle ACFS File System" > /u01/app/acfsmounts/myacfs/myfile
   ```

7. List the contents of the test file that was created in the file system.

   For example:

   ```
   $ cat /u01/app/acfsmounts/myacfs/myfile
   Oracle ACFS File System
   ```

Accessing an Oracle ACFS File System on a Different Node in the Cluster

If the node is part of a cluster, perform the following steps on node 2 to view the test file you created on node 1.

> **Note:** If the file system has been registered with the Oracle ACFS mount registry, you can skip steps 1 to 3.

1. Enable the volume that was previously created and enabled on node 1.

   Start ASMCMD connected to the Oracle ASM instance. You must be a user in the OSASM operating system group. See "About Privileges for Oracle ASM" on page 3-26.

   For example:

   ```
   ASMCMD [+] > volenable -G data volume1
   ```

   See "volenable" on page 12-65.

2. View information about the volume that you created on node 1.

For example:

```
ASMCMD [+] > volinfo -G data volume1
```

See "volinfo" on page 12-66.

3. Mount the file system using the Oracle ACFS `mount` command.

For example:

```
# /bin/mount -t acfs /dev/asm/volume1-123 /u01/app/acfsmounts/myacfs
```

See "mount" on page 13-24 (Linux or UNIX) or "acfsmountvol" on page 13-45 (Windows). The `root` privilege is required run the `mount` command and the Windows `Administrator` privilege is required to run the `acfsmountvol` command.

After the file system has been mounted, ensure that the permissions are set to allow access for the appropriate users.

4. List the contents of the test file you previously created on the file system.

For example:

```
$ cat /u01/app/acfsmounts/myacfs/myfile
Oracle ACFS File System
```

The contents should match the file created previously on node 1.

Managing Oracle ACFS Snapshots

To create and verify a snapshot on node 1:

1. Create snapshot of the new file system created on node 1.

For example:

```
$ /sbin/acfsutil snap create mysnapshot_20090725 /u01/app/acfsmounts/myacfs
```

See "acfsutil snap create" on page 13-108.

2. Update the test file in the file system so that it is different than the snapshot.

For example:

```
$ echo "Modifying a file in Oracle ACFS File System" >
/u01/app/acfsmounts/myacfs/myfile
```

3. List the contents of the test file and the snapshot view of the test file.

For example:

```
$ cat /u01/app/acfsmounts/myacfs/myfile
```

```
$ cat /u01/app/acfsmounts/myacfs/.ACFS/snaps/mysnapshot_20090725/myfile
```

The contents of the test file and snapshot should be different. If node 1 is in a cluster, then you can perform the same list operation on node 2.

To manage snapshots with Oracle Enterprise Manager, see "Managing Oracle ACFS Snapshots with Oracle Enterprise Manager" on page 10-5.

Securing Oracle ACFS File Systems

This section discusses the basic operations to manage security for an Oracle ACFS file system on Linux.

The scenario in this section shows how could you use Oracle ACFS security to ensure that only the maintenance user can access medical history files during the maintenance period. Also, Oracle ACFS encryption is also enabled on the same file system.

The scenario in this section assumes you are not planning to use Oracle ACFS replication on the file system. You cannot use Oracle ACFS security with replication.

The disk group on which the volume is created for the file system has compatibility attributes for `ASM` and `ADVM` set to `11.2.0.3`. For information about disk group compatibility, refer to "Disk Group Compatibility" on page 4-33.

The basic steps to manage security are:

1. Initialize security for Oracle ACFS.

Run the `acfsutil sec init` command to configure storage for security credentials and identify an operating system user as the first security administrator and the operating system security group. The security administrator must belong to the operating system group. This command must be run before any other security command and requires root or Windows `Administrator` privileges to run.

The `acfsutil sec init` command is only run once to set up Oracle ACFS security for each cluster and can be run from any node in the cluster. Other security commands can also be run from any node in a cluster. Security administrators are common for all Oracle ACFS file systems in a cluster.

For example, the following command initializes security for a cluster and creates the first security administrator (`medHistAdmin1`).

```
# /sbin/acfsutil sec init -u medHistAdmin1 -g medHistAdminGrp
```

The `medHistAdmin1` security administrator must belong to the `medHistAdminGrp` operating system group. That group is identified as the security group for the security administrators.

When the root user or Windows `Administrator` user runs the command, the user assigns a temporary security password to the first security administrator. The first security administrator should now change the temporary password with the `acfsutil sec admin password` command. The valid password format is described in "acfsutil sec init" on page 13-71.

```
$ /sbin/acfsutil sec admin password
```

Note that all `acfsutil sec` commands (other than `acfsutil sec init`) must be run by an Oracle ACFS security administrator and the administrator is prompted for the security administrator's password when each command is run.

> **Note:** When prompting for the security administrator's password, the following text displays: `Realm management password`
>
> The password required is the Oracle ACFS security administrator's password, not the operating system password of the user.

Security administrators are allowed to browse all directories in an Oracle ACFS file system whether they have the underlying operating system permissions and whether any realm checks allow it. This functionality enables a security administrator to check the location of the files when securing them with Oracle ACFS security realms. However, a security administrator cannot view the contents

of individual files without the appropriate operating system and security realm permissions.

For more information, refer to "acfsutil sec init" on page 13-71 and "acfsutil sec admin password" on page 13-66.

2. Add additional security administrators as necessary.

The first security administrator can add additional security administrators to administer Oracle ACFS security with the `acfsutil sec admin add` command.

For example, add a new security administrator `medHistAdmin2`.

```
$ /sbin/acfsutil sec admin add medHistAdmin2
```

The `medHistAdmin2` user must belong to the operating system group (`medHistAdminGrp`) identified as the security administrator group with the `acfsutil sec init` command.

The `medHistAdmin2` security administrator should change the assigned temporary security password with the `acfsutil sec admin password` command. The `medHistAdmin2` administrator can add new security administrators.

For more information, refer to "acfsutil sec admin add" on page 13-65 and "acfsutil sec admin password" on page 13-66.

3. Prepare an Oracle ACFS file system for security.

Run the `acfsutil sec prepare` on an Oracle ACFS file system before adding any security realms.

For example, prepare the Oracle ACFS file system mounted on `/acfsmounts/acfs1` for Oracle ACFS security.

```
$ /sbin/acfsutil sec prepare -m /acfsmounts/acfs1
```

By default, security is enabled for a file system after running this command. You can explicitly disable or enable security with the `acfsutil sec disable` or `acfsutil sec enable` commands. For more information, refer to "acfsutil sec disable" on page 13-68 and "acfsutil sec enable" on page 13-69.

This command automatically creates several security realms, such as the `SYSTEM_BackupOperators` security realm. Administrators can add users to the `SYSTEM_BackupOperators` realm which gives those users permissions to make backups of realm-secured files in the Oracle ACFS file system.

For more information, refer to "acfsutil sec prepare" on page 13-73.

4. Provide encryption for this file system.

Encrypting the file system is optional, but is enabled in this scenario.

a. First, run the `acfsutil encr init` command to initialize encryption and create the storage necessary for the encryption keys. This command must be run one time for each cluster on which encryption is set up.

For example, the following command initializes encryption for a cluster.

```
# /sbin/acfsutil encr init
```

This command must be run before any other encryption command and requires root or administrator privileges to run. For more information, refer to "acfsutil encr init" on page 13-90.

    b. Next, run the `acfsutil encr set` command to set encryption for the Oracle ACFS file system.

      For example, the following command sets encryption for the file system mounted on the `/acfsmounts/acfs1` directory.

```
# /sbin/acfsutil encr set -m /acfsmounts/acfs1/
```

      The `acfsutil encr set` command transparently generates a volume encryption key which is stored in the key store that was previously configured with the `acfsutil encr init` command. This command requires root or administrator privileges to run.

    For more information, refer to "acfsutil encr init" on page 13-90 and "acfsutil encr set" on page 13-94.

5. Create a security realm on the file system.

    Run the `acfsutil sec realm create` command to create a security realm for a file system.

    For example, create a security realm named `medHistRealm` which contains medical records files with all files encrypted in the realm.

```
$ /sbin/acfsutil sec realm create medHistRealm -m /acfsmounts/acfs1/
                                   -e on -a AES -k 128
```

    The `-e` option specifies that all the files in the realm are encrypted with the `AES` algorithm and the key length set to 128 bits. The file system must first be prepared for encryption with the `acfsutil encr init` and `acfsutil encr set` commands. Note that you do not have to enter the same value for the `-k` option with `acfsutil sec realm create` as you have entered with the `acfsutil encr set` command.

    For more information, refer to "acfsutil sec realm create" on page 13-79.

6. Create security rules.

    Run the `acfsutil sec rule create` command to creates rules which determine access to the files and directories of a security realm.

    For example, create rules that allow the `medMaintenance` user to access medical records for the time period 10 PM to 2 AM for file maintenance.

```
$ /sbin/acfsutil sec rule create medHistRule1a —m /acfsmounts/acfs1/
     —t time 22:00:00,02:00:00 —o ALLOW

$ /sbin/acfsutil sec rule create medHistRule1b —m /acfsmounts/acfs1/
     —t username medMaintenance —o ALLOW
```

    You can edit rules with the `acfsutil sec rule edit` command.

    For more information, refer to "acfsutil sec rule create" on page 13-83 and "acfsutil sec rule edit" on page 13-85.

7. Create security rule sets and add rules to rule sets.

    Run the `acfsutil sec ruleset create` command to create a rule set to which rules can be added.

    For example, create a rule set named `medRuleSet1` that includes rules for operations on the files and directories of the security `medHistRealm` realm.

```
$ /sbin/acfsutil sec ruleset create medRuleSet1 —m /acfsmounts/acfs1/
```

    Add the rules to the `medRuleSet1` rule set.

```
$ /sbin/acfsutil sec ruleset edit medRuleSet1 –m /acfsmounts/acfs1/
          -a medHistRule1a,medHistRule1b -o ALL_TRUE
```

The `ALL_TRUE` option is the default action, but is added here to emphasize that both rules in each rule set must be true.

For more information, refer to "acfsutil sec ruleset create" on page 13-86 and "acfsutil sec ruleset edit" on page 13-87.

8. Add objects to a security realm.

Run the `acfsutil sec realm add` command to add objects, such as command rules, rule sets, and files, to a security realm.

For example, add the `medRuleSet1` rule set and all the files in the `/acfsmounts/acfs1/medicalrecords` directory to the `medHistRealm`.

```
$ /sbin/acfsutil sec realm add medHistRealm –m /acfsmounts/acfs1/
        -l ALL:medRuleSet1
        -f -r /acfsmounts/acfs1/medicalrecords
```

When adding a rule set to a realm, the rule set is added with a command rule, such as `ALL:medRuleSet1`. Only one rule set can be included with each command rule. To display a list of the command rules, use `acfsutil sec info` with the `-c` option. Refer to "acfsutil sec info" on page 13-70.

Add backup operators to the `SYSTEM_BackupOperators` security realm that was automatically created with the `acfsutil sec prepare` command.

```
$ /sbin/acfsutil sec realm add SYSTEM_BackupOperators –m /acfsmounts/acfs1/
        -G sysBackupGrp
```

Users that belong to the `sysBackupGrp` operating system group can now make backups of realm-secured files in the Oracle ACFS file system.

For more information, refer to "acfsutil sec realm add" on page 13-75 and "acfsutil sec realm delete" on page 13-80.

9. Display security information.

Run the `acfsutil sec info` command to display information for a security realm. For example, display security information for the `medHistRealm` realm.

```
$ /sbin/acfsutil sec info -m /acfsmounts/acfs1/ –n medHistRealm
```

To display the security realms to which a file or a directory belongs, run the `acfsutil sec info file` command. For example:

```
$ /sbin/acfsutil sec info file -m /acfsmounts/acfs1/
                              /acfsmounts/acfs1/medicalrecords
```

For more information, refer to "acfsutil sec info" on page 13-70 and "acfsutil sec info file" on page 13-71.

10. Save security metadata as a backup.

Run the `acfsutil sec save` command to save the security metadata of a file system.

For example, save the security metadata of the `/acfsmounts/acfs1` file system to the `acfs1_backup.xml` file.

```
$ /sbin/acfsutil sec save –m /acfsmounts/acfs1
                        –p acfs1_backup.xml
```

The `acfs1_backup.xml` security metadata backup file is saved in the `/acfsmounts/acfs1/.Security/backup/` directory. The saved XML file can be loaded with the `acfsutil sec load` command.

For more information, refer to "acfsutil sec save" on page 13-88 and "acfsutil sec load" on page 13-72.

You can run some `acfsutil sec` commands in a batch file with the `acfsutil sec batch` command. For example, you could create a batch file that contains a group of `acfsutil sec rule` and `acfsutil sec ruleset` commands. For more information, refer to "acfsutil sec batch" on page 13-67.

Auditing and diagnostic data for Oracle ACFS security is saved to log files. For more information about Oracle ACFS security, including the log files, refer to "Oracle ACFS Security" on page 5-13.

Encrypting Oracle ACFS File Systems

This section discusses the basic operations to manage encryption on an Oracle ACFS file system on Linux. The examples in this section show a scenario in which the medical history files are encrypted in an Oracle ACFS file system.

The steps in this section assume Oracle ACFS security is not configured for the file system; however, you can use both Oracle ACFS security and encryption on the same file system. If you decide to use both security and encryption, then both encryption and security must be initialized for the cluster containing the file system. After security is initialized on the file system, then an Oracle ACFS security administrator runs `acfsutil sec` commands to provide encryption for the file system. For information about setting up security with encryption, refer to "Securing Oracle ACFS File Systems" on page 13-5.

The steps in this section assume you are not planning to use Oracle ACFS replication on the file system. You cannot use Oracle ACFS encryption with replication.

Because the `acfsutil encr set` and `acfsutil encr rekey -v` commands modify the encryption key store, you should back up the Oracle Cluster Registry (OCR) after running these commands to ensure there is an OCR backup that contains all of the volume encryption keys (VEKs) for the file system.

The disk group on which the volume is created for the file system has compatibility attributes for `ASM` and `ADVM` set to `11.2.0.3`. For information about disk group compatibility, refer to "Disk Group Compatibility" on page 4-33.

The basic steps to manage encryption are:

1. Initialize encryption.

   Run the `acfsutil encr init` command to initialize encryption and create the storage necessary for the encryption keys. This command must be run one time for each cluster on which encryption is set up.

   For example, the following command initializes encryption for a cluster.

   ```
   # /sbin/acfsutil encr init
   ```

   This command must be run before any other encryption command and requires root or administrator privileges to run.

   For more information, refer to "acfsutil encr init" on page 13-90.

2. Set encryption parameters.

Run the `acfsutil encr set` command to set the encryption parameters for the entire Oracle ACFS file system.

For example, the following command sets the `AES` encryption algorithm and a file key length of `128` for a file system mounted on the `/acfsmounts/acfs1` directory.

```
# /sbin/acfsutil encr set -a AES -k 128 -m /acfsmounts/acfs1/
```

The `acfsutil encr set` command also transparently generates a volume encryption key which is stored in the key store that was previously configured with the `acfsutil encr init` command.

This command requires root or administrator privileges to run.

For more information, refer to "acfsutil encr set" on page 13-94.

3. Enable encryption.

Run the `acfsutil encr on` command to enable encryption for directories and files.

For example, the following command enables encryption recursively on all files in the `/acfsmounts/acfs1/medicalrecords` directory.

```
# /sbin/acfsutil encr on -r /acfsmounts/acfs1/medicalrecords
                         -m /acfsmounts/acfs1/
```

For users that have appropriate permissions to access files in the `/acfsmounts/acfs1/medicalrecords` directory, they can still read the decrypted files.

This command can be run by an administrator or the file owner.

For more information, refer to "acfsutil encr on" on page 13-92.

4. Display encryption information.

Run the `acfsutil encr info` command to display encryption information for directories and files.

```
# /sbin/acfsutil encr info -m /acfsmounts/acfs1/
                            -r /acfsmounts/acfs1/medicalrecords
```

This command can be run by an administrator or the file owner.

For more information, refer to "acfsutil encr info" on page 13-89.

Auditing and diagnostic data for Oracle ACFS encryption is saved to log files. For more information about Oracle ACFS encryption, including the log files, refer to "Oracle ACFS Encryption" on page 5-15.

Tagging Oracle ACFS File Systems

This section discusses the operations to manage tagging on directories and files in an Oracle ACFS file system on Linux.

The disk group on which the volume is created for the file system has compatibility attributes for `ASM` and `ADVM` set to `11.2.0.3`. For information about disk group compatibility, refer to "Disk Group Compatibility" on page 4-33.

Oracle ACFS implements tagging with Extended Attributes. There are some requirements for using Extended Attributes; refer to "Oracle ACFS Tagging" on page 5-19.

The steps to manage tagging are:

1. Specify tag names for directories and files.

   Run the `acfsutil tag set` command to set tags on directories or files. You can use these tags to specify which objects are replicated.

   For example, add the `comedy` and `drama` tags to the files in the subdirectories of the `/acfsmounts/repl_data/films` directory.

   ```
   $ /sbin/acfsutil tag set -r comedy /acfsmounts/repl_data/films/comedies
   ```

   ```
   $ /sbin/acfsutil tag set -r drama /acfsmounts/repl_data/films/dramas
   ```

   ```
   $ /sbin/acfsutil tag set -r drama /acfsmounts/repl_data/films/mysteries
   ```

   In this example, the `drama` tag is purposely used twice and that tag is changed in a later step.

   You must have system administrator privileges or be the file owner to run this command.

   For more information, refer to "acfsutil tag set" on page 13-51.

2. Display tagging information.

   Run the `acfsutil tag info` command to display the tag names for directories or files in Oracle ACFS file systems. Files without tags are not be displayed.

   For example, display tagging information for files in the `/acfsmounts/repl_data/films` directory.

   ```
   $ /sbin/acfsutil tag info -r /acfsmounts/repl_data/films
   ```

   Display tagging information for files with the `drama` tag in the `/acfsmounts/repl_data/films` directory.

   ```
   $ /sbin/acfsutil tag info -t drama -r /acfsmounts/repl_data/films
   ```

   You must have system administrator privileges or be the file owner to run this command.

   For more information, refer to "acfsutil tag info" on page 13-50.

3. Remove and change tag names if necessary.

   Run the `acfsutil tag unset` command to remove tags on directories or files. For example, unset the `drama` tag on the files in the `mysteries` subdirectory of the `/acfsmounts/repl_data/films` directory to apply a different tag to the subdirectory.

   ```
   $ /sbin/acfsutil tag unset -r drama /acfsmounts/repl_data/films/mysteries
   ```

   Add the `mystery` tag to the files in the `mysteries` subdirectory of the `/acfsmounts/repl_data/films` directory.

   ```
   $ /sbin/acfsutil tag set -r mystery /acfsmounts/repl_data/films/mysteries
   ```

   You must have system administrator privileges or be the file owner to run these commands.

   For more information, refer to "acfsutil tag unset" on page 13-52.

For more information about tagging an Oracle ACFS file system, refer to "Oracle ACFS Tagging" on page 5-19.

Replicating Oracle ACFS File Systems

This section discusses the operations to manage replication on an Oracle ACFS file system on Linux.

The steps in this section assume you are not planning to use Oracle ACFS security or encryption on the file system. You cannot use Oracle ACFS replication with security or encryption.

The disk groups on which volumes are created for the primary and standby file systems must have compatibility attributes for ASM and ADVM set to 11.2.0.3. For information about disk group compatibility, refer to "Disk Group Compatibility" on page 4-33.

The steps to manage replication are:

1. Determine the storage capacity necessary for replication on the sites hosting the primary and standby file systems. The primary file system must have a minimum size of 4 GB for each node that is mounting the file system. The standby file system must have a minimum size of 4 GB and should be sized appropriately for the amount of data being replicated and the space necessary for the replication logs sent from the primary file system.

   Calculate the replication-related storage requirement for the primary file system, then use the same size requirement for the standby file system. If Oracle ACFS tagging is used to replicate only a subset of the files in the primary file system, then the size requirement for the standby file system is proportional to that subset of the primary file system.

   Run the `acfsutil info fs` command with the `-s interval` option on the node where the primary file system is mounted to display the amount and rate of change to the primary file system for the node. The amount of change includes all user and metadata modifications to the primary file system. This amount approximates the size of replication logs that are generated when recording changes to the file system. Changes are stored in temporary files called replication logs which are kept in a special directory in the primary file system until they can be sent to the standby to be applied. After confirmation is received that the changes contained in a replication log have been successfully applied to the standby file system, the replication logs on the primary file system are deleted.

   To approximate the extra storage capacity necessary for the replication logs, determine the following:

   - The time interval during which the site hosting the primary file system may experience network connectivity problems or slowdowns when accessing the site hosting the standby file system.

   - The time interval during which the site hosting the standby file system may be taken offline for maintenance.

   These time intervals are used in calculating the amount and rate of change in storage space. You must account for the time interval when the primary file system cannot send the replication logs over to the standby file system at its usual rate or when standby file systems are inaccessible while undergoing maintenance. The replication logs will accumulate on the site hosting the primary file system and may eventually cause that site to run out of space.

   For the following scenario, assume $t = 60$ minutes is the time interval in your environment that would adequately account for network problems or maintenance on site hosting the standby file system.

Run `acfsutil info fs -s 900` on the primary file system to collect the average rate of change over a 24 hour period with a 15 minute (900 seconds) interval. Note that `t/4` (60/4) is the value for the sampling interval. Do not exceed a value of `t/2` for the time interval as you may miss some important peaks.

```
$ /sbin/acfsutil info fs -s 900 /acfsmounts/repl_data
```

With the output, you can determine the average rate of change, the peak rate of change, and how long the peaks last. However, the command displays information only for the node on which the command is run. To collect the total amount of change in the file system the command must be run on every node that is modifying the file system. Note that the maximum number of supported nodes is eight.

The following formula approximates the extra storage capacity needed:

```
Extra storage capacity to hold replication logs =
    (Number-nodes-on-primary * 1GB) + P
```

where `P` is the peak amount of change generated across all nodes for time `t` as reported by the `acfsutil info fs -s` output.

In the example, we need to total the changes from four 15-minute intervals to find the total amount of change that could occur in 60 minutes. You may choose to use the single hour that generated the largest amount of change, or you could select the top four 15-minute intervals even if they did not occur together to prepare for the worst-case scenario.

Assume that you have four nodes modifying the primary file system, and that during the measured interval, the peak amount of change reported for the 60 minutes is approximately 20 GB for all nodes. Using the storage capacity formula, 24 GB of excess storage capacity on each site hosting the primary file system is required for storage of the replication logs.

```
Extra storage capacity to hold replication logs = (4 * 1GB per node) +
 20GB maximum change per hour = 24GB of extra storage capacity
```

Next, check that the network transfer rate is greater than or equal to the rate of change observed during the monitoring period. In the previous example, the peak of 20 GB of changed data per hour is equivalent to a peak rate of change of about 5.5 MB/sec. To keep up with this rate of change, you must ensure that the network can reliably transfer at least this amount of data per second without negatively impacting your existing network workloads.

To estimate your current actual network transfer rate, calculate the elapsed time required to FTP a 1 GB file from the primary file system to the intended standby file system during a period of time when network usage is low. For example, if the 1 GB file transfers in 30 seconds, then your current FTP transfer rate is 33 MB per seconds (1000 MB/30 seconds = 33 MB per second). Because of various delays inherent in the transfers, for planning purposes you should reduce this measured FTP transfer rate by 20%, and then by an additional 5% per node.

In the previous example with 4 nodes, the FTP transfer rate used for planning is:

```
33 MB/sec * (1 − (0.2) − (4 * 0.05)) = 33 * (0.6) = ~20MB/sec
```

Because the peak rate of change was only 5.5 MB/second, you can expect the network to be able to handle this additional workload in this example. However, if the network capacity was already close to being fully utilized, you might want to consider increasing network capacity before implementing replication for this file system and workload.

In addition, insure you have sufficient network capacity to allow replication to catch up after times when network problems prevent a primary file system from sending replication logs to the standby file system.

For more information, refer to "acfsutil info fs" on page 13-99.

2. Set up tags, user names, and service names.

When starting replication on an Oracle ACFS file system, first perform the following steps:

- Determine the user name and password that the sites hosting the primary and standby file systems use to connect to the remote Oracle ASM instance as the Oracle ASM and DBA administrator. All nodes that have the file system mounted must support this user name and password. The user must have SYSASM and SYSDBA privileges. For example:

  ```
  SQL> CREATE USER primary_admin IDENTIFIED BY primary_passwd;
  SQL> GRANT sysasm,sysdba TO primary_admin;
  ```

  Oracle wallets can also be used to manage security credentials.

  See Also:

  - *Oracle Database Advanced Security Administrator's Guide* for information about Oracle wallets
  - *Oracle Database SecureFiles and Large Objects Developer's Guide* for information about wallet management
  - *Oracle Database Net Services Reference* for information about wallet parameters in the SQLNET.ORA file

- Determine a unique service name for the replicated file system.

  When both the primary and standby file systems are located in different clusters for disaster tolerance, then the service names for the primary and standby file systems can be the same. However, if the both file systems are mounted on the same node, such as a test configuration, then unique service names must be used for the primary and standby file systems. Using unique service names for the primary and standby file systems requires the use of the -c option during replication initialization. Service names are limited to a maximum of 128 bytes.

  Notes:

  - You must specify a service name other than +ASM because that service name is already in use by the Oracle ASM instance.
  - You must specify a unique service name for each file system that you want to replicate when there are multiple replicated file systems on a node or cluster.

  Using this service name, create a net service alias on the sites hosting the primary and standby file system that connects to the remote site. This alias along with the user name and password are used as the connection string in the replication initialization commands.

For example, the following are examples of connect descriptors with net service aliases for the sites hosting the primary and standby file systems.

```
primary_repl_site=(DESCRIPTION=
  (ADDRESS=(PROTOCOL=tcp)(HOST=primary1.example.com)(PORT=1521))
  (ADDRESS=(PROTOCOL=tcp)(HOST=primary2.example.com)(PORT=1521))
  (CONNECT_DATA=(SERVICE_NAME=primary_service)))

standby_repl_site=(DESCRIPTION=
  (ADDRESS=(PROTOCOL=tcp)(HOST=standby1.example.com)(PORT=1521))
  (CONNECT_DATA=(SERVICE_NAME=standby_service)))
```

If you want to perform replication using a single client access name (SCAN) VIP, you must update the `REMOTE_LISTENER` initialization parameter in the Oracle ASM instance before initializing replication. You can update the parameter in the initialization file or with the `ALTER SYSTEM` SQL statement.

For example:

```
SQL> ALTER SYSTEM SET remote_listener='SCAN_NAME:1521' sid='*' scope=both;
```

> **See Also:** *Oracle Database Net Services Administrator's Guide* for information about connect descriptors

- Optionally set tags on directories and files to replicate only selected files in an Oracle ACFS file system. You can also add tags to files after replication has already started. For information about the steps to tag files, refer to "Tagging Oracle ACFS File Systems" on page 13-11.

3. Configure the site hosting the standby file system.

   Before replicating an Oracle ACFS file system, configure the site hosting the standby file system by performing the following procedures.

   - Create a new file system of adequate size to hold the replicated files and associated replication logs from the primary file system. For example: `/standby/repl_data`

   - Mount the file system on one node only.

   - Run the `acfsutil repl init standby` command. If this command is interrupted for any reason, the user must re-create the file system, mount it on one node only, and rerun the command. This command requires the following configuration information:

     – The connect string to be used to connect to the site hosting the primary file system. For example:

       `primary_admin/primary_passwd@primary_repl_site`

       The user `primary_admin` must have `SYSASM` and `SYSDBA` privileges.

     – If the standby file system is using a different service name than the primary file system, then the use `-c` option. This option specifies the service name for the standby file system. For example:

       `standby_repl_service`

     – The mount point of the standby file system. For example:

       `/standby/repl_data`

For example, run the following `acfsutil repl init standby` command on the site hosting the standby file system.

```
$ /sbin/acfsutil repl init standby
    -p primary_admin/primary_passwd@primary_repl_site
    -c standby_repl_service /standby/repl_data
```

The `acfsutil repl init standby` command requires root or system administrator privileges to run.

For more information, refer to "acfsutil repl init" on page 13-57.

4. Configure the site hosting the primary file system.

   After the standby file system has been set up, configure the site hosting the primary file system and start replication by performing the following procedures.

   Run the `acfsutil repl init primary` command. This command requires the following configuration information:

   ■ The connect string to be used to connect to the site hosting the standby file system. For example:

   ```
   standby_admin/standby_passwd@standby_repl_site
   ```

   The user `standby_admin` must have `SYSASM` and `SYSDBA` privileges.

   ■ The mount point of the primary file system. For example: `/acfsmounts/repl_data`

   ■ If the primary file system is using a different service name than the standby file system, then use the `-c` option. This option specifies the service name on the site hosting the primary file system. For example:

   ```
   primary_repl_service
   ```

   ■ If the mount point is different on the site hosting the standby file system than it is on the site hosting the primary file system, specify the mount point on the standby file system with the `-m standby_mount_point` option. For example:

   ```
   -m /standby/repl_data
   ```

   For example, run the following `acfsutil repl init primary` command on the site hosting the primary file system.

   ```
   $ /sbin/acfsutil repl init primary
       -s standby_admin/standby_passwd@standby_repl_site
       -m /standby/repl_data -c primary_repl_service
       /acfsmounts/repl_data
   ```

   The `acfsutil repl init primary` command requires root or system administrator privileges to run.

   For more information, refer to "acfsutil repl init" on page 13-57.

5. Monitor information about replication on the file system.

   The `acfsutil repl info` command displays information about the state of the replication processing on the primary or standby file system.

   For example, run the following `acfsutil repl info` command on the site hosting the primary file system to display configuration information.

   ```
   $ /sbin/acfsutil repl info -c -v /acfsmounts/repl_data
   ```

You must have system administrator or Oracle ASM administrator privileges to run this command.

For information, refer to "acfsutil repl info" on page 13-56.

6. Manage replication background processes.

   Run the `acfsutil repl bg` command to start, stop, or retrieve information about replication background processes.

   For example, the following example displays information about the replication processes for the `/acfsmounts/repl_data` file system.

   ```
   $ /sbin/acfsutil repl bg info /acfsmounts/repl_data
   ```

   You must have system administrator or Oracle ASM administrator privileges to run the `acfsutil repl bg info` command.

   For more information, refer to "acfsutil repl bg" on page 13-53.

7. Pause replication momentarily only if necessary.

   Run the `acfsutil repl pause` to momentarily stop replication. You should run the `acfsutil repl resume` command soon as possible to resume replication.

   For example, the following command pauses replication on the `/acfsmounts/repl_data` file system.

   ```
   $ /sbin/acfsutil repl pause /acfsmounts/repl_data
   ```

   The following command resumes replication on the `/acfsmounts/repl_data` file system.

   ```
   $ /sbin/acfsutil repl resume /acfsmounts/repl_data
   ```

   You must have system administrator or Oracle ASM administrator privileges to run these commands.

   For more information, refer to "acfsutil repl pause" on page 13-59 and "acfsutil repl resume" on page 13-60.

   > **Note:** On an Oracle ACFS file system `df` reports space usage by internal metadata plus user files and directories. `du` only reports the space usage of user files and directories. Depending on the size of the volume and number of the nodes, internal metadata is allocated in varying sizes. Additionally, with replication enabled an internal replication log is allocated for each node that is used to record changes to the file system before exposing the replication log to user space daemons to transport to the standby.

   For more information about replicating an Oracle ACFS file system, refer to "Oracle ACFS Replication" on page 5-17.

Deregistering, Dismounting, and Disabling Volumes and Oracle ACFS File Systems

This section discusses the operations to deregister or dismount a file system and disable a volume. This section contains these topics:

- Deregistering an Oracle ACFS File System
- Dismounting an Oracle ACFS File System

- [Disabling a Volume](#)

Deregistering an Oracle ACFS File System

You can deregister an Oracle ACFS file system if you do not want the file system to be automatically mounted.

For example:

```
$ /sbin/acfsutil registry -d /u01/app/acfsmounts/myacfs
```

If you deregister a file system, then you must explicitly mount the file system after Oracle Clusterware or the system is restarted.

For more information about the registry, see "About the Oracle ACFS Mount Registry" on page 5-8. For information about acfsutil registry, see "acfsutil registry" on page 13-104.

Dismounting an Oracle ACFS File System

You can dismount a file system without deregistering the file system or disabling the volume on which the file system is mounted.

For example, you can dismount a file system and run fsck to check the file system.

```
# /bin/umount /u01/app/acfsmounts/myacfs
```

```
# /sbin/fsck -a -v -y -t acfs /dev/asm/volume1-123
```

After you dismount a file system, you must explicitly mount the file system.

Use umount on Linux systems or acfsdismount on Windows systems. For information about the commands to dismount a file system, see "umount" on page 13-26 or "acfsdismount" on page 13-43.

Use fsck on Linux systems or acfschkdsk on Windows systems to check a file system. For information about the commands to check a file system, see "fsck" on page 13-21 or "acfschkdsk" on page 13-41.

Disabling a Volume

To disable a volume, you must first dismount the file system on which the volume is mounted.

For example:

```
# /bin/umount /u01/app/acfsmounts/myacfs
```

After a file system is dismounted, you can disable the volume and remove the volume device file.

For example:

```
ASMCMD> voldisable -G data volume1
```

Dismounting the file system and disabling a volume does not destroy data in the file system. You can enable the volume and mount the file system to access the existing data. For information about voldisable and volenable, see "voldisable" on page 12-64 and "volenable" on page 12-65.

Removing an Oracle ACFS File System and a Volume

To permanently remove a volume and Oracle ACFS file system, perform the following steps. These steps destroy the data in the file system.

1. Deregister the file system with `acfsutil registry -d`.

   For example:

   ```
   $ /sbin/acfsutil registry -d /oracle/acfsmounts/acfs1
   acfsutil registry: successfully removed ACFS mount point
      /oracle/acfsmounts/acfs1 from Oracle Registry
   ```

   For information about running `acfsutil registry`, see "acfsutil registry" on page 13-104.

2. Dismount the file system.

   For example:

   ```
   # /bin/umount /oracle/acfsmounts/acfs1
   ```

   You must dismount the file system on all nodes of a cluster.

   Use `umount` on Linux systems or `acfsdismount` on Windows systems. For information about running `umount` or `acfsdismount`, see "umount" on page 13-26 or "acfsdismount" on page 13-43.

3. Remove the file system with `acfsutil rmfs`.

   If you were not planning to remove the volume in a later step, this step is necessary to remove the file system. Otherwise, the file system is removed when the volume is deleted.

   For example:

   ```
   $ /sbin/acfsutil rmfs /dev/asm/volume1-123
   ```

   For information about running `acfsutil rmfs`, see "acfsutil rmfs" on page 13-106.

4. Optionally you can disable the volume with the ASMCMD `voldisable` command.

   For example:

   ```
   ASMCMD> voldisable -G data volume1
   ```

   For information about running `voldisable`, see "voldisable" on page 12-64.

5. Delete the volume with the ASMCMD `voldelete` command.

   For example:

   ```
   ASMCMD> voldelete -G data volume1
   ```

   For information about running `voldelete`, see "voldelete" on page 12-64.

Oracle ACFS Command-line Tools for Linux and UNIX Environments

Table 13–1 contains a summary of the Oracle ACFS commands for Linux and UNIX. The commands in Table 13–1 have been extended with additional options to support Oracle ACFS. All other Linux and UNIX file system commands operate without change for Oracle ACFS.

For example, Oracle ACFS adds a set of Oracle ACFS-specific mount options to those provided with the base operating system platform. You should review both the mount

options for the Linux and UNIX platforms in addition to the Oracle ACFS-specific options for the complete set of file system mount options.

File systems on Oracle ADVM volumes that are not Oracle ACFS file systems, such as ext3, are managed with the same Linux commands that are listed in Table 13–1 using the file-specific options for the type of file system. You can refer to the man pages for options available for the Linux commands in Table 13–1.

Note: Security-Enhanced Linux (SELinux) is not supported on Oracle ACFS file systems.

Table 13–1 Summary of Oracle ACFS commands for Linux and UNIX

Command	Description
fsck	Checks and repairs an Oracle ACFS file system.
mkfs	Creates an Oracle ACFS file system.
mount	Mounts an Oracle ACFS file system.
umount	Dismounts an Oracle ACFS file system.

fsck

Purpose

Checks and repairs an Oracle ACFS file system.

Syntax and Description

```
fsck -t acfs -h /dev/null
fsck [-a|-f] [-v] [-vv] -t acfs [-n|-y] volume_device
```

fsck -t acfs -h dev/null displays usage text and exits.

Table 13–2 contains the options available with the fsck command.

Table 13–2 Options for the Linux fsck command

Option	Description
-a	Specifies to automatically repair the file system.
-f	Forces the file system into mountable state without completing a file system check or repair.
-v	Specifies verbose mode. The progress is displayed as the operation occurs.
-vv	Specifies very verbose mode. The progress is displayed as the operation occurs.
-h	Displays the usage help text and exits.
-n	Answers no to any prompts.
-y	Answers yes to any prompts.
volume_device	Specifies the primary Oracle ADVM volume device.

fsck checks and repairs an existing Oracle ACFS. This command can only be run on a dismounted file system. root privileges are required to run fsck. The Oracle ACFS driver must be loaded for fsck to work.

By default, fsck only checks for and reports any errors. The -a flag must be specified to instruct fsck to repair errors in the file system.

In a few cases, fsck prompts for questions before proceeding to check a file system. These cases include:

- If fsck detects that another fsck is in progress on the file system
- If fsck detects that the Oracle ACFS driver is not loaded
- If the file system does not appear to be Oracle ACFS

In checking mode, fsck also prompts if there are transaction logs that have not been processed completely due to an incomplete shutdown. To run in a non-interactive mode, include either the -y or -n options to answer yes or no to any questions.

fsck creates working files before it checks a file system. These working files are created in /usr/tmp if space is available. /tmp is used if /usr/tmp does not exist. If insufficient space is available in the tmp directory, fsck attempts to write to the current working directory. The files that fsck creates are roughly the size of the file system being checked divided by 32K. At most two such files are allocated. For example, a 2 GB file system being checked causes fsck to generate one or two 64K working files in the /usr/tmp directory. These files are deleted after fsck has finished.

In the event that fsck finds a file or directory in the file system for which it cannot determine its name or intended location (possibly due to a corruption in its parent directory), it places this object in the /lost+found directory when fsck is run in fix mode. For security reasons only the root user on Linux can read files in /lost+found. If the administrator can later determine the original name and location of the file based on its contents, the file can be moved or copied into its intended location.

The file names in the /lost+found directory are in the following formats:

```
parent.id.file.id.time-in-sec-since-1970
parent.id.dir.id.time-in-sec-since-1970
```

The id fields are the internal Oracle ACFS numeric identifiers for each file and directory in the file system.

You can use acfsutil info id id mount_point to attempt to determine the directory associated with parent.id. This directory is assumed to be where the deleted object originated. For information about acfsutil info, see "acfsutil info file" on page 13-99.

If the parent directory is not known, the parent id field is set to UNKNOWN.

> **Note:** It is not possible to see the contents of the /lost+found directory from a snapshot.

Examples

The following example shows how to check and repair an Oracle ACFS file system.

Example 13–1 Using the fsck command

```
# /sbin/fsck -a -v -y -t acfs /dev/asm/volume1-123
```

mkfs

Purpose
Creates an Oracle ACFS file system.

Syntax and Description
```
mkfs -t acfs -h
mkfs [-v] [-f] -t acfs [-b blocksize] [-n name] volume_device [size]
```

mkfs -t acfs -h displays usage text and exits.

Table 13–3 contains the options available with the mkfs command.

Table 13–3   Options for the Linux mkfs command

Option	Description
-t acfs	Specifies the type of file system on Linux. acfs designates the Oracle ACFS type.
-v	Specifies verbose mode. The progress is displayed as the operation occurs.
-n name	Specifies the name for the file system. A name can be a maximum of 64 characters. acfsutil info fs returns the name if one was specified.
-f	Specifies the force option. This action creates the file system even if there is an existing Oracle ACFS on the volume device, although only if the file system is dismounted. This option overwrites structures on the original file system. Use this option with caution.
-h	Displays the usage help text and exits.
-b blocksize	The default block size is 4K and this is the only size supported in 11g Release 2 (11.2).
size	Specifies the size that the file system should consume on the named device. The quantity specified can be in units of K (kilobytes), M (megabytes), G (gigabytes), T (terabytes), or P (petabytes). If a unit is not specified, the default is bytes.
	If the size specified is not a multiple of the block size, then the value is rounded up to the closest multiple. If this option is not specified, the entire device is consumed.
volume_device	Specifies an Oracle ADVM device file that is to be formatted.

mkfs is used to create the on disk structure needed for Oracle ACFS file system to be mounted. The mkfs command is the traditional UNIX command used to build a file system. After mkfs runs successfully, the USAGE column in the V$ASM_VOLUME view displays ACFS. root privilege is not required. The ownership of the volume device file dictates who can run this command. The minimum file system size is 200 MB. The Oracle ACFS driver must be loaded for mkfs to work.

Examples
Before creating an Oracle ACFS file system, first determine which Oracle ADVM volume devices are available. You can use the ASMCMD volinfo command to display information about the volumes and volume devices.

```
ASMCMD [+] > volinfo -a
...
```

```
                  Volume Name: VOLUME1
                  Volume Device: /dev/asm/volume1-123
                  State: ENABLED
...
```

See "volinfo" on page 12-66.

Next create an Oracle ACFS file system on the volume device file.

Example 13–2   Using the mkfs command

```
$ /sbin/mkfs -t acfs /dev/asm/volume1-123
```

mount

Purpose
Mounts an Oracle ACFS file system.

Syntax and Description
```
mount -h
mount [-v] -t acfs [-o options] volume_device mount_point
mount
```

`mount -h` displays usage text and exits.

Table 13–4 contains the options available with the `mount` command.

Table 13–4   Options for the Linux mount command

Option	Description
-h	Displays the usage help text and exits.
-t acfs	Specifies the type of file system on Linux. acfs designates the Oracle ACFS type.
-v	Specifies verbose mode. The progress is displayed as the operation occurs.

Table 13–4   (Cont.) Options for the Linux mount command

Option	Description
-o	Options are specified with the -o flag followed by a comma-delimited string of options. The following options are available:
	■ all
	Reads the Oracle ACFS mount registry created with the acfsutil registry command and mounts the file systems in it. A mount -t acfs -o all command is automatically run at Oracle ACFS startup.
	When the -o all option is specified, other -o options are ignored. To specify mount options for a registry entry, include those options with the acfsutil registry command when you add the entry to the registry.
	■ ro
	Mounts the file system in read-only mode.
	■ norootsuid
	Fails the execution of binaries by non-root users whose permissions allow set user Id execution, and are owned by root. An attempt to run these executables as a non-root user fails with a permission denied error.
	■ rootsuid
	Allows the execution of binaries by non-root users of set user Id files owned by root. This is the default action.
volume_device	Specifies an Oracle ADVM volume device file that has been formatted by mkfs. *device* is required but can be a dummy value.
mount_point	Specifies the directory where the file system is mounted. This directory must exist before you run the mount command.

mount attaches a file system to the Oracle ACFS hierarchy at the mount point that is the name of a directory. The mount happens on the node where the mount command was issued. The mount command returns an error if the file system is not in a dismounted state on this node.

It is not always possible to return the cause of a mount failure to the mount command. When this happens Oracle ACFS writes the cause of the failure to the system console and associated system log file.

After mount runs successfully, the MOUNTPATH field in the V$ASM_VOLUME view displays the directory name on which the file system is now mounted.

An Oracle ACFS file system should only be mounted on one mount point. The same mount point name should be used on all cluster members.

The mount command lists all mounted file systems if it is run with no parameters.

root privilege is required to run mount.

Examples

The first example shows how to mount volume1-123 on the mount point /u01/app/acfsmounts/myacfs. The second example shows how to mount all the registered Oracle ACFS file systems. The dummy names (none) have been entered for the device and directory as they are required, but not used, when the all option is specified.

Example 13–3   Using the mount command

```
# /bin/mount -t acfs /dev/asm/volume1-123 /u01/app/acfsmounts/myacfs

# /bin/mount -t acfs -o all none none
```

umount

Purpose
Dismounts an Oracle ACFS file system.

Syntax and Description

```
umount -h
umount [-v] volume_device |mount_point
umount -a [-t acfs]
```

umount -h displays usage text and exits.

Table 13–5 contains the options available with the umount command.

Table 13–5   Options for the Linux umount command

Option	Description
-h	Displays the usage help text and exits.
-t acfs	Specifies the type of file system on Linux. acfs designates the Oracle ACFS type.
-v	Specifies verbose mode. The progress is displayed as the operation occurs.
-a	Specifies to dismount all Oracle ACFS file systems on this node.
volume_device	Specifies an Oracle ADVM volume device file that has been formatted by mkfs.
mount_point	Specifies the directory where the file system is mounted. This directory must exist before you run the mount command.

umount detaches an Oracle ACFS from the file system hierarchy on the current node. If the file system is busy, umount fails.

root privileges are required to run the umount command.

Examples
The following examples show how to dismount an Oracle ACFS file system. The first example uses the volume device file and the second example uses the file system.

Example 13–4   Using the umount command

```
# /bin/umount /dev/asm/volume1-123

# /bin/umount /u01/app/acfsmounts/myacfs
```

Oracle ACFS Command-line Tools for the Solaris Environment

> **Note:** If a disk group is dismounted with the force option on Solaris, any Oracle ADVM volumes device files that were on that disk group remain on the system. These files are removed when the disk group is remounted.

Table 13–6 contains a summary of the Oracle ACFS commands for Solaris. The commands in Table 13–6 have been extended with additional options to support Oracle ACFS on Solaris.

Table 13–6   Summary of Oracle ACFS commands for Solaris

Command	Description
fsck	Checks and repairs an Oracle ACFS file system on Solaris.
mkfs	Creates an Oracle ACFS file system on Solaris.
mount	Mounts an Oracle ACFS file system on Solaris.
umount/umountall	Dismounts an Oracle ACFS file system on Solaris.

fsck

Purpose
Checks and repairs an Oracle ACFS file system on the Solaris operating system.

Syntax and Description
```
fsck -F acfs -o h /dev/null
fsck -F acfs [{-n|N}|{-y|Y}] [-o options] volume_device
```

fsck -F acfs -o h /dev/null displays usage text and exits.

Table 13–7 contains the options available with the fsck command.

Table 13–7   Options for the Solaris fsck command

Option	Description
-F acfs	Specifies the type of file system on Solaris. acfs designates the Oracle ACFS type.
-n |N	Answers no to any prompts.
-y|Y	Answers yes to any prompts.

Table 13–7   (Cont.) Options for the Solaris fsck command

Option	Description
-o	Specifies that options follow (a, f, h, v). Options are preceded with the -o flag and entered as a comma-delimited string. For example: -o a,v
	- a Specifies to automatically fix the file system. - f Forces the file system into mountable state without completing a file system check or fix. - h Displays the usage text and exits. - v Specifies verbose mode. The progress is displayed as the operation occurs. - vv Specifies very verbose mode. The progress is displayed as the operation occurs.
volume_device	Specifies an Oracle ADVM device file.

fsck checks and repairs an existing Oracle ACFS file system. This command can only be run on a dismounted file system. root privileges are required to run fsck. The Oracle ACFS driver must be loaded for fsck to work.

By default, fsck only checks for and reports any errors. The -o a option must be specified to instruct fsck to fix errors in the file system.

In a few cases, fsck prompts for questions before proceeding to check a file system. These cases include:

- If fsck detects that another fsck is in progress on the file system
- If fsck detects that the Oracle ACFS driver is not loaded
- If the file system does not appear to be Oracle ACFS

In checking mode, fsck also prompts if there are transaction logs that have not been processed completely due to an incomplete shutdown. To run in a non-interactive mode, include either the -y or -n options to answer yes or no to any questions.

fsck creates working files before it checks a file system. These working files are created in /usr/tmp if space is available. /tmp is used if /usr/tmp does not exist. If insufficient space is available in the tmp directory, fsck attempts to write to the current working directory. The files that fsck creates are roughly the size of the file system being checked divided by 32K. At most two such files are allocated. For example, a 2 GB file system being checked causes fsck to generate one or two 64K working files in the /usr/tmp directory. These files are deleted after fsck has finished.

In the event that fsck finds a file or directory in the file system for which it cannot determine its name or intended location (possibly due to a corruption in its parent directory), it places this object in the /lost+found directory when fsck is run in fix mode. For security reasons only the root user on Linux can read files in /lost+found. If the administrator can later determine the original name and location of the file based on its contents, the file can be moved or copied into its intended location.

The file names in the /lost+found directory are in the following formats:

```
parent.id.file.id.time-in-sec-since-1970
parent.id.dir.id.time-in-sec-since-1970
```

The *id* fields are the internal Oracle ACFS numeric identifiers for each file and directory in the file system.

You can use `acfsutil info id` *id mount_point* to attempt to determine the directory associated with `parent.id`. This directory is assumed to be where the deleted object originated. For information about `acfsutil info`, see "acfsutil info file" on page 13-99.

If the parent directory is not known, the parent `id` field is set to `UNKNOWN`.

> **Note:** It is not possible to see the contents of the `/lost+found` directory from a snapshot.

Examples
The following example shows how to check and repair an Oracle ACFS file system.

Example 13–5 Using the fsck command

```
# /usr/sbin/fsck -F acfs -y -o a,v /dev/asm/volume1-123
```

mkfs

Purpose
Creates an Oracle ACFS file system on the Solaris operating system.

Syntax and Description

```
mkfs -F acfs -o h /dev/null
mkfs -F acfs [-o options] volume_device size
```

`mkfs -F acfs -o h /dev/null` displays usage text and exits.

Table 13–8 contains the options available with the `mkfs` command.

Table 13–8 Options for the Solaris mkfs command

Option	Description
-F acfs	Specifies the type of file system on Solaris. `acfs` designates the Oracle ACFS type.

Table 13–8 (Cont.) Options for the Solaris mkfs command

Option	Description
-o *options*	Specifies that options follow (b, f, h, n, v). Options are preceded with the -o flag and entered as a comma-delimited string. For example: -o f,v
	■ b=*blocksize*
	Specifies the block size of the new file system. The default block size is 4K and this is the only size supported.
	■ h
	Displays the usage text and exits.
	■ f
	Specifies the force option. This action creates the file system even if there is an existing Oracle ACFS on the volume device, although only if the file system is dismounted. This option overwrites structures on the original file system. Use this option with caution.
	■ n=*name*
	Specifies the name for the file system. A name can be a maximum of 64 characters. acfsutil info fs returns the name if one was specified.
	■ v
	Specifies verbose mode. The progress is displayed as the operation occurs.
volume_device	Specifies an Oracle ADVM device file.
size	Specifies the size of the file system in 512-byte units or in units of K|M|G|T|P. Units specified are in K (kilobytes), M (megabytes), G (gigabytes), T (terabytes), or P (petabytes).

mkfs is used to create the on disk structure needed for Oracle ACFS file system to be mounted. The mkfs command is the traditional UNIX command used to build a file system. After mkfs runs successfully, the USAGE column in the V$ASM_VOLUME view displays ACFS. root privilege is not required. The ownership of the volume device file dictates who can run this command. The minimum file system size is 200 MB. The Oracle ACFS driver must be loaded for mkfs to work.

Examples

Before creating an Oracle ACFS file system, first determine which Oracle ADVM volume devices are available. You can use the ASMCMD volinfo command to display information about the volumes and volume devices.

```
ASMCMD [+] > volinfo -a
...
        Volume Name: VOLUME1
        Volume Device: /dev/asm/volume1-123
        State: ENABLED
...
```

See "volinfo" on page 12-66.

Next create an Oracle ACFS file system on the volume device file.

Example 13–6   Using the mkfs command

```
$ /usr/sbin/mkfs -F acfs /dev/asm/volume1-123
```

mount

Purpose
Mounts an Oracle ACFS file system on the Solaris operating system.

Syntax and Description
```
mount -F acfs -o h /tmp /dev/null
mount -F acfs [-r] [-o options] volume_device mount_point
```

mount -F acfs -o h /tmp /dev/null displays usage text and exits.

Table 13–9 contains the options available with the mount command.

Table 13–9   Options for the Solaris mount command

Option	Description
-F acfs	Specifies the type of file system on Solaris. acfs designates the Oracle ACFS type.
-r	Mounts in read-only mode.

Table 13–9   (Cont.) Options for the Solaris mount command

Option	Description
-o	Specifies that options follow. Options are preceded with the -o flag followed by a comma-delimited string of options. For example: -o all,v
	The following options are available:
	■ all
	Reads the Oracle ACFS mount registry created with the acfsutil registry command and mounts all the file systems in it. A mount -o all command is automatically run at Oracle ACFS startup.
	The -o all option requires two placeholders to satisfy the command arguments: a dummy argument for the volume device and any valid directory.
	When the -o all option is specified, other -o options are ignored. To specify mount options for a registry entry, include those options with the acfsutil registry command when you add the entry to the registry.
	■ devices/nodevices
	Allows or disallows the opening of any character or block special devices from this mount.
	■ exec/noexec
	Allows or disallows the execution of programs in this file system.
	■ h
	Displays the usage help text and exits.
	■ rootsuid/norootsuid
	Allows or disallows the execution of setuid to root executables (binaries by non-root users whose permissions allow set user Id execution, and are owned by root). rootsuid is the default action. If norootsuid is specified, an attempt to run these executables as a non-root user fails with a permission denied error.
	■ ro
	Mounts the file system in read-only mode.
	■ rw
	Mounts the file system in read-write mode.
	■ setuid/nosetuid
	Allows or disallows the execution of setuid and setgid programs.
	■ suid/nosuid
	Allows or disallows the execution of setuid and setgid programs and the opening of any character or block special devices from this mount.
	■ v
	Specifies verbose mode. The progress is displayed as the operation occurs.
volume_device	Specifies an Oracle ADVM volume device file that has been formatted by mkfs. *device* is required but can be a dummy value.
mount_point	Specifies the directory where the file system is mounted. This directory must exist before you run the mount command.

mount attaches a file system to the Oracle ACFS hierarchy at the mount point that is the name of a directory. The mount occurs on the node where the mount command was issued. The mount command returns an error if the file system is not in a dismounted state on this node.

It is not always possible to return the cause of a mount failure to the mount command. When this happens Oracle ACFS writes the cause of the failure to the system console and associated system log file.

After mount runs successfully, the MOUNTPATH field in the V$ASM_VOLUME view displays the directory name on which the file system is now mounted.

An Oracle ACFS file system should only be mounted on one mount point. The same mount point name should be used on all cluster members.

root privilege is required to run mount.

Examples

The first example shows how to mount volume1-123 on the mount point /u01/app/acfsmounts/myacfs. The second example shows how to mount all the registered Oracle ACFS file systems. Placeholder arguments must be provided for the volume device name and mount point when specifying the -o all option. The volume device can be a dummy value, such as none. Any valid directory can be specified for the mount point, such as /tmp.

Example 13–7   Using the mount command

```
# /sbin/mount -F acfs /dev/asm/volume1-123 /u01/app/acfsmounts/myacfs

# /sbin/mount -F acfs -o all none /tmp
```

umount/umountall

Purpose

Dismounts an Oracle ACFS file system on the Solaris operating system.

Syntax and Description

```
umount -V [mount_point | volume_device]
umountall -F acfs
```

unmountall -F acfs dismounts all Oracle ACFS file systems.

Table 13–10 contains the options available with the umount command.

Table 13–10   Options for the Solaris umount command

Option	Description
-V	Echoes the complete command line, but does not run the command. Use this option is used to verify and validate the command line before execution. Valid only with umount.
mount_point	Specifies the directory where the file system is mounted. Valid only with umount.
volume_device	Specifies the Oracle ADVM volume device name associated with the file system. Valid only with umount.
-F acfs	Specifies the type of file system on Solaris. acfs designates the Oracle ACFS type. Valid only with umountall.

umount and `umountall` detach an Oracle ACFS from the file system hierarchy on the current node. If a file system is busy, `umount` and `umountall` fail.

`root` privileges are required to run the `umount` and `umountall` commands.

Examples

The following examples show how to dismount an Oracle ACFS file system. The first example specifies the mount point of the file system to dismount. The second example specifies the volume device associated with the file system to dismount. The third example dismounts all Oracle ACFS file systems.

Example 13–8   Using the umount command

```
# /sbin/umount /dev/asm/volume1-123

# /sbin/umount /u01/app/acfsmounts/myacfs

# /sbin/umountall -F acfs
```

Oracle ACFS Command-line Tools for the AIX Environment

During the installation of Oracle ACFS and Oracle ADVM with Oracle Grid Infrastructure on AIX, several system security classes (authorizations) are created for Oracle ACFS and Oracle ADVM operations. These authorizations enable the following operations:

- Configuration of Oracle ACFS and Oracle ADVM devices

- Removal of Oracle ACFS and Oracle ADVM devices

- Definition of Oracle ACFS and Oracle ADVM devices

These authorizations are collected in a role (`oracle_devmgmt`) which is assigned by default to the Oracle Grid Infrastructure user and Oracle ASM administrator. You can run the `lsrole` or `rolelist` command after installation to see this new role. These commands can be run at any time to ensure that the user has the proper authorization to manage Oracle ACFS and Oracle ADVM.

For example, run the following as the Oracle ASM administrator:

```
$ rolelist -e
 oracle_devmgmt
```

After the initial installation on an AIX operating system, the shell from which the root script was run will not inherit the new role. If the role is not present for the user, then the `swrole` command must be run before performing Oracle ACFS or Oracle ADVM operations.

For example, run the following as the Oracle ASM administrator:

```
$ swrole oracle_devmgmt
```

Alternatively, you can open a new shell to perform Oracle ACFS or Oracle ADVM operations.

During the removal of Oracle Grid Infrastructure, the `oracle_devmgmt` role and its associated authorizations are deleted from the system.

> **See Also:** Oracle Grid Infrastructure on AIX, refer to *Oracle Grid Infrastructure Installation Guide for AIX Based Systems* for more information about installing

> **Note:** If a disk group is dismounted with the force option on AIX, any Oracle ADVM volumes device files that were on that disk group remain on the system. These files are removed when the disk group is remounted.

Table 13–11 contains a summary of the Oracle ACFS commands for the AIX operating system. The commands in Table 13–11 have been extended with additional options to support Oracle ACFS on AIX.

Table 13–11   Summary of Oracle ACFS commands for AIX

Command	Description
fsck	Checks and repairs an Oracle ACFS file system on AIX.
mkfs	Creates an Oracle ACFS file system on AIX.
mount	Mounts an Oracle ACFS file system on AIX.
umount/umountall	Dismounts an Oracle ACFS file system on AIX.

fsck

Purpose
Checks and repairs an Oracle ACFS file system on the AIX operating system.

Syntax and Description
fsck -V acfs [-n|-y] [-o *options*] *volume_device*

Table 13–12 contains the options available with the AIX fsck command.

Table 13–12   Options for the AIX fsck command

Option	Description
-V acfs	Specifies an Oracle ADVM volume on AIX. acfs designates the Oracle ACFS type.
-n	Answers no to any prompts.
-y	Answers yes to any prompts.

-o *options*	Specifies that options follow (a, f, v). Options are preceded with the -o flag and entered as a comma-delimited string. For example: -o a,v
	
	■ a
	
	  Specifies to automatically fix the file system.
	
	■ f
	
	  Forces the file system into mountable state without completing a file system check or fix.
	
	■ v
	
	  Specifies verbose mode. The progress is displayed as the operation occurs.
	
	■ vv
	
	  Specifies very verbose mode. The progress is displayed as the operation occurs.

volume_device	Specifies the primary Oracle ADVM volume device.

fsck checks and repairs an existing Oracle ACFS. This command can only be run on a dismounted file system. root privileges are required to run fsck. The Oracle ACFS driver must be loaded for fsck to work.

By default, fsck only checks for and reports any errors. The -a flag must be specified to instruct fsck to fix errors in the file system.

In a few cases, fsck prompts for questions before proceeding to check a file system. These cases include:

- If fsck detects that another fsck is in progress on the file system

- If fsck detects that the Oracle ACFS driver is not loaded

- If the file system does not appear to be Oracle ACFS

In checking mode, fsck also prompts if there are transaction logs that have not been processed completely due to an incomplete shutdown. To run in a non-interactive mode, include either the -y or -n options to answer yes or no to any questions.

fsck creates working files before it checks a file system. These working files are created in /usr/tmp if space is available. /tmp is used if /usr/tmp does not exist. If insufficient space is available in the tmp directory, fsck attempts to write to the current working directory. The files that fsck creates are roughly the size of the file system being checked divided by 32K. At most two such files are allocated. For example, a 2 GB file system being checked causes fsck to generate one or two 64K working files in the /usr/tmp directory. These files are deleted after fsck has finished.

In the event that fsck finds a file or directory in the file system for which it cannot determine its name or intended location (possibly due to a corruption in its parent directory), it places this object in the /lost+found directory when fsck is run in fix mode. For security reasons only the root user on Linux can read files in /lost+found. If the administrator can later determine the original name and location of the file based on its contents, the file can be moved or copied into its intended location.

The file names in the /lost+found directory are in the following formats:

```
parent.id.file.id.time-in-sec-since-1970
parent.id.dir.id.time-in-sec-since-1970
```

The id fields are the internal Oracle ACFS numeric identifiers for each file and directory in the file system.

You can use acfsutil info id id mount_point to attempt to determine the directory associated with parent.id. This directory is assumed to be where the deleted object originated. For information about acfsutil info, see "acfsutil info file" on page 13-99.

If the parent directory is not known, the parent id field is set to UNKNOWN.

> **Note:** It is not possible to see the contents of the /lost+found directory from a snapshot.

Examples

The following example shows how to check and repair an Oracle ACFS file system.

Example 13–9   Using the fsck command

```
# /usr/sbin/fsck -V acfs -y -o a,v /dev/asm/volume1-123
```

mkfs

Purpose
Creates an Oracle ACFS file system on the AIX operating system.

Syntax and Description
```
mkfs -V acfs -o h /dev/null
mkfs -V acfs [-v name ] [-s size][-o options] volume_device
```

`mkfs -V acfs -o h /dev/null` displays usage text and exits.

Table 13–13 contains the options available with the AIX `mkfs` command.

Table 13–13   Options for the AIX mkfs command

Option	Description
`-V acfs`	Specifies the type of file system on AIX. `acfs` designates the Oracle ACFS type.
`-v name`	Specifies the name for the file system. A name can be a maximum of 64 characters. `acfsutil info fs` returns the name if one was specified.
`-s size`	Specifies the size of the file system in 512-byte units or in units of `K|M|G|T|P`. Units specified are in `K` (kilobytes), `M` (megabytes), `G` (gigabytes), `T` (terabytes), or `P` (petabytes).
`-o options`	Specifies that options follow (b, f, h, v). Options are preceded with the `-o` flag and entered as a comma-delimited string. For example: `-o f,v` ■ `b=blocksize` Specifies the block size of the new file system. The default block size is 4K and this is the only size supported. ■ `f` Specifies the force option. This action creates the file system even if there is an existing Oracle ACFS on the volume device, although only if the file system is dismounted. This option overwrites structures on the original file system. Use this option with caution. ■ `h` Displays the usage text and exits. ■ `v` Specifies verbose mode. The progress is displayed as the operation occurs.
`volume_device`	Specifies an Oracle ADVM device file that is to be formatted.

`mkfs` is used to create the on disk structure needed for Oracle ACFS file system to be mounted. The `mkfs` command is the traditional UNIX command used to build a file system. After `mkfs` runs successfully, the `USAGE` column in the `V$ASM_VOLUME` view displays `ACFS`. `root` privilege is not required. The ownership of the volume device file dictates who can run this command. The minimum file system size is 200 MB. The Oracle ACFS driver must be loaded for `mkfs` to work.

Examples

Before creating an Oracle ACFS file system, first determine which Oracle ADVM volume devices are available. You can use the ASMCMD `volinfo` command to display information about the volumes and volume devices.

```
ASMCMD [+] > volinfo -a
...
        Volume Name: VOLUME1
        Volume Device: /dev/asm/volume1-123
        State: ENABLED
...
```

See "volinfo" on page 12-66.

Next create an Oracle ACFS file system on the volume device file.

Example 13–10   Using the mkfs command

```
$ /usr/sbin/mkfs -V acfs /dev/asm/volume1-123
```

mount

Purpose

Mounts an Oracle ACFS file system on the AIX operating system.

Syntax and Description

```
mount -V acfs -o h /dev/null /xxx
mount -v acfs [-o options] volume_device mount_point
```

`mount -V acfs -o h /dev/null /xxx` displays usage text and exits.

Table 13–14 contains the options available with the `mount` command.

Table 13–14   Options for the AIX mount command

Option	Description
-v acfs	Specifies the type of file system on AIX. `acfs` designates the Oracle ACFS type.

Table 13–14   (Cont.) Options for the AIX mount command

Option	Description
`-o options`	Specifies that options follow. Options are preceded with the `-o` flag followed by a comma-delimited string of options. For example: `-o all,ro,v`
	The following options are available:
	■ `all`
	Reads the Oracle ACFS mount registry created with the `acfsutil registry` command and mounts the file systems in it. A `mount -V acfs -o all` command is automatically run at Oracle ACFS startup.
	The `-o all` option requires two placeholders to satisfy the command arguments: a dummy argument for the volume device and directory.
	When the `-o all` option is specified, other `-o` options are ignored. To specify mount options for a registry entry, include those options with the `acfsutil registry` command when you add the entry to the registry.
	■ `h`
	Displays the usage help text and exits.
	■ `nosuid`
	Specifies that the execution of `setuid` and `setgid` programs by way of this mount is not allowed.
	■ `nodev`
	Specifies that you cannot open devices from this mount.
	■ `norootsuid`
	Fails the execution of binaries by non-root users whose permissions allow set user Id execution, and are owned by root. An attempt to run these executables as a non-root user fails with a permission denied error.
	■ `rootsuid`
	Allows the execution of binaries by non-root users of set user Id files owned by root. This is the default action.
	■ `ro`
	Mounts the file system in read-only mode.
	■ `rw`
	Mounts the file system in read-write mode.
	■ `v`
	Specifies verbose mode. The progress is displayed as the operation occurs.
`volume_device`	Specifies an Oracle ADVM volume device file that has been formatted by `mkfs`. `device` is required but can be a dummy value.
`mount_point`	Specifies the directory where the file system is mounted. This directory must exist before you run the `mount` command.

`mount` attaches a file system to the Oracle ACFS hierarchy at the mount point that is the name of a directory. The mount happens on the node where the mount command was issued. The mount command returns an error if the file system is not in a dismounted state on this node.

It is not always possible to return the cause of a mount failure to the `mount` command. When this happens Oracle ACFS writes the cause of the failure to the system console and associated system log file.

After `mount` runs successfully, the `MOUNTPATH` field in the `V$ASM_VOLUME` view displays the directory name on which the file system is now mounted.

An Oracle ACFS file system should only be mounted on one mount point. The same mount point name should be used on all cluster members.

`root` privilege is required to run `mount`.

Examples

The first example shows how to mount `volume1-123` on the mount point `/u01/app/acfsmounts/myacfs`. The second example shows how to mount all the registered Oracle ACFS file systems. The dummy names (`none`) have been entered as placeholders for the device and directory as these arguments are required, but not used, when the `all` option is specified.

Example 13–11   Using the mount command

```
# /usr/sbin/mount -v acfs /dev/asm/volume1-123 /u01/app/acfsmounts/myacfs

# /usr/sbin/mount -v acfs -o all none none
```

umount/umountall

Purpose

Dismounts an Oracle ACFS file system on the AIX operating system.

Syntax and Description

```
umount [mount_point | volume_device]
umountall -F acfs
```

`unmountall -F acfs` dismounts all Oracle ACFS file systems.

Table 13–15 contains the options available with the `umount` command.

Table 13–15   Options for the AIX umount command

Option	Description
mount_point	Specifies the directory where the file system is mounted. Valid only with `umount`.
volume_device	Specifies the Oracle ADVM volume device name associated with the file system. Valid only with `umount`.
-F acfs	Specifies the type of file system on AIX. `acfs` designates the Oracle ACFS type. Valid only with `umountall`.

`umount` and `umountall` detach an Oracle ACFS from the file system hierarchy on the current node. If a file system is busy, `umount` and `umountall` fail.

`root` privileges are required to run the `umount` and `umountall` commands.

Examples

The following examples show how to dismount an Oracle ACFS file system. The first example specifies the mount point of the file system to dismount. The second example

specifies the volume device associated with the file system to dismount. The third example dismounts all Oracle ACFS file systems.

Example 13–12   Using the umount command

```
# /usr/sbin/umount /dev/asm/volume1-123

# /usr/sbin/umount /u01/app/acfsmounts/myacfs

# /usr/sbin/umountall -F acfs
```

Oracle ACFS Command-line Tools for Windows Environments

Table 13–16 contains a summary of the Oracle ACFS file system commands for Windows. These commands have been extended to support Oracle ACFS. All other Windows file system commands operate without change for Oracle ACFS.

Table 13–16   Summary of Oracle ACFS file system commands for Windows

Command	Description
acfschkdsk	Checks and repairs an Oracle ACFS file system on Windows.
acfsdismount	Dismounts an Oracle ACFS on Windows.
acfsformat	Creates an Oracle ACFS file system on Windows.
acfsmountvol	Mounts an Oracle ACFS file system on Windows.
acfsutil detach	Prepares the Oracle ACFS driver for module unload on Windows.

Table 13–17 contains a summary of the non-Oracle ACFS file system commands for Windows.

Table 13–17   Summary of Non-Oracle ACFS file system commands for Windows

Command	Description
advmutil dismount	Dismounts Oracle ASM volume devices for file systems that are not Oracle ACFS on Windows.
advmutil list	Lists Oracle ASM volume devices for file systems that are not Oracle ACFS on Windows.
advmutil mount	Mounts Oracle ASM volume devices for file systems that are not Oracle ACFS on Windows.

acfschkdsk

Purpose
Checks and repairs an Oracle ACFS file system.

Syntax and Description
```
acfschkdsk /h
acfschkdsk [/a|/f] [/v] [/vv] [/n|/y] volume_device
```

Table 13–18 contains the options available with the acfschkdsk command.

Table 13–18   Options for the acfschkdsk command

Option	Description
/a	Specifies to automatically repair the file system.
/f	Forces the file system into mountable state without completing a file system check or repair.
/v	Specifies verbose mode. The progress is displayed as the operation occurs.
/vv	Specifies very verbose mode. The progress is displayed as the operation occurs.
/h	Displays the usage help text and exits.
/n	Answers no to any prompts.
/y	Answers yes to any prompts.
volume_device	Specifies the Oracle ADVM volume device.

acfschkdsk checks and repairs an existing Oracle ACFS. This command can only be run on a file system that has been dismounted clusterwide.

The Oracle ACFS driver must be loaded for acfschkdsk to work. If the driver is not loaded, the administrator is prompted to ensure this is intentional. For information about loading drivers, see "Oracle ACFS Drivers Resource Management" on page B-3.

The Oracle ACFS driver normally ensures that acfschkdsk is the only user of the file system clusterwide. In extreme cases it may be necessary to fix a file system without loading the driver if the file system automount causes the system to fail. The verifications that are normally done by the driver for exclusive access are bypassed in this case.

By default acfschkdsk only checks for and only reports errors. The /a flag must be specified to instruct acfschkdsk to repair errors in the file system.

In a few cases, acfschkdsk prompts for questions before proceeding to check a file system. For example, if acfschkdsk detects that another acfschkdsk is in progress on the file system, or if acfschkdsk detects that the Oracle ACFS driver is not loaded, or if the file system does not appear to be Oracle ACFS. In checking mode, acfschkdsk also prompts if there are transaction logs that have not been processed completely due to an incomplete shutdown. To run in a non-interactive mode, include either the /y or /n options to answer yes or no to any questions.

acfschkdsk creates working files before it checks a file system. These working files are created in the temp directory if space is available. If insufficient space is available, acfschkdsk attempts to write in the current working directory. The files acfschkdsk creates are roughly the size of the file system being checked divided by 32K. At most two files are allocated. These files are deleted when acfschkdsk has finished.

In the event that acfschkdsk finds a file or directory in the file system for which it cannot determine its name or intended location (possibly due to a corruption in its parent directory), it places this object in the \lost+found directory when acfschkdsk is run in fix mode. For security reasons, only the Windows Administrator can read files in \lost+found. If the administrator can later determine the original name and location of the file based on its contents, the file can be moved or copied into its intended location.

The file names in the \lost+found directory are in the following formats:

parent.*id*.file.*id*.*time-in-sec-since-1970*

parent.*id*.dir.*id*.*time-in-sec-since-1970*

The *id* fields are the internal Oracle ACFS numeric identifiers for each file and directory in the file system.

You can use `acfsutil info id` *id mount_point* to attempt to determine the directory associated with `parent.id`. This directory is assumed to be where the deleted object originated. For information about `acfsutil info`, see "acfsutil info file" on page 13-99.

If the parent directory is not known, the parent *id* field is set to `UNKNOWN`.

> **Note:** It is not possible to see the contents of the `\lost+found` directory from a snapshot.

Examples
The following example shows how to check an Oracle ACFS file system on Windows. The verbose mode is enabled in this example.

Example 13–13   Using the acfschkdsk command
```
C:\> acfschkdsk /v asm-volume1-311
```

acfsdismount

Purpose
Dismounts an Oracle ACFS file system and removes its mount point (name space attachment).

Syntax and Description
```
acfsdismount /h
acfsdismount [/v] {drive_letter | mount_point}
acfsdismount [/v] /all
```

Table 13–19 contains the options available with the `acfsdismount` command.

Table 13–19   Options for the acfsdismount command

Option	Description
/v	Specifies verbose mode. The progress is displayed as the operation occurs.
/h	Displays help text and exits.
/all	Specifies to dismount all Oracle ACFS file systems on this node.
drive_letter	Specifies the drive letter followed by a colon.
mount_point	Specifies the full path of the mount point including the drive letter.

`acfsdismount` removes the mount point (name space attachment) for the specified file system on the current node and dismounts the file system if it is mounted. When the file system is in the dismounted state, the data is preserved on the device and the file system can be re-mounted with the `acfsmountvol` command.

Windows Administrator privileges are required to use the `acfsdismount` command.

Examples

The following examples show how to dismount an Oracle ACFS file system. The first example dismounts a file system using only the drive letter. The second example dismounts a file system using the full path name. The third examples dismounts all mounted file systems. All the examples enable the verbose mode.

Example 13–14   Using the acfsdismount command

```
C:\> acfsdismount /v O:

C:\> acfsdismount /v O:\mnt\

C:\oracle>acfsdismount /v /all
acfsdismount: valid path name: C:\oracle\acfsmounts\acfs1\
acfsdismount: opening handle to volume C:\oracle\acfsmounts\acfs1
acfsdismount: locking volume C:\oracle\acfsmounts\acfs1
acfsdismount: dismounting volume C:\oracle\acfsmounts\acfs1
acfsdismount: removing volume mount point on C:\oracle\acfsmounts\acfs1\
acfsdismount: closing handle to volume C:\oracle\acfsmounts\acfs1 and exiting
cleanly
```

acfsformat

Purpose

Creates an Oracle ACFS file system.

Syntax and Description

```
acfsformat /h
acfsformat [/v] [/f] [/b blocksize] [/n name] volume_device [blocks]
```

Table 13–20 contains the options available with the acfsformat command.

Table 13–20   Options for the acfsformat command

Option	Description
/v	Specifies verbose mode. The progress is displayed as the operation occurs.
/n name	Specifies the name for the file system. This is also known as a volume label. acfsutil info fs returns the name if one was specified. This can also be displayed from the VOL_LABEL column in the V$ASM_ACFSVOLUMES view.
/f	Specifies the force option. This action creates the file system even if there is an existing Oracle ACFS on the device, although only if the file system is dismounted. This option overwrites structures on the original file system. Use this option with caution.
/h	Displays the usage help text.
/b blocksize	The default block size is 4K and this is the only size supported in 11g Release 2 11.2).

Table 13–20   (Cont.) Options for the acfsformat command

Option	Description
blocks	Specifies the number of blocks that the file system should consume on the named device. The quantity specified can be in units of K (kilobytes), M (megabytes), G (gigabytes), or T (terabytes). If a unit is not specified, the default is bytes. If number of blocks specified is not a multiple of the block size, than the value is rounded up to the closest multiple. If this option is not specified, the entire device is consumed.
volume_device	Specifies an Oracle ADVM device file to be formatted.

acfsformat creates the on disk structure needed for Oracle ACFS to be mounted. acfsformat performs a quick format on the target volume. After acfsformat runs successfully, the USAGE column in the V$ASM_VOLUME view displays ACFS.

Windows Administrator privileges are not required to use this command. The minimum file system size is 200 MB. The Oracle ACFS driver must be loaded for acfsformat to work.

Examples
This example shows how to create an Oracle ACFS file system. The verbose option is specified in this example.

Example 13–15   Using the acfsformat command

```
C:\> acfsformat /v asm-volume1-311
```

acfsmountvol

Purpose
Mounts an Oracle ACFS file system.

Syntax and Description
```
acfsmountvol /h
acfsmountvol
acfsmountvol [/v] /all
acfsmountvol [/v] { drive_letter | dir } volume_device
```

Table 13–21 contains the options available with the acfsmountvol command.

Table 13–21   Options for the acfsmountvol command

Option	Description
/v	Specifies verbose mode. The progress is displayed as the operation occurs.
/h	Displays the help usage text and exits.
/all	Reads the ACFS mount registry created with acfsutil registry and mounts the file systems listed in it on this node.
drive_letter	The Windows drive letter to link to an Oracle ADVM volume device.
dir	Specifies the full directory of the mount point including the drive letter.

Table 13–21   (Cont.) Options for the acfsmountvol command

Option	Description
volume_device	Specifies an Oracle ADVM volume device file that has been formatted by `acfsformat`.

`acfsmountvol` attaches an Oracle ACFS to the file system hierarchy at the specified path name or drive letter. *dir* must be an empty directory. Oracle ACFS mount points can be created on any empty directory and they can be hierarchical (nested).

After `acfsmountvol` runs successfully, the `MOUNTPATH` column in the `V$ASM_VOLUME` view displays the full path or drive letter on which the file system is now mounted. If no arguments are specified, the command lists all mounted Oracle ACFS file systems on this node.

It is not always possible to return the cause of a mount failure to the `acfsmountvol` command. When this happens Oracle ACFS writes the cause of the failure to the Windows system event logger.

An Oracle ACFS file system should only be mounted on one mount point. The same mount point name should be used on all cluster members.

Windows Administrator privileges are required to mount an Oracle ACFS.

Examples

The first example shows how to mount `asm-volume1-311` on the mount point identified as the `F:` drive. The second example shows how to mount `asm-volume1-215` on the mount point specified by `c:\oracle\acfsmounts\acfs1\`. The first two examples enable verbose mode. The third example lists all the mounted file systems on the node.

Example 13–16   Using the acfsmountvol command

```
C:\> acfsmountvol  /v  F:  asm-volume1-215

C:\> acfsmountvol /v  c:\oracle\acfsmounts\acfs1  asm-volume1-311

C:\oracle>acfsmountvol
   asm-volume1-311
       C:\oracle\acfsmounts\acfs1
```

acfsutil detach

Purpose
Prepares the Oracle ACFS driver for module unload.

Syntax and Description
`acfsutil detach`

`acfsutil detach` prepares Oracle ACFS for having its driver unloaded. Oracle ACFS registers itself as a base file system upon driver load. In order for the Windows service control manager to send a stop command to the driver, Oracle ACFS must unregister itself as a base file system in addition to deleting any device objects associated with the driver.

After `acfsutil detach` has completed successfully, and all volumes have been dismounted, then the administrator can run the Windows service control manager's

stop command to the Oracle ACFS driver. If the `acfsutil detach` command is not used, then the `stop` command does not call the Oracle ACFS driver unload routine. This command can only be run by the Windows `Administrator`. It is called automatically by the Oracle ACFS shutdown scripts.

Examples

The following examples show how to detach all Oracle ACFS file systems and unload drivers on Windows. The first example dismounts all the Oracle ACFS file systems volume devices. The second example detaches the volume devices.

Example 13–17   Using the acfsutil detach command

```
C:\> acfsdismount /all

C:\> acfsutil detach
```

advmutil dismount

Purpose

The `advmutil dismount` command dismounts Oracle ADVM volume devices for file systems that are not Oracle ACFS.

Syntax and Description

```
advmutil dismount [/unregister] {drive_letter | dir}
advmutil dismount [/unregister] [/all | /dg=disk_group]
```

Table 13–22 contains the options available with the Windows `advmutil dismount` command.

Table 13–22   Options for the advmutil dismount command

Option	Description
drive_letter	Specifies the drive letter to dismount.
dir	Specifies the directory of the mount point, including the drive letter.
/all	Specifies to dismount all file systems on this node.
/unregister	Optionally unregister an Oracle ADVM volume device and its drive letter from the system registry. Upon startup of the Oracle ASM instance and Oracle ACFS, the unregistered drive letter and Oracle ADVM volume symbolic links are not created.
/dg=*disk_group*	Optional name of the disk group containing the registered Oracle ADVM volume devices to create drive letters or possibly mount or dismount file systems.

> **Note:**   To format an Oracle ADVM volume device with NTFS, it must be mounted on a drive letter. After formatting, Oracle ADVM volume devices with NTFS can be mounted on either a drive letter or a directory.

To successfully run this command, the local Oracle ASM instance must be running and the disk groups required by this command must have been created and mounted in the Oracle ASM instance and volumes enabled.

Examples

These examples show how to dismount a file system that is not Oracle ACFS.

1. The first example shows how to remove a drive letter for an Oracle ADVM volume device which disallows access to the file system.

2. The second example shows how to remove a drive letter for an Oracle ADVM volume device which disallows access to the file system and removes it from the registry.

3. The third example shows how to remove a drive letter for all registered Oracle ADVM volume devices which disallows access to all file systems.

4. The fourth example shows how to remove a drive letter for all registered Oracle ADVM volume devices in a particular Oracle ASM disk group which disallows access to all related file systems.

Example 13–18   Using the advmutil dismount command

```
C:\> advmutil dismount H:

C:\> advmutil dismount H: /unregister

C:\> advmutil dismount /all

C:\> advmutil dismount /dg=data
```

advmutil list

Purpose

The `advmutil list` command lists the Oracle ADVM volume devices that were registered with `advmutil mount /register`.

Syntax and Description

```
advmutil list /registry
```

Table 13–23 contains the options available with the Windows `advmutil list` command.

Table 13–23   Options for the advmutil list command

Option	Description
list /registry	Lists the drive letter and corresponding Oracle ADVM volume device name contained in the Windows registry.

To successfully run this command, the local Oracle ASM instance must be running and the disk groups required by this command must have been created and mounted in the Oracle ASM instance and volumes enabled.

For information about listing Oracle ADVM volume devices for Oracle ACFS, see "acfsutil info file" on page 13-99.

Examples

The following example shows how to list all entries in the system registry.

Example 13–19   Using the advmutil list command

```
C:\> advmutil list /registry

  Drive Letter      Asm Volume
  ===================================
  H:                asm-volume1-123
  K:                asm-volume2-245
```

advmutil mount

Purpose

The `advmutil mount` command mounts Oracle ADVM volume devices for file systems that are not Oracle ACFS.

Syntax and Description

```
advmutil mount [/register] {drive_letter | dir} volume_device
advmutil mount /all
```

Table 13–24 contains the options available with the Windows `advmutil mount` command.

Table 13–24   Options for the advmutil mount command

Option	Description
drive_letter	Specifies the drive letter to mount.
dir	Specifies the directory of the mount point, including the drive letter.
volume_device	Indicates the Oracle ADVM volume device.
/all	Specifies to mount all non-Oracle ACFS file systems on this node.
/register	Optionally register an Oracle ADVM volume device and its drive letter in the Windows system registry. After startup of the Oracle ASM instance and Oracle ACFS, all registered drive letters and Oracle ADVM volume symbolic links are created. This allows for automatic file system mounting by Windows as is also done for Oracle ACFS file systems.

> **Note:** To format an Oracle ADVM volume device with NTFS, it must be mounted on a drive letter. After formatting, Oracle ADVM volume devices with NTFS can be mounted on either a drive letter or a directory.

To successfully run this command, the local Oracle ASM instance must be running and the disk groups required by this command must have been created and mounted in the Oracle ASM instance and volumes enabled.

For information about registering Oracle ADVM volume devices for Oracle ACFS, see "acfsutil registry" on page 13-104.

Examples

These examples show how to mount a file system that is not Oracle ACFS.

1. The first example shows how to create a drive letter for an Oracle ASM volume device for use as a non-Oracle ACFS access which you could then use to format the volume for use with NTFS.

2. The second example shows how to create a drive letter for an Oracle ASM volume device for use as a non-Oracle ACFS access which you could then use to format the volume for use with NTFS and save it in the Windows system registry.

3. The third example shows how to create a drive letter for all Oracle ASM volume devices which were saved in the system registry which also allows a file system to be accessible.

Example 13–20   Using the advmutil mount command

```
C:\> advmutil mount H: asm-volume1-123

C:\> advmutil mount H: asm-volume1-123 /register

C:\> advmutil mount /all
```

Oracle ACFS Command-Line Tools for Tagging

Table 13–25 contains a summary of the commands for Oracle ACFS tagging.

You can run `acfsutil help` on all platforms to display help text. You can run `acfsutil version` on all platforms to display the Oracle ACFS version.

When the options are entered with commands on a Windows platform, use / instead of - with the option. For example, you can display help for `acfsutil` on a Linux platform with `acfsutil -h`. On a Windows platform, use `acfsutil /h`.

Note that a mount point on a Windows operating system can be a drive letter or a directory including the drive letter.

Table 13–25   Summary of commands for Oracle ACFS tagging

Command	Description
acfsutil tag info	Displays the tags for directories or files in Oracle ACFS file systems.
acfsutil tag set	Adds a tag to directories or files in an Oracle ACFS file system.
acfsutil tag unset	Removes a tag or all tag names from directories or files in an Oracle ACFS file system.

acfsutil tag info

Purpose

Displays the tag names for tagged directories or file names in Oracle ACFS file systems.

Syntax and Description

```
acfsutil tag info -h
acfsutil tag info [-r] [-t tagname] path [path ...]
acfsutil tag info [-t tagname]
```

`acfsutil tag info -h` displays help text and exits.

Table 13–26 contains the options available with the `acfsutil tag info` command.

Table 13–26   Options for the acfsutil tag info command

Option	Description
-t *tagname*	Specifies the tag name string to display. Enclose the string in quotes if the string contains a space.
path	Specifies the path name to one or more files or directories.
-r	Specifies recursive action on a specified path.

The `acfsutil tag info` command can recursively apply the operation to all child files and subdirectories if the `-r` option is included with specified directory path names.

If no path names are specified, the `acfsutil tag info` command searches all Oracle ACFS mount points for tagged files.

Only the paths of tagged files in an Oracle ACFS file system are displayed. If the `-t` *tagname* option is included, only tag names matching *tagname* are displayed.

Any user may use this command to display tag info on a directory that the user has read access privileges.

Examples
The following example show the use of the `acfsutil tag info` command.

Example 13–21   Using the acfsutil tag info command

```
$ /sbin/acfsutil tag info -r /u01/app/acfsmounts/myacfs/myrepfiles/
```

acfsutil tag set

Purpose
Adds the given tag to the specified files or directories in an Oracle ACFS file system

Syntax and Description
```
acfsutil tag set -h
acfsutil tag set [-v] [-r] tagname path [path ...]
```

`acfsutil tag set -h` displays help text and exits.

Table 13–27 contains the options available with the `acfsutil tag set` command.

Table 13–27   Options for the acfsutil tag set command

Option	Description
tagname	Specifies a tag name string. Enclose the string in quotes if the string contains a space.
	The tag string can be composed of ASCII characters that include: a-b, A-Z, 0-9, the space character, the hyphen, and the underscore. The maximum length of a tag name is 32 characters.
	`all` is a reserved tag name and cannot be set on any file or directory.
path	Specifies the path string to one or more files or directories.
-r	Specifies recursive action on the specified path.
-v	Displays the progress of the operation.

The command can recursively apply the operation to all child files and subdirectories for the specified directory path names.

When adding a tag name to a file or directory, the existing tags on that file or directory remain unchanged. New files that are created after a directory is assigned a tag implicitly inherit tags from their parent directories. Existing files in the directory do not inherit the new tag; these files must be explicitly assigned the tag. Renaming a file or moving a file to a subdirectory within the same file system does not cause the file to inherit tag names from the new parent directory. Moving a file between file systems uses a copy operation and the newly created file does inherit the tag names of the parent directory.

Any user who has the privilege to modify the target file or directory may run this command.

Examples

The following example show the use of the `acfsutil tag set` command.

Example 13–22   Using the acfsutil tag set command

```
$ /sbin/acfsutil tag set repl_grp1 -r /u01/app/acfsmounts/myacfs/myrepfiles/*.dat
```

acfsutil tag unset

Purpose

Removes the given tag name or all tag names from the specified file or directory.

Syntax and Description

```
acfsutil tag unset -h
acfsutil tag unset [-v] [-r] {all | tagname} path [path ...]
```

`acfsutil tag unset -h` displays help text and exits.

Table 13–28 contains the options available with the `acfsutil tag unset` command.

Table 13–28   Options for the acfsutil tag unset command

Option	Description
all	Specifies to remove all tags in the path string.
tagname	Specifies a tag name string. Enclose the string in quotes if the string contains a space.
path	Specifies the path string to one or more files or directories.
-r	Specifies recursive action on the specified path.
-v	Displays the progress of the operation.

The command can recursively apply the operation to all child files and subdirectories for the specified directory path names.

When removing a tag name from a file or directory, other existing tags on that file or directory remain unchanged.

Any user who has privilege to modify the target file or directory may run this command.

Examples

The following example show the use of the `acfsutil tag unset` command.

Example 13–23   Using the acfsutil tag unset command

```
$ /sbin/acfsutil tag unset repl_grp1 -r
       /u01/app/acfsmounts/myacfs/myrepfiles/*.log
```

Oracle ACFS Command-Line Tools for Replication

Table 13–29 contains a summary of the commands for Oracle ACFS replication.

You can run `acfsutil help` on all platforms to display help text. You can run `acfsutil version` on all platforms to display the Oracle ACFS version.

When the options are entered with commands on a Windows platform, use `/` instead of `-` with the option. For example, you can display help for `acfsutil` on a Linux platform with `acfsutil -h`. On a Windows platform, use `acfsutil /h`.

Note that a mount point on a Windows operating system can be a drive letter or a directory including the drive letter.

You must use a trailing back slash when specifying a drive letter mount point with replication commands on Windows platforms. For example, use `p:\` (not `p:`) for a drive letter mount point. If the trailing back slash is omitted, this could trigger a Windows path substitution mechanism that replaces the drive letter with the last accessed path on that drive.

Table 13–29   Summary of commands for Oracle ACFS replication

Command	Description
acfsutil repl bg	Starts, stops, or displays information about the Oracle ACFS replication background processes.
acfsutil repl compare	Verifies that files have been replicated on an Oracle ACFS file system.
acfsutil repl info	Displays information about replication processing on an Oracle ACFS file system.
acfsutil repl init	Initiates replication on files in an Oracle ACFS file system.
acfsutil repl pause	Pauses replication on an Oracle ACFS file system.
acfsutil repl resume	Resumes replication on an Oracle ACFS file system.
acfsutil repl sync	Synchronizes primary and standby file systems.
acfsutil repl terminate	Stops all replication activity on the Oracle ACFS file system.
acfsutil repl trace	Specifies the trace level for replication information.
acfsutil repl update	Updates replication information while replication is running.

acfsutil repl bg

Purpose

Starts, stops, or displays information about the Oracle ACFS replication background processes.

Syntax and Description

```
acfsutil repl bg -h
acfsutil repl bg {start|stop|info} mount_point
```

`acfsutil repl bg -h` displays help text and exits.

Table 13–30 contains the options available with the `acfsutil repl bg` command.

Table 13–30   Options for the acfsutil repl bg command

Option	Description
start	Starts the replication background processing.
stop	Stops the replication background processing
info	Displays information about the replication background processes.
mount_point	Specifies the directory where the file system is mounted.

Oracle ACFS replication uses background processes to transport file system changes, to communicate between the sites hosting the primary and standby file systems, to monitor the state of replication processing, and to apply file system changes to the standby file system. These background processes must be running for replication to function.

When replication is started with the `acfsutil repl init` command, these background processes are started and are registered with Oracle Clusterware so that they are automatically restarted after a reboot or system crash. Usually it is not necessary to run the `acfsutil repl bg` command because Oracle Clusterware automatically starts and stops the background processes as file systems are mounted and unmounted or node membership changes after replication is initialized. However, replication cannot progress if one of the daemons is not running. In this case running `repl bg stop` followed by `repl bg start` ensures that all the daemons are running.

`acfsutil bg start` ensures that all the necessary daemons for replication are running. The daemons may run on different cluster nodes. `acfsutil bg stop` ensures that no daemons for the file system are running on the local node and relocates the daemons to another node if available.

You must have system administrator privileges to run the `acfsutil repl bg start` or `acfsutil repl bg stop` commands. You must have system administrator or Oracle ASM administrator privileges to run the `acfsutil repl bg info` command.

Examples

Example 13–24 shows the use of the `acfsutil repl bg` command.

Example 13–24   Using the acfsutil repl bg command

```
$ /sbin/acfsutil repl bg start /u01/app/acfsmounts/myacfs

$ /sbin/acfsutil repl bg stop /u01/app/acfsmounts/myacfs

$ /sbin/acfsutil repl bg info /u01/app/acfsmounts/myacfs
```

acfsutil repl compare

Purpose

Verifies that files have been replicated on an Oracle ACFS file system.

Syntax and Description

```
acfsutil repl compare -h
acfsutil repl compare [-a] [-t { all | tagname, ...}]
          primary_mount_point standby_mount_point
```

`acfsutil repl compare -h` displays help text and exits.

Table 13–31 contains the options available with the `acfsutil repl compare` command.

Table 13–31   Options for the acfsutil repl compare command

Option	Description
-a	Compares all files and their extended attribute names.
-t { all | *tagname, ...* }	Compares files with all or some of the tag names specified in the `acfsutil repl init` operation. The `all` option specifies all tag names supplied in the `acfsutil repl init` operation. A comma separated list of tag names must only use tag names specified during the `acfsutil repl init` operation.
primary_mount_point	Specifies the directory where the primary file system is mounted.
standby_mount_point	Specifies the directory where the standby file system is mounted.

The `acfsutil repl compare` command verifies all or part of the primary mount point has been replicated to the standby mount point. The standby mount point must be mounted locally for comparison. Best results are obtained when the primary file system is changing as little as possible.

The `-a` option is typically used when no tag names were specified during the `acfsutil repl init` operation. The `-a` option compares all files on the primary file system against those on the standby file system. The `-a` option also tests for extra files on the standby file system that do not exist on the primary.

The `-t` option is used when tags were specified during the `acfsutil repl init` operation. The `-t` operation locates all file names on the primary file system with the specified tag names and compares them to the corresponding files on the standby. The `-t` option takes as arguments either `all` option or a comma separated list of tag names. If the `-t` option is used with `all`, then all tag names supplied during the `acfsutil repl init` operation are selected. Otherwise, any specified tag names must exactly match a tag name supplied during the `acfsutil repl init` operation. The `-t` option also tests for extra files on the standby file system that do not have an associated tag name that was specified during the `acfsutil repl init` operation. The `acfsutil repl info -c` option can be used to determine what tags were specified during the `acfsutil repl init` operation.

If neither the `-a` nor `-t` option is provided, a primary to standby file comparison is done without testing tag names nor extended attributes.

If NFS is used to mount the standby file system locally, the standby cannot be checked for matching tag names and extended attributes due to limitations in the NFS protocol.

Examples

Example 13–25 shows the use of the `acfsutil repl compare` command.

Example 13–25   Using the acfsutil repl compare command

```
$ /sbin/acfsutil repl compare /u01/app/acfsmounts/myacfs /nfs_mounted_standby
```

acfsutil repl info

Purpose
Displays information about replication processing on an Oracle ACFS file system.

Syntax and Description

```
acfsutil repl info -h
acfsutil repl info -c [-v] mount_point
acfsutil repl info -s [-v][-l] [-n number{m|h|d|w|y}]
          [-r start_time[#stop_time]] [-f eventlog] mount_point
acfsutil repl info [-a|-e|-t] [-v][-l]
          [-r start_time[#stop_time]] [-f eventlog] mount_point
```

acfsutil repl info -h displays help text and exits.

Table 13–32 contains the options available with the acfsutil repl info command.

Table 13–32   Options for the acfsutil repl info command

Option	Description
-a	Displays only apply records from the replication event log. An apply record contains the date and time that the set of file system changes were captured on the primary file system and the date and time that they were applied on the standby file system.
-c	Displays configuration information.
-e	Displays only error records from the replication event log.
-f eventlog	Specifies the path to the replication event log to be used as the source of the information. The default location is: mount_point/.ACFS/repl/logs/ReplicationEventsLog
-h	Displays help text.
-l	Displays last event only.
-n number{m|h|d|w|y}	Specifies the interval to display information. number specifies the number of minutes (m), hours (h), days (d), weeks (w), or years (y). For example: -n 5d The statistics are summarized in the units of time specified.
-r start_time[#stop_time]	Specifies the start and stop time to display information. The format for the time period is yyyy-nn-ddThh:mm:ss. yyyy - year nn - month (01..12) dd - day (01..31) hh - hour (00..23) mm - minute (00..59) ss - second (00..59) For example: -r 2010-05-05T08:30:00#2010-05-05T15:30:00
-s	Displays statistical information. This option is valid only when run on the primary file system. It is important that the primary and standby nodes are running a network time service to ensure that the statistics are meaningful.
-t	Displays only transport records from the replication event log.
-v	Displays verbose output.

Table 13–32   (Cont.) Options for the acfsutil repl info command

Option	Description
mount_point	Specifies the directory where the file system is mounted.

To display information about replication configuration, use `acfsutil repl info` with the `-c` option. To display information about replication statistics, use `acfsutil repl info` with the `-s` option. To display information about replication events, use `acfsutil repl info` with the `-a`, `-e`, or `-t` options.

You must have system administrator or Oracle ASM administrator privileges to run this command.

Examples

Example 13–26 shows the use of the `acfsutil repl info` command.

Example 13–26   Using the acfsutil repl info command

```
$ /sbin/acfsutil repl info -c /u01/app/acfsmounts/myacfs

$ /sbin/acfsutil repl info -s -n 5d /u01/app/acfsmounts/myacfs

$ /sbin/acfsutil repl info -a -v /u01/app/acfsmounts/myacfs
```

acfsutil repl init

Purpose

Initiates replication on all the files in an Oracle ACFS file system or only those files with a specified list of tags.

Syntax and Description

```
acfsutil repl init -h

acfsutil repl init primary -s standby_connect_string [tagname...]
     [-m standby_mount_point] [-c primary_service]
     [-d trace_level] [-z {on|off}] mount_point

acfsutil repl init standby -p primary_connect_string
     [-c standby_service] [-d trace_level] mount_point
```

`acfsutil repl init -h` displays help text and exits.

Table 13–33 contains the options available with the `acfsutil repl init` command.

Table 13–33   Options for the acfsutil repl init command

Option	Description
`primary`	Specifies replication of the primary file system.
`standby`	Specifies replication of the standby file system.
`-s standby_conncect_string`	Specifies the connection string for the site hosting the standby file system. The user in the connection string must have `SYSASM` and `SYSDBA` privileges.

Table 13–33   (Cont.) Options for the acfsutil repl init command

Option	Description
-p *primary_conncect_string*	Specifies the connection string for the site hosting the primary file system. The user in the connection string must have SYSASM and SYSDBA privileges.
tagname	Specifies the tag names of the files to be replicated.
-m *standby_mount_point*	Specifies the directory where the standby file system is mounted if it is different than the primary file system. This is valid only for the primary file system initialization.
-c *primary_service*	Specifies the service name for the primary file system. Required if the primary file system is using a different service name than the standby file system.
-c *standby_service*	Specifies the service name for the standby file system. Required if the standby file system is using a different service name than the primary file system.
-d *trace_level*	Specifies the trace level setting (0..6) for replication logs.
-z on|off	Turn on or off compression on replication logs. Compression is off by default.
mount_point	Specifies the directory where the file system is mounted. For repl init primary, *mount_point* specifies the primary file system. For repl init standby, *mount_point* specifies the standby file system.

When the acfsutil repl init primary command completes successfully, processing begins to replicate initial copies of all specified files to the standby file system. In addition, any changes to these files and any new files subsequently created in the file system on the primary file system are captured and transported to the standby file system. However, successful completion of the command does not indicate that the specified files have been copied in full to the standby file system. To monitor the progress of the initial replication of all specified files to the standby file system, users run the acfsutil repl info -c command.

The first phase of the initial replication copy process begins with a complete copy of the file system directory tree structure, not the files within the directory tree, to the standby file system. During this time, attempts to rename directories may fail. After the directory tree structure has been successfully copied to the standby file system, replication initialization begins the second phase of the initial copy process: copying all files to be replicated to the standby file system. While a file is being copied to the standby file system during this initial replication phase, writes and truncates to that file are blocked.

Symbolic link files are replicated as is. If the symbolic link resolves to an absolute path name, and that path name does not exist on the standby file system, then referencing the symbolic link results in errors.

A new service name must be created for replication. You cannot use the +ASM service name. You must specify a unique service name for each file system that you want to replicate when there are multiple replicated file systems on a node or cluster.

A file system being replicated can only be mounted on one path for the primary or standby file system. Otherwise, dismounting any one of the paths on a node where it is mounted on multiple paths stops the replication background threads.

A replicated Oracle ACFS file system must be mounted on the same directory within a cluster. The file system does not have to be mounted on all of the nodes in a cluster,

but if it is mounted, it must be mounted on the directory specified with the `acfsutil repl init` command. In addition, no other file system can be mounted on that directory on other nodes.

A single Oracle ACFS file system cannot be configured both as a primary and a standby file system. If replication is configured on the same host or within the same cluster, then the primary and standby file systems cannot use the same named mount point.

The primary file system must have a minimum size of 4 GB for each node that is mounting the file system. The standby file system must have a minimum size of 4 GB and should be sized appropriately for the amount of data being replicated and the space necessary for the replication logs sent from the primary file system.

The sites hosting the primary and standby file systems must be running the same operating system and must have the same machine architecture.

Oracle wallets can also be used to manage security credentials.

> **See Also:**
>
> - *Oracle Database Advanced Security Administrator's Guide* for information about Oracle wallets
>
> - *Oracle Database SecureFiles and Large Objects Developer's Guide* for information about wallet management
>
> - *Oracle Database Net Services Reference* for information about wallet parameters in the `SQLNET.ORA` file

You must have system administrator privileges to run this command.

Examples

Example 13–27 shows the use of the `acfsutil repl init` command.

Example 13–27   Using the acfsutil repl init command

```
$ /sbin/acfsutil repl init primary
    -s standby_admin/standby_passwd@standby_repl_site
    -m /standby/repl_data -c primary_repl_service
    /acfsmounts/repl_data

$ /sbin/acfsutil repl init standby
    -p primary_admin/primary_passwd@primary_repl_site
    -c standby_repl_service /standby/repl_data
```

acfsutil repl pause

Purpose

Pauses replication on an Oracle ACFS file system.

Syntax and Description

```
acfsutil repl pause -h
acfsutil repl pause mount_point
```

`acfsutil repl pause -h` displays help text and exits.

Table 13–34 contains the options available with the `acfsutil repl pause` command.

Table 13–34   Options for the acfsutil repl pause command

Option	Description
`mount_point`	Specifies the directory where the file system is mounted.

The `acfsutil repl pause` command temporarily stops the replication process until the `acfsutil repl resume` command is run. The command can be run at either the site hosting the primary or a standby file system. `acfsutil repl resume` should be run soon after `acfsutil repl pause` to resume replication.

The `acfsutil repl pause` command can be used to temporarily stop the transportation of replication logs from the primary file system to the standby file system, but logs continue to be generated on the primary file system. Note that logs are not deleted until they are transported to the standby file system and applied to the standby file system.

The `acfsutil repl pause` command can also be used to temporarily stop the application of replication logs to a standby file system, but the command does not stop the transportation of logs to the standby file system. Note that logs are not deleted from the standby file system until they are applied to the file system.

When replication is paused on either the site hosting the primary or standby file system, replication logs still accumulate on the standby and primary file system. These logs can consume all available storage space in the file system so you must check available storage on the sites hosting the file systems if you have paused replication. Ensure that you run `acfsutil repl resume` before the accumulated replication logs consume all available space in the file system.

You must have system administrator or Oracle ASM administrator privileges to run this command.

Examples

Example 13–28 shows the use of the `acfsutil repl pause` command.

Example 13–28   Using the acfsutil repl pause command

```
$ /sbin/acfsutil repl pause /u01/app/acfsmounts/myacfs
```

acfsutil repl resume

Purpose

Resumes replication on an Oracle ACFS file system where replication has been paused.

Syntax and Description

```
acfsutil repl resume -h
acfsutil repl resume mount_point
```

`acfsutil repl resume -h` displays help text and exits.

Table 13–35 contains the options available with the `acfsutil repl resume` command.

Table 13–35   Options for the acfsutil repl resume command

Option	Description
`mount_point`	Specifies the directory where the file system is mounted.

This command resumes the replication process after the `acfsutil repl pause` command has been run. This command should be run soon after replication has been paused with the `acfsutil repl pause` command.

You must have system administrator or Oracle ASM administrator privileges to run this command.

Examples

Example 13–29 shows the use of the `acfsutil repl resume` command.

Example 13–29   Using the acfsutil repl resume command

```
$ /sbin/acfsutil repl resume /u01/app/acfsmounts/myacfs
```

acfsutil repl sync

Purpose

Synchronizes primary and standby file systems.

Syntax and Description

```
acfsutil repl sync -h
acfsutil repl sync [apply] mount_point
```

`acfsutil repl sync -h` displays help text and exits.

Table 13–36 contains the options available with the `acfsutil repl sync` command.

Table 13–36   Options for the acfsutil repl sync command

Option	Description
apply	Specifies that all changes be applied to the standby file system.
mount_point	Specifies the directory where the primary file system is mounted.

The `acfsutil repl sync` command can be used to synchronize the state of the primary and standby file system. This command can only be run on the site hosting the primary file system.

You should first quiesce your applications and run a sync(1) call so that the synchronized state is known and meaningful to you. The `acfsutil repl sync` command then causes all outstanding replication data to be shipped from the primary to the standby file system.

The `acfsutil repl sync` command returns success when this transfer is complete or, if the `apply` parameter is supplied, when all of these changes have been successfully applied to the standby file system. At this point, unless an unmount of the last node on the primary site cluster is about to occur, applications can be restarted. `acfsutil repl sync` can be used before dismounting the file system on the last node to ensure that all modifications have been sent to the standby file system.

If `acfsutil repl info -c` reports the primary file system is in the `Initializing` state, the `acfsutil repl sync` command does not guarantee that all files to be replicated in the file system have been sent to the standby file system. Files that existed in the file system before running the `acfsutil repl init` command are sent to the standby file system in the background.

You must have system administrator or Oracle ASM administrator privileges to run this command.

Examples

Example 13–30 shows the use of the `acfsutil repl sync` command.

Example 13–30   Using the acfsutil repl sync command

```
$ /sbin/acfsutil repl sync /u01/app/acfsmounts/myacfs
```

acfsutil repl terminate

Purpose

Stops all replication activity on the Oracle ACFS file system at the site where it is run.

Syntax and Description

```
acfsutil repl terminate -h
acfsutil repl terminate primary mount_point
acfsutil repl terminate standby [immediate] mount_point
```

`acfsutil repl terminate -h` displays help text and exits.

Table 13–37 contains the options available with the `acfsutil repl terminate` command.

Table 13–37   Options for the acfsutil repl terminate command

Option	Description
primary	Stops replication on the primary file system.
standby	Stops replication on the standby file system.
immediate	Stops replication processing immediately on the standby file system.
mount_point	Specifies the directory where the file system is mounted.

The `acfsutil repl terminate` command stops all replication processing. When terminating replication, you should terminate replication for the primary file system first, and then the standby file system. If you want to ensure all changes are sent to the standby file system before terminating the primary site, ensure that all applications are quiesced and run the `acfsutil repl sync` command. The terminate command must be run on both sites if both sites hosting the file systems are available.

After `acfsutil repl terminate standby` has been run, you can use the standby file system in read-write mode. After `acfsutil repl terminate` has been run on either the primary file system or standby file system, you must start replication from the beginning with the `acfsutil repl init` command.

The `repl terminate standby` command waits until replication data at the standby file system has been applied. If you want to terminate replication immediately without applying all the replication data, use the `immediate` option. However, this option can leave some files at the standby file system in an indeterminate state.

To momentarily stop Oracle ACFS replication, you can use the `acfsutil repl pause` command, followed soon after by the `acfsutil repl resume` command.

You must have system administrator privileges to run this command.

Examples

Example 13–31 shows the use of the `acfsutil repl terminate` command.

Example 13–31   Using the acfsutil repl terminate command

```
$ /sbin/acfsutil repl terminate /u01/app/acfsmounts/myacfs
```

acfsutil repl trace

Purpose

Sets the replication trace level for gathering trace information on an Oracle ACFS file system.

Syntax and Description

```
acfsutil repl trace -h
acfsutil repl trace level mount_point
```

`acfsutil repl trace -h` displays help text and exits.

Table 13–39 contains the options available with the `acfsutil repl trace` command.

Table 13–38   Options for the acfsutil repl trace command

Option	Description
level	Specifies the trace level setting (0..6). The default level is 2.
mount_point	Specifies the directory where the file system is mounted.

Increasing the trace level can have a performance impact and should be done at the recommendation of Oracle support services.

Trace files for Oracle ACFS replication are stored in the following locations:

```
GRID_HOME/log/host_name/acfsrepl*
GRID_HOME/log/host_name/client/acfsutil.log
```

You must have system administrator or Oracle ASM administrator privileges to run this command.

Examples

Example 13–32 shows the use of the `acfsutil repl trace` command.

Example 13–32   Using the acfsutil repl trace command

```
$ /sbin/acfsutil repl trace 5 /u01/app/acfsmounts/myacfs
```

acfsutil repl update

Purpose

Updates replication information while replication is running on an Oracle ACFS file system.

Syntax and Description

```
acfsutil repl update -h
acfsutil repl update
```

[-p *primary_connect_string*] [-s *standby_connect_string*] [-z {on|off}]
  mount_point

`acfsutil repl update -h` displays help text and exits.

Table 13–39 contains the options available with the `acfsutil repl update` command.

Table 13–39   Options for the acfsutil repl update command

Option	Description
-p *primary_connect_string*	Specifies the connection string for the primary file system. The user in the connection string must have `SYSASM` and `SYSDBA` privileges.
-s *standby_connect_string*	Specifies the connection string for the standby file system. The user in the connection string must have `SYSASM` and `SYSDBA` privileges.
-z on|off	Turn on or off compression on replication logs. Compression is off by default.
mount_point	Specifies the directory where the file system is mounted.

The `acfsutil repl update` command updates replication information after replication has been started. This command can only be run on the site hosting the primary file system. The connection strings for the primary and standby file systems can be altered.

You can specify any combination of the -p, -s, and -z options, but you must specify at least one of the options.

While replication is running, compression can be turned on or off with the -z option.

You must have system administrator privileges to run this command.

Examples

Example 13–33 shows the use of the `acfsutil repl update` command.

Example 13–33   Using the acfsutil repl update command

```
$ /sbin/acfsutil repl update -s mystandby@oracle.com /u01/app/acfsmounts/myacfs
```

Oracle ACFS Command-Line Tools for Security

Table 13–40 contains a summary of the commands for Oracle ACFS security.

You can run `acfsutil help` on all platforms to display help text. You can run `acfsutil version` on all platforms to display the Oracle ACFS version.

When the options are entered with commands on a Windows platform, use / instead of - with the option. For example, you can display help for `acfsutil` on a Linux platform with `acfsutil -h`. On a Windows platform, use `acfsutil /h`.

Note that a mount point on a Windows operating system can be a drive letter or a directory including the drive letter.

Table 13–40   Summary of commands for Oracle ACFS security

Command	Description
acfsutil sec admin add	Adds a security administrator.
acfsutil sec admin password	Changes the password of a security administrator.
acfsutil sec admin remove	Removes a security administrator.
acfsutil sec batch	Runs a batch file.
acfsutil sec disable	Disables Oracle ACFS security.
acfsutil sec enable	Enables Oracle ACFS security.
acfsutil sec info	Displays Oracle ACFS file system security information.
acfsutil sec info file	Lists the security realms that a specified file or directory belongs to.
acfsutil sec init	Initializes Oracle ACFS file system security.
acfsutil sec load	Loads Oracle ACFS file system security metadata.
acfsutil sec prepare	Prepares an Oracle ACFS file system for security.
acfsutil sec realm add	Adds objects to an Oracle ACFS file system realm.
acfsutil sec realm clone	Clones an Oracle ACFS file system realm.
acfsutil sec realm create	Creates an Oracle ACFS file system realm.
acfsutil sec realm delete	Removes objects from an Oracle ACFS file system realm.
acfsutil sec realm destroy	Removes an Oracle ACFS file system realm.
acfsutil sec rule clone	Clones an Oracle ACFS file system security rule.
acfsutil sec rule create	Creates an Oracle ACFS file system security rule.
acfsutil sec rule destroy	Removes an Oracle ACFS file system security rule.
acfsutil sec rule edit	Updates an Oracle ACFS file system security rule.
acfsutil sec ruleset clone	Clones an Oracle ACFS file system security rule set.
acfsutil sec ruleset create	Creates an Oracle ACFS file system security rule set.
acfsutil sec ruleset destroy	Removes an Oracle ACFS file system rule set.
acfsutil sec ruleset edit	Updates an Oracle ACFS file system rule set.
acfsutil sec save	Saves Oracle ACFS file system security metadata.

acfsutil sec admin add

Purpose

Adds a new security administrator for an Oracle ACFS file system.

Syntax and Description

```
acfsutil sec admin add -h
acfsutil sec admin add admin
```

acfsutil sec admin add -h displays help text and exits.

Table 13–41 contains the options available with the acfsutil sec admin add command.

Table 13–41   Options for the acfsutil sec admin add command

Option	Description
`admin`	Specifies a security administrator user name. The user specified must be an existing operating system user and a member of the security group specified with the `acfsutil sec init` command.
	On Windows, a security administrator user name must be specified with a fully qualified domain user name in the form of `domain_name\username`.

Security administrators are common for all Oracle ACFS file systems in a cluster. A temporary password must be provided for the new security administrator. The password must conform to the format that is described in "acfsutil sec init" on page 13-71.

The new security administrator can change the password with the `acfsutil sec admin password` command. For information, refer to "acfsutil sec admin password" on page 13-66.

Security administrators are allowed to browse all directories in an Oracle ACFS file system whether they have the underlying operating system permissions and whether any realm checks allow it. This enables a security administrator to check the location of the files when securing them with Oracle ACFS security realms. However, a security administrator cannot view the contents of individual files without the appropriate operating system and security realm permissions.

Only an existing security administrator can run this command.

Examples

The following example shows the use of the `acfsutil sec admin add` command.

Example 13–34   Using the acfsutil sec admin add command

```
$ /sbin/acfsutil sec admin add sec_admin_three
```

acfsutil sec admin password

Purpose

Changes the password of a security administrator for an Oracle ACFS file system.

Syntax and Description

```
acfsutil sec admin password -h
acfsutil sec admin password
```

`acfsutil sec admin password -h` displays help text and exits.

The `acfsutil sec admin password` command changes the security password for the administrator that is running the command. When you run this command, you are prompted to enter a new password. The password must conform to the format that is described in "acfsutil sec init" on page 13-71.

Every time a security administrator runs an `acfsutil sec` command, the administrator is prompted for the security administrator's password.

> **Note:** When prompting for the security administrator's password, the following text displays: `Realm management password`
>
> The password required is the Oracle ACFS security administrator's password, not the operating system password of the user.

Only a security administrator can run this command.

Examples

The following example shows the use of the `acfsutil sec admin password` command.

Example 13–35 Using the acfsutil sec admin password command

```
$ /sbin/acfsutil sec admin password
```

acfsutil sec admin remove

Purpose

Removes a security administrator from an Oracle ACFS file system.

Syntax and Description

```
acfsutil sec admin remove -h
acfsutil sec admin remove admin
```

`acfsutil sec admin remove -h` displays help text and exits.

Table 13–42 contains the options available with the `acfsutil sec admin remove` command.

Table 13–42 Options for the acfsutil sec admin remove command

Option	Description
`admin`	Specifies an existing security administrator user name.
	On Windows, the security administrator user name must be specified with a fully qualified user name in the form of `domain_name\username`.

Only a security administrator can run this command.

Examples

The following example shows the use of the `acfsutil sec admin remove` command.

Example 13–36 Using the acfsutil sec admin remove command

```
$ /sbin/acfsutil sec admin remove sec_admin_three
```

acfsutil sec batch

Purpose

Runs a specified batch file.

Syntax and Description

```
acfsutil sec batch -h
acfsutil sec batch batch_file
```

`acfsutil sec batch -h` displays help text and exits.

Table 13–43 contains the options available with the `acfsutil sec batch` command.

Table 13–43   Options for the acfsutil sec batch command

Option	Description
batch_file	Specifies an existing batch file name. The batch file contains a list of `acfsutil sec` commands.

The batch file can only contain security realm management commands. Interactive commands are not recommended. The `acfsutil sec admin add`, `acfsutil sec admin password`, and `acfsutil sec init` commands are not supported in the batch file. Also, other `acfsutil` commands, such as `acfsutil encr` commands, are not allowed in the batch file. If a command in the batch file fails, subsequent commands in the batch file are not run.

The following are examples of commands that can be in a batch file:

```
acfsutil sec realm create my_realm1 -m /mnt1 -e off
acfsutil sec realm create my_realm2 -m /mnt2 -e off
```

Only a security administrator can run this command. When the command is run, the administrator is prompted once for a password.

Examples

The following example shows the use of the `acfsutil sec batch` command.

Example 13–37   Using the acfsutil sec batch command

```
$ /sbin/acfsutil sec batch my_batch_file
```

acfsutil sec disable

Purpose

Disables Oracle ACFS security on a mount point or a realm in a mount point.

Syntax and Description

```
acfsutil sec disable -h
acfsutil sec disable -m mount_point [-S snap_name] [realm]
```

`acfsutil sec disable -h` displays help text and exits.

Table 13–44 contains the options available with the `acfsutil sec disable` command.

Table 13–44   Options for the acfsutil sec disable command

Option	Description
-m mount_point	Specifies the directory where the file system is mounted.
realm	Specifies the name of the security realm in the Oracle ACFS file system.

Table 13–44   (Cont.) Options for the acfsutil sec disable command

Option	Description
-S *snap_name*	Disables security for the specified read-write snapshot.

The `acfsutil sec disable -m `*mount_point* command disables security functionality on the Oracle ACFS file system specified by the mount point. When security is disabled on the file system, security realms do not enforce realm authorization.

The `acfsutil sec disable -m `*mount_point realm* command disables security for a specific realm.

Only a security administrator can run this command.

Examples

The following example shows the use of the `acfsutil sec disable` command.

Example 13–38   Using the acfsutil sec disable command

```
$ /sbin/acfsutil sec disable -m /u01/app/acfsmounts/myacfs my_realm
```

acfsutil sec enable

Purpose

Enables Oracle ACFS security on a mount point or a realm in a mount point.

Syntax and Description

```
acfsutil sec enable -h
acfsutil sec enable -m mount_point [-S snap_name] [realm]
```

`acfsutil sec enable -h` displays help text and exits.

Table 13–45 contains the options available with the `acfsutil sec enable` command.

Table 13–45   Options for the acfsutil sec enable command

Option	Description
-m *mount_point*	Specifies the directory where the file system is mounted.
realm	Specifies the name of the security realm.
-S *snap_name*	Enables security for the specified read-write snapshot.

The `acfsutil sec enable -m `*mount_point* command enables security functionality on the Oracle ACFS file system specified by the mount point. When security is enabled on the file system, security realms that have been enabled enforce realm authorization. You should run this command before enabling any individual security realm.

The `acfsutil sec enable -m `*mount_point realm* command enables security for a specific realm. The realm enforces authorization if security has been enabled on the file system.

Only a security administrator can run this command.

Examples

These example shows the use of the `acfsutil sec enable` command.

Example 13–39   Using the acfsutil sec enable command

```
$ /sbin/acfsutil sec enable -m /u01/app/acfsmounts/myacfs

$ /sbin/acfsutil sec enable -m /u01/app/acfsmounts/myacfs my_realm
```

acfsutil sec info

Purpose
Displays information about Oracle ACFS security.

Syntax and Description

```
acfsutil sec info -h
acfsutil sec info -m mount_point
     [{-n [realm] | -l [rule] | -s [ruleset] | -c }] [-S snap_name]
```

acfsutil sec info -h displays help text and exits.

Table 13–46 contains the options available with the acfsutil sec info command.

Table 13–46   Options for the acfsutil sec info command

Option	Description
-m mount_point	Specifies the directory where the file system is mounted.
-n realm	Displays information about the specified security realm. If the realm name is omitted, a list of all realms is displayed.
-l rule	Displays information about the specified rule. If the rule name is omitted, a list of all rules is displayed.
-s ruleset	Displays information about the specified rule set. If the rule set name is omitted, a list of all rule sets is displayed.
-c	Lists all the command rules.
-S snap_name	Displays information about the realms, rules, and rule sets in the specified snapshot.

The acfsutil sec info command retrieves information about the list of realms, rules, and rule sets on the specified mount point. By specifying a particular realm, rule, or ruleset, you can retrieve information specific about the specified realm, rule, or ruleset. You can also display information about a specified snapshot.

If the -m option is specified without any other options, then the security enabled status and prepared status are displayed for the specified mount point.

Only a security administrator can run this command.

Examples
The following example shows the use of the acfsutil sec info command.

Example 13–40   Using the acfsutil sec info command

```
$ /sbin/acfsutil sec info -m /u01/app/acfsmounts/myacfs -n my_realm
```

acfsutil sec info file

Purpose
Lists the names of the Oracle ACFS security realms that the specified file or directory belongs to.

Syntax and Description
```
acfsutil sec info file -h
acfsutil sec info file -m mount_point path
```

`acfsutil sec info file -h` displays help text and exits.

Table 13–47 contains the options available with the `acfsutil sec info file` command.

Table 13–47   Options for the acfsutil sec info file command

Option	Description
`-m mount_point`	Specifies the directory where the file system is mounted.
`path`	Specifies the path of the file or directory in the file system.

This command also displays the encryption status of files.

Only a security administrator can run this command.

Examples
The following example shows the use of the `acfsutil sec info file` command.

Example 13–41   Using the acfsutil sec info file command

```
$ /sbin/acfsutil sec info file -m /u01/app/acfsmounts/myacfs
                               /u01/app/acfsmounts/myacfs/myfiles
```

acfsutil sec init

Purpose
Initializes Oracle ACFS security.

Syntax and Description
```
acfsutil sec init -h
acfsutil sec init -u admin -g admin_sec_goup
```

`acfsutil sec init -h` displays help text and exits.

Table 13–48 contains the options available with the `acfsutil sec init` command.

Table 13–48   Options for the acfsutil sec init command

Option	Description
`-u admin`	Specifies the first security administrator user name. The user specified must be an existing operating system user and a member of the operating system group specified by the `-g` option.
	On Windows, the security administrator user name must be specified with a fully qualified user name in the form of *domain_name\username*.

Table 13–48   (Cont.) Options for the acfsutil sec init command

Option	Description
`-g admin_sec_group`	Specifies the name of the security group for the administrator. The group specified must be an existing operating system group.
	On Windows, the group name must be specified with a fully qualified domain group name in the form of `domain_name\groupname`. If the `domain_name\groupname` contains a space, then enclose the string in quotes (" ").

The `acfsutil sec init` command creates the storage necessary for security credentials and identifies an operating system user as the first security administrator. The command also identifies the operating system group that is the designated security group. All users that are security administrators must be members of the designated security group. Security administrators are common for all Oracle ACFS file systems.

The `acfsutil sec init` command is run once to set up Oracle ACFS security for each cluster and can be run from any node in the cluster. Other security commands can also be run from any node in a cluster.

Only the root user or Windows `Administrator` user can run this command. The user specifies a temporary password for the security administrator. The security administrator password must conform to the following format:

- The maximum number of characters is 20.
- The minimum number of characters is 8.
- The password must contain at least one digit.
- The password must contain at least one letter.

The new security administrator can change the password with the `acfsutil sec admin password` command. For information, refer to "acfsutil sec admin password" on page 13-66.

Security administrators are allowed to browse all directories in an Oracle ACFS file system whether they have the underlying operating system permissions and whether any realm checks allow it. This enables a security administrator to check the location of the files when securing them with Oracle ACFS security realms. However, a security administrator cannot view the contents of individual files without the appropriate operating system and security realm permissions.

Examples
The following example shows the use of the `acfsutil sec init` command.

Example 13–42   Using the acfsutil sec init command

```
$ /sbin/acfsutil sec init -u grid -g asmadmin
```

acfsutil sec load

Purpose
Loads Oracle ACFS security metadata into a file system identified by a mount point.

Syntax and Description
```
acfsutil sec load -h
acfsutil sec load -m mount_point -p file
```

`acfsutil sec load -h` displays help text and exits.

Table 13–49 contains the options available with the `acfsutil sec load` command.

Table 13–49   Options for the acfsutil sec load command

Option	Description
`-m` *mount_point*	Specifies the directory where the file system is mounted.
`-p` *file*	Specifies the name of an existing saved security metadata file.

The `acfsutil sec load` command loads the security metadata in a saved XML file into the specified Oracle ACFS file system.

To run the `acfsutil sec load` command, the destination mount point must have a file system that has been prepared for security and does not contain any user-created security objects.

If the file system mounted on destination mount point contains security objects, then you must run `acfsutil sec prepare -u` to remove all previously created security objects on the file system. After successfully running `acfsutil sec prepare -u`, you must run `acfsutil sec prepare` to prepare the file system for security. After successfully running `acfsutil sec prepare`, you can run `acfsutil sec load` on the file system. For information about preparing security on or removing security from a file system, refer to "acfsutil sec prepare" on page 13-73.

The `acfsutil sec load` command does not load system security realms from the backup file. System security realms are created with the `acfsutil sec prepare` command; `acfsutil sec load` does not recreate these realms. For information about the system-created security realms, refer to "acfsutil sec prepare" on page 13-73.

Only a security administrator can run this command.

Examples
The following example shows the use of the `acfsutil sec load` command.

Example 13–43   Using the acfsutil sec load command

```
$ /sbin/acfsutil sec load -m /u01/app/acfsmounts/myacfs -p my_metadata_file.xml
```

acfsutil sec prepare

Purpose
Prepares an Oracle ACFS file system for security features.

Syntax and Description
```
acfsutil sec prepare -h
acfsutil sec prepare [-u] -m mount_point
```

`acfsutil sec prepare -h` displays help text and exits.

Table 13–50 contains the options available with the `acfsutil sec prepare` command.

Table 13–50   Options for the acfsutil sec prepare command

Option	Description
-m *mount_point*	Specifies the directory where the file system is mounted.
-u	Backs out security for the specified mount point.
	This command removes security from in the file system and reverts the file system to the state before acfsutil sec prepare was run on the file system.
	This command removes all realm-secured files and directories from the realms and then destroys all Oracle ACFS security rules, rule sets and realms from the file system. However, the .Security directory and its contents, including log files and the security metadata backup files, are not deleted.
	If you want to remove encryption and security is being used, then this command must be run before encryption is backed out. To back out encryption, refer to "acfsutil encr set" on page 13-94.

The acfsutil sec prepare command must be run before any of the realm management commands. This command prepares the specified Oracle ACFS file system for security and by default turns security on for the file system.

This command creates the /*mount_point*/.Security, /*mount_point*/.Security/backup, and /*mount_point*/.Security/logs directories where *mount_point* is the option specified in the command line.

This command creates the following system security realms:

- SYSTEM_Logs

  This is a system-created realm to protect the Oracle ACFS security log files in the directory .Security/realm/logs/ directory.

- SYSTEM_SecurityMetadata

  This is a system-created realm to protect the Oracle ACFS metadata XML file in the directory .Security/backup/ directory.

- SYSTEM_Antivirus

  This is a system-created realm that allows access for the antivirus software that is running on an Oracle ACFS file system. For every realm protected file or directory, the SYSTEM_Antivirus realm is evaluated when authorization checks are performed to determine if the SYSTEM_Antivirus realm allows access to the file or directory.

  To allow the antivirus process to access realm-protected files or directories, you must add the LocalSystem or SYSTEM group to the realm with the acfsutil sec realm add command, as shown in Example 13–45. If other antivirus processes are running as Administrator, then the user Administrator must be added to the SYSTEM_Antivirus realm to allow access to realm protected files and directories.

  If no Antivirus products have been installed, do not add any users or groups to the SYSTEM_Antivirus realm. Because users or groups added to the SYSTEM_Antivirus realm have READ and READDIR access, limit the users or groups added to this realm. You can restrict the time window when the users or groups of this realm can access the realm protected files or directories with time-based rules. You can also have application-based rules if you can identify the process name for the antivirus installation that scans the files.

The SYSTEM_Antivirus realm can only perform the following operations on a file or directory: OPEN, READ, READDIR, and setting time attributes. To remove or delete files or directories, you may need to disable security to clean up the infected files.

This realm is set up only for Windows systems.

■ SYSTEM_BackupOperators

This is a system-created realm that can be used to authorize users that can back up realm-secured files and directories. You can add users, groups, rule sets, and command rules to this realm to provide fine-grain authorization for backing up realm-secured files and directories. A user must be added to this realm to back up realm-secured files and directories.

You can add users, groups, rule sets, and command rules to system-created realms with the acfsutil sec realm add command, the same as for user-created realms. However, adding files and directories to system realms is not recommended. You can use the acfsutil sec realm delete command to delete objects from the system-created realms.

System-created security realms cannot be removed by a security administrator with the acfsutil sec admin destroy command. These realms are only removed when security is backed out of a file system when executing the acfsutil sec prepare command with the -u option.

The acfsutil sec prepare -u command is not allowed if any snapshots exist in the file system.

Only a security administrator can run the acfsutil sec prepare command.

Examples
The following example shows the use of the acfsutil sec prepare command.

Example 13–44   Using the acfsutil sec prepare command

```
$ /sbin/acfsutil sec prepare -m /u01/app/acfsmounts/myacfs
```

acfsutil sec realm add

Purpose
Adds objects to an Oracle ACFS security realm.

Syntax and Description
```
acfsutil sec realm add -h
acfsutil sec realm add realm -m mount_point
     { [-u user, ...] [-G os_group,...]
     [-l commandrule:ruleset,commandrule:ruleset,...]
     [-e [-a {AES}] [-k {128|192|256}] ]
     [-f [ -r] path ...] }
```

acfsutil sec realm add -h displays help text and exits.

Table 13–51 contains the options available with the acfsutil sec realm add command.

Table 13–51   Options for the acfsutil sec realm add command

Option	Description
realm	Specifies the realm name to add.

Table 13–51 (Cont.) Options for the acfsutil sec realm add command

Option	Description
-m *mount_point*	Specifies the directory where the file system is mounted.
-u *user*	Specifies user names to add.
-G *os_group*	Specifies the operating system groups to add.
-l *commandrule*:*ruleset*	Specifies the filters to add. The *commandrule* switch is used to add one or more command rules to the realm with a rule set.
	ruleset specifies the rule set associated with the command rule for this realm. Only one rule set can be included with each command rule.
	For a list of command rules, refer to Table 13–52. To display a list of the command rules, use acfsutil sec info with the -c option. Refer to "acfsutil sec info" on page 13-70.
-e	Enables encryption on the realm. Turning encryption on for the realm causes all files contained in the realm to be encrypted. These files remain encrypted until they are no longer part of an encrypted realm.
	Files that are encrypted are not re-encrypted to match the new specified encryption parameters.
-a {AES}	Specifies the encryption algorithm for the realm.
-k {128|192|256}	Specifies the encryption key length.
-f [-r] *path* ...	Adds files specified by *path* to the realm. -r specifies a recursive operation. File paths must be separated by spaces and must be placed at the end of the command.
	If a specified file is not realm secured, the file is encrypted or decrypted to match the encryption status for the realm.

The acfsutil sec realm add command adds objects to the specified realm. The objects to be added include users, groups, command rules, rule sets, and files. If the command encounters an error when adding an object, a message is displayed and the command continues processing the remaining objects.

Multiple entries can be added in a comma separated list when adding users, operating system groups, or command rules. Do not use spaces in the comma separated list. If spaces are added, then enclose the list in quotes.

If the -e option is specified, then encryption must have been initialized for the cluster and set on the file system. For more information, refer to "acfsutil encr init" on page 13-90 and "acfsutil encr set" on page 13-94.

If the entire mount point, which includes the .Security directory, is added to the realm then the security administrator operating system group should be added to the realm to maintain security logging and backing up operations.

The supported command rules are listed in Table 13–52. These command rules restrict or protect against file system operations on realm-secured files and directories.

Table 13–52 Command Rules

Rule	Description
ALL	Protects against all file system operations on files and directories.

Table 13–52 (Cont.) Command Rules

Rule	Description
APPENDFILE	Restricts against additions to the end of a file. Restrictions include writes that start within the current file size, but proceed beyond the end of the file.
CHGRP	Protects from changing the group ownership on a file or directory.
CHMOD	Protects from changing the permissions on a file or directory.
CHOWN	Protects from changing the owner information of a file or directory.
CREATEFILE	Protects from creation of new file in a directory.
DELETEFILE	Protects from deletion of a file from a directory.
EXTEND	Restricts the extension operation of a file size. A file size may still be modifiable with other operations. EXTEND does not protect against a truncate followed by an append operation.
IMMUTABLE	Denies any changes to the files and directories in the realm except changes to extended attributes as a result of commands such as acfsutil tag and acfsutil encr.
	Includes the following protection for a file or directory: APPENDFILE, CHGRP, CHMOD, CHOWN, DELETEFILE, EXTEND, OVERWRITE, RENAME, RMDIR, TRUNCATE, and WRITE.
	IMMUTABLE does not deny any changes to the atime attribute. The atime attribute changes when a user accesses the file.
	Can be set to archive the files and directories in a security realm.
LINKFILE	Restricts the creation of hard links to files.
MKDIR	Protects from the creation of new directory in a directory.
MMAPREAD	Protects a file from being memory mapped for a read operation using mmap() on Linux or using CreateFileMapping followed by MapViewOfFile() on Windows.
MMAPWRITE	Protects a file from being memory mapped for a write operation. Setting MMAPWRITE also protects a file from mapping for read as the operating system maps a file for both read and write.
OPENFILE	Protects from the opening of a file.
OVERWRITE	Prevents existing content in a file from being overwritten with a write operation whose start and end offsets are within the current file size.
	If the operations on a file are truncate followed by append, OVERWRITE does not protect the file. To provide additional protection from both append and overwrite operations, use the WRITE command rule.
READDIR	Restricts for a directory listing, except for use by the security administrator group.
READ	Protects from reading the contents of a file. READ also protects against read operations using mmap(2).
RENAME	Protects against renaming a file or directory.
RMDIR	Protects against removing a directory.
SYMLINK	Restricts the creation of symbolic links in the directories protected by a security realm. When creating symbolic links, it does not matter whether the source file is protected by a security realm.

Table 13–52   (Cont.) Command Rules

Rule	Description
TRUNCATE	Restricts the truncation of a file.
WRITE	Protects a file against the write system call. WRITE also protects against append and overwrite operations, plus write operations using mmap(2).
	A file may still be modifiable with other file operations. To protect the file from other modifications, also use the TRUNCATE and DELETEFILE command rules.

Only a security administrator can run this command.

Examples

Example 13–45 shows the use of the acfsutil sec realm add command. The first acfsutil sec command adds a user group to a security realm. The second and third commands add the LocalSystem or SYSTEM group to the SYSTEM_Antivirus realm in a Windows environment.

Example 13–45   Using the acfsutil sec realm add command

```
$ /sbin/acfsutil sec realm add my_security_realm -m /u01/app/acfsmounts/myacfs
    -G my_os_group

$ /sbin/acfsutil sec realm add SYSTEM_Antivirus /m e: /G "NT AUTHORITY\\SYSTEM"

$ /sbin/acfsutil sec realm add SYSTEM_Antivirus /m e: /G "SYSTEM"
```

acfsutil sec realm clone

Purpose

Clones an Oracle ACFS security realm.

Syntax and Description

```
acfsutil sec realm clone -h
acfsutil sec realm clone realm -s src_mount_point new_realm
    [-e] [-f] [-G] [-l] [-u]
acfsutil sec realm clone realm -s src_mount_point
    [new_realm] -d destination_mount_point
    [-e] [-G] [-l] [-u]
```

acfsutil sec realm clone -h displays help text and exits.

Table 13–53 contains the options available with the acfsutil sec realm clone command.

Table 13–53   Options for the acfsutil sec realm clone command

Option	Description
realm	Specifies the realm name to be cloned.
-s src_mount_point	Specifies the directory where the source file system is mounted.
new_realm	Specifies the new realm name.

Table 13–53   (Cont.) Options for the acfsutil sec realm clone command

Option	Description
`-d destination_mount_ point`	Specifies the directory for the destination mount point for the new realm.
`-e`	Copy encryption attributes to the new realm.
`-f`	Copy file objects to the new realm.
`-G`	Copy operating system groups to the new realm.
`-l`	Copy filters to the new realm.
`-u`	Copy users to the new realm.

The `acfsutil sec realm clone` makes a copy of the specified realm in the destination mount point. If the source and mount points are different and the new realm name is not specified, then the realm is cloned using the existing realm name in the Oracle ACFS file system specified by destination mount point. If the destination mount point is not specified, then the cloned realm is located in the source mount point and a new unique realm name must be specified.

If the `-l` option is specified and the destination mount point is different than the source mount point, then the rules and rule sets must be cloned first.

If the `-e` option is specified and the destination mount point is different than the source mount point, then encryption must be set on destination mount point. For more information, refer to "acfsutil encr set" on page 13-94.

The `-f` option can only be used if the destination mount point is the same as the source mount point.

Only a security administrator can run this command.

Examples
The following example shows the use of the `acfsutil sec realm clone` command.

Example 13–46   Using the acfsutil sec realm clone command

```
$ /sbin/acfsutil sec realm clone my_security_realm -s /u01/app/acfsmounts/myacfs
      my_new_security_realm -d /u02/app/acfsmounts/myacfs -G
```

acfsutil sec realm create

Purpose
Creates an Oracle ACFS security realm.

Syntax and Description
```
acfsutil sec realm create -h
acfsutil sec realm create realm -m mount_point
     -e { on -a {AES} -k {128|192|256} | off }
     [-o {enable|disable}] [-d "description"]
```

`acfsutil sec realm create -h` displays help text and exits.

Table 13–54 contains the options available with the `acfsutil sec realm create` command.

Table 13–54   Options for the acfsutil sec realm create command

Option	Description
`realm`	Specifies the realm name.
`-m mount_point`	Specifies the mount point for the file system. A mount point is specified as a path on Linux/Unix.
`-e {on|off}`	Specifies encryption on or off for the realm.
`-a {AES}`	Specifies the encryption algorithm.
`-k {128|192|256}`	Specifies the encryption key length.
`-o {enable|disable}`	Specifies where security is on or off for the realm.
`-d "description"`	Specifies a realm description.

The `acfsutil sec create realm` creates a new realm in the specified Oracle ACFS file system. The new realm name must be unique in the file system identified by the mount point.

A maximum of 500 Oracle ACFS security realms can be created, including any default system realms created by the `acfsutil sec prepare` command.

The realm is enabled by default unless the `-o disable` option is specified.

If the `-e on` option is specified, then encryption must have been initialized for the cluster and set on the file system. For more information, refer to "acfsutil encr init" on page 13-90 and "acfsutil encr set" on page 13-94.

If the `-e off` option is specified, you cannot specify the `-a` and `-k` options.

Only a security administrator can run this command.

Examples

The following example shows the use of the `acfsutil sec realm create` command.

Example 13–47   Using the acfsutil sec realm create command

```
$ /sbin/acfsutil sec realm create my_security_realm -m /u01/app/acfsmounts/myacfs
    -e on -a AES -k 192 -o enable
```

acfsutil sec realm delete

Purpose

Deletes objects from an Oracle ACFS security realm.

Syntax and Description

```
acfsutil sec realm delete -h
acfsutil sec realm delete realm -m mount_point
    { [-u user, ...] [-G os_group,...]
    [-l commandrule:ruleset,commandrule:ruleset,...]
    [-f [ -r] path,...] ] [-e ] }
```

`acfsutil sec realm delete -h` displays help text and exits.

Table 13–55 contains the options available with the `acfsutil sec realm delete` command.

Table 13–55   Options for the acfsutil sec realm delete command

Option	Description
realm	Specifies the realm name.
-m *mount_point*	Specifies the directory where the file system is mounted.
-u *user*	Specifies user names to delete.
-G *os_group*	Specifies the operating system groups to delete.
-l *commandrule*:*ruleset*	Specifies the filters to delete from the realm. To display a list of the command rules, use acfsutil sec info with the -c option. *ruleset* specifies the rule set associated with the command rule for this realm.
-f [-r] *path* ...	Deletes files specified by *path* from the realm. -r specifies a recursive operation. File paths must be separated by spaces. If this is the last realm securing the file, the file is encrypted or decrypted to match the file system level encryption state.
-e	Disables encryption on the realm. When disabling encryption, this option decrypts any files in the realm that do not belong to any other encrypted realms. If a file is part of another realm which is encrypted or if encryption is turned on for the file system, then the file remains encrypted.

The acfsutil sec realm delete command removes objects from the specified realm. The objects to be deleted include users, groups, rule sets, and files. If the command encounters an error when deleting an object, a message is displayed and the command continues processing the remaining objects.

Multiple entries can be added in a comma separated list when adding users, operating system groups, or command rules. Do not use spaces in the comma separated list. If spaces are added, then enclose the list in quotes.

Only a security administrator can run this command.

Examples
The following example shows the use of the acfsutil sec realm delete command.

Example 13–48   Using the acfsutil sec realm delete command

```
$ /sbin/acfsutil sec realm delete my_security_realm -m /u01/app/acfsmounts/myacfs
    -f -r /u01/app/acfsmounts/myacfs/myoldfiles/*.log
```

acfsutil sec realm destroy

Purpose
Destroys an Oracle ACFS security realm.

Syntax and Description
```
acfsutil sec realm destroy -h
acfsutil sec realm destroy realm -m mount_point
```

acfsutil sec realm destroy -h displays help text and exits.

Table 13–56 contains the options available with the acfsutil sec realm destroy command.

Table 13–56   Options for the acfsutil sec realm destroy command

Option	Description
realm	Specifies the realm name.
-m *mount_point*	Specifies the directory where the file system is mounted.

The `acfsutil sec destroy realm` removes a security realm from the specified Oracle ACFS file system. Destroying the realm does not destroy the objects in the realm; this command simply removes the security associated with the realm from the objects.

Only a security administrator can run this command.

Examples
The following example shows the use of the `acfsutil sec realm destroy` command.

Example 13–49   Using the acfsutil sec realm destroy command

```
$ /sbin/acfsutil sec realm destroy my_security_realm -m /u01/app/acfsmounts/myacfs
```

acfsutil sec rule clone

Purpose
Clones a security rule.

Syntax and Description
```
acfsutil sec rule clone -h
acfsutil sec rule clone rule -s src_mount_point new_rule
acfsutil sec rule clone rule -s src_mount_point
     [new_rule] -d mount_point
```

`acfsutil sec rule clone -h` displays help text and exits.

Table 13–57 contains the options available with the `acfsutil sec rule clone` command.

Table 13–57   Options for the acfsutil sec rule clone command

Option	Description
rule	Specifies the existing name of the rule. If the name contains a space, enclose in quotes (" ").
-s *src_mount_point*	Specifies the directory where the source file system is mounted.
-d *mount_point*	Specifies the directory for the destination mount point of the file system.
new_rule	Specifies the new name of the rule. If the name contains a space, enclose in quotes (" ").

If the source and mount points are different and the new rule name is not specified, then the rule is cloned using the existing rule name in the Oracle ACFS file system specified by destination mount point. If the destination mount point is not specified, then the cloned rule is located in the source mount point and a new unique rule name must be specified.

Only a security administrator can run this command.

Examples

The following example shows the use of the `acfsutil sec rule clone` command.

Example 13–50   Using the acfsutil sec rule clone command

```
$ /sbin/acfsutil sec rule clone my_security_rule -s /u01/app/acfsmounts/myacfs
      my_new_security_rule -d /u02/app/acfsmounts/myacfs
```

acfsutil sec rule create

Purpose

Creates a security rule.

Syntax and Description

```
acfsutil sec rule create -h
acfsutil sec rule create rule -m mount_point
    -t rule_type rule_value
    [-o {ALLOW|DENY}]
```

`acfsutil sec rule create -h` displays help text and exits.

Table 13–58 contains the options available with the `acfsutil sec rule create` command.

Table 13–58   Options for the acfsutil sec rule create command

Option	Description
rule	Specifies the name of the rule. If the name contains a space, enclose in quotes (`" "`).
`-m` *mount_point*	Specifies the directory where the file system is mounted.
`-t` *rule_type rule_value*	Specifies a rule type and a rule value. The rule type can be `application`, `hostname`, `time`, or `username`. The rule value depends on the type of rule. The valid rule types and values are described in this section.
`-o` *option*	Specifies options preceded by `-o`. The option specified can be `ALLOW` or `DENY`. The default value is `DENY`.

The `acfsutil sec rule create` command creates a new rule in the Oracle ACFS file system specified by the mount point. The new rule can be added to a rule set and that rule set can be added to a security realm.

A maximum of 500 Oracle ACFS security rules can be created.

The rule types and associated rule values are:

- `application`

  This rule type specifies the name of an application which is allowed or denied access to the objects protected by a realm.

- `hostname`

  This rule type specifies the name of a machine from which a user accesses the objects protected by a realm. Access from a node can be allowed or denied using this rule. The `hostname` should be one of the cluster node names and not any other external nodes which could have mounted the Oracle ACFS file system as a network File System (NFS) mount.

- time

    This rule type specifies the time interval in the form *start_time,end_time*. This time interval specifies access to a realm. Access can be allowed or denied to objects protected by a realm only during certain times of the day by setting this rule in a realm. The time is based on the local time of the host.

- username

    This rule type specifies the name of a user to be added or deleted from a realm. This option can be used to deny access for any user that belongs to a security group that is part of a realm.

Only a security administrator can run this command.

Examples

The following example shows the use of the `acfsutil sec rule create` command.

Example 13–51   Using the acfsutil sec rule create command

```
$ /sbin/acfsutil sec rule create my_security_rule -m /u01/app/acfsmounts/myacfs
      -t username security_user_one -o ALLOW
```

acfsutil sec rule destroy

Purpose

Removes a security rule.

Syntax and Description

```
acfsutil sec rule destroy -h
acfsutil sec rule destroy rule -m mount_point
```

`acfsutil sec rule destroy -h` displays help text and exits.

Table 13–59 contains the options available with the `acfsutil sec rule destroy` command.

Table 13–59   Options for the acfsutil sec rule destroy command

Option	Description
rule	Specifies the name of the rule. If the name contains a space, enclose in quotes (`" "`).
-m *mount_point*	Specifies the directory where the file system is mounted.

The `acfsutil sec rule destroy` command removes a rule from the rule sets in the Oracle ACFS file system specified by the mount point. A rule set is not destroyed if all the rules are destroyed. The empty rule set must be explicitly destroyed.

Only a security administrator can run this command.

Examples

The following example shows the use of the `acfsutil sec rule destroy` command.

Example 13–52   Using the acfsutil sec rule destroy command

```
$ /sbin/acfsutil sec rule destroy my_security_rule -m /u01/app/acfsmounts/myacfs
```

acfsutil sec rule edit

Purpose

Updates a security rule.

Syntax and Description

```
acfsutil sec rule edit -h
acfsutil sec rule edit rule -m mount_point
      { [ -t rule_type rule_value ] [ -o {ALLOW|DENY}] }
```

`acfsutil sec rule edit -h` displays help text and exits.

Table 13–60 contains the options available with the `acfsutil sec rule edit` command.

Table 13–60   Options for the acfsutil sec rule edit command

Option	Description
rule	Specifies the name of the rule. If the name contains a space, enclose in quotes (" ").
-m *mount_point*	Specifies the directory where the file system is mounted.
-t *rule_type rule_value*	Specifies a rule type and a rule value. The rule type can be `application`, `hostname`, `time`, or `username`. Rule value depends on the type of rule. For information on the rule type and rule value, refer to "acfsutil sec rule create" on page 13-83.
-o *option*	Specifies options preceded by -o. The option specified can be `ALLOW` or `DENY`.

The `acfsutil sec rule edit` updates a rule.

Only a security administrator can run this command.

Examples

The following example shows the use of the `acfsutil sec rule edit` command.

Example 13–53   Using the acfsutil sec rule edit command

```
$ /sbin/acfsutil sec rule edit my_security_rule -m /u01/app/acfsmounts/myacfs
      -t username security_user_three -o ALLOW
```

acfsutil sec ruleset clone

Purpose

Clones a security rule set.

Syntax and Description

```
acfsutil sec ruleset clone -h
acfsutil sec ruleset clone ruleset -s mount_point new_ruleset
acfsutil sec ruleset clone ruleset -s mount_point
      [new_ruleset] -d mount_point
```

`acfsutil sec ruleset clone -h` displays help text and exits.

Table 13–61 contains the options available with the `acfsutil sec ruleset clone` command.

Table 13–61   Options for the acfsutil sec ruleset clone command

Option	Description
rule_set	Specifies the existing name of the rule set. If the name contains a space, enclose in quotes (" ").
-s mount_point	Specifies the directory where the source file system is mounted.
-d mount_point	Specifies the directory for the destination mount point of the file system.
new_rule_set	Specifies the new name of the rule set. If the name contains a space, enclose in quotes (" ").

If the source mount point is different from destination mount point, the rules in the rule set must be cloned first.

If the source and mount points are different and the new rule set name is not specified, then the rule set is cloned using the existing rule set name in the Oracle ACFS file system specified by destination mount point. If the destination mount point is not specified, then the cloned rule set is located in the source mount point and a new unique rule set name must be specified.

Only a security administrator can run this command.

Examples

The following example shows the use of the acfsutil sec ruleset clone command.

Example 13–54   Using the acfsutil sec ruleset clone command

```
$ /sbin/acfsutil sec ruleset clone
     my_security_ruleset -s /u01/app/acfsmounts/myacfs
     my_new_security_ruleset -d /u02/app/acfsmounts/myacfs
```

acfsutil sec ruleset create

Purpose

Creates a security rule set.

Syntax and Description

```
acfsutil sec ruleset create -h
acfsutil sec ruleset create rule_set -m mount_point
     [-o {ALL_TRUE|ANY_TRUE}]
```

acfsutil sec ruleset create -h displays help text and exits.

Table 13–62 contains the options available with the acfsutil sec ruleset create command.

Table 13–62   Options for the acfsutil sec ruleset create command

Option	Description
rule_set	Specifies the name of the rule set. If the name contains a space, enclose in quotes (" ").
-m mount_point	Specifies the directory where the file system is mounted.
-o option	Specifies options preceded by -o. The option specified can be ALL_TRUE or ANY_TRUE. The default value is ALL_TRUE.

The `acfsutil sec ruleset create` command creates a new rule set in the specified mount point.

A maximum of 500 Oracle ACFS security rule sets can be created.

Only a security administrator can run this command.

Examples

The following example shows the use of the `acfsutil sec ruleset create` command.

Example 13–55 Using the acfsutil sec ruleset create command

```
$ /sbin/acfsutil sec ruleset create
      my_security_ruleset -m /u01/app/acfsmounts/myacfs -o ANY_TRUE
```

acfsutil sec ruleset destroy

Purpose

Removes a security rule set.

Syntax and Description

```
acfsutil sec ruleset destroy -h
acfsutil sec ruleset destroy rule_set -m mount_point
```

`acfsutil sec ruleset destroy -h` displays help text and exits.

Table 13–63 contains the options available with the `acfsutil sec ruleset destroy` command.

Table 13–63 Options for the acfsutil sec ruleset destroy command

Option	Description
rule_set	Specifies the name of the rule set. If the name contains a space, enclose in quotes (" ").
-m *mount_point*	Specifies the directory where the file system is mounted.

The `acfsutil sec ruleset destroy` command removes a rule set from the Oracle ACFS file system specified by the mount point. Only a security administrator can run this command.

Examples

The following example shows the use of the `acfsutil sec ruleset destroy` command.

Example 13–56 Using the acfsutil sec ruleset destroy command

```
$ /sbin/acfsutil sec ruleset destroy
      my_security_ruleset -m /u01/app/acfsmounts/myacfs
```

acfsutil sec ruleset edit

Purpose

Updates a security rule set.

Syntax and Description

```
acfsutil sec ruleset edit -h
acfsutil sec ruleset edit rule_set -m mount_point
      { [-a rule,...] [-d rule,...] [-o {ALL_TRUE|ANY_TRUE}]}
```

`acfsutil sec ruleset edit -h` displays help text and exits.

Table 13–64 contains the options available with the `acfsutil sec ruleset edit` command.

Table 13–64   Options for the acfsutil sec ruleset edit command

Option	Description
rule_set	Specifies the name of the rule set. If the name contains a space, enclose in quotes (`" "`).
-m *mount_point*	Specifies the directory where the file system is mounted.
-a *rule*	Specifies the rule to add.
-d *rule*	Specifies the rule to remove.
-o *option*	Specifies options preceded by `-o`. The option specified can be `ALL_TRUE` or `ANY_TRUE`.

The `acfsutil sec ruleset edit` command updates a rule set in the Oracle ACFS file system specified by the mount point.

Only a security administrator can run this command.

Examples

The following example shows the use of the `acfsutil sec ruleset edit` command.

Example 13–57   Using the acfsutil sec ruleset edit command

```
$ /sbin/acfsutil sec ruleset edit
      my_security_ruleset -m /u01/app/acfsmounts/myacfs
      -a my_new_rule -o ANY_TRUE
```

acfsutil sec save

Purpose

Saves Oracle ACFS file system security metadata.

Syntax and Description

```
acfsutil sec save -h
acfsutil sec save -m mount_point -p file
```

`acfsutil sec save -h` displays help text and exits.

Table 13–65 contains the options available with the `acfsutil sec save` command.

Table 13–65   Options for the acfsutil sec save command

Option	Description
-m *mount_point*	Specifies the directory where the file system is mounted.
-p *file*	Specifies a file name to store the security metadata. The file is saved in the `/mount_point/.Security/backup/` directory.

The `acfsutil sec save` command saves the security metadata for an Oracle ACFS file system to an XML file. By default, the file is saved in the `/mount_point/.Security/backup` directory.

This file can be backed up as a regular file by a backup application. System realms protect this file and allow only members of these realms to access this file and prevent all other users including the root user and system administrator from access. For information about the system-created security realms, refer to "acfsutil sec prepare" on page 13-73.

Only a security administrator can run this command.

Examples

The following example shows the use of the `acfsutil sec save` command.

Example 13–58   Using the acfsutil sec save command

```
$ /sbin/acfsutil sec save -m /u01/app/acfsmounts/myacfs -p my_metadata_file.xml
```

Oracle ACFS Command-Line Tools for Encryption

Table 13–66 contains a summary of the commands for Oracle ACFS encryption.

You can run `acfsutil help` on all platforms to display help text. You can run `acfsutil version` on all platforms to display the Oracle ACFS version.

When the options are entered with commands on a Windows platform, use `/` instead of `-` with the option. For example, you can display help for `acfsutil` on a Linux platform with `acfsutil -h`. On a Windows platform, use `acfsutil /h`.

Note that a mount point on a Windows operating system can be a drive letter or a directory including the drive letter.

Table 13–66   Summary of commands for Oracle ACFS encryption

Command	Description
acfsutil encr info	Displays encryption-related information about Oracle ACFS file systems.
acfsutil encr init	Creates storage for encryption keys.
acfsutil encr off	Disables encryption for an Oracle ACFS file system.
acfsutil encr on	Encrypts an Oracle ACFS file system.
acfsutil encr rekey	Generates a new key and re-encrypts an Oracle ACFS file system.
acfsutil encr set	Sets or changes encryption parameters for an Oracle ACFS file system.

acfsutil encr info

Purpose

Displays encryption-related information about Oracle ACFS file systems, directories, or files.

Syntax and Description

```
acfsutil encr info -h
acfsutil encr info -m mount_point [[-r] path [path...]]
```

`acfsutil encr info -h` displays help text and exits.

Table 13–67 contains the options available with the `acfsutil encr info` command.

Table 13–67   Options for the acfsutil encr info command

Option	Description
-m *mount_point*	Specifies the directory where the file system is mounted.
-r	Specifies recursive action under an existing directory folder identified by *path*.
path	Specifies the absolute or relative path of a directory. Multiple path values are allowed.

If -m is specified without a *path*, the encryption status, algorithm, and key length are displayed for the file system level.

If -r is specified with a *path*, the encryption status, algorithm, and key length are displayed for all objects under the directory specified by *path*.

The `acfsutil encr info` command displays encryption status and parameters for files in a snapshot if the files are specified with the *path* option.

This command fails when it is run on realm-secured objects.

Any user can run this command to display encryption information about a file system, directory, or file.

Examples

The following are examples of the use of `acfsutil encr info`.

Example 13–59   Using the acfsutil encr info command

```
# /sbin/acfsutil encr info -m /u01/app/acfsmounts/myacfs

# /sbin/acfsutil encr info -m /u01/app/acfsmounts/myacfs
                           -r /u01/app/acfsmounts/myacfs/myfiles
```

acfsutil encr init

Purpose
Creates storage for encryption keys.

Syntax and Description
acfsutil encr init -h
acfsutil encr init [-p]

`acfsutil encr init -h` displays help text and exits.

Table 13–68 contains the options available with the `acfsutil encr init` command.

Table 13–68   Options for the acfsutil encr init command

Option	Description
-p	Creates PKCS (password-protected) storage for keys.

The `acfsutil encr init` command must be run before any other encryption `acfsutil` commands can be run. This command must be run once for each cluster on which Oracle ACFS encryption is run.

If the `-p` option is specified, you must provide a password when prompted. The password must conform to the format that is described in "acfsutil sec init" on page 13-71.

If the `-p` option is not specified, a single sign-on (SSO) wallet is created.

Only a user with root or system administrator privileges can run this command.

Examples

The following is an example of the use of `acfsutil encr init`.

Example 13–60   Using the acfsutil encr init command

```
# /sbin/acfsutil encr init
```

acfsutil encr off

Purpose

Disables encryption for an Oracle ACFS file system, directories, or individual files.

Syntax and Description

```
acfsutil encr off -h
acfsutil encr off -m mount_point [[-r] path [ path...]]
```

`acfsutil encr off -h` displays help text and exits.

Table 13–70 contains the options available with the `acfsutil encr off` command.

Table 13–69   Options for the acfsutil encr off command

Option	Description
`-m mount_point`	Specifies the directory where the file system is mounted.
`-r`	Specifies to disable encryption recursively under an existing directory identified by `path`.
`path`	Specifies the absolute or relative path of a directory. Multiple path values are allowed.

This command cannot be run on security realm-protected files.

Only an administrator can run this command on an Oracle ACFS file system (`-m` option without a `path` specified). When the `-m` option is specified without a `path`, all the files under the mount point are decrypted.

The `path` option can specify a path to a file or directory in a read-write snapshot. If the `-r` option is specified with the command on the root directory, the command does not transverse the snapshots under the `.ACFS` directory. If a decryption operation is specified at the file system level, then the operation does not process files and directories of snapshots in the `.ACFS/snaps/` directory.

Only a user with root or system administrator privileges can run this command to disable encryption on a file system. The file owner can also run this command to disable encryption on a directory or file.

Examples

The following are examples of the use of `acfsutil encr off`.

Example 13–61   Using the acfsutil encr off command

```
# /sbin/acfsutil encr off -m /u01/app/acfsmounts/myacfs

# /sbin/acfsutil encr off -m /u01/app/acfsmounts/myacfs
                          -r /u01/app/acfsmounts/myacfs/myfiles
```

acfsutil encr on

Purpose

Encrypts an Oracle ACFS file system, directories, or individual files.

Syntax and Description

```
acfsutil encr on -h
acfsutil encr on -m mount_point
     [-a {AES} -k {128|192|256}] [[-r] path [ path...]]
```

`acfsutil encr on -h` displays help text and exits.

Table 13–70 contains the options available with the `acfsutil encr on` command.

Table 13–70   Options for the acfsutil encr on command

Option	Description
-m mount_point	Specifies the directory where the file system is mounted.
-a algorithm	Specifies the encryption algorithm type for a directory or file. Advanced Encryption Standard (AES) is the only encryption algorithm supported for this release.
-k key_length	Specifies the encryption key length for a directory or file.
-r	Specifies encryption recursively under existing directory folder identified by path.
path	Specifies the absolute or relative path of a directory. Multiple path values are allowed.

This command cannot be run on realm-protected files.

The default values for the -a and -k are determined by the volume parameters specified when `acfsutil encr set` was run. To set the key length at the volume level, use the `acfsutil encr set` command.

Only an administrator can run this command on an Oracle ACFS file system (-m option without a path specified). When the -m option is specified without a path, all the files under the mount point are encrypted.

The path option can specify a path to a file or directory in a read-write snapshot. If the -r option is specified with the command on the root directory, the command does not transverse the snapshots under the .ACFS directory. If an encryption operation is specified at the file system level, then the operation does not process files and directories of snapshots in the .ACFS/snaps/ directory.

Only a user with root or system administrator privileges can run this command to enable encryption on a file system. The file owner can also run this command to enable encryption on a directory or file.

Examples

The following are examples of the use of `acfsutil encr on`.

Example 13–62   Using the acfsutil encr on command

```
# /sbin/acfsutil encr on -m /u01/app/acfsmounts/myacfs

# /sbin/acfsutil encr on -m /u01/app/acfsmounts/myacfs
                         -a AES -k 128 -r /u01/app/acfsmounts/myacfs/myfiles
```

acfsutil encr rekey

Purpose

Generates a new key and re-encrypts volume or file.

Syntax and Description

```
acfsutil encr rekey -h
acfsutil encr rekey -m mount_point
     {-f [-r] path [path...] | -v } [-a {AES} -k {128 | 192 | 256}]
```

`acfsutil encr rekey -h` displays help text and exits.

Table 13–71 contains the options available with the `acfsutil encr rekey` command.

Table 13–71   Options for the acfsutil encr rekey command

Option	Description
`-m mount_point`	Specifies the directory where the file system is mounted.
`-f [-r] path ...`	Generates a new file encryption key for the specified path and then encrypts the data with the new key.
	If -r is specified, the rekey operation is performed recursively under `path`.
	`path` specifies the absolute or relative path of a directory. Multiple path values are allowed.
`-v`	Generates a new volume encryption key (VEK) for the specified mount point and then encrypts all the file encryption keys in file system with the new key. Prompts for the wallet password because the wallet must be accessed to store the new VEK.
`-a algorithm`	Specifies the algorithm. Advanced Encryption Standard (AES) is the only encryption supported for this release.
`-k key_length`	Specifies the key length for the directory or file specified by `path`.

This command cannot be run on security realm-protected files.

The default values for the `-a` and `-k` are determined by the volume parameters specified when `acfsutil encr set` was run.

The `path` option can specify a path to a file or directory in a read-write snapshot. If the `-r` option is specified with the command on the root directory, the command does not transverse the snapshots under the `.ACFS` directory. If a rekey operation is specified at the file system level, then the operation does not process files and directories of snapshots in the `.ACFS/snaps/` directory.

Only a user with root or system administrator privileges can run this command with the -v option. The file owner can also run this command with the -f option to rekey encryption on the directory or file.

You should back up the Oracle Cluster Registry (OCR) after running this command with the -v option to ensure there is an OCR backup that contains all of the volume encryption keys (VEKs) for the file system.

Examples

The following are examples of the use of acfsutil encr rekey.

Example 13–63   Using the acfsutil encr rekey command

```
# /sbin/acfsutil encr rekey -m /u01/app/acfsmounts/myacfs -v

# /sbin/acfsutil encr rekey -m /u01/app/acfsmounts/myacfs -f
                            -r /u01/app/acfsmounts/myacfs/myfiles
```

acfsutil encr set

Purpose

Sets or changes encryption parameters for an Oracle ACFS file system.

Syntax and Description

```
acfsutil encr set -h
acfsutil encr set [-a {AES } -k {128|192|256} | -u] -m mount_point
```

acfsutil encr set -h displays help text and exits.

Table 13–72 contains the options available with the acfsutil encr set command.

Table 13–72   Options for the acfsutil encr set command

Option	Description
-a algorithm	Specifies the algorithm. Advanced Encryption Standard (AES) is the default value and the only encryption supported for this release. The algorithm must be specified if -k is specified.
-k {128|192|256}	Specifies the key length. The key length is set at the volume level. The default is 192. Must be specified if -a is specified.
-u	Backs out encryption. Decrypts all encrypted files in the file system and reverts the file system to the state before acfsutil encr set was run on the file system. If security is being used, then this command can only be run after security has been backed out. To remove security, refer to "acfsutil sec prepare" on page 13-73.
-m mount_point	Specifies the directory where the file system is mounted.

Before running the acfsutil encr set command, you must first run the acfsutil encr init command.

The acfsutil encr set command configures encryption parameters for a file system, transparently generates a volume encryption key, and stores that the generated key in the key store that was previously configured with the acfsutil encr init command.

In addition `acfsutil encr set` creates the `mount_point/.Security/encryption/logs/` directory that contains the log file (`encr-host_name.log`) that collects auditing and diagnostic data.

Password requirements when storing the key are dependent on how the encryption key storage was configured. If `-p` was specified with `acfsutil encr init`, then a password is required to run this command.

The `acfsutil encr set -u` command is not allowed if any snapshots exist in the file system.

Only a user with root or system administrator privileges can run the `acfsutil encr set` command.

You should back up the Oracle Cluster Registry (OCR) after running this command to ensure there is an OCR backup that contains all of the volume encryption keys (VEKs) for the file system.

Examples

The following example shows the use of `acfsutil encr set` command.

Example 13–64   Using the acfsutil encr set command

```
# /sbin/acfsutil encr set -a AES -k 256 -m /u01/app/acfsmounts/myacfs
```

Oracle ACFS Command-Line Utilities for Multiple Environments

Table 13–73 contains a summary of the Oracle ACFS command utilities for multiple environments.

You can use `acfsutil help` on all platforms to display help text. You can run `acfsutil version` on all platforms to display the Oracle ACFS version.

When the options are entered with commands on a Windows platform, use `/` instead of `-` with the option. For example, you can display help for `acfsutil` on a Linux platform with `acfsutil -h`. On a Windows platform, use `acfsutil /h`.

Note that a mount point on a Windows operating system can be a drive letter or a directory including the drive letter.

Table 13–73   Summary of Oracle ACFS command utilities for multiple environments

Command	Description
acfsdbg	Debugs an Oracle ACFS file system.
acfsutil info file	Displays information for a file in an Oracle ACFS file system.
acfsutil info fs	Displays detailed Oracle ACFS file system information.
acfsutil info id	Displays Oracle ACFS file system information for a given identifier and mount point.
acfsutil registry	Registers an Oracle ACFS file system with the Oracle ACFS mount registry.
acfsutil rmfs	Removes an Oracle ACFS file system.
acfsutil size	Resizes an Oracle ACFS file system.
acfsutil snap create	Creates a snapshot of an Oracle ACFS file system.
acfsutil snap delete	Deletes a snapshot of an Oracle ACFS file system.
acfsutil snap info	Displays information about Oracle ACFS file system snapshots.

Table 13–73   (Cont.) Summary of Oracle ACFS command utilities for multiple environments

Command	Description
acfsutil tune	Modifies or displays Oracle ACFS tunable parameters.
advmutil canonical	Displays the canonical name of an Oracle ADVM volume.
advmutil tune	Modifies or displays Oracle ADVM parameters.
advmutil volinfo	Displays information about Oracle ADVM volumes.

acfsdbg

Purpose

Debugs an Oracle ACFS file system.

Syntax and Description

```
acfsdbg -h
acfsdbg [-r] [-l] volume_device
```

Table 13–74 contains the options available with the `acfsdbg` command.

Table 13–74   Options for the acfsdbg command

Option	Description
-h	Prints out the usage message which displays the various options that are available when invoking the `acfsdbg` command, then exits.
-r	Operates in read-only mode. No data is modified on the file system and all write commands are disabled. If the device is mounted anywhere, `acfsdbg` may not display the latest data because some of it is cached by the file system mounts.
-l	Processes kernel log files. The default is to not process the log files.
volume_device	Specifies the device name of the volume.

`acfsdbg` is an extension to `fsck` and `acfschkdsk`, and is an interactive program that views and modifies on disk structures of the file system. This command is intended for experienced development and support engineers to examine on disk structures to diagnose problems. Use with caution.

When `acfsdbg` is started, it displays a command prompt. At the command prompt, you can enter the subcommands listed in Table 13–75.

You can also use `acfsdbg` for scripts by echoing `acfsdbg` subcommands with a shell pipe to the `acfsdbg` binary.

By default the file system is not modified when running the tool. If the `-l` option is used, the file system metadata in the transaction logs is applied before the interactive disk block dump session. The volume device must specify a volume with a dismounted Oracle ACFS file system. If the volume device has a mounted Oracle ACFS file system, `acfsdbg` displays an error message and exits.

You must be the administrator or a member of the Oracle ASM administrator group to run `acfsdbg`.

Subcommands

Table 13–75 lists the subcommands of acfsdbg.

Table 13–75 Subcommands for acfsdbg

Option	Description	Syntax
calculate	Calculates simple arithmetic expressions Valid operators: + - * / % & | ^ ~ << >> White space starts a new expression 0-1 represents a negative 1	calculate [-v] expr […] -v Verbose mode expr Simple 2+2 expression
cksum	Generates and replaces checksum in header Header offset can be an expression as used by the calculate subcommand White space starts a new header offset Command is disabled in read-only mode	cksum [-C | -CE] header_offset […] -C Regenerate for normal structure checksum -CE Re-generate for Extent structure checksum header_offset Offset of the on disk structure header. The value can be an expression as used by the calculate subcommand
close	Closes the open handle to the device	close
echo	Echoes text on command line to stdout	echo
fenum	Displays the specified File Entry TAble (FETA) entry	fenum [-f | -e | -d] FETA_entry_number -f Displays all on disk structures related to this structure -e Displays all on disk extent information related to this structure -d Casts the structure as a directory and displays its contents FETA_entry_number The File Entry Table number used to identify a file on the file system
help	Displays help message	help
offset	Displays structure at disk offset	offset [-c cast] [-f | -d] disk_offset -f Displays all on disk structures related to this structure -d Casts the structure as a directory and displays its contents disk_offset Disk offset to display. The value can be an expression as used by the calculate subcommand
open	Opens a handle to a device. The default is the volume device name entered on the command line	open [volume_device]
primary	Sets the context of commands to the primary file system	primary
prompt	Sets the prompt to the specified string	prompt "prompt_string"
quit	Exits the acfsdbg debugger command	quit

Table 13–75   (Cont.) Subcommands for acfsdbg

Option	Description	Syntax
read	Reads value from offset The default size to read in is 8 bytes The default count to read is 1	read [-1 | -2 | -4 | -8 | -s] [*count*] *offset* -1 Read byte value -2 Read 2 byte (short) value -4 Read 4 byte (int) value -8 Read 8 byte (long) value -s Read null- terminated string *count* Number of values to read. If not specified, the default is 1 *offset* Disk offset to read. The value can be an expression as used by the calculate subcommand
snapshot	Sets the context of commands to the specified snapshot	snapshot *snapshot_name*
write	Writes hexadecimal, octal, or decimal values at the disk offset, estimating how many bytes to write based on value size or number of digits in leading 0 hexadecimal values The disk offset can be an expression used by the calculate subcommand Numeric values can also be an expression as used by the calculate subcommand This command is disabled in read-only mode	write [-1 | -2 | -4 | -8 | -c | -s] [-C | -CE] *offset value* -1 Write byte value -2 Write 2 byte (short) value -4 Write 4 byte (int) value -8 Write 8 byte (long) value -c Write text (no null termination). Enclose string in single-quotes (') -s Write null-terminated string. Enclose string in quotes (") -C Regenerate normal structure checksum -CE Regenerate extent structure checksum *offset* Disk offset to write. The value can be an expression used by the calculate subcommand *value* The value to write. If numeric, the value can be an expression as used by the calculate subcommand

Examples

Example 13–65 shows the use of the acfsdbg subcommand.

Example 13–65   Using the acfsdbg command

```
$ /sbin/acfsdbg /dev/asm/voume1-123
acfsdbg: version              = 11.2.0.3.0
Oracle ASM Cluster File System (ACFS) On-Disk Structure Version: 39.0
The ACFS volume was created at  Mon Mar  2 14:57:45 2011
acfsdbg>

acfsbdg> calculate 60*1024
    61,440
    61440
    61440
    0xf000
    0170000
    1111:0000:0000:0000

acfsdbg> prompt "acfsdbg test>"
acfsdbg test>

echo "offset 64*1024" | acfsdbg /dev/asm/volume1-123
```

acfsutil info file

Purpose

Displays information for a file in an Oracle ACFS file system.

Syntax and Description

```
acfsutil info file -h
acfsutil info file path
```

acfsutil info file -h displays help text and exits.

Table 13–76 contains the options available with the acfsutil info file command.

Table 13–76   Options for the acfsutil info file command

Option	Description
path	Specifies the full path name of a file.

The acfsutil info file command displays information about a file in an Oracle ACFS file system. The information includes the extent map which details the locations of the blocks comprising the file and the used versus allocated storage for a file. Tagging information is also displayed for a file.

You must have read access to the specified file to run acfsutil info file.

Examples

The following is an example of acfsutil info file.

Example 13–66   Using the acfsutil info file command

```
$ /sbin/acfsutil info file /u01/app/acfsmounts/myacfs/myfile
```

acfsutil info fs

Purpose

Displays detailed Oracle ACFS file system information.

Syntax and Description

```
acfsutil info fs -h
acfsutil info fs [ {-o item | -s [interval [count] ] } ] [mount_point]
```

acfsutil info fs -h displays help text and exits.

Table 13–77 contains the options available with the acfsutil info fs command.

Table 13–77   Options for the acfsutil info fs command

Option	Description
-o	Displays the specific file system item from the following list:
	■ freespace - Amount of free space available in the file system
	■ totalspace - Size of the file system
	■ mountpoints - Mount points associated with file systems.
	■ volumes - Number of volumes associated with the file system
	■ primaryvolume - Path name of the device associated with file system
	■ available - Length of time the file system has been available
	■ isadvm - 1 if an Oracle ADVM volume is present, 0 if not
	■ ismountpoint - 1 if *path* is the mount point, 0 if *path* is not a mount point
	■ isavailable - 1 if the file system is available, 0 if file system is not available
	■ iscorrupt - 1 if the file system is corrupt, 0 if file system is not corrupt
	■ isreplprimary - 1 if primary file system, 0 if not
	■ isreplstandby - 1 if standby file system, 0 if not
	■ diskgroup - Name of the disk group associated with *path*
	■ redundancy - Type of redundancy used by the Oracle ADVM volume
	■ replication - 1 if file system has been initialized for replication as the primary or standby file system, 0 if not initialized for replication
	■ resizeincrement - Allocation unit size in bytes of the Oracle ADVM volume used by *path*
	■ stripewidth - Stripe width for the Oracle ADVM volume used by *path*
	■ stripecolumns - Stripe set columns for the Oracle ADVM volume used by *path*
-s [*interval* [*count*]]	Displays file system statistics.
	The optional *interval* parameter specifies the amount of time in seconds between each report. The first report contains statistics for the time since the file system mount. Each subsequent report contains statistics collected during the interval since the previous report.
	The optional *count* parameter can be specified in conjunction with the *interval* parameter. If the *count* parameter is specified, the value of count determines the number of reports generated at interval seconds apart.
	If the *interval* parameter is specified without the *count* parameter, the command generates reports continuously until you break out of the command with Ctrl+c.
mount_point	Specifies the directory where the file system is mounted. Not a valid option for replication data.

`acfsutil info fs` displays information and statistics about Oracle ACFS file systems. If replication has been initiated on a the file system, additional flags and status information are displayed. The values for the **replication status** field are primary, standby, or disabled.

The `-o` *item* option displays file system information for the *item* specified.

The `-s` option displays the amount and rate of change currently on a file system for the node that the command is run on.

With no option specified, the command displays file system information that includes the volume device name, the size of the file system, the amount of space available on the volume device, the file system mount time on this node, the state of the file system, the user specified block size, the number of Oracle ACFS snapshots, the space consumed by snapshots in the file system, and the optional name or volume label associated with the file system. The possible file system states are displayed in the `flags` line. These states include:

- `Offline` indicates that the underlying devices are not accessible, possibly due to an Oracle ASM instance failure, disk group forced dismount, or an irrecoverable I/O error. The file system on this node can only be dismounted. All other attempts at access result in errors.

- `Available` indicates that the file system is on line and operational.

- `Corrupt` indicates that the file system should be checked and repaired at the earliest possible convenience to correct a detected inconsistency. For example, run the `fsck` command on Linux or the `acfschkdsk` command on Windows to check and repair the file system. Ensure that you run the command in repair mode to correct the problem. For more information, refer to "fsck" on page 13-21 and "acfschkdsk" on page 13-41.

Any user can run `acfsutil info fs`.

Examples

The following are examples of the use of `acfsutil info fs`.

Example 13–67 displays information about an Oracle ACFS file system in a Linux environment.

Example 13–67   Using the acfsutil info fs command on Linux

```
$ /sbin/acfsutil info fs
/primary
    ACFS Version: 11.2.0.2.0
    flags:        MountPoint,Available,Replication
    mount time:   Mon Oct 25 12:11:03 2010
    volumes:      1
    total size:   5368709120
    total free:   4144230400
    primary volume: /dev/asm/pvol-74
        label:
        flags:                Primary,Available,ADVM
        on-disk version:      40.0
        allocation unit:      4096
        major, minor:         252, 37889
        size:                 5368709120
        free:                 4144230400
        ADVM diskgroup        REPLDG
        ADVM resize increment: 268435456
        ADVM redundancy:      unprotected
```

```
                ADVM stripe columns:   4
                ADVM stripe width:     131072
        number of snapshots:  0
        snapshot space usage: 0
        replication status: primary

/standby
        ACFS Version: 11.2.0.2.0
        flags:          MountPoint,Available,Replication
        mount time:     Mon Oct 25 12:11:03 2010
        volumes:        1
        total size:     5368709120
        total free:     5263945728
        primary volume: /dev/asm/svol-74
                label:
                flags:                  Primary,Available,ADVM
                on-disk version:        40.0
                allocation unit:        4096
                major, minor:           252, 37890
                size:                   5368709120
                free:                   5263945728
                ADVM diskgroup          REPLDG
                ADVM resize increment: 268435456
                ADVM redundancy:        unprotected
                ADVM stripe columns:   4
                ADVM stripe width:     131072
        number of snapshots:  0
        snapshot space usage: 0
        replication status: standby

$ /sbin/acfsutil info fs -o mountpoints,replication
/primary
1
/standby
1

$ /sbin/acfsutil info fs -o mountpoints,isreplprimary
/primary
1
/standby
0

$ /sbin/acfsutil info fs -o mountpoints,isreplstandby
/primary
0
/standby
1
```

Example 13–68 illustrates the use of `acfsutil info fs` in a Windows environment.

Example 13–68   Using the acfsutil info fs command on Windows

```
C:\oracle>acfsutil info fs /o freespace c:\oracle\acfsmounts\acfs1
968667136

C:\oracle>acfsutil info fs /o totalspace c:\oracle\acfsmounts\acfs1
1073741824

C:\oracle>acfsutil info fs /o volume c:\oracle\acfsmounts\acfs1
1
```

```
C:\oracle>acfsutil info fs /o primaryvolume c:\oracle\acfsmounts\acfs1
asm-volume1-311

C:\oracle>acfsutil info fs /o diskgroup c:\oracle\acfsmounts\acfs1
DATA

C:\oracle>acfsutil info fs /o redundancy c:\oracle\acfsmounts\acfs1
mirror
```

Example 13–69 illustrates the use of `acfsutil info fs` with `-s` option to display the current amount and rate of change on a file system.

Example 13–69   Using the acfsutil info fs command with the -s option

```
$ /sbin/acfsutil info fs -s /u01/app/acfsmounts/myacfs
    amount of change since mount:       359.22 MB
    average rate of change since mount: 3 KB

$ /sbin/acfsutil info fs -s 60 4 /u01/app/acfsmounts/myacfs
    amount of change since mount:       359.22 MB
    average rate of change since mount: 3 KB/s

    amount of change: 15.02 MB   rate of change: 256 KB/s
    amount of change: 9.46 MB   rate of change: 161 KB/s
    amount of change: 7.32 MB   rate of change: 125 KB/s
    amount of change: 6.89 MB   rate of change: 117 KB/s
...
```

acfsutil info id

Purpose
Displays Oracle ACFS file system information for a given identifier and mount point.

Syntax and Description
```
acfsutil info id -h
acfsutil info id num mount_point
```

`acfsutil info id -h` displays help text and exits.

Table 13–78 contains the options available with the `acfsutil info id` command.

Table 13–78   Options for the acfsutil info id command

Option	Description
mount_point	Specifies the directory where the file system is mounted.
num	Specifies the Oracle ACFS file identifier number reported by the Oracle ACFS driver. The number should be specified in decimal format.

`acfsutil info id` is used to translate an internal numeric Oracle ACFS file identifier to a path name in the file system. This is useful when the Oracle ACFS driver reports I/O errors to the system event logger associated with a particular file in an Oracle ACFS and identifies it by its internal identifier. You must have administrator privileges or you must be a member of the Oracle ASM administrator group to run `acfsutil info id`.

Examples

The following is an example of acfsutil info id.

Example 13–70   Using the acfsutil info id command

```
$ /sbin/acfsutil info id 117 /u01/app/acfsmounts/myacfs
```

acfsutil registry

Purpose

Registers an Oracle ACFS file system with the Oracle ACFS mount registry.

Syntax and Description

```
acfsutil registry -h
acfsutil registry
acfsutil registry -a [-f] [-n { nodes | all }]
         [-o moptions] device mount_point
acfsutil registry -d {device | mount_point}
acfsutil registry -l [device | mount_point]
acfsutil registry -m device
```

acfsutil registry -h displays help text and exits.

Table 13–79 contains the options available with the acfsutil registry command.

Table 13–79   Options for the acfsutil registry command

Option	Description
-a	Specifies to add the device, mount point, and associated *moptions* to the Oracle ACFS mount registry. The Oracle ASM volume device specified must exist on the local node to add the information to the mount registry.
	The arguments represent all the information needed to mount the file system. At Oracle ACFS startup time these file systems are automatically mounted. On Windows the volume mount points on the specified directories are created if they do not exist.
	Duplicate device entries are not allowed. Duplicate mount points are allowed but must be include the -n option for disjoint node-specific mounts.
-d	Deletes the device or mount point from the Oracle ACFS mount registry. If a mount point is specified and it is not unique in the Oracle ACFS mount registry, the command fails and you must run the command again specifying the device as an argument.
-f	This is used in combination with -a when the specified device might exist in the registry and the administrator wants to replace the registration.
-n	This is used in combination with -a and specifies an optional comma-delimited list of nodes, or the all keyword. This specifies which nodes should attempt to mount this device on the specified mount point. Host names should be specified. The all keyword is functionally equivalent to not specifying a list of nodes and indicates that the device should be mounted on all nodes. Mounting on all nodes is the default behavior.

Table 13–79   (Cont.) Options for the acfsutil registry command

Option	Description
`-o moptions`	Specifies the mount options for use when mounting the file system. Valid for Linux, Solaris, and AIX. For specific `-o moptions`, refer to the `mount` command for each operating system. All the options for each operating system are available except the `all` option.
`-l`	Lists all the mount points currently in the registry on a single line, with fields separated by a colon (`:`).
`-m`	Lists the registered mount point, if one exists, associated with the specified device. The mount point is only returned if the Oracle ACFS file system has been registered or has been previously mounted.
`mount_point`	Specifies the directory where the file system is mounted.
`device`	Specifies an Oracle ACFS device file that has been formatted.

`acfsutil registry` adds or deletes a file system from the Oracle ACFS persistent mount registry. The mount registry is a global registry that is used at Oracle ACFS startup on each node to mount all file systems specified in it. `root` or `asmadmin` privileges are required to modify the registry. For information about operating system group privileges, see "About Privileges for Oracle ASM" on page 3-26.

Any user is allowed to display the contents of the registry. To mount all the file systems in the Oracle ACFS mount registry, use the platform specific mount command with the `all` option. This is done automatically at Oracle ACFS startup on each node.

If no options are specified, the command displays all of the Oracle ACFS mounts in the registry.

> **Note:**   Do not register an Oracle ACFS file system that has had an individual file system Cluster Ready Services (CRS) resource added. For example, do not register a file system with `acfsutil registry` if the Oracle ACFS file system has been registered using `srvctl add filesystem`.

See "About the Oracle ACFS Mount Registry" on page 5-8.

Examples

The following examples show the use of `acfsutil registry`. The first example shows how to add the volume device file and file system mount point to the registry. The second example shows how to list the registered mount point associated with the specified volume device file. The third example shows how to delete the specified volume device file from the registry.

Example 13–71   Using the acfsutil registry command

```
$ /sbin/acfsutil registry -a /dev/asm/volume1-123 /u01/app/acfsmounts/myacfs

$ /sbin/acfsutil registry -m /dev/asm/volume1-123

$ /sbin/acfsutil registry -d /dev/asm/volume1-123
```

acfsutil rmfs

Purpose
Removes an Oracle ACFS file system.

Syntax and Description
```
acfsutil rmfs -h
acfsutil rmfs device
```

`acfsutil rmfs -h` displays help text and exits.

Table 13–80 contains the options available with the `acfsutil rmfs` command.

Table 13–80   Options for the acfsutil rmfs command

Option	Description
device	Specifies an Oracle ACFS device file that has been formatted.

`acfsutil rmfs` is used to remove an Oracle ACFS that is dismounted. When the command is run, the superblock of the file system is disabled. `root` or `asmadmin` privileges are required to run this command. For information about operating system group privileges, see "About Privileges for Oracle ASM" on page 3-26.

After `acfsutil rmfs` runs successfully, the `MOUNTPATH` and `USAGE` columns in the `V$ASM_VOLUME` view are cleared for the device. The removed Oracle ACFS can be restored using `fsck` or `acfschkdsk`. The device can be reformatted with a new Oracle ACFS using the `mkfs` or `acfsformat` commands.

Examples
The following example shows the use of `acfsutil rmfs` to remove the specified volume device file and associated file system.

Example 13–72   Using the acfsutil rmfs command
```
$ /sbin/acfsutil rmfs /dev/asm/volume1-123
```

acfsutil size

Purpose
Resizes an Oracle ACFS file system.

Syntax and Description
```
acfsutil size -h
acfsutil size [+|-]n[K|M|G|T|P ] [device] mount_point
```

`acfsutil size -h` displays help text and exits.

Table 13–81 contains the options available with the `acfsutil size` command.

Table 13–81   Options for the acfsutil size command

Option	Description
[+|-]*n*	Specifies the new size for the Oracle ACFS file system where *n* is a valid positive whole number greater than zero. The number can be preceded by a + or - to indicate the amount to add or decrease. If no operand exists, the new size is the absolute size.
K|M|G|T|P	Specifies that the integer supplied for size is in the units of K (Kilobytes), M (Megabytes), G (Gigabytes), T (Terabytes), or P (Petabytes). If the unit indicator is specified, then it must be appended to the integer. If omitted, the default unit is bytes.
device	Specifies the optional volume device file.
mount_point	Specifies the directory where the file system is mounted

`acfsutil size` grows or shrinks the mounted Oracle ACFS and its underlying Oracle ADVM storage to match the new size specified. This operation also resizes the underlying Oracle ASM Volume file to match the new length that is specified. The disk group must have enough free storage to accommodate any requested increase to the file system size.

Reducing a file system size returns unused storage space located at the end of the file system to the disk group. Shrinking in this release is intended for accidents when the wrong initial size or resize increment was specified, and before the storage was actually used. After storage has been used for user data or file system metadata, it may not be possible to shrink the file system, even if the files using that storage have been deleted.

> **Note:**   Defragmenting a file system is not supported in this release; only unused storage can be deallocated from the end of the file system.

The size value is rounded up based on the block size of the file system and the allocation unit of the Oracle ASM volume device file. To determine the Oracle ASM volume device resize increment, examine the `RESIZE_UNIT_MB` field in the `V$ASM_VOLUME` view, or look for `Resize Unit` in the output of `asmcmd volinfo` or `acfsutil info fs`.

There is a limit of 5 extents for the file system's internal storage bitmap. This causes any attempts to increase the file system to fail after it has been increased four or more times. However, if after increasing the file system four times or more times the file system size is decreased, then you may be able to increase the file system size again if the size of the increase is less than the size of the decrease.

When the limit on a file system expansion has been reached, running `fsck` or `acfschkdsk` with the `-a` option may consolidate the internal storage bitmap, allowing future file system expansion.

`root` or users who are members of the `asmadmin` group can run this command. For information about operating system group privileges, see "About Privileges for Oracle ASM" on page 3-26.

Examples

The following example shows the use of `acfsutil size`. This example increases the primary device file of `/u01/app/acfsmounts/myacfs` file system by 500 MB.

Example 13–73   Using the acfsutil size command

```
$ /sbin/acfsutil size +500M /u01/app/acfsmounts/myacfs
```

acfsutil snap create

Purpose
Creates a read-only or read-write snapshot of an Oracle ACFS file system.

Syntax and Description
```
acfsutil snap create -h
acfsutil snap create [-r|-w] snapshot mount_point
```

`acfsutil snap create -h` displays help text and exits.

Table 13–82 contains the options available with the `acfsutil snap create` command.

Table 13–82   Options for the acfsutil snap create command

Option	Description
-r	Creates a read-only snapshot. This is the default setting.
-w	Creates a read-write snapshot.
snapshot	Specifies a name for the snapshot. The name provided must be a valid directory name. The `.ACFS/snaps` directory itself cannot be snapped.
mount_point	Specifies the directory where the file system is mounted.

`acfsutil snap create` creates a read-only or read-write snapshot of the Oracle ACFS mounted on *mount_point*. You can specify `-r` for read-only or `-w` for read-write; read-only is the default if neither `-r` or `-w` are specified.

The read-write snapshot enables the fast creation of an Oracle ACFS snapshot image that can be both read and written without impacting the state of the Oracle ACFS file system hosting the snapshot images. The read-write functionality can be used for testing new versions of application software or running test scenarios on production file data without modifying the original file system.

Snapshots are not separate file systems. The snapshot appears in the `.ACFS/snaps/snapshot` directory and initially is a complete replica of the file system at the time the snapshot command was given. A read-only snapshot continues to preserve that initial point-in-time view. A read-write snapshot can be modified by updates written directly to the files located within the `.ACFS/snaps/snapshot` hierarchy.

Any user can access the snapshot directory by specifying the path name. However, the `.ACFS` directory itself is hidden from directory listings of the root of the file system. This prevents recursive commands, such as `rm -rf` or `acfsutil tag set -r`, from the root of the file system inadvertently operating on snapshot files.

Snapshots usually use very little storage initially as they share file system blocks with the original file system until a file changes.

Tools such as `du` report the total disk space usage of the snapshotted files, which includes the storage shared with the original versions of the files. To determine the total space used for the snapshots, use the `acfsutil snap info` or `acfsutil info fs` command. See "acfsutil info fs" on page 13-99.

Oracle ACFS snapshots are immediately available for use after they are created. They are always online under the .ACFS/snaps directory when the original file system is mounted. No separate command is needed to mount them.

Administrator privileges are required to use this command or you must be a member of the Oracle ASM administrator group.

For more information about Oracle ACFS snapshots, see "About Oracle ACFS Snapshots" on page 5-8.

Examples

Example 13–74 shows the use of the acfsutil snap create command.

Example 13–74   Using the acfsutil snap create command

```
$ acfsutil snap create -w midday_test1 /u01/critical_apps

$ acfsutil snap create -w midday_test2 /u01/critical_apps

$ /sbin/acfsutil snap create payroll_report1 /u01/critical_apps

$ /sbin/acfsutil snap create payroll_report2 /u01/critical_apps

$ ls /u01/critical_apps/.ACFS/snaps
midday_test1  midday_test2  payroll_report1  payroll_report2
```

acfsutil snap delete

Purpose

Deletes a snapshot of an Oracle ACFS file system.

Syntax and Description

```
acfsutil snap delete -h
acfsutil snap delete snapshot mount_point
```

acfsutil snap delete -h displays help text and exits.

Table 13–83 contains the options available with the acfsutil snap delete command.

Table 13–83   Options for the acfsutil snap delete command

Option	Description
snapshot	Specifies a name for the snapshot.
mount_point	Specifies the directory where the file system is mounted.

acfsutil snap delete deletes the snapshot named *snapshot* in the Oracle ACFS mounted on *mount_point*. After successful completion of the command, the representation of the snapshot in the .ACFS/snaps directory is removed. The command fails if any file within the snapshot is open on any cluster node. Note that the disk space used by the snapshot being deleted is freed by a background task after the completion of the acfsutil snap delete command. If one of these background threads is running to clean up a deleted snapshot, then the acfsutil snap info command shows a pending delete operation.

Administrator privileges are required to use this command or you must be a member of the Oracle ASM administrator group.

Examples

Example 13–75 shows the use of the `acfsutil snap delete` command.

Example 13–75   Using the acfsutil snap delete command

```
$ /sbin/acfsutil snap delete midday_test1 /u01/critical_apps
acfsutil snap delete: Snapshot operation is complete.
```

acfsutil snap info

Purpose

Displays information about Oracle ACFS file system snapshots.

Syntax and Description

```
acfsutil snap info -h
acfsutil snap info [snapshot] mount_point
```

`acfsutil snap info -h` displays help text and exits.

Table 13–84 contains the options available with the `acfsutil snap delete` command.

Table 13–84   Options for the acfsutil snap info command

Option	Description
snapshot	Specifies a name for the snapshot.
mount_point	Specifies the directory where the file system is mounted.

The `acfsutil snap info` command displays information about an individual specified snapshot or all snapshots in the specified Oracle ACFS file system. The snapshot name, snapshot type (RO or RW), and creation date and time are displayed.

The snapshot space usage amount includes snapshot metadata. If all the files are deleted from a snapshot, some of the metadata still remains and that amount is displayed with `acfsutil snap info`.

The `acfsutil snap info` command also shows pending delete operations.

Examples

Example 13–76 shows the use of the `acfsutil snap info` command when there is a pending delete operation. When the background process finishes the clean up of the deleted snapshot, the delete pending row does not appear in the output.

Example 13–76   Using the acfsutil snap info command

```
$ /sbin/acfsutil snap info /u01/critical_apps
snapshot name:             midday_test2
RO snapshot or RW snapshot:  RW
snapshot creation time:     Fri Feb 18 06:10:59 2011

snapshot name:             payroll_report1
RO snapshot or RW snapshot:  RO
snapshot creation time:     Tue Feb 22 06:56:34 2011

snapshot name:             payroll_report2
RO snapshot or RW snapshot:  RO
snapshot creation time:     Tue Feb 22 06:57:21 2011
```

```
number of snapshots:  3 (active)
                      1 (delete pending)
snapshot space usage: 8383348736
```

acfsutil tune

Purpose

The `acfsutil tune` command displays the value of a specific tunable parameter or all Oracle ACFS tunable parameters, or sets the value of a tunable parameter in a persistent manner on a particular node.

Syntax and Description

```
acfsutil tune -h
acfsutil tune [tunable_name]
acfsutil tune tunable_name=value
```

`acfsutil tune -h` displays help text and exits.

Table 13–85 contains the options available with the `acfsutil tune` command.

Table 13–85   Options for the acfsutil tune command

Option	Description
tunable_name	Specifies the name of the tunable parameter.
value	Specifies the value for a tunable parameter.

The only Oracle ACFS tunable parameter is `AcfsMaxOpenFiles`, which limits the number of open Oracle ACFS files on Windows and AIX. Normally you do not have to change the value of this tunable parameter; however, you may want to consider increasing the value if you have a large working set of files in your Oracle ACFS file systems.

Changing a tunable parameter has an immediate effect and persists across restarts. You must be a root user or the Windows `Administrator` to change the value of a tunable parameter.

Examples

The first example prints Oracle ACFS tunable parameters. The second example changes the value of a tunable parameter.

Example 13–77   Using the acfsutil tune command

```
$ /sbin/acfsutil tune

$ acfsutil tune AcfsMaxOpenFiles=50000
```

advmutil canonical

Purpose

`advmutil canonical` displays the canonical name of the specified Oracle ADVM device name.

Syntax and Description

```
advmutil -h
advmutil canonical volume_device
```

`advmutil -h` displays help text and exits.

Table 13–86 contains the options available with the `advmutil canonical` command.

Table 13–86   Options for the advmutil canonical command

Option	Description
volume_device	Specifies a string identifying a Oracle ADVM volume device.

There are several different formats that can identify an Oracle ADVM volume device, but a normalized, unambiguous (canonical) name should be used when the volume device name is used with other commands such as `SRVCTL`.

For example, on the Windows operating system you can use the following prefixes with a volume device name: \\.\, \??\, \\?\

The `advmutil canonical` command would return the canonical name that another utility would recognize without having to strip off extra characters. The command would most likely be used in a script.

Examples

The following examples show the use of `advmutil canonical` on a Windows operating system. For each example, the command returns the canonical name of the volume device.

Example 13–78   Using advmutil canonical

```
[C:\]advmutil canonical asm-volume1-274
asm-volume1-274

[C:\]advmutil canonical \\.\asm-volume1-274
asm-volume1-274

[C:\]advmutil canonical \\?\asm-volume1-274
asm-volume1-274

[C:\]advmutil canonical \??\asm-volume1-274
asm-volume1-274
```

advmutil tune

Purpose

`advmutil tune` displays the value of a specific Oracle ADVM parameter or sets the value of a specific Oracle ADVM parameter.

Syntax and Description

```
advmutil -h
advmutil tune parameter [= value]
```

`advmutil -h` displays help text and exits.

Table 13–87 contains the options available with the `advmutil tune` command.

Table 13–87   Options for the advmutil tune command

Option	Description
parameter	Specifies the parameter for which you want to set or display the value.
value	Optional value provided to set the value of the specified parameter.

If a value is not provided, the `advmutil tune` command displays the value that is currently assigned to the specified parameter.

The parameters that can be specified with `advmutil tune` are the maximum kernel memory (`max_memory`) or the maximum time in minutes for the deadlock timer (`deadlock_timer`).

The maximum kernel memory (`max_memory`) specifies the maximum operating system (OS) kernel memory in megabytes that can be consumed by the Oracle ASM Dynamic Volume Manager driver to cache Oracle ASM extent maps.

> **Note:**   The `deadlock_timer` parameter should only be set by Oracle Support Services.

Examples

The first example sets the maximum kernel memory. The second example queries the current setting for the maximum kernel memory to be consumed by the Oracle ADVM. The third example changes the maximum time in minutes for the deadlock timer. The fourth example queries the current setting of a parameter.

Example 13–79   Using advmutil tune

```
$ /sbin/advmutil tune max_memory = 40

$ /sbin/advmutil tune max_memory
  max_memory = 40 (0x28)

$ /sbin/advmutil tune deadlock_timer = 20

$ /sbin/advmutil tune deadlock_timer
  deadlock_timer = 20 (0x14)
```

advmutil volinfo

Purpose

`advmutil volinfo` displays information about Oracle ADVM volume devices.

Syntax and Description

```
advmutil -h
advmutil volinfo [-l][-L] [volume_device]
```

`advmutil -h` displays help text and exits.

Table 13–88 contains the options available with the `advmutil volinfo` command.

Table 13–88   Options for the advmutil volinfo command

Option	Description
volume_device	Specifies an optional volume device name.
-l	Separates the Oracle ADVM volume device information by field descriptions and colons on one line.
-L	Separates the Oracle ADVM volume device information by spaces on one line.

`advmutil volinfo` displays information about Oracle ADVM volume devices in a list format by default. The `-l` option on Linux (`/l` on Windows) formats the display into a colon-separated string with field descriptions. The `-L` option on Linux (`/L` on Windows) formats the display into a space-separated string in a format suitable for scripting.

Examples

The first example displays information about an Oracle ADVM volume device, using the `advmutil volinfo` command with the volume device name. The second example displays information about the volume device using the `-l` option. The third example displays information about the volume device using the `-L` option.

Example 13–80   Using advmutil volinfo

```
$ /sbin/advmutil volinfo /dev/asm/volume1-123
Device : /dev/asm/volume1-228
Interface Version: 1
Size (MB): 256
Resize Increment (MB): 32
Redundancy: mirror
Stripe Columns: 4
Stripe Width (KB): 128
Disk Group: DATA
Volume: VOLUME1
Compatible.advm : 11.2.0.0.0

$ /sbin/advmutil volinfo -l /dev/asm/volume1-228
Device : /dev/asm/volume1-228 : Interface Version : 1 : Size (MB) : 256 :
Resize Increment (MB) : 32 : Redundancy : mirror : Stripe Columns : 4 :
Stripe Width (KB) : 128 : Disk Group : DATA : Volume : VOLUME1 :
Compatible.advm : 11.2.0.0.0

$ /sbin/advmutil volinfo -L /dev/asm/volume1-228
/dev/asm/volume1-228  1  256 32  mirror  4  128  DATA  VOLUME1  11.2.0.0.0
```

A

Creating Database Files in Oracle ASM Using SQL*Plus

This appendix describes how to deploy Oracle Automatic Storage Management (Oracle ASM) using SQL*Plus under the following topics:

- Creating a Database that Uses Oracle ASM
- Creating Tablespaces in Oracle ASM
- Creating Redo Logs in Oracle ASM
- Creating Control Files in Oracle ASM
- Creating Archive Log Files in Oracle ASM

> **See Also:** *Oracle Database Administrator's Guide* for information about using Oracle Managed Files (OMF)

Creating a Database that Uses Oracle ASM

The recommended method of creating your database is to use Database Configuration Assistant (DBCA). However, if you create your database manually using the `CREATE DATABASE` statement, then Oracle ASM enables you to create a database and all of its underlying files with a minimum of input.

The following section describes an example of using the `CREATE DATABASE` statement, where database files are created and managed automatically by Oracle ASM.

Creating a Database that Uses Oracle ASM: Example

This example creates a database with the following Oracle ASM files:

- A `SYSTEM` tablespace data file in disk group `data`.
- A `SYSAUX` tablespace data file in disk group `data`. The tablespace is locally managed with automatic segment-space management.
- A multiplexed online redo log with two online log groups, one member of each in `data` and `fra` (fast recovery area).
- If automatic undo management mode is enabled, then an undo tablespace data file in directory `data`.
- If you do not specify a value for the `CONTROL_FILES` initialization parameter, then two control files, one in `data` and another in `fra` (fast recovery area). The control file in `data` is the primary control file.

The following initialization parameter settings are included in the initialization parameter file:

```
DB_CREATE_FILE_DEST = +data
DB_RECOVERY_FILE_DEST = +fra
DB_RECOVERY_FILE_DEST_SIZE = 10G
```

The following statement is run at the SQL prompt:

```
CREATE DATABASE orcl;
```

Creating Tablespaces in Oracle ASM

When Oracle ASM creates a data file for a permanent tablespace (or a temporary file for a temporary tablespace), the data file is set to auto-extensible with an unlimited maximum size and 100 MB default size. You can use the `AUTOEXTEND` clause to override this default extensibility and the `SIZE` clause to override the default size.

Oracle ASM applies attributes to the data file, as specified in the system default template for a data file as shown in the table in "Managing Disk Group Templates" on page 7-15. You can also create and specify your own template.

Files in a tablespace might be in both Oracle ASM files and non-Oracle ASM files due to the tablespace history. RMAN commands enable non-Oracle ASM files to be relocated to an Oracle ASM disk group and enable Oracle ASM files to be relocated as non-Oracle ASM files.

For example, if there is a disk group named `data`, then you can create a tablespace `tblspace` in that disk group with the following SQL statement:

```
CREATE TABLESPACE tblspace DATAFILE +data;
```

The following example illustrates the usage of Oracle ASM with defaults. This example enables Oracle ASM to create and manage the tablespace data file for you, using Oracle supplied defaults that are adequate for most situations.

Assume the following initialization parameter setting:

```
DB_CREATE_FILE_DEST = +data
```

The following statement creates the tablespace and its data file:

```
CREATE TABLESPACE tblspace;
```

Creating Redo Logs in Oracle ASM

Online redo logs can be created in multiple disk groups, either implicitly in an initialization parameter file or explicitly in an `ALTER DATABASE...ADD LOGFILE` statement. Each online log should have one log member in multiple disk groups. The filenames for log file members are automatically generated. All partially created redo log files, created because of a system error, are automatically deleted.

For example, if there are disk groups named `data1` and `data2`, you can create a log file with a member in each disk group with the following SQL statement:

```
ALTER DATABASE ADD LOGFILE (+data1,+data2);
```
The following example creates a log file with a member in each of the disk groups `data1` and `data2`. The following parameter settings are included in the initialization parameter file:

```
DB_CREATE_ONLINE_LOG_DEST_1 = +data1
DB_CREATE_ONLINE_LOG_DEST_2 = +data2
```

The following statement is run at the SQL prompt:

```
ALTER DATABASE ADD LOGFILE;
```

Creating Control Files in Oracle ASM

Control files can be explicitly created in multiple disk groups. The filenames for control files are automatically generated.

If you must specify a control file by name, then you can use alias filenames to reference Oracle ASM files and use easy to understand names. Using aliases in the specification of a control file during its creation enables you to later refer to the control file with a user-specified name. An alias can also be specified as a control file name in the CONTROL_FILES initialization parameter.

When creating a control file, data files and log files that are stored in Oracle ASM disk groups should be given to the CREATE CONTROLFILE command using the file reference form of their Oracle ASM filenames. However, the use of the RESETLOGS option requires the use of a file creation form for the specification of the log files.

Creating Control Files in Oracle ASM: Example 1

The following CREATE CONTROLFILE statement is generated by an ALTER DATABASE BACKUP CONTROLFILE TO TRACE command for a database with data files and log files created on disk groups data and fra:

```
CREATE CONTROLFILE REUSE DATABASE "ORCL" NORESETLOGS  ARCHIVELOG
    MAXLOGFILES 16
    MAXLOGMEMBERS 3
    MAXDATAFILES 100
    MAXINSTANCES 8
    MAXLOGHISTORY 292
LOGFILE
  GROUP 1 (
    '+DATA/orcl/onlinelog/group_1.261.685366203',
    '+FRA/orcl/onlinelog/group_1.257.685366205'
  ) SIZE 50M BLOCKSIZE 512,
  GROUP 2 (
    '+DATA/orcl/onlinelog/group_2.262.685366207',
    '+FRA/orcl/onlinelog/group_2.258.685366211'
  ) SIZE 50M BLOCKSIZE 512,
  GROUP 3 (
    '+DATA/orcl/onlinelog/group_3.263.685366213',
    '+FRA/orcl/onlinelog/group_3.259.685366215'
  ) SIZE 50M BLOCKSIZE 512
-- STANDBY LOGFILE
DATAFILE
  '+DATA/orcl/datafile/system.256.685366089',
  '+DATA/orcl/datafile/sysaux.257.685366089',
  '+DATA/orcl/datafile/undotbs1.258.685366091',
  '+DATA/orcl/datafile/users.259.685366091'
CHARACTER SET WE8MSWIN1252
;
```

Creating Control Files in Oracle ASM: Example 2

This example is a CREATE CONTROLFILE statement for a database with data files, but uses a RESETLOGS clause, and thus uses the creation form for log files:

```
CREATE CONTROLFILE REUSE DATABASE "ORCL" RESETLOGS  ARCHIVELOG
    MAXLOGFILES 16
    MAXLOGMEMBERS 3
    MAXDATAFILES 100
    MAXINSTANCES 8
    MAXLOGHISTORY 292
LOGFILE
  GROUP 1 (
    '+DATA/orcl/onlinelog/group_1.261.685366203',
    '+FRA/orcl/onlinelog/group_1.257.685366205'
  ) SIZE 50M BLOCKSIZE 512,
  GROUP 2 (
    '+DATA/orcl/onlinelog/group_2.262.685366207',
    '+FRA/orcl/onlinelog/group_2.258.685366211'
  ) SIZE 50M BLOCKSIZE 512,
  GROUP 3 (
    '+DATA/orcl/onlinelog/group_3.263.685366213',
    '+FRA/orcl/onlinelog/group_3.259.685366215'
  ) SIZE 50M BLOCKSIZE 512
-- STANDBY LOGFILE
DATAFILE
  '+DATA/orcl/datafile/system.256.685366089',
  '+DATA/orcl/datafile/sysaux.257.685366089',
  '+DATA/orcl/datafile/undotbs1.258.685366091',
  '+DATA/orcl/datafile/users.259.685366091'
CHARACTER SET WE8MSWIN1252
;
```

Creating Archive Log Files in Oracle ASM

Disk groups can be specified as archive log destinations in the LOG_ARCHIVE_DEST and LOG_ARCHIVE_DEST_n initialization parameters. When destinations are specified in this manner, the archive log filenames are unique, even if archived twice.

If LOG_ARCHIVE_DEST is set to a disk group name, LOG_ARCHIVE_FORMAT is ignored. Unique filenames for archived logs are automatically created by the Oracle database. If LOG_ARCHIVE_DEST is set to a directory in a disk group, LOG_ARCHIVE_FORMAT has its normal semantics.

The following sample archive log names might be generated with DB_RECOVERY_FILE_DEST set to +data. ORCL is the value of the DB_UNIQUE_NAME parameter:

```
+DATA/ORCL/ARCHIVELOG/2009_09_23/thread_1_seq_38.614.541956473
+DATA/ORCL/ARCHIVELOG/2009_09_23/thread_4_seq_35.609.541956477
+DATA/ORCL/ARCHIVELOG/2009_09_23/thread_2_seq_34.603.541956487
+DATA/ORCL/ARCHIVELOG/2009_09_25/thread_3_seq_100.621.541956497
+DATA/ORCL/ARCHIVELOG/2009_09_25/thread_1_seq_38.614.541956511
```

B

Oracle ACFS Advanced Topics

This appendix discusses limits, advanced administration, and troubleshooting for Oracle Automatic Storage Management Cluster File System (Oracle ACFS).

> **See Also:** Articles available at My Oracle Support (`https://support.oracle.com`) for information about Oracle ACFS and Oracle ADVM.

This appendix contains the following topics:

- Limits of Oracle ACFS
- Limits of Oracle ADVM
- Oracle ACFS Drivers Resource Management
- Oracle ACFS Registry Resource Management
- Oracle ACFS Individual File System Resource Management
- Oracle ACFS and Oracle Restart
- Understanding Oracle ACFS I/O Failure Console Messages

For information about Oracle ACFS, see Chapter 5, "Introduction to Oracle ACFS".

Limits of Oracle ACFS

The limits of Oracle ACFS are discussed in this section.

The topics contained in this section are:

- Oracle ACFS Disk Space Usage
- Oracle ACFS Error Handling
- Oracle ACFS and NFS

> **Note:** Oracle ACFS does not support hard links on directories.

Oracle ACFS Disk Space Usage

Oracle ACFS supports 2^40 (1 trillion) files in a file system. More than 4 billion files have been tested. 32-bit Linux systems are limited to 2^26 (67 million) files. Oracle ACFS supports 63 snapshots. These snapshots can be any combination of read-only and read-write. Oracle ACFS supports 64 mounted file systems on 32-bit systems, and

256 mounts on 64-bit systems. However, more file systems can be mounted if there is adequate memory.

Oracle ACFS preallocates large user files to improve performance when writing data. This storage is not returned when the file is closed, but it is returned when the file is deleted. Oracle ACFS also allocates local metadata files as nodes mount the file system for the first time. This can result in a mount failing due to an out of space error, and much of this storage must be contiguous. This storage is approximately 64-128 megabytes per node.

Oracle ACFS also keeps local bitmaps available to reduce contention on the global storage bitmap when searching for free space. This disk space is reported as in use by tools such as the UNIX df command even though some of it may not actually be allocated as of yet. This local storage pool can be as large as 128 megabytes per node and can allow space allocations to succeed, even though commands, such as df, report less space available than what is being allocated.

Oracle ACFS Error Handling

Oracle ASM instance failure or forced shutdown while Oracle ACFS or another file system is using an Oracle ADVM volume results in I/O failures. The volumes must be closed and re-opened to access the volume again. This requires dismounting any file systems that were mounted when the local Oracle ASM instance failed. After the instance is restarted, the corresponding disk group must be mounted with the volume enabled followed by a remount of the file system. See "Deregistering, Dismounting, and Disabling Volumes and Oracle ACFS File Systems" on page 13-18.

If any file systems are currently mounted on Oracle ADVM volume files, the SHUTDOWN ABORT command should not be used to terminate the Oracle ASM instance without first dismounting those file systems. Otherwise, applications encounter I/O errors and Oracle ACFS user data and metadata being written at the time of the termination may not be flushed to storage before the Oracle ASM storage is fenced. If there is not time to permit the file system to dismount, then you should run two sync (1) commands to flush cached file system data and metadata to persistent storage before issuing the SHUTDOWN ABORT operation.

Oracle ACFS does not interrupt the operating system environment when a metadata write fails, whether due to Oracle ASM instance failure or storage failure. Instead, Oracle ACFS isolates errors to an individual file system, putting it in an offline error state. The only operation that succeeds on that node for that file system from that point forward is a dismount operation. Another node recovers any outstanding metadata transactions, assuming it can write the metadata out to the storage. It is possible to remount the file system on the offlined node after the I/O condition is resolved.

It might not be possible for an administrator to dismount a file system while it is in the offline error state if there are processes referencing the file system, such as a directory of the file system being the current working directory for a process. To dismount the file system in this case it would be necessary to identify all processes on that node with references to files and directories on the file system and cause them to exit. The Linux fuser or lsof commands or Window handle command list information about processes and open files.

If Oracle ACFS detects inconsistent file metadata returned from a read operation, based on checksum or expected type comparisons, Oracle ACFS takes the appropriate action to isolate the affected file system components and generate a notification that fsck or acfschkdsk should be run as soon as possible. Each time the file system is mounted a notification is generated with a system event logger message until fsck or acfschkdsk is run.

Oracle ACFS and NFS

When exporting file systems through NFS on Linux, use the `-fsid=num` exports option. This option forces the file system identification portion of the file handle used to communicate with NFS clients to be the specified number instead of a number derived from the major and minor number of the block device on which the file system is mounted. You can use any 32-bit number for `num`, but it must be unique among all the exported file systems. In addition, `num` must be unique among members of the cluster and must be the same `num` on each member of the cluster for a given file system. This is needed because Oracle ASM DVM block device major numbers are not guaranteed to be the same across reboots of the same node or across different nodes in the cluster.

Limits of Oracle ADVM

The limits of Oracle ADVM and these discussed in this section.

The default configuration for an Oracle ADVM volume is four columns of 64 megabytes (MB) extents in length and a 128 KB stripe width. Oracle ADVM writes data as 128 kilobytes (KB) stripe chunks in round robin fashion to each column and fills a stripe set of four 64 MB extents with 2000 stripe chunks before moving to a second stripe set of four 64 MB extents for volumes greater than 256 MB. Note that setting the number of columns on an Oracle ADVM dynamic volume to 1 effectively turns off striping for the Oracle ADVM volume.

On Linux platforms Oracle ASM Dynamic Volume Manager (Oracle ADVM) volume devices are created as block devices regardless of the configuration of the underlying storage in the Oracle ASM disk group. Do not use `raw (8)` to map Oracle ADVM volume block devices into raw volume devices.

Oracle ACFS Drivers Resource Management

The Oracle ACFS drivers resource is supported only for Oracle Grid Infrastructure cluster configurations; it is not supported for Oracle Restart configurations. See "Oracle ACFS and Oracle Restart" on page B-6.

The Oracle ACFS drivers resource (`ora.drivers.acfs`) is created by the Grid Infrastructure root script that is run following the Grid Infrastructure installation. The Oracle ASM instance resource (`ora.asm`) names the drivers resource as a weak dependency. Consequently, the start action for the drivers resource is also called whenever the start action for the `ora.asm` resource is issued. The start action for the drivers resource includes support for loading the Oracle ACFS, Oracle ADVM, and Oracle Kernel Services Driver (OKS) drivers into the operating system.

Following an Oracle Grid Infrastructure installation on Linux and UNIX platforms, a root script is run that includes actions for copying the Oracle ACFS components; including the Oracle ACFS, OKS, and Oracle ADVM drivers; into operating system-specific locations.

The Oracle ASM instance is started during the Grid Infrastructure installation process whenever Oracle Clusterware Registry (OCR) and the voting files are configured within an Oracle ASM disk group. In that case, the Oracle ACFS drivers are initially loaded during Grid Infrastructure Installation based on the resource dependency. The Oracle ASM instance can also be started using the Oracle ASM Configuration Assistant and the Oracle ACFS drivers are loaded based on that action. In steady state mode, the Oracle ACFS drivers are automatically loaded during Oracle Clusterware initialization when the Oracle High Availability Services Daemon (OHASD) calls the start action for the Oracle ASM instance resource that also results in loading the Oracle ACFS drivers due to the resource dependency relationship. The start action for the

Oracle ACFS drivers resource attempts to load the Oracle ACFS, Oracle ADVM, and OKS drivers into the native operating system.

The policy for the Oracle ACFS drivers is that they remain loaded until Oracle Clusterware is shut down. The `ora.drivers.acfs` resource is managed automatically by Oracle High Availability Services Daemon (OHASD) and its state cannot be manually manipulated by `srvctl` or `crsctl`.

Oracle ACFS Registry Resource Management

The Oracle ACFS registry resource is supported only for Oracle Grid Infrastructure cluster configurations; it is not supported for Oracle Restart configurations. See "Oracle ACFS and Oracle Restart" on page B-6.

The Oracle ACFS registry resource (`ora.registry.acfs`) is created by the root script that is run following Grid Infrastructure installation. The start action for the Oracle ACFS mount registry resource is automatically called during Grid Infrastructure initialization to activate the local node state of the clusterwide Oracle ACFS mount registry. If this initialization is successful, the state of this resource is set to `online`; otherwise, the state of the resource is set to `offline`. The state of the Oracle ACFS registry resource is determined only by the active state of the mount registry. The `online` status is independent of any registry contents or the current state of any individual registered file systems that may exist within the Oracle ACFS registry.

In addition to activating the local node state of the mount registry, the Oracle ACFS registry resource start action assists in establishing a clusterwide Oracle ACFS file name space. On each node, the resource start action scans the contents of the clusterwide mount registry and mounts any file systems designated for mounting on the local cluster member. Before mounting a registered file system, the resource start action confirms that the associated file system storage stack is active and will mount the disk group, enable the volume file, and create the mount point if necessary to complete the mount operation.

The check action for the Oracle ACFS registry resource assists in maintaining the clusterwide Oracle ACFS file system name space. On each node, the check action scans the contents of the mount registry for newly created entries and mounts any Oracle ACFS file systems registered for mounting on the local node. Consequently, a new Oracle ACFS file system can be created and registered on one node of the cluster, and is automatically mounted on all cluster members designated by the Oracle ACFS registry entry.

The Oracle ACFS registry resource check action also assists with file system recoveries. Recovering a file system from an offline state requires dismounting and remounting the file system. As the Oracle ACFS registry resource check action scans the mount registry searching for newly created file systems, it also checks for any offline file systems on the local node and if found, attempts to dismount and remount each offline file system. If the remount is successful, the file system transitions from offline to fully active status.

The Oracle ACFS registry resource stop action is usually called during the Grid Infrastructure shutdown sequence of operations. To transition the registry resource to an offline state, all file systems on this cluster member that are configured with Oracle ADVM devices must be dismounted. A mounted file system maintains an open reference on its Oracle ADVM device special file and associated dynamic volume file that must be closed before the Oracle ASM instance can be shutdown normally. The registry resource stop action scans the operating system's internal mount table searching for any mounted file system that is configured with an Oracle ADVM device file. If any is found, the stop action attempts to dismount that file system. However, if

there are open references resulting from applications or users of that file system, then the file system cannot be dismounted until these are closed. If the dismount operation fails, the process IDs of any processes holding an open reference on the file system are displayed and logged to enable the administrator to resolve the open references and dismount the file systems. The internal mount table entries can include registered and unregistered Oracle ACFS file systems, and other local file systems that were mounted on an Oracle ADVM device file.

The Oracle ACFS registry resource clean action is called implicitly if the resource stop action fails to transition the resource to the offline state. In that case, the registry resource clean action can be called to effectively force the resource offline. The registry resource clean action scans the operating system internal mount table searching for any file system that is mounted upon an Oracle ADVM device. If any is found, the resource clean action attempts to `umount` the file system as in the resource stop action. However, if there are open references that prevent the file system from being dismounted, the clean resource action displays and logs the Process Identifiers of any process holding a reference, terminates the referencing processes, and then dismounts the file system. At the completion of the clean action, the registry resource is set to an offline state and other participants in the Grid Infrastructure shutdown sequence can now be stopped.

Whenever Oracle Clusterware is started on a cluster node, the Oracle ACFS startup operations for the node consult the cluster mount registry and attempt to mount all Oracle ACFS file systems that are registered for this node. Following each file system addition to the mount registry, the newly registered file system is automatically mounted on each node designated by the registry entry. If a registered file system is automatically mounted and is later dismounted, it is not automatically remounted until the system is rebooted or Oracle Clusterware is restarted. It can be manually remounted using the `mount` command or Oracle Enterprise Manager.

The Oracle ACFS cluster mount registry action routines attempt to mount each Oracle ACFS file system on its registered mount point and create the mount point if it does not exist. The registry action routines also mount any Oracle ASM disk groups and enable any Oracle ADVM volumes required to support the Oracle ACFS mount operation. In the event that a file system enters into an offline error state, the registry action routines attempt to recover the file system and return it to an on-line state by dismounting and remounting the file system. For information about the offline error state, see "About Oracle ACFS Integration with Oracle ASM" on page 5-10.

Oracle ACFS Individual File System Resource Management

The Oracle ACFS individual file system resource is supported only for Oracle Grid Infrastructure cluster configurations; it is not supported for Oracle Restart configurations. See "Oracle ACFS and Oracle Restart" on page B-6.

Oracle ASM Configuration Assistant (ASMCA) facilitates the creation of Oracle ACFS individual file system resources (`ora.`*`diskgroup`*`.`*`volume`*`.acfs`). During database creation with Database Configuration Assistant (DBCA), the individual file system resource is included in the dependency list of its associated disk group so that stopping the disk group also attempts to stop any dependent Oracle ACFS file systems.

An Oracle ACFS individual file system resource is typically created for use with application resource dependency lists. For example, if an Oracle ACFS file system is configured for use as an Oracle Database home, then a resource created for the file system can be included in the resource dependency list of the Oracle Database

application. This dependency causes the file system and stack to be automatically mounted due to the start action of the database application.

An Oracle ACFS file system that is to be mounted from a dependency action should not be included in the Oracle ACFS mount registry.

The start action for an Oracle ACFS individual file system resource is to mount the file system. This individual file system resource action includes confirming that the associated file system storage stack is active and mounting the disk group, enabling the volume file, and creating the mount point if necessary to complete the mount operation. If the file system is successfully mounted, the state of the resource is set to `online`; otherwise, it is set to `offline`.

The check action for an individual file system resource verifies that the file system is mounted. It sets the state of the resource to `online` status if mounted, otherwise the status is set to `offline`.

The stop action for an Oracle ACFS individual file system resource attempts to dismount the file system. If the file system cannot be dismounted due to open references, the stop action displays and logs the process identifiers for any processes holding a reference.

Use of the `srvctl start` and `stop` actions to manage the individual file system resources maintains their correct resource state.

Oracle ACFS and Oracle Restart

Oracle Restart does not support root-based Oracle ACFS resources for this release. Consequently, the following operations are not automatically performed:

- Loading Oracle ACFS drivers

- Mounting Oracle ACFS file systems listed in the Oracle ACFS mount registry

- Mounting resource-based Oracle ACFS database home file systems

The Oracle ACFS resources associated with these actions are not created for Oracle Restart configurations.

While Oracle ACFS resource management is fully supported for Oracle Grid Infrastructure configurations, the Oracle ACFS resource-based management actions must be replaced with alternative, sometimes manual, operations in Oracle Restart configurations.

Understanding Oracle ACFS I/O Failure Console Messages

Oracle ACFS logs information for I/O failures in the operating-specific system event log.

A console message has the following format:

```
[Oracle ACFS]: I/O failure (error_code) with device device_name during a operation_name op_type.
file_entry_num Starting offset: offset. Length of data transfer: io_length bytes.
Impact: acfs_type   Object: object_type   Oper.Context: operation_context
Snapshot?: yes_or_no AcfsObjectID: acfs_object_id . Internal ACFS Location: code_location.
```

The italicized variables in the console message syntax correspond to the following:

- I/O failure

The operating system specific error code, in Hex, seen by Oracle ACFS for a failed I/O. This may indicate a hardware problem, or it might indicate a failure to initiate the I/O for some other reason.

- Device

  The device involved, usually the ADVM device file, but under some circumstances it might be a string indicating the device minor number

- Operation name

  The kind of operation involved:

  `user data`, `metadata`, or `paging`

- Operation type

  The type of operation involved:

  `synch read`, `synch write`, `asynch read`, or `asynch write`

- File entry number

  The Oracle ACFS File entry number of the file system object involved, as a decimal number. The `acfsutil info fileid` tool finds the corresponding file name.

- Offset

  The disk offset of the I/O, as a decimal number.

- Length of I/O

  The length of the I/O in bytes, as decimal number.

- File system object impacted

  An indication that the file system object involved is either node-local, or is a resource accessed clusterwide. For example:

  `Node` or `Cluster`

- Type of object impacted

  A string indicating the kind of file system object involved, when possible. For example:

  `Unknown`, `User Dir.`, `User Symlink`, `User File`, `Sys.Dir`, `Sys.File`, or `MetaData`

  - `Sys.Dir.`

    Oracle ACFS-administered directory within the visible namespace

  - `sys.File`

    Oracle ACFS-administered file within the visible namespace

  - `MetaData`

    Oracle ACFS-administered resources outside of the visible namespace

- Operational context

  A higher-level view of what code context was issuing the I/O. This is for use by Oracle Support Services. For example:

  `Unknown`, `Read`, `Write`, `Grow`, `Shrink`, `Commit`, or `Recovery`

- Snapshot

  An indication of whether, if possible to determine, the data involved was from a Snapshot. For example:

Yes, No, or ?

- Object type of the file system

  An internal identifier for the type of file system object. For use by Oracle Support Services.

- Location of the code

  An internal identifier of the code location issuing this message. For use by Oracle Support Services.

The following is an example from /var/log/messages in a Linux environment:

```
[Oracle ACFS]: I/O failure (0xc0000001) with device /dev/sdb during a metadata synch write .
Fenum Unknown. Starting offset: 67113984. Length of data transfer: 2560 bytes.
Impact: Node   Object: MetaData   Oper.Context: Write
Snapshot?: ?  AcfsObjectID: 8  . Internal ACFS Location: 5 .
```

Glossary

Oracle Automatic Storage Management Cluster

An Oracle Automatic Storage Management (Oracle ASM) cluster is a collection of interconnected nodes, each with an Oracle ASM instance, operating as a unified cluster using Oracle Clusterware. An Oracle ASM cluster presents a shared pool of storage to one or more Oracle Databases that are also operating on the nodes. The databases can also be clustered using Oracle Real Application Clusters, but that is not a requirement. The disks that provide the shared storage pool must be accessible from all of the nodes in the Oracle ASM cluster.

ASMLib

ASMLib is an application programming interface (API) developed by Oracle to simplify the operating system–to-database interface and to exploit the capabilities of vendors' storage arrays on Linux-based systems.

Cluster File System (CFS)

A Cluster File System, or CFS, is a file system that is distributed across multiple computer systems. Generally, the computer systems share access to a disk connected through a **Storage Area Network (SAN)**. The CFS component on each individual computer system coordinates access to the disks using a global communication mechanism.

Cluster Synchronization Services (CSS)

Cluster Synchronization Services (CSS) provide the foundation for coordinated, inter-process communication activities across a set of nodes. CSS services include group services, lock services, node information and cluster configuration services, and node monitoring services that detect nodes joining and leaving the cluster.

CSS Cluster

A CSS cluster is the cluster configuration defined by the CSS cluster membership services. CSS maintains the authoritative membership for an Oracle cluster and monitors nodes joining and leaving the cluster configuration. There is at most one CSS cluster defined for a given set of nodes.

disk group

An Oracle ASM disk group is a collection of disks that Oracle ASM manages as a unit. Within a disk group, Oracle ASM exposes a file system interface for Oracle Database files. The content of files that are stored in a disk group are evenly distributed, or striped, to eliminate hot spots and to provide uniform performance across the disks. Oracle ASM files may also be optionally mirrored within a disk group. The performance of disks in a disk group is comparable to the performance of raw devices.

Direct Attached Storage (DAS)

Direct Attached Storage, or DAS, consists of storage devices that attach directly to host without an intervening network. DAS generally costs less than SAN attached storage, but also offers less flexibility and functionality. Examples of DAS include SATA disks, common to most desktop systems, and SCSI disks that are found on many servers.

extent

An extent, also called data extent, is the raw storage used to hold the contents of a file. Each extent consists of one or more allocation units (AU) on a specific disk.

extent map

An extent map is a list of extent pointers that point to all the data extents of a file. This is the raw storage for the file. Each extent pointer gives the disk and allocation unit of the data extent. For reliability, each extent pointer also includes a check byte to ensure it has not been damaged. This is particularly important when using an in-memory copy of the extent map to direct file I/O to a particular disk location.

fibre channel

Fibre channel is a set of standards that define the interface between computer systems and peripherals. The fibre channel interface is a serial bus interface originally designed to supersede the SCSI standard. However, both the fibre channel and SCSI interfaces have evolved independently of each other and have benefited from the existence of the other. Fibre is spelled with **re** rather than an **er** to indicate a distinction from terms such as *fiber optics*. Both copper and fiber optics are commonly used as a media for fibre channel.

file system

A file system is a software component providing structured access to disks. File systems present objects, such as files, to application programs. Access to files is generally specified with standard API defining operating system calls such as Open/Close and Read/Write that the application program uses for accessing files. File systems are usually provided as a component of an operating system, but may be provided as an independent software component.

file

A file is an object presented to an application program by a **file system**. A file is subdivided into blocks by the file system. A file system typically places what appears to the application program as consecutive blocks, into nonconsecutive locations on disks. The mapping of the file blocks to the locations on disks is kept in what is known as an **extent map**.

host bus adapter (HBA)

A host bus adapter (HBA), or host adapter, connects a host system to other network and storage devices, such as, eSATA, and SCSI devices.

Logical Unit Number (LUN)

In storage technology, the term LUN is often used to denote a disk presented to a computer system by a storage array. In SCSI terminology, a Logical Unit Number, or LUN, is a number representing a device address presented to a computer system.

mirroring

In storage systems, mirroring is a means for protecting the integrity of data by storing copies of data on multiple disks. If a disk fails, then a secondary copy of the data is available on a second or third disk. Generally, mirroring can be deployed in one of two

ways. In the most common case, as with a RAID storage array, a fixed amount of physical storage space on a single disk is uniformly copied on multiple disks. Through software on the storage array, the physically mirrored space on multiple devices is presented to the host as a single disk. Another approach to mirroring is to copy fixed segments of a file onto multiple disks. Oracle ASM uses the second approach for maximum flexibility.

Network Attached Storage (NAS)

Network Attached Storage, or NAS, comprises a class of systems providing file services to host computers. A device driver in the operating system accesses data using protocols such as NFS or Common Internet File System (CIFS). From the application program's point of view, Network Attached Storage appears as a file system local to the host computer on which the application program resides.

Network File System (NFS)

A network file system, or NFS, is an industry standard providing Network Attached Storage on many platforms. NFS extends the local file system framework to provide device sharing to users on an IP network. The user view of an NFS is that a remote file system on a host appears as if it were locally mounted.

partition

Operating systems typically provide a means for splitting a disk into sections called partitions. To an application, a partition appears as an independent disk, even though that is not the case. The operating system provides a command for managing the size and locations of partitions on a disk. This command writes a partition map onto the disk in an area that is not visible to the application program.

Redundant Array of Inexpensive Disks (RAID)

Redundant Array of Inexpensive Disks, or RAID, is a means for combining multiple disks through software and presenting them to a host as a collection of apparently distinct disks. Mirroring and striping are two means used for combining the disks as a collection. RAID can be implemented as a software component of a computer system or as software operating inside of a storage array. RAID operates on the physical presentation of storage as opposed to Oracle ASM mirroring and striping at the logical file level.

Storage Area Network (SAN)

Storage Area Network, or SAN, is a specialized network designed for attaching storage devices, such as disk arrays and tape drives, to computer systems. The most common network infrastructure used in SANs is **fibre channel**. SANs provide great flexibility by allowing a common interconnect infrastructure in which any storage device can be connected to any computer system. Zoning and LUN masking are commonly used for access control for SANs. Zoning is a mechanism where the SAN infrastructure determines the access rights of storage devices to computer systems. LUN masking is a mechanism where the storage device determines the access rights of what computer systems are permitted to connect to the device.

Small Computer System Interface (SCSI)

Small Computer System Interface, or SCSI, is a standard defining the interface between computer systems and peripherals, most commonly storage devices. The SCSI standard defines both the logical and physical aspects of the interface between computer systems and peripherals.

striping

In storage systems, striping is a means for spreading data across multiple disks as opposed to storing the data on a single disk. Striping is usually done to improve performance. Generally, striping can be deployed in one of two ways. In the most common case, as with a RAID storage array, a fixed amount of physical storage space that could have been stored on a single disk is uniformly striped across multiple disks. Through software on the storage array, the physically striped space on multiple devices is presented to the host as a single disk. Another approach to striping is to stripe fixed segments of a file across multiple disks connected to the host. Oracle ASM uses the second approach for maximum performance.

volume

In the storage realm, the meaning of volume has many related definitions. Volumes are often thought of as the virtual entity represented as a **Logical Unit Number (LUN)**. Volumes often are presented as an aggregation of pieces from several disks. A volume is managed by a software component called a **volume manager**.

volume manager

A volume manager is a software component that manages the mapping of the collection of the pieces of the disks into a **volume**.

Index