Oracle® AI Database

Oracle® Automatic Storage Management Administrator's Guide

26ai

G43239-01

October 2025

ORACLE®

Oracle AI Database Oracle® Automatic Storage Management Administrator's Guide, 26ai

G43239-01

Contents

Preface

1 Introducing Oracle Automatic Storage Management

2 Exploring Considerations for Oracle ASM Storage

3   Administering Oracle ASM Instances

4   Administering Oracle ASM Disk Groups

5 Administering Oracle ASM Files, Directories, and Templates

6   Using Views to Display Oracle ASM Information

7   Administering Oracle ASM with Oracle Enterprise Manager

8 Performing Oracle ASM Data Migration with RMAN

9 Managing Oracle ASM With ASMCA

10 Managing Oracle ASM with ASMCMD

11 Managing Oracle Flex ASM

12 Creating Database Files in Oracle ASM Using SQL*Plus

Preface

The *Oracle Automatic Storage Management Administrator's Guide* describes how to administer Oracle Automatic Storage Management (Oracle ASM) for Oracle AI Database.

This Preface contains the following topics:

- [Audience](#)
- [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 AI 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](#).

Related Documents

For more information, refer to the following Oracle resources:

- *Oracle Advanced Cluster File System Administrator's Guide*
- *Oracle AI Database Administrator's Guide*
- *Oracle AI Database Concepts*
- *Oracle AI Database Reference*
- *Oracle AI Database SQL Language Reference*
- *Oracle Clusterware Administration and Deployment Guide*
- *Oracle Real Application Clusters Administration and Deployment Guide*
- Platform-specific guides, including Oracle AI Database, Oracle Grid Infrastructure, and Oracle Real Application Clusters installation guides

> ⓘ **See Also**
>
> - *Oracle AI Database Licensing Information User Manual* to determine whether a feature is available on your edition of Oracle AI Database
>
> - *Oracle AI Database New Features* for a complete description of the new features in this release
>
> - *Oracle AI Database Upgrade Guide* for a complete description of the deprecated and desupported features in this release

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.

1
Introducing Oracle Automatic Storage Management

Oracle Automatic Storage Management (Oracle ASM), Oracle ASM Cluster File System (Oracle ACFS), and Oracle ASM Dynamic Volume Manager (Oracle ADVM) are key components of storage management.

This chapter provides an overview of Oracle Automatic Storage Management (Oracle ASM), Oracle ASM Cluster File System (Oracle ACFS), and Oracle ASM Dynamic Volume Manager (Oracle ADVM) concepts and features. This chapter contains the following topics:

- [Overview of Oracle Automatic Storage Management](#)
- [About Oracle Advanced Cluster File System](#)
- [Understanding Oracle ASM Concepts](#)
- [Understanding Oracle ASM Disk Group Administration](#)

> ### ⓘ See Also
>
> - The Oracle ASM technologies page at `https://www.oracle.com/database/technologies/rac/asm.html`
> - The Oracle ACFS technologies page at `https://www.oracle.com/database/technologies/rac/acfs.html`
> - "Note for Automatic Storage Management (ASM) (Doc ID 1187723.1)" at [My Oracle Support](#) (`https://support.oracle.com`)

Changes in Oracle Automatic Storage Management 26ai

New Features

These are the new features for Oracle Automatic Storage Management (Oracle ASM) 26ai.

- With the introduction of Oracle AI Database 26ai, it is now possible to scrub individual extents or a range of extents.

  Scrubbing specific extent sets greatly reduces the scrubbing turn-around time, improves the data availability, and minimizes the performance impact. See: Extent-Based Scrubbing

These are desupported features for Oracle ASM 26ai.

- Oracle ASM Filter Driver (ASMFD) is desupported on both Linux and Oracle Solaris beginning with Oracle AI Database 26ai.
  On Linux systems running kernel 5.14 or higher (including Oracle Linux, RedHat Enterprise Linux, and SUSE Linux Enterprise Server), ASMFD filtering was disabled and is unsupported. Oracle recommends moving to ASMlib v3 for those environments, as it provides equivalent functionality. For details, see [My Oracle Support Document 2806979.1](#).

Migration steps are available for <u>Oracle Restart</u> and <u>Oracle Grid Infrastructure</u>. On Oracle Solaris, ASMFD is now desupported.

> ⓘ **See Also**
>
> - *Oracle AI Database Licensing Information User Manual* to determine whether a feature is available on your edition of Oracle AI Database
>
> - *Oracle AI Database New Features* for a complete description of the new features in this release
>
> - *Oracle AI Database Upgrade Guide* for a complete description of the deprecated and desupported features in this release

Overview of Oracle Automatic Storage Management

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

Oracle ASM is a volume manager and a file system for Oracle AI Database files that supports single-instance Oracle AI Database and Oracle Real Application Clusters (Oracle RAC) configurations.

- A volume manager is a software component that manages the mapping of the collection of the pieces of the disks into a 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. 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.

- 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.

Oracle ASM uses disk groups 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 AI 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.

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. When using OMF to create new files, new names are created for the new files so overwriting files is avoided.

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 ASM has easy to use management interfaces such as SQL*Plus, the Oracle ASM Command Line Utility (ASMCMD) command-line interface, and Oracle ASM Configuration Assistant (ASMCA).

> ⓘ **See Also**
>
> - [Administering Oracle ASM Disk Groups](#) for information about administering disk groups
> - [Managing Oracle ASM With ASMCA](#) for information about Oracle ASM Configuration Assistant
> - [Managing Oracle ASM with ASMCMD](#) for information about the ASMCMD command-line interface
> - *Oracle AI Database Administrator's Guide* for information about Oracle AI Database structure and storage

About Oracle Advanced Cluster File System

Oracle Advanced Cluster File System (Oracle ACFS) and Oracle ASM Dynamic Volume Manager (Oracle ADVM) extend Oracle ASM functionality.

Oracle Advanced 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 all customer files. The Oracle ASM Dynamic Volume Manager (Oracle ADVM) provides volume management services and a standard disk device driver interface to clients.

> ⓘ **See Also**
>
> *Oracle Advanced Cluster File System Administrator's Guide* for more information about Oracle ACFS and Oracle ADVM

Understanding Oracle ASM Concepts

The concepts for the key Oracle ASM components are introduced in this topic.

The following topics are discussed:

- [About Oracle ASM Instances](#)

- [About Oracle ASM Disk Groups](#)
- [About Mirroring and Failure Groups](#)
- [About Oracle ASM Disks](#)
- [About Oracle ASM Allocation Units](#)
- [About Oracle ASM Files](#)

> ⓘ **See Also**
>
> [Exploring Considerations for Oracle ASM Storage](#) for information about preparing your storage environment.

About Oracle ASM Instances

An Oracle ASM instance is built on the same technology as an Oracle AI Database instance.

An Oracle ASM instance has a System Global Area (SGA) and background processes that are similar to those of Oracle AI 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.

Oracle ASM is installed in the Oracle Grid Infrastructure home before Oracle AI 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 file names of the files in a disk group
- The location of disk group data file extents
- 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 in a Standard Oracle ASM cluster fails, then all of the database instances on that node also fail. However, in an Oracle Flex ASM configuration, Oracle 12*c* database instances would not fail as they would be able to access another Oracle ASM instance remotely on another node.
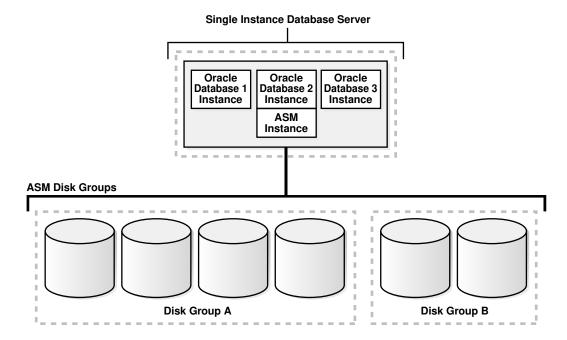
Unlike a file system driver failure, an Oracle ASM instance failure does not require restarting the operating system. In an Oracle RAC environment, the Oracle ASM and database instances on the surviving nodes automatically recover from an Oracle ASM instance failure on a node.

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 AI Databases

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

A clustered storage pool can be shared by multiple single-instance Oracle AI 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 AI Databases

> ⓘ **See Also**
>
> - [Administering Oracle ASM Instances](#) for information about managing an Oracle ASM instance.
> - [Managing Oracle Flex ASM](#) for information about Oracle Flex ASM configurations.

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.

> ⓘ **See Also**
>
> [Administering Oracle ASM Disk Groups](#) for more information about managing 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.

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.

> ⓘ **See Also**
>
> - [Managing Disk Group Templates](#) for information about disk group types and templates
>
> - [Mirroring, Redundancy, and Failure Group Options](#) for more information about mirroring and failure groups

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.

About Oracle ASM 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.

> ⓘ **See Also**
>
> - [Example 4-1](#) for an example that shows how the `AU_SIZE` is specified with the `CREATE DISKGROUP` SQL statement
>
> - [Features Enabled By Disk Group Compatibility Attribute Settings](#) for information about allocation unit sizes and disk group compatibility attributes
>
> - Articles at [My Oracle Support](#) (`https://support.oracle.com`) for additional information about allocation units and Oracle ASM

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 AI Database communicates with Oracle ASM in terms of files. This is similar to the way Oracle AI 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 topics describe Oracle ASM file components:

- [Understanding Oracle ASM Extents](#)
- [Understanding Oracle ASM Striping](#)
- [About Oracle ASM File Templates](#)

Understanding Oracle ASM Extents

The contents of Oracle ASM files are stored in a disk group as a set, or collection, of extents that are stored on individual disks within disk groups.

Each extent resides on an individual disk. An extent, also called data extent, is the raw storage used to hold the contents of a file. 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. The various extent sizes are described in this topic.

For disk groups with AU size less than 4 MB:

- 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+).

For disk groups with AU size greater than or equal to 4 MB and the disk group RDBMS compatibility greater than or equal to `11.2.0.4`, the counts for extents of sizes (the disk group AU size, 4*AU size, or 16*AU size) are calculated using the application block size to support maximum file size.

The extent sizing 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](#).

[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](#)).
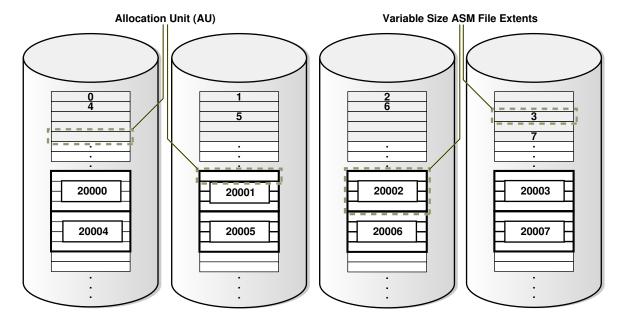
Figure 1-4   Oracle ASM File Allocation in a Disk Group

Understanding Oracle ASM Striping

Oracle ASM striping has two primary purposes: balance loads across all of the disks in a disk group and 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).
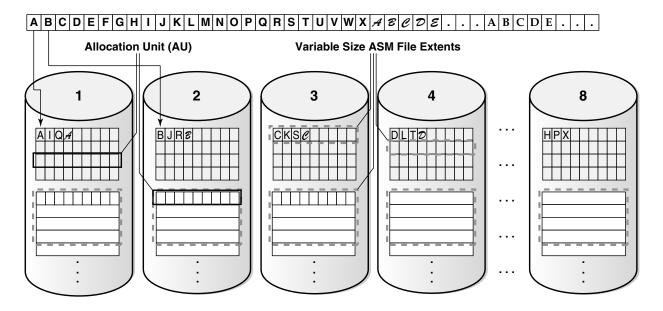
Figure 1-5 and Figure 1-6 are illustrations of Oracle ASM file striping. In both illustrations, the allocation unit size has been set to 1 M (`AU_SIZE = 1M`) for the disk group which consists of 8 disks. The instance is Oracle ASM 11*g* Release 2 (11.2) and the disk group compatibility attributes for 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.
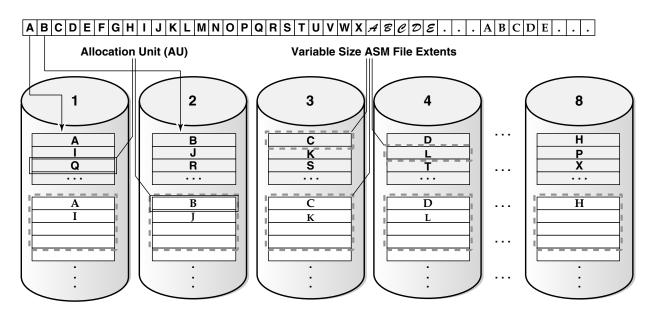
Figure 1-5   Oracle ASM Fine-Grained Striping

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

About Oracle ASM 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.

> ⓘ **See Also**
>
> Managing Disk Group Templates for more information about Oracle ASM templates

Understanding Oracle ASM Disk Group Administration

Oracle ASM disk group administration is introduced in this topic.

The following topics are discussed:

- 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 various factors, such as the platform and whether Oracle Exadata disks are used. The path names that an operating system accepts are always usable as discovery strings.

> ⓘ **See Also**
>
> - `ASM_DISKSTRING` for information about the `ASM_DISKSTRING` initialization parameter
> - Oracle ASM Disk Discovery for information about disk discovery

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.

> ⓘ **See Also**
>
> [Mounting and Dismounting Disk Groups](#) for more information about disk groups

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.

> ⓘ **See Also**
>
> [Altering Disk Groups](#) for more information about altering disk group membership

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 `ASM_POWER_LIMIT` initialization parameter.

> ⓘ **See Also**
>
> - [ASM_POWER_LIMIT](#) for more information about the power limit setting
> - [Manually Rebalancing Disk Groups](#) for more information about disk rebalancing

2

Exploring Considerations for Oracle ASM Storage

Several issues should be considered 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 Device Path and Permission Persistence](#)

Storage Resources for Disk Groups

There are various storage resources that can be used to create Oracle ASM 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 managers 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). 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.

You can use NFS, with or without Direct NFS, to store data files. However, NFS is not supported for Oracle Clusterware files. To install Oracle Real Application Clusters (Oracle RAC) on Windows using NFS, you must also have access to a shared storage method other than NFS for Oracle Clusterware files.

NFS-based quorum disks (quorum failure groups) should not use Direct NFS (dNFS) because dNFS does not support soft mounts. Instead, use a soft mount of a NFS mount point for quorum disks. When using a soft mount, Oracle ASM handles an I/O failure gracefully, and sets only the associated quorum disk offline.

With hard mounts, the Oracle ASM or the database instance may hang if the NFS server becomes unavailable. Note that these hang situations can occur whether or not Direct NFS is used and whether or not Oracle ASM is used for mirroring.

> ⓘ **See Also**
>
> *Oracle Grid Infrastructure Installation and Upgrade Guide* for your operating system for information about Oracle Direct NFS and storage requirements for Oracle ASM

> ⓘ **Note**
>
> - Oracle ASM Dynamic Volume Manager (Oracle ADVM) volumes and Oracle Advanced 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 in some cases. Samba/CIFS clients on Windows cannot use ACLs when interfacing with Oracle ACFS Linux, Solaris, or AIX servers.
>
> - 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).

The procedures for preparing storage resources for Oracle ASM are:

- 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 device names are typically presented from the `/dev` directory with the `/dev/device_name_identifier` name syntax.

- 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.

  – Change the device permissions to read/write

> ⓘ **Note**
>
> To ensure that ownership and permission settings are persistent, you can use `udev` to ensure that the disks do not revert to root ownership when the systems restart.

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.

> ⓘ **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.

For detailed information about preparing disks for an Oracle ASM installation, refer to your platform-specific installation guide for Oracle AI 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 redundant physical path 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.

> ⓘ **See Also**
>
> - [Oracle ASM Disk Discovery](#) for information about disk discovery
> - [My Oracle Support](#) (`https://support.oracle.com`) for articles with information about Oracle ASM and Multipathing
> - Your platform-specific installation guide for information about configuring multipathing for your system

Recommendations for Storage Preparation

Recommendations for storage preparation with Oracle ASM are discussed in this topic.

> ⓘ **Note**
>
> Oracle ASM Filter Driver (ASMFD) is desupported on both Linux and Oracle Solaris beginning with Oracle AI Database 26ai.

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

- Configure a separate disk group for the following:
  - Oracle Cluster Registry (OCR) and voting files
  - Grid Infrastructure Management Repository (GIMR) files
  - Database data files
  - Fast recovery area
- The number of LUNs (Oracle ASM disks) for each disk group should be at least equal to four times the number of active I/O paths. For example, if a disk group has two active I/O paths, then minimum of eight LUNs should be used. The LUNs should be of equal size and performance for each disk group.

  An I/O path is a distinct channel or connection between storage presenting LUNs and the server. An *active* I/O path is an I/O path in which the I/O load on a LUN is multiplexed through multipathing software.

- 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 flash memory and hard disk drives (HDD), 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.

- Configure a minimum of three failure groups for normal redundancy disk groups and five failure groups for high redundancy disk groups to maintain the necessary number of copies of the Partner Status Table (PST) to ensure robustness with respect to storage hardware failures.

- External redundancy disk groups can be used when external hardware, such as a storage array, provides redundancy. 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.

- Use Oracle ASMLib to provide consistent device naming and permission persistency.

> ### ⓘ See Also
>
> - [Creating Disk Groups for a New Oracle Installation](#) for information about configuring disk groups for a new installation
>
> - The Oracle Cloud Storage page on the Oracle Technology Network website at `http://www.oracle.com/technetwork/database/cloud-storage/index.html` for more information about Oracle ASM
>
> - *Oracle AI Database Backup and Recovery User's Guide* for information about configuring the fast recovery area
>
> - *Oracle AI Database Administrator's Guide* for information about specifying a fast recovery area
>
> - [Oracle ASM Failure Groups](#) for information about failure groups

Storage Device Path and Permission Persistence

Before installation, or before configuring new storage devices to use with Oracle ASM, administrators must configure storage device names and ownership to ensure that storage paths and ownership persist after system restarts.

> ### ⓘ Note
>
> Oracle ASM Filter Driver (ASMFD) is desupported on both Linux and Oracle Solaris beginning with Oracle AI Database 26ai.

Use Oracle ASMLib to maintain permissions and manage device paths. On Oracle Solaris, you can also use the Solaris I/O multipathing features to maintain permissions and device paths.

> ### ⓘ See Also
>
> *Oracle Grid Infrastructure Installation and Upgrade Guide* for your operating system for more information about configuring storage devices for path and permission persistence.

3

Administering Oracle ASM Instances

Administering Automatic Storage Management (Oracle ASM) instances is similar to administering Oracle AI Databases.

This chapter describes how to administer Automatic Storage Management (Oracle ASM) instances, configure Oracle ASM instance parameters, and set Oracle AI 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 AI Database instance, but the process requires fewer procedures. You can use Oracle ASM Command Line Utility (ASMCMD) command-line interface, Oracle ASM Configuration Assistant (ASMCA), and SQL*Plus to perform Oracle ASM instance administration tasks.

This chapter contains the following topics:

- [Operating with Different Releases of Oracle ASM and Database Instances Simultaneously](#)
- [Initialization Parameters for Oracle ASM Instances](#)
- [Administration of Oracle ASM Instances](#)
- [Upgrading and Patching Oracle ASM](#)
- [Authentication for Accessing Oracle ASM Instances](#)
- [Migration of a Database to Use Oracle ASM](#)
- [Administering Oracle ASM audit trail](#)

> ⓘ **See Also**
>
> - [About Oracle ASM Instances](#) for a description of an Oracle ASM instance

Operating with Different Releases of Oracle ASM and Database Instances Simultaneously

Oracle Automatic Storage Management (Oracle ASM) 26ai supports Oracle AI Database 26ai or earlier software versions.

> ⓘ **Note**
>
> - An Oracle ASM instance must be at Oracle ASM 26ai to support Oracle AI Database 26ai.
>
> - Refer to *Oracle Exadata documentation* for information about the Oracle AI Database versions that Oracle ASM 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.

When using different software versions, the database instance supports Oracle ASM functionality of the earliest release in use. For example, an Oracle Database 19*c* instance operating with an Oracle ASM 26ai instance only supports Oracle ASM 19*c* 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.

> ⓘ **See Also**
>
> - Disk Group Compatibility for information about disk group compatibility attributes settings
>
> - Example 6-5 for an example showing a query on the `V$ASM_CLIENT` view
>
> - Views Containing Oracle ASM Information for more information about the `V$ASM_CLIENT` and `V$ASM_*` views

Initialization Parameters for Oracle ASM Instances

Settings for Oracle ASM instance initialization parameters are discussed in this topic.

To install and initially configure an Oracle ASM instance, use Oracle Universal Installer (OUI) and Oracle ASM Configuration Assistant (ASMCA).

After an Oracle ASM instance has been installed on a single-instance Oracle AI 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.

The following topics are discussed:

- About Initialization Parameter Files for an Oracle ASM Instance
- About Backing Up, Copying, and Moving an Oracle ASM Initialization Parameter File
- About Oracle ASM Initialization Parameter Settings

- [About Automatic Memory Management for Oracle ASM](#)
- [Recommended Settings for Oracle ASM Initialization Parameters](#)
- [About Setting Database Initialization Parameters for Use with Oracle ASM](#)

> ⓘ **See Also**
>
> - *Oracle AI Database Reference* for information about initialization parameters
> - *Oracle AI Database Administrator's Guide* for information about initialization parameter files
> - *Oracle Grid Infrastructure Installation and Upgrade Guide* for information about installing and configuring Oracle ASM

About 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, then 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 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 nondefault initialization parameters for the Oracle ASM instance.

You can administer Oracle ASM initialization parameter files with SQL*Plus, ASMCA, and ASMCMD commands.

> ⓘ **See Also**
>
> - [Managing Oracle ASM With ASMCA](#) for information about the ASMCA GUI and command-line interfaces
> - [ASMCMD Instance Management Commands](#) for information about ASMCMD commands for managing an Oracle ASM SPFILE; such as `spbackup`, `spcopy`, and `spmove`
> - *Oracle AI Database Administrator's Guide* for more information about creating and maintaining initialization parameter files
> - *Oracle AI Database SQL Language Reference* for information about creating an SPFILE with the `CREATE SPFILE` SQL statement

About 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.

In addition, you can use the SQL `CREATE SPFILE` to create an Oracle ASM SPFILE when connected to the Oracle ASM instance.

You can also 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.

The following topics are discussed:

- [Creating, Copying, and Moving an SPFILE Into a Disk Group](#)
- [Making a Back Up Copy of an Oracle ASM SPFILE in a Disk Group](#)

> ⓘ **See Also**
>
> - spbackup, spcopy, and spmove for information about ASMCMD commands for managing an SPFILE
>
> - *Oracle AI Database Administrator's Guide* for more information about creating and maintaining initialization parameter files
>
> - *Oracle AI 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)

Creating, Copying, and Moving an SPFILE Into a Disk Group

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 instance from Oracle ASM 11*g* Release 1 (11.1) to Oracle ASM 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`. For information about disk group compatibility attributes, see Disk Group Compatibility.

In the following steps, assume an Oracle ASM 11*g* Release 2 (11.2) instance is using a PFILE stored in `$ORACLE_HOME/dbs/asmpfile.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.

   > ⓘ **Note**
   >
   > If you include the `AS COPY` option, then the `CREATE SPFILE FROM PFILE` statement does not update the clusterwide SPFILE location.

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 About Starting Up an Oracle ASM Instance and About Shutting Down an Oracle ASM Instance.

Making a Back Up Copy of an Oracle ASM SPFILE in a Disk Group

This section describes the steps to make a back up copy of an Oracle ASM SPFILE in another disk group using the ASMCMD commands. If necessary, then the backup copy can be used to restore the Oracle ASM SPFILE.

The source and target disk groups must have the disk group attribute `COMPATIBLE.ASM` set to `11.2` or higher.

To make a copy of the Oracle ASM SPFILE in another disk group with the `spcopy` command perform the following steps:

1. Locate the Oracle ASM SPFILE using the ASMCMD `spget` command.

   For example:

   ```
   ASMCMD [+] > spget
   +DATA/ASM/ASMPARAMETERFILE/registry.253.849343867
   ```

2. Copy the Oracle ASM SPFILE to another disk group with `spcopy` command.

   For example:

```
ASMCMD [+] > spcopy +DATA/ASM/ASMPARAMETERFILE/registry.253.849343867 +FRA/spfileCopyASM.ora
```

   Running `spcopy` without the `-u` option does not update the location of the Oracle ASM SPFILE. You can use `spset` to set the location of the Oracle ASM SPFILE in the Grid Plug and Play (GPnP) profile.

3. List all the copies of the Oracle ASM SPFILE file contained in the FRA disk group using the ASMCMD `ls` command.

   For example:

```
ASMCMD [+] > ls -l --absolutepath FRA/ASM/ASMPARAMETERFILE
Type               Redund  Striped  Time            Sys  Name
ASMPARAMETERFILE   MIRROR  COARSE   JUN 06 13:00:00  Y    +FRA/spfileCopyASM.ora =>
REGISTRY.253.849533009
```

4. Verify the current location of the Oracle ASM SPFILE file with the `spget` command.

   For example:

   ```
   ASMCMD [+] > spget
   +DATA/ASM/ASMPARAMETERFILE/registry.253.849343867
   ```

In the event that the current Oracle ASM SPFILE file in a disk group has been corrupted or that disk group is not accessible, you can use `spset` or `spcopy` with the `-u` option to restore the Oracle ASM SPFILE file using the backup copy that you have previously created.

For example:

```
ASMCMD [+] > spcopy -u +FRA/spfileCopyASM.ora +DATA2/ASM/spfileASM.ora
```

About Oracle ASM Initialization Parameter Settings

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 SQL `ALTER SYSTEM` or `ALTER SESSION` statements.

The Oracle ASM parameters use suitable defaults for most environments. You cannot use parameters with names that are prefixed with `ASM_*` in database instance parameter files.

About 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.

An Oracle ASM instance can automatically increase the values set for `MEMORY_TARGET` and `MEMORY_MAX_TARGET` if an ORA-04031 error is raised and automatic memory management is enabled. If `MEMORY_MAX_TARGET` has been explicitly set to a value, then every time ORA-04031 is raised, the `MEMORY_TARGET` value is increased by 10% of the existing `MEMORY_TARGET` value or 128 MB, whichever is greater, but not greater than the customer specified `MEMORY_MAX_TARGET` value. If `MEMORY_MAX_TARGET` is not explicitly set, then both `MEMORY_TARGET` and `MEMORY_MAX_TARGET` are increased by 10% of the existing `MEMORY_TARGET` value or 128 MB, whichever is greater, for a maximum of five increases. The Oracle ASM instance must be rebooted to use the new `MEMORY_TARGET` and `MEMORY_MAX_TARGET` settings.

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 automatic shared memory management and automatic PGA memory management. Unless specified, the behaviors of the automatic memory management parameters in Oracle ASM instances behave the same as in Oracle AI Database instances.

> ⓘ **Note**
>
> - For a Linux environment, automatic memory management cannot work if `/dev/shm` is not available or is undersized. For more information, see *Oracle AI Database Administrator's Reference for Linux and UNIX-Based Operating Systems*. For information about platforms that support automatic memory management, see *Oracle AI Database Administrator's Guide*.
>
> - The minimum `MEMORY_TARGET` for Oracle ASM is 1.75 GB (1792 MB). If you set `MEMORY_TARGET` lower, then Oracle increases the value for `MEMORY_TARGET` to 1.75 GB automatically.
>
> - For the recommended settings of memory initialization parameters in an Oracle Exadata environment, refer to the Oracle Exadata documentation.

> ⓘ **See Also**
>
> - *Oracle AI Database Administrator's Guide* for more information about the functionality of automatic memory management for database instances, which varies from Oracle ASM
> - *Oracle AI Database Concepts* for an overview of memory management methods

Recommended Settings for Oracle ASM Initialization Parameters

> ⓘ **Note**
>
> Oracle ASM Filter Driver (ASMFD) is desupported on both Linux and Oracle Solaris beginning with Oracle AI Database 26ai.

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

- ASM_DISKGROUPS
- ASM_DISKSTRING
- ASM_IO_PROCESSES
- 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

> ⓘ **See Also**
>
> - *Oracle AI Database Reference* for more information about initialization parameters
> - *Oracle AI Database Administrator's Guide* for more information about creating and maintaining an initialization parameter file

ASM_DISKGROUPS

The `ASM_DISKGROUPS` initialization parameter specifies a list of disk group names that an Oracle ASM instance mounts at startup when the SQL `ALTER DISKGROUP ALL MOUNT` statement is issued.

The Oracle ASM instance startup process executes `ALTER DISKGROUP ALL MOUNT` unless the `NOMOUNT` startup option is specified.

The default value of the `ASM_DISKGROUPS` parameter is a `NULL` string.

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` in most situations. Oracle ASM automatically adds a disk group to the parameter when the disk group is successfully created or mounted. Oracle ASM also automatically removes a disk group from the parameter when the disk group is dropped. However, the SPFILE is not updated on a manual dismount.

> ⓘ **Note**
>
> In Oracle Flex ASM configurations, disk groups are not removed from the SPFILE on dismount. Disk groups must always be auto-mounted so that remote instances can access them. A remote instance is an Oracle AI Database, Oracle IOServer, or Oracle ASM proxy instance located on a different node than the Oracle ASM instance.

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 Oracle Database 12*c* Release 1 or later, Oracle ASM configurations support up to 511 disk groups. Oracle ASM configurations with Oracle Database releases before 12*c* Release 1 can only support up to 63 disk groups.

> ⓘ **See Also**
>
> - [About Mounting Disk Groups at Startup](#) for information about disk groups that are mounted at startup time
> - [Mounting and Dismounting Disk Groups](#) for additional information about mounting Oracle ASM disk groups
> - [Managing Oracle Flex ASM](#) for information about Oracle Flex ASM
> - *Oracle AI 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, to limit the discovery process to only include disks that are in the `/dev/rdsk/mydisks` directory for an Oracle ASM instance that does not use Oracle ASMLib, 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`, you could set `ASM_DISKSTRING` as follows on a Linux system:

```
ASM_DISKSTRING = '/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.

Note that the `ASM_DISKSTRING` initialization parameter value may also include strings that include prefixes. For example:

- `orcl:` - Oracle ASMLib
- `n:` - NVMe (Non-Volatile Memory Express)
- `o:` - Oracle Exadata

> ⓘ **See Also**
>
> - [Oracle ASM Disk Discovery](#) for more information about discovering disks
> - *Oracle Grid Infrastructure Installation and Upgrade Guide* for information about configuring storage with NVMe devices and Oracle ASMLIB
> - *Oracle Exadata documentation* for information about the Oracle ASM discovery string format for Oracle Exadata
> - *Oracle AI Database Reference* for more information about the `ASM_DISKSTRING` initialization parameter

ASM_IO_PROCESSES

The `ASM_IO_PROCESSES` initialization parameter specifies the number of IO worker processes to be started in an Oracle IOServer (IOS) server instance.

This parameter is only applicable for an Oracle IOServer instance, which runs in an Oracle Grid Infrastructure home.

IOServer instances are started automatically in the Oracle Grid Infrastructure configuration when generating the credential file for a client cluster. You can configure the number of IOServer instances running in the client cluster.

The default value should work in most cases. However, under heavy IO loads a higher value than the default may be appropriate.

> ⓘ **See Also**
>
> - [Overview of Oracle Flex ASM](#) for information about Oracle IOServer and Oracle Flex ASM
> - *Oracle AI Database Reference* for more information about the `ASM_IO_PROCESSES` 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 higher (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.

The specification of the power of the rebalancing operation in a disk group only affects rebalance operations, not new allocations to a disk group.

> ⓘ **Note**
>
> - Oracle recommends that you do not set the `ASM_POWER_LIMIT` initialization parameter to `0`.
>
> - You cannot set the `ASM_POWER_LIMIT` initialization parameter to `0` in an Oracle Exadata or Oracle Data Appliance environment.

> ⓘ **See Also**
>
> - [Manually Rebalancing Disk Groups](#) and [Tuning Rebalance Operations](#) for information about the `ASM_POWER_LIMIT` initialization parameter, and the `POWER` clause
>
> - [Disk Group Compatibility](#) for information about disk group compatibility
>
> - *Oracle AI Database Reference* for more information about the `ASM_POWER_LIMIT` initialization parameter
>
> - *Oracle AI 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.

> ⓘ **Note**
>
> The `ASM_PREFERRED_READ_FAILURE_GROUPS` initialization parameter has been deprecated in Oracle ASM 12*c* Release 2 (12.2) and may be desupported in a future release.

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, ...`

> ⓘ **Note**
>
> In an Oracle extended cluster, which contains nodes that span multiple physically separated sites, the `PREFERRED_READ.ENABLED` disk group attribute controls whether preferred read functionality is enabled for a disk group. If preferred read functionality is enabled, then this functionality enables an instance to determine and read from disks at the same site as itself, which can improve performance. Whether or not `PREFERRED_READ.ENABLED` has been enabled, preferred read can be set at the failure group level on an Oracle ASM instance or a client instance in a cluster with the `ASM_PREFERRED_READ_FAILURE_GROUPS` initialization parameter, which is available for backward compatibility.
>
> The failure group values set by `ASM_PREFERRED_READ_FAILURE_GROUPS` take precedence over the `PREFERRED_READ.ENABLED` site values. The values of `ASM_PREFERRED_READ_FAILURE_GROUPS` on the client instances take precedence over the values on the Oracle ASM instances.

> ⓘ **See Also**
>
> - [Preferred Read Failure Groups](#) for more information about preferred read failure groups
>
> - [PREFERRED_READ.ENABLED](#) for information about the `PREFERRED_READ.ENABLED` disk group attribute
>
> - *Oracle Real Application Clusters Administration and Deployment Guide* for more information about configuring preferred disks in extended clusters
>
> - *Oracle AI Database Reference* for more information about the `ASM_PREFERRED_READ_FAILURE_GROUPS` initialization parameter

DB_CACHE_SIZE

The setting for the `DB_CACHE_SIZE` parameter determines the size of the buffer cache.

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

This buffer cache stores metadata blocks. The default value for this parameter is suitable for most environments.

> ⓘ **See Also**
>
> - *Oracle AI Database Administrator's Guide* for more information about setting the `DB_CACHE_SIZE` initialization parameter
>
> - *Oracle AI Database Performance Tuning Guide* for more information about memory configuration
>
> - *Oracle AI 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.

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

INSTANCE_TYPE

The `INSTANCE_TYPE` initialization parameter specifies whether the instance is a database instance, an Oracle Automatic Storage Management (Oracle ASM) instance, or an Oracle ASM Proxy instance.

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

```
INSTANCE_TYPE = ASM
```

In addition to values `asm` and `rdbms`, `INSTANCE_TYPE` can be set to `asmproxy` in an Oracle Flex ASM configuration.

> ⓘ **See Also**
>
> - [Managing Oracle Flex ASM](#) for information about Oracle Flex ASM
> - *Oracle AI Database Reference* for more information about the `INSTANCE_TYPE` parameter

LARGE_POOL_SIZE

The setting for the `LARGE_POOL_SIZE` parameter is used for large allocations.

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

The default value for this parameter is suitable for most environments.

Chapter 3
Initialization Parameters for Oracle ASM Instances

> ⓘ **See Also**
>
> - *Oracle AI Database Administrator's Guide* for more information about setting the `LARGE_POOL_SIZE` initialization parameter
> - *Oracle AI Database Performance Tuning Guide* for more information about memory configuration
> - *Oracle AI 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, then you can use the following formulas, where n is the number of database instances connecting to the Oracle ASM instance.

In a non-Exadata environment, the recommended settings are:

- For n < 10, `PROCESSES` = 50*n + 50
- For n >= 10, `PROCESSES` = 10*n + 450

In an Oracle Exadata environment, the recommended setting is `PROCESSES` = `MAX`(450 + 10*n, 1024).

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

SHARED_POOL_SIZE

The setting for the `SHARED_POOL_SIZE` parameter determines the amount of memory required to manage the instance.

You do not have to set a value for the `SHARED_POOL_SIZE` initialization parameter if you use automatic memory management. 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 AI Database Administrator's Guide* for more information about setting the `SHARED_POOL_SIZE` initialization parameter
> - *Oracle AI Database Performance Tuning Guide* for more information about memory configuration
> - *Oracle AI Database Reference* for more information about the `SHARED_POOL_SIZE` parameter

About 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.

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

Administration of Oracle ASM Instances

Oracle ASM is typically installed in an Oracle Grid Infrastructure home separate from the Oracle AI Database home. Only one Oracle ASM instance is supported on a server in a standard configuration; however, Oracle Flex ASM provides additional configuration options.

When managing an Oracle ASM instance, the administration activity should be performed in the Oracle Grid Infrastructure home.

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

- Using Oracle Grid Infrastructure for a Standalone Server
- About Starting Up an Oracle ASM Instance
- About Shutting Down an Oracle ASM Instance
- About Administering Oracle ASM Instances with Server Control Utility
- About Out of Place Upgrades
- About Configuring Oracle Grid Infrastructure with the Configuration Wizard
- About Active Session History Sampling for Oracle ASM
- About Oracle Home User on Windows

> ⓘ **See Also**
>
> - Managing Oracle Flex ASM for information about Oracle Flex ASM
> - Creating Database Files in Oracle ASM Using SQL*Plus for information about Oracle ASM and creating database files

Using Oracle Grid Infrastructure for a Standalone Server

Oracle Grid Infrastructure for a standalone server (Oracle Restart) improves the availability of your Oracle AI Database.

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

Oracle Restart provides managed startup and restart of a single-instance (non-clustered) Oracle AI 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 AI 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 and Upgrade Guide* for information about installation of Oracle Grid Infrastructure

About 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 AI Database instance with some minor differences.

When starting an Oracle ASM instance with SQL*Plus, 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 `+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.

  > ⓘ **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.

After the Oracle ASM instance has started, you can mount disk groups with the `ALTER DISKGROUP...MOUNT` command.

- The associated Oracle AI 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 Advanced 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.

- `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.

  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
```

> ⓘ **See Also**
>
> - [Authentication for Accessing Oracle ASM Instances](#) for more information about user authentication
>
> - [startup](#) for information about starting an Oracle ASM instance with ASMCMD `startup` command
>
> - [About Restricted Mode](#) for information about restricted mode
>
> - [About Mounting Disk Groups at Startup](#) for information about disk groups that are mounted at startup time
>
> - [Mounting and Dismounting Disk Groups](#) for information about mounting disk groups
>
> - *Oracle Advanced Cluster File System Administrator's Guide* for information about dismounting an Oracle ACFS file system
>
> - *Oracle AI Database Administrator's Guide* for more information about using environment variables to select instances
>
> - *Oracle AI 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 AI Database Concepts* for information about Oracle AI Database background processes
>
> - *Oracle AI 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 `/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=/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](#).

   For example:

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

> ⓘ **See Also**
>
> *Oracle AI Database Administrator's Guide* for more information about using `STARTUP` with a nondefault 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](#).

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.

About Shutting Down an Oracle ASM Instance

You can initiate the Oracle ASM shutdown process by running 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.

If you are not using Oracle Flex ASM, 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 you are using Oracle Flex ASM, Oracle Flex ASM clients move to other running Oracle ASM instances if an Oracle ASM instance is shut down.

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`.

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
```

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 stops.

  If any Oracle Advanced 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.

> ⓘ **See Also**
>
> - [Authentication for Accessing Oracle ASM Instances](#) for more information about user authentication
> - [Authentication for Accessing Oracle ASM Instances](#) for more information about user authentication on Oracle ASM instance
> - *Oracle Advanced Cluster File System Administrator's Guide* for information about dismounting an Oracle ACFS file system
> - *Oracle AI Database Administrator's Guide* for more information about using environment variables to select instances
> - *Oracle AI 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)

About 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 Oracle Clusterware (CRS) resource in Oracle Cluster Registry (OCR)
- Enable, disable, start, and stop Oracle ASM instances
- Display the Oracle ASM instance configuration and status
- Add, modify, and remove Oracle ASM network resources
- Add, modify, remove, enable, disable, start, and stop Oracle ASM listener resources

> ⓘ **See Also**
>
> *Oracle Clusterware Administration and Deployment Guide* and *Oracle Real Application Clusters Administration and Deployment Guide* for information about SRVCTL commands

About 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 11*g* Release 2 (11.2) is not supported. For example, an upgrade of Oracle Grid Infrastructure 11*g* Release 2 (11.2.0.1) to Oracle Grid Infrastructure 11*g* Release 2 (11.2.0.2) must be an out of place upgrade.

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

About 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.

About 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 AI Database Performance Tuning Guide* for more information about gathering performance statistics
> - *Oracle AI Database Reference* for a description of the `V$ACTIVE_SESSION_HISTORY` view

About Oracle Home User on Windows

Oracle AI Database supports the use of an Oracle home user, which can be specified at installation time. The Oracle home user is associated with an Oracle home and it cannot be changed after installation. Different Oracle homes on a system can share the same Oracle home user or use different Oracle home user names.

In previous releases on Windows operating systems, Oracle services were required to run as `Local System` privileges, which are fully privileged. This feature enables the database, listener, and job scheduler services to run with low and non-administrative user privileges to allow tighter control of security. The Oracle home user can be a built-in account or a Windows user account. A Windows user account should be a low privileged (non-Administrator) account to ensure that the Oracle home user has a limited set of privileges, ensuring that Oracle AI Database services have only those privileges required to run Oracle products. The Windows user account can be a Local User, a Domain User, or a Managed Services Account in general.

However, Oracle RAC, Oracle Restart, and Oracle Grid Infrastructure installations require the use of the Domain User as the Oracle home user because a clusterwide identity is necessary.

> ⓘ **Note**
>
> Oracle RAC, Oracle Restart, and Oracle Grid Infrastructure installations require the use of the Domain user as the Oracle home user because a clusterwide identity is necessary.

> ⓘ **See Also**
>
> *Oracle AI Database Platform Guide for Microsoft Windows* for information about running Oracle services on Windows platforms and different types of Windows user accounts

Upgrading and Patching Oracle ASM

This section contains the following topics:

- [About Oracle ASM Rolling Upgrade](#)
- [About Oracle ASM Rolling Patches](#)

> ⓘ **Note**
>
> - For Oracle RAC environments, the Oracle Clusterware version number must be at least equal to the version number of the patch that you are applying to the Oracle AI Database.
>
> - You must apply the patch to the Oracle Grid Infrastructure home before you apply it to the Oracle AI Database home.

About Oracle ASM Rolling Upgrade

Oracle ASM rolling upgrade enables you to independently upgrade or patch clustered Oracle ASM nodes without affecting database availability which provides 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.

To perform a rolling upgrade, 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 Oracle ASM is upgraded with Oracle Clusterware for Oracle 11*g* Release 2 (11.2) or later as both are in the Oracle Grid Infrastructure home.

> ⓘ **Note**
>
> - Rolling upgrades only apply to clustered Oracle ASM instances.
> - 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 and Upgrade Guide* for information about performing, and restrictions related to, a rolling upgrade of Oracle ASM
> - *Oracle AI Database Upgrade Guide* for information about upgrading Oracle AI Database
> - *Oracle AI Database SQL Language Reference* for information about the rolling migration clause of the `ALTER SYSTEM` command

About Oracle ASM Rolling Patches

You can apply patches in a clustered Oracle ASM environment to update one node at a time to the latest patch level without affecting the overall availability of the Oracle ASM cluster or the database clusters using Oracle ASM for storage.

The `ALTER SYSTEM ROLLING PATCH` SQL statement enables you to start and stop rolling patches. For example:

```
SQL> ALTER SYSTEM START ROLLING PATCH;

SQL> ALTER SYSTEM STOP ROLLING PATCH;
```

You can determine if the cluster is in rolling patch mode by executing a `SYS_CONTEXT` SQL query for Cluster State. A new state (In Rolling Patch) is added to informing the user that the cluster is in rolling patch mode.

The queries in display information about rolling patches. To run these queries, you must be connected to the Oracle ASM instance in the Grid home, and the Grid Infrastructure home must be configured with the Oracle Clusterware option for an Oracle RAC environment.

You can view all the patch Ids applied on the node and cluster by querying the `V$PATCHES` view.

ASMCMD commands for rolling patches include:

- `showclusterstate`
- `showpatches`
- `showversion`

Example 3-2   Determining rolling patch mode and patch level

```
SELECT SYS_CONTEXT('SYS_CLUSTER_PROPERTIES', 'CLUSTER_STATE') FROM DUAL;

SELECT SYS_CONTEXT('SYS_CLUSTER_PROPERTIES', 'CURRENT_PATCHLVL') FROM DUAL;
```

> ⓘ **See Also**
>
> - **ASMCMD Instance Management Commands** for information about ASMCMD commands to monitor upgrade operations on an Oracle ASM instance
>
> - *Oracle AI Database SQL Language Reference* for information about the rolling patch clause of the `ALTER SYSTEM` command
>
> - *Oracle Grid Infrastructure Installation and Upgrade Guide* for information about performing, and restrictions related to, a rolling migration of Oracle ASM and Oracle Grid Infrastructure homes

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

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

- Managing a Shared Password File in a Disk Group

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.

> ⓘ **See Also**
>
> - **Managing Oracle ASM File Access Control for Disk Groups** for information about file permissions and Oracle ASM File Access Control
>
> - *Oracle Grid Infrastructure Installation and Upgrade Guide* for your operating system-specific information about how to ensure that the Oracle ASM and database instances have member disk access
>
> - *Oracle AI Database Security Guide* for information about maintaining database security, including assigning passwords

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

> ⓘ **See Also**
>
> *Oracle Advanced Cluster File System Administrator's Guide* for information about privileges and Oracle ACFS

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

Table 3-1   (Cont.) One operating system group and one set of privileges for all Oracle ASM users

Role/Software Owner	User	Group/Privilege
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 (for example: `asmadmin`)

  This group is granted the SYSASM privilege, which provides full administrative privileges for the Oracle ASM instance.

- OSDBA for Oracle ASM group (for example: `asmdba`)

  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 (`dba`).

- OSOPER for Oracle ASM group (for example: `asmoper`)

  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.

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 AI 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.

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.
> - Oracle ASM does not support the creation of external and global users.

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 AI Database password file. You must run `DROP USER` to delete a user with no privileges in an Oracle ASM password file.

> ⓘ **See Also**
>
> - [orapwusr](#) for information about creating a user with Oracle ASM command-line utility (ASMCMD)
> - *Oracle AI Database SQL Language Reference* for information about the `CREATE USER` and `GRANT` commands
> - *Oracle AI Database Administrator's Guide* for information about creating and maintaining a password file

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_ASMADMIN`, `ORA_ASMDBA`, and `ORA_ASMOPER` are the operating system groups designated for `OSASM`, `OSDBA` and `OSOPER` respectively for Oracle ASM.

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

> ⓘ **See Also**
>
> - *Oracle AI Database Administrator's Guide* for more information about using operating system authentication
> - *Oracle Grid Infrastructure Installation and Upgrade Guide* for information about installation of the Oracle Grid Infrastructure
> - *Oracle AI Database Installation Guide* for your specific operating system for information about default operating system groups

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.

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.

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 use password file authentication.

> ⓘ **See Also**
>
> - [About Privileges for Oracle ASM](#) for information about privileges for Oracle ASM
> - *Oracle AI Database Administrator's Guide* for information about creating and maintaining a password file
> - *Oracle AI Database SQL Language Reference* for information about the `CREATE USER` and `GRANT` commands
> - *Oracle AI Database Security Guide* for information about database security
> - *Oracle AI 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.

Managing a Shared Password File in a Disk Group

You can manage a shared password file on a disk group enabling access to the single file across the cluster system.

This sections contains these topics:

- [About a Shared Password File in a Disk Group](#)
- [Creating a Password File in a Disk Group](#)
- [Backing Up and Restoring an Oracle ASM Password File in a Disk Group](#)

> ⓘ **Note**
>
> After a change has been made to the location of the password file, the following SQL statement should be executed to flush the password file metadata cache.
>
> ```
> SQL> ALTER SYSTEM FLUSH PASSWORDFILE_METADATA_CACHE
> ```
>
> The location of the password file can be changed by running `orapwd`, or the ASMCMD `pwcopy`, `pwcreate`, `pwdelete`, `pwmove`, or `pwset` command.

For information about ASMCMD commands to manage an Oracle ASM or database instance password file in a disk group, refer to [ASMCMD Instance Management Commands](#).

> ⓘ **See Also**
>
> - *Oracle AI Database Administrator's Guide* for information about maintaining a password file
> - *Oracle AI Database Reference* for information about the `REMOTE_LOGIN_PASSWORDFILE` initialization parameter
> - *Oracle Clusterware Administration and Deployment Guide* and *Oracle Real Application Clusters Administration and Deployment Guide* for information about SRVCTL commands

About a Shared Password File in a Disk Group

An individual password file for Oracle AI Database or Oracle ASM can reside on a designated Oracle ASM disk group. Having the password files reside on a single location accessible across the cluster reduces maintenance costs and situations where passwords become out of sync.

You can use a password file located on a disk group for authentication only if the Oracle ASM instance is running and the designated disk group is mounted. Otherwise, operating system authentication must be used to bootstrap the startup of the Oracle ASM instance and stack.

The `COMPATIBLE.ASM` disk group attribute must be set to `12.1` or higher for the disk group where the password is to be located. The `SYSASM` privilege is required to manage the Oracle ASM password file. The `SYSDBA` privilege on Oracle ASM is required to manage the database password file.

The shared password file in a disk group is managed by ASMCMD commands, the `ORAPWD` tool, and SRVCTL commands. `ORAPWD` supports the creation of password files on an Oracle ASM disk group. All other password file manipulation is performed with ASMCMD or SRVCTL commands.

Before running commands, such as `ORAPWD`, to create a password file, ensure that the `ORACLE_SID` and `ORACLE_HOME` environmental variables have been set properly. For example, before setting the password file for Oracle ASM, set the `ORACLE_SID` and `ORACLE_HOME` environmental variables to ensure that you can connect to the local Oracle ASM instance. For information about environmental variables and connecting to an Oracle ASM instance, refer to [Connecting To and Starting Up an Oracle ASM Instance](#).

SRVCTL provides commands to manage a password file in a disk group, such as the following commands for updating and displaying the location of the password file:

```
$ srvctl modify asm -pwfile location
$ srvctl modify database -db dbname -pwfile location
$ srvctl config asm
```

Creating a Password File in a Disk Group

> ⓘ **Note**
>
> Do not use `orapwd` to create a new Oracle ASM password file when Oracle ASM is configured in a cluster. If the Oracle ASM password file is inaccessible in a cluster configuration, you must restore the password file using an available backup rather than create a new file. For articles about recreating a shared Oracle ASM password file in a cluster, such as document 1929673.1, refer to articles at [My Oracle Support](#) (`https://support.oracle.com`).

You can use `orapwd` to create a password file in a specified Oracle ASM disk group.

- To create an Oracle AI Database password file in a disk group from the existing password file on the file system, run `orapwd` with the `dbuniquename` switch. The `dbuniquename` switch specifies to create an Oracle AI Database password file for the database instance identified by `orcl`.

  1. Determine that the Oracle AI Database password file is currently located on the file system.

```
SQL>  SELECT file_name FROM V$PASSWORDFILE_INFO;
FILE_NAME
----------------------------------------------------------------------
 /u01/app/oracle/product/19.1.0/db_home1/dbs/orapworcl
```

2. Run `orapwd` to move the password file to an Oracle ASM disk group.

```
$ orapwd file='+data/ORCL/orapwdb' dbuniquename='orcl'

Enter password for SYS:
```

3. After the password file has been moved, update the password metadata cache with the `ALTER SYSTEM` statement and then confirm the new password file location.

```
SQL>  ALTER SYSTEM FLUSH PASSWORDFILE_METADATA_CACHE;
System altered.

SQL>  SELECT file_name FROM v$passwordfile_info;
FILE_NAME
----------------------------------------------------------------------
+DATA/ORCL/orapwdb
```

- To create an Oracle ASM password file in a disk group, run `orapwd` with the `asm` switch set to `y`.

```
$ orapwd file='+data/ASM/orapwasm' asm=y

Enter password for SYS:
```

- To create an Oracle ASM password file in an Oracle ASM disk group from an existing password file, run `orapwd` with the `input_file` switch and the `asm` switch set to `y`.

```
$ orapwd input_file='/oraclegrid/dbs/orapwasm' file='+data/ASM/orapwasm' asm=y

Enter password for SYS:
```

Backing Up and Restoring an Oracle ASM Password File in a Disk Group

You can make a backup of the Oracle ASM password file, and if the Oracle ASM password file is lost or the disk group becomes inaccessible, then you can restore the backup password file.

This section describes the steps to back up the Oracle ASM password file to a disk group and the steps to restore the Oracle ASM password file.

The source and target disk groups must have the disk group attribute `COMPATIBLE.ASM` set to `12.1` or higher.

1. Locate the password file using the ASMCMD `pwget` command.

   For example:

   ```
   ASMCMD [+] > pwget --asm
   +DATA/orapwasm
   ```

2. Back up the password file to another disk group with the `pwcopy` command.

   For example:

   ```
   ASMCMD [+] > pwcopy +DATA/orapwasm +FRA/my_pwfile_backup
   ```

   Using `pwcopy` without the `--asm` or `--dbuniquename` option does not change the current location of the password file. If necessary after the copy is made, you can set the current password file location with the `pwset` command.

3. Verify which password file is in the current location after making a backup with the `pwcopy` command.

For example:

```
ASMCMD [+] > pwget --asm
+DATA/orapwasm
```

4. Verify the backup password file was created.

For example:

```
ASMCMD [+] > ls +fra/my_pwfile_backup
my_pwfile_backup
```

5. To restore the Oracle ASM password file, you can use `pwset` or `pwcopy`.

To restore the Oracle ASM password file from the backup and use the existing location, use the `pwset` command with the `--asm` option.

For example:

```
ASMCMD [+] > pwset --asm +FRA/my_pwfile_backup
```

To restore the Oracle ASM password file from the backup to another disk group, use the `pwcopy` command with the `--asm` option.

For example:

```
ASMCMD [+] > pwcopy --asm +FRA/my_pwfile_backup +DATA2/my_orapwasm
```

The `--asm` option with the `pwset` and `pwcopy` command sets the password location for the Oracle ASM instance.

6. Verify the location of the current password file with the `pwget` command if you have changed the location.

For example:

```
ASMCMD [+] > pwget --asm
+DATA2/my_orapwasm
```

If the location of the current password file displayed is not the expected location, then flush the password metadata cache to refresh the password file information.

```
SQL>  ALTER SYSTEM FLUSH PASSWORDFILE_METADATA_CACHE;
System altered.
```

Migration of a Database to Use Oracle ASM

With a new installation of Oracle AI Database and Oracle ASM, you can initially create your database and select the Oracle ASM storage option. If you have an existing Oracle AI Database that stores database files in the operating system file system, 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:

• About Using Oracle Recovery Manager to Migrate Databases to Oracle ASM

• Best Practices Technical Briefs on Migrating to Oracle ASM

About 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 Performing Oracle ASM Data Migration with RMAN.

Best Practices Technical Briefs on Migrating to Oracle ASM

The Oracle Maximum Availability Architecture (MAA) website provides excellent best practices technical technical briefs based on different scenarios, such as:

- Minimal Downtime Migration to Oracle ASM

- Platform Migration using Transportable Tablespaces

- Platform Migration using Transportable Database

> ⓘ **See Also**
>
> - Information about best practices for migrating to Oracle ASM on the Oracle Maximum Availability Architecture (MAA) website at `https://www.oracle.com/database/technologies/high-availability/maa.html`
>
> - Additional information on the Oracle ASM website at `https://www.oracle.com/database/technologies/rac/asm.html`

Administering Oracle ASM audit trail

This document explains how to manage the audit trail records in Oracle ASM instances.

Oracle ASM audit records with Syslog

- Oracle ASM audit trail records are redirected to the Syslog facility.

- With this facility, the Oracle ASM audit trail records are written to */var/log/oraasmaudit.log* file.

- Log rotation is configured for Oracle ASM audit trail records in */etc/logrotate.d/oraasmaudit* configuration file. Audit logs are rotated once every four weeks and will be compressed after they are rotated.

Managing Oracle ASM audit records in Operating system audit trail

Operating system audit trail

Enabling Operating system audit trail

To disable syslog auditing and enable Operating system audit trail, set `AUDIT_SYSLOG_LEVEL` initialization parameter to `NULL` and `AUDIT_TRAIL` initialization parameter to 'OS'.

Purging Operating system audit trail files

Refer to the `audcreatejob`, `audsettimestamp`, and `audcleanaudittrail` ASMCMD commands for information about managing Oracle ASM audit trails. See <u>ASMCMD Audit Files Management Commands</u>.

Related Topics

- AUDIT_FILE_DEST initialization parameter
- AUDIT_TRAIL initialiation parameter
- Administering the Audit Trail

Managing Oracle ASM Audit Records in Unified Audit Trail

Unified Audit Trail

- See Oracle AI Database Security Guide for more information about unified auditing.
- Unified audit trail records are available through
    - `GV$UNIFIED_AUDIT_TRAIL` view for Oracle ASM RAC instances.

Enabling Unified audit trail

- See Oracle AI Database Security Guide for more details on enabling unified audit trail.

Purging Unified audit trail files

- See Oracle AI Database Security Guide for more information on purging audit trail files.
- <u>ASMCMD Audit Files Management Commands</u> provides detailed information about ASMCMD commands to manage Oracle ASM audit trail files.

When viewing RAC ASM generated audit records and the internal table `AUDSYS.AUD$UNIFIED` is not accessible for any reason then only audit records will be written to spillover files which can be queried using `GV$UNIFIED_AUDIT_TRAIL`. You should query the `UNIFIED_AUDIT_TRAIL` view if an instance is OPEN (either READ ONLY or READ WRITE). The `UNIFIED_AUDIT_TRAIL` view internally fetches records from both the `AUDSYS.AUD$UNIFIED` table and the `GV$UNIFIED_AUDIT_TRAIL` view.

> ⓘ **Note**
>
> The `SYS.AUDIT_ACTIONS` table which logs information regarding the `ACTION CODE` and `ACTION NAME` is not accessible from non-database instances. Users can cross verify the `ACTION_CODE` logged by `GV$UNIFIED_AUDIT_TRAIL` against the `ACTION_NAME` in `SYS.AUDIT_ACTIONSACTION_CODE` and `ACTION_NAME` items that are relevant for ASM instances include:
>
> **Figure 3-1   Action Codes and Names**
>
> ```
> ACTION_CODE       ACTION_NAME
> -----------       ----------
>           3       SELECT
>          42       ALTER SESSION
>          43       ALTER USER
>          47       PL/SQL EXECUTE
>          49       ALTER SYSTEM
>          51       CREATE USER
>          53       DROP USER
>         100       LOGON
>         101       LOGOFF
>         114       GRANT ROLE
>         115       REVOKE ROLE
>         138       DATABASE STARTUP
>         139       DATABASE SHUTDOWN
>         187       CREATE SPFILE
>         188       CREATE PFILE
>         193       ALTER DISKGROUP
>         194       CREATE DISKGROUP
>         195       DROP DISKGROUP
> ```

With the introduction of Oracle AI Database 26ai, two new static `init.ora` parameters are included to configure auditing in ASM Instances:

- `ASM_UNIFIED_AUDIT_LEVEL`: Defines what kind of actions are to be audited.

- `ASM_UNIFIED_AUDIT_DESTINATION`: Specifies where to write the audit records.

After modifying these parameters, the instance must be restarted for the changes to take effect. The allowed set of values for the parameters are as follows:

`ASM_UNIFIED_AUDIT_LEVEL`:

- `ALL` - ALL actions done on ASM instance will be audited.

- `BASIC` - Only specific important actions, i.e: `CONNECT`, `SHUTDOWN` and STARTUP will be audited. (DEFAULT VALUE)

`ASM_UNIFIED_AUDIT_DESTINATION` :

- `FILES` - Writes the generated audit records to spillover files.

- `SYSTEMLOG` - Writes audit records to configured system log (SYSLOG) location. (DEFAULT VALUE)

> **ⓘ Note**
>
> These parameters are applicable to ASM, ASM PROXY and REMOTE ASM as well.

`ASM_UNIFIED_AUDIT_DESTINATION` works in conjunction with `UNIFIED_AUDIT_SYSTEMLOG` `init.ora` parameter which is used to determine the 'facility' and 'priority' values. When `ASM_UNIFIED_AUDIT_DESTINATION` has `SYSTEMLOG` set, we use the values in `UNIFIED_AUDIT_SYSTEMLOG` to determine where/how to write to `SYSTEMLOG`.

> **ⓘ Note**
>
> With the release of Oracle AI Database 26ai, `UNIFIED_AUDIT_SYSTEMLOG` is now modifiable in ASM instances as well.

Audit Trail Properties in Operating System and Unified Audit Trail

Table 3-3   Audit Trail Size and Age Properties

Property Name	Description
Audit file max size	Audit file max size can have a minimum value of 1 KB and maximum value of 2000000 KB. The default value is 10000 KB.
	Oracle ASM instance will stop writing audit records to the audit files upon reaching the file max size limit. The files are renamed and a new file will be created for subsequent audit records.
Audit file max age	Audit file max age can have a minimum value of 1 day and maximum value of 497 days. The default value is 5 days.
	Oracle ASM instance will stop writing audit records to the audit files upon reaching the file max age limit. The files are renamed and a new file will be created for subsequent audit records.
Audit purge job interval	Audit purge job interval can have a minimum value of 1 hour and maximum value of 999 hours.

> **ⓘ Note**
>
> Unified Auditing, when enabled in ASM instances, will mandatorily audit only `CONNECT` and `SHUTDOWN`. This is the same restriction that existed in Oracle 19c. The `UNIFIED_AUDIT_SYSTEMLOG` `init.ora` parameter [when set in ASM instances] will redirect the audit records to syslog and will NOT write a copy of the full audit record to the OS files.

> **ⓘ See Also**
>
> - [ASMCMD Audit Files Management Commands](#) for information about setting the properties
> - *Oracle AI Database PL/SQL Packages and Types Reference* for more details about audit trail properties.

4

Administering Oracle ASM Disk Groups

Administration of Oracle ASM disk groups includes creating, altering, dropping, mounting, and dismounting tasks.

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:

- [Managing 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](#)
- [Maintaining Disk Groups](#)
- [Managing Oracle ASM Flex Disk Groups](#)

> ⓘ **See Also**
>
> - [About Starting Up an Oracle ASM Instance](#) for information about starting up an Oracle ASM instance
>
> - [Managing Oracle ASM With ASMCA](#) for information about administering Oracle ASM disk groups with Oracle ASM Configuration Assistant (ASMCA)
>
> - [Managing Oracle ASM with ASMCMD](#) for information about administering Oracle ASM disk groups with ASMCMD
>
> - The Oracle Cloud Storage page on the Oracle Technology Network website at `http://www.oracle.com/technetwork/database/cloud-storage/index.html` for more information about Oracle ASM

Managing Disk Group Attributes

Disk group attributes are parameters that are bound to a disk group, rather than an Oracle ASM instance. Some disk group attributes can be set when a disk group is created or altered. Other disk attributes can only be set when a disk group is created or only be set when a disk group is altered.

This section contains the following topics:

- [About Viewing and Setting Disk Group Attributes](#)
- [Summary of Disk Group Attributes](#)

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](#).

About Viewing and Setting Disk Group Attributes

You can set and display disk group attribute settings.

You can display disk group attributes with the `V$ASM_ATTRIBUTE` view and the ASMCMD `lsattr` command.

Disk group attributes can be set with the `ATTRIBUTE` clause of the `ALTER DISKGROUP` or `CREATE DISKGROUP` SQL statement and the ASMCMD `setattr` and `mkdg` commands.

> ⓘ **See Also**
>
> - [Example 6-1](#) for an example of the use of the `V$ASM_ATTRIBUTE` view
> - [lsattr](#) for information about the `lsattr` command
> - [setattr](#) for information about the `setattr` command
> - [mkdg](#) for information about the `mkdg` command
> - [Managing Disk Groups with ASMCA](#) for information about setting disk group attributes with Oracle ASM Configuration Assistant (ASMCA)

Summary of Disk Group Attributes

The Oracle ASM disk group attributes are listed in this section.

- [ACCESS_CONTROL.ENABLED](#)
- [ACCESS_CONTROL.UMASK](#)
- [AU_SIZE](#)
- [CELL.SMART_SCAN_CAPABLE](#)
- [CELL.SPARSE_DG](#)
- [COMPATIBLE.ASM](#)
- [COMPATIBLE.RDBMS](#)
- [COMPATIBLE.ADVM](#)

- [CONTENT.CHECK](#)

- [CONTENT.TYPE](#)

- [CONTENT_HARDCHECK.ENABLED](#)

- [DISK_REPAIR_TIME](#)

- [FAILGROUP_REPAIR_TIME](#)

- [IDP.BOUNDARY and IDP.TYPE](#)

- [LOGICAL_SECTOR_SIZE](#)

- [PHYS_META_REPLICATED](#)

- [PREFERRED_READ.ENABLED](#)

- [SCRUB_ASYNC_LIMIT](#)

- [SECTOR_SIZE](#)

- [STORAGE.TYPE](#)

- [VAM_MIGRATION_DONE](#)

ACCESS_CONTROL.ENABLED

This attribute configures Oracle ASM File Access Control and 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](#).

ACCESS_CONTROL.UMASK

This attribute configures Oracle ASM File Access Control and 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](#).

AU_SIZE

This attribute specifies allocation unit (AU) size and can only be set when creating a disk group. [Example 4-1](#) shows how the `AU_SIZE` is specified with the `CREATE DISKGROUP` SQL statement.

For information about allocation units, refer to [About Oracle ASM Allocation Units](#) .

CELL.SMART_SCAN_CAPABLE

This attribute is only applicable to the configuration of Oracle Exadata storage.

> ⓘ **See Also**
>
> *Oracle Exadata documentation*

CELL.SPARSE_DG

This attribute is only applicable to the configuration of Oracle Exadata storage.

> ⓘ **See Also**
>
> *Oracle Exadata documentation*

COMPATIBLE.ASM

This attribute specifies the Oracle ASM compatibility level of a disk group. Compatibility attributes can be set when the disk group is created and advanced when a disk group is altered, but compatibility attributes cannot be reversed. For information about the `COMPATIBLE.ASM` attribute, refer to [Disk Group Compatibility Attributes](#).

COMPATIBLE.RDBMS

This attribute specifies the database compatibility level of a disk group. Compatibility attributes can be set when the disk group is created and advanced when a disk group is altered, but compatibility attributes cannot be reversed. For information about the `COMPATIBLE.RDBMS` attribute, refer to [Disk Group Compatibility Attributes](#).

COMPATIBLE.ADVM

This attribute specifies the Oracle Advanced Cluster File System (Oracle ACFS) and Oracle ASM Dynamic Volume Manager (Oracle ADVM) compatibility level of a disk group. Compatibility attributes can be set when the disk group is created and advanced when a disk group is altered, but compatibility attributes cannot be reversed. For information about the `COMPATIBLE.ADVM` attribute, refer to [Disk Group Compatibility Attributes](#).

> ⓘ **See Also**
>
> *Oracle Advanced Cluster File System Administrator's Guide* for more information about Oracle ACFS and Oracle ADVM

CONTENT.CHECK

The `CONTENT.CHECK` attribute enables or disables content checking when performing data relocation operations for a disk group.

The attribute value can be set to `true` (enabled) or `false` (disabled).

When `CONTENT.CHECK` is enabled, an Oracle ASM relocation process (rebalance, resync, or resilver) detects corruptions during a data copy operation and performs automatic block corruption recovery by replacing these corruptions with an uncorrupted mirror copy if one is available.

The content check process detects and repairs corruptions for situations when the I/O operation is successful, but the blocks have invalid content. The process also performs a

Hardware Assisted Resilient Data (HARD) check for all supported files and a block header check for data files.

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

> ⓘ **See Also**
>
> Manually Rebalancing Disk Groups and Tuning Rebalance Operations for information about disk group rebalancing

CONTENT.TYPE

This attribute identifies the disk group type: `data`, `recovery`, or `system`. The type value determines where Oracle ASM places the nearest neighbor disk in the failure group where Oracle ASM mirrors copies of the data. Oracle ASM uses this attribute value to make it less likely that a double failure in the storage medium causes disk groups of different content types to become unavailable. Specifying this attribute enables Oracle ASM to better deploy all disk groups in the available storage.

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.

This attribute can be set with ASMCMD `mkdg` or SQL `CREATE` and `ALTER DISKGROUP` statements.

For example:

```
SQL> CREATE DISKGROUP data NORMAL REDUNDANCY
       FAILGROUP controller1 DISK
       '/devices/diska1' NAME diska1,
...
       ATTRIBUTE
       'compatible.asm' = '12.1.0.1',
       'content.type' = 'recovery',
...

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

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

For information about disk resync, see Oracle ASM Fast Mirror Resync.

> ⓘ **See Also**
>
> - *Oracle Exadata documentation*
> - [My Oracle Support](https://support.oracle.com) (`https://support.oracle.com`) for information about using the `CONTENT.TYPE` attribute with Oracle Exadata storage

CONTENT_HARDCHECK.ENABLED

The `CONTENT_HARDCHECK.ENABLED` disk group attribute enables or disables Hardware Assisted Resilient Data (HARD) checking when performing data copy operations for rebalancing a disk group.

The attribute value can be set to `true` or `false`. This attribute can only be set when altering a disk group.

- When the `CONTENT.CHECK` disk group attribute is set to enabled (`true`), the setting of `CONTENT_HARDCHECK.ENABLED` is ignored and checking is done on the content of user data, including HARD checks.

- When the `CONTENT.CHECK` attribute is set to disabled (`false`) and the `CONTENT_HARDCHECK.ENABLED` attribute is set to enabled (`true`), only HARD checking is performed.

- When the `CONTENT.CHECK` attribute is set to disabled (`false`) and the `CONTENT_HARDCHECK.ENABLED` attribute is set to disabled (`false`), no checking is performed.

> ⓘ **See Also**
>
> [Manually Rebalancing Disk Groups](#) and [Tuning Rebalance Operations](#) for information about disk group rebalancing

DISK_REPAIR_TIME

This attribute specifies the time interval to repair a disk and bring it back online before initiating the drop operation. This attribute can only be set when altering a disk group and is only applicable to normal and high redundancy disk groups.

For information about disk repair time, see [Oracle ASM Fast Mirror Resync](#).

FAILGROUP_REPAIR_TIME

This attribute specifies a default repair time for the failure groups in the disk group. The failure group repair time is used if Oracle ASM determines that an entire failure group has failed. The default value is 24 hours (`24h`). If there is a repair time specified for a disk, such as with the `DROP AFTER` clause of the SQL `ALTER DISKGROUP OFFLINE DISK` statement, that disk repair time overrides the failure group repair time.

This attribute can only be set when altering a disk group and is only applicable to normal and high redundancy disk groups.

For information about failure group repair time, see [Oracle ASM Fast Mirror Resync](#).

IDP.BOUNDARY and IDP.TYPE

> ⓘ **Note**
>
> The Intelligent Data Placement (IDP) feature has been deprecated since Oracle ASM 12*c* Release 2 (12.2) and is planned for desupport in a future release.

These attributes are used to configure Oracle Exadata storage.

> ⓘ **See Also**
>
> *Oracle Exadata documentation*

LOGICAL_SECTOR_SIZE

The `LOGICAL_SECTOR_SIZE` disk group attribute specifies the logical sector size for a disk group. The value can be set when creating or altering a disk group.

The value of `LOGICAL_SECTOR_SIZE` specifies the smallest unit of I/O allowed on the disk group. The value must be less than or equal to the size of the `SECTOR_SIZE` disk group attribute.

`COMPATIBLE.ASM` must be set to `12.2` or greater to use the `LOGICAL_SECTOR_SIZE` disk group attribute.

For information about setting the `LOGICAL_SECTOR_SIZE` disk group attribute, see Specifying the Logical and Physical Sector Size.

For information about the `SECTOR_SIZE` disk group attribute, refer to SECTOR_SIZE.

PHYS_META_REPLICATED

This disk group attribute tracks the replication status of a disk group. When the Oracle ASM compatibility of a disk group is advanced to 12.1 or higher, the physical metadata of each disk is replicated. This metadata includes the disk header, free space table blocks, and allocation table blocks. The replication is performed online asynchronously. This attribute value is set to `true` by Oracle ASM if the physical metadata of every disk in the disk group has been replicated.

This disk group attribute is only defined in a disk group with the Oracle ASM disk group compatibility (`COMPATIBLE.ASM`) set to `12.1` and higher. This attribute is read-only and is intended for information only. A user cannot set or change its value. The values are either `true` or `false`.

PREFERRED_READ.ENABLED

In an Oracle extended cluster, which contains nodes that span multiple physically separated sites, the `PREFERRED_READ.ENABLED` disk group attribute controls whether preferred read functionality is enabled for a disk group.

If preferred read functionality is enabled, then this functionality enables an instance to determine and read from disks at the same site as itself, which can improve performance.

Each Oracle ASM instance in an Oracle extended cluster has the information to determine the site in which it is located. If preferred read functionality is enabled, then based on the site information, the preferred status for a disk is determined to be true if the disk is in the same site as the instance. Otherwise, if the disk is not located at the same site as the Oracle ASM instance, the preferred status for a disk is set to false.

For extended clusters, the preferred read setting is enabled by default with a `TRUE` setting for the `PREFERRED_READ.ENABLED` disk group attribute of each disk group. For clusters that are not extended (only one physical site), preferred read is disabled. Preferred read status applies to extended, normal, high, and flex redundancy disk groups.

Oracle Database 12*c* Release 2 (12.2) is required and `COMPATIBLE.ASM` must be set to `12.2` or greater to use `PREFERRED_READ.ENABLED`.

Whether or not `PREFERRED_READ.ENABLED` has been enabled, preferred read can be set at the failure group level on an Oracle ASM instance or a client instance in a cluster with the `ASM_PREFERRED_READ_FAILURE_GROUPS` initialization parameter, which is available for backward compatibility.

To determine whether a disk has preferred read status, you can check the `PREFERRED_READ` column of the `V$ASM_DISK` view.

> ⓘ **See Also**
>
> - [ASM_PREFERRED_READ_FAILURE_GROUPS](#) for information about the `ASM_PREFERRED_READ_FAILURE_GROUPS` initialization parameter
> - [About Oracle ASM Extended Disk Groups](#) for information about Oracle ASM extended disk groups
> - *Oracle Grid Infrastructure Installation and Upgrade Guide* for information about Oracle extended clusters
> - *Oracle AI Database Reference* for information about all of the `V$ASM*` dynamic performance views

SCRUB_ASYNC_LIMIT

The `SCRUB_ASYNC_LIMIT` disk group attribute sets the maximum allowed outstanding parallel read requests for asynchronous scrubbing.

The range of values is `0` to `1024`. The default value is `1`.

`COMPATIBLE.ASM` must be set to `12.2` or greater to use this disk group attribute.

> ⓘ **Note**
>
> ASM allows users to pause the currently running scrub operation on the diskgroup or disable any future scrub operation by setting the diskgroup attribute value for `SCRUB_ASYNC_LIMIT` to '0'.

For information about the scrubbing operation, refer to [Scrubbing Disk Groups](#).

SECTOR_SIZE

The `SECTOR_SIZE` disk group attribute specifies the sector size for a disk group. The value can be set when creating or altering a disk group.

The value of `SECTOR_SIZE` is used for the block size of newly-created redo logs and Oracle ACFS file systems. The value can be changed to facilitate migration of disks to different sector sizes. The value must be greater than or equal to the size of the `LOGICAL_SECTOR_SIZE` disk group attribute.

`COMPATIBLE.ASM` must be set to `12.2` or greater to use the 12.2 functionality of `SECTOR_SIZE`, which includes the ability to alter the sector size after a disk group has been created.

For information about setting the `SECTOR_SIZE` disk group attribute, see [Specifying the Logical and Physical Sector Size](#).

For information about the `LOGICAL_SECTOR_SIZE` disk group attribute, refer to [LOGICAL_SECTOR_SIZE](#).

STORAGE.TYPE

This 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**
>
> 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 maximum support with ZFS storage, set the `COMPATIBLE.ASM` and `COMPATIBLE.RDBMS` disk group attributes to `11.2.0.4` or higher. For information about disk group compatibility attributes, refer to [Disk Group Compatibility](#).

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 AI Database Concepts* for more information about Hybrid Columnar Compression. Hybrid Columnar Compression is a feature of certain Oracle storage systems.

VAM_MIGRATION_DONE

Virtual Allocation Metadata (VAM) supports flexible storage allocation. The `VAM_MIGRATION_DONE` disk group attribute must be enabled and the migration completed before migrating a `NORMAL` or `HIGH` redundancy disk group to a `FLEX` redundancy disk group without the use of a restrictive mount.

Creating Disk Groups

This topic introduces information about creating disk groups.

You can use the `CREATE DISKGROUP` SQL statement to create a disk group.

This section contains the following topics:

- [About Creating a Disk Group](#)
- [Creating Disk Groups Using the CREATE DISKGROUP SQL Statement](#)
- [Creating Disk Groups for a New Oracle Installation](#)
- [Specifying the Logical and Physical Sector Size](#)
- [Storing Oracle Cluster Registry and Voting Files in Oracle ASM Disk Groups](#)

> ⓘ **See Also**
>
> - *Oracle Grid Infrastructure Installation and Upgrade Guide* for information about storage requirements for Oracle Automatic Storage Management
> - *Oracle AI Database SQL Language Reference* for information about the `CREATE DISKGROUP` SQL statement

About Creating a Disk Group

There are multiple items you need to specify when you create a disk group.

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**
  >
  > Do not use identifiers for Oracle ASM disk group names that must be quoted, including `#` and `$` characters. While the use of quoted identifiers for Oracle AI Database objects may be valid as names in some tools, the names may not be valid when using other tools that manage that object. For more information about naming Oracle AI Database objects, refer to *Oracle AI Database SQL Language Reference*.

- Determine the redundancy level of the disk group.

For Oracle ASM to mirror files, specify a redundancy level such as `NORMAL REDUNDANCY` (two-way mirroring by default for most file types), `HIGH REDUNDANCY` (three-way mirroring for all files), `FLEX REDUNDANCY`, or `EXTENDED REDUNDANCY`.

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 with `NORMAL` or `HIGH` redundancy, you can migrate the disk group to a `FLEX` redundancy disk group with the `CONVERT REDUNDANCY TO FLEX` clause of the `ALTER DISKGROUP` SQL statement. You cannot migrate an `EXTERNAL` disk group to a `FLEX` redundancy disk group. When migrating to a flex disk group, the disk group must have a minimum of three failure groups and must be mounted in restricted mode (`MOUNTED RESTRICTED`) for the duration of the migration.

To change the redundancy level for `NORMAL`, `HIGH`, or `EXTERNAL` disk groups to a `NORMAL`, `HIGH`, or `EXTERNAL`, you must create another disk group with the appropriate redundancy and then move the files to the new disk group.

For `FLEX` and `EXTENDED` redundancy disk groups, the disk group redundancy cannot be changed. However, the redundancy of the files in the disk group can be changed by modifying the `REDUNDANCY` property of the respective file group.

- Identify 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 indicate that the disks belong to specific failure groups.

  You must create failure groups of equal size to maintain space balance and even distribution of mirror data.

- Optionally determine the type of failure group, such as quorum (`QUORUM`) or regular (default).

- Optionally provide disk group attribute values, such as the values for software compatibility or allocation unit size.

All disks in the disk group must be of equal size. Oracle ASM programmatically determines the size of each disk. If for some reason the size cannot be determined, or to restrict the amount of space used on a disk, you can specify a `SIZE` clause for a disk; however, the same size value must be specified for every disk in the disk group.

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.

You can use the ASMCMD `stamp` command to provision disks for Oracle ASM before creating a disk group that uses them. If a disk is provisioned, then the disk name, failure group, and site information are specified at the time the disk is provisioned and do not need to be specified when the disk group is created.

> ⓘ **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. 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 ASM Configuration Assistant (ASMCA) and the ASMCMD `mkdg` command.

> ⓘ **See Also**
>
> - [Managing Oracle ASM Flex Disk Groups](#) for information about flex disk groups and file groups
>
> - [Mirroring, Redundancy, and Failure Group Options](#) and [Understanding Oracle ASM Concepts](#) for information about redundancy levels and failure groups
>
> - [Storing Oracle Cluster Registry and Voting Files in Oracle ASM Disk Groups](#) for information about `QUORUM` and `REGULAR` failure groups
>
> - [stamp](#) for information about the ASMCMD `stamp` command
>
> - [Managing Disk Groups with ASMCA](#) for information about creating disk groups with Oracle ASM Configuration Assistant (ASMCA)
>
> - [mkdg](#) for information about creating disk groups with the ASMCMD `mkdg` command
>
> - *Oracle AI Database SQL Language Reference* for information about the `CREATE DISKGROUP` SQL statement

Creating Disk Groups Using the CREATE DISKGROUP SQL Statement

You can create disk groups for data and the fast recovery area using the `CREATE DISKGROUP` SQL statement.

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.

The example assumes that the `ASM_DISKSTRING` initialization parameter is set to the '/devices/*' string. Oracle ASM disk discovery identifies `diska1` through `diska4` and `diskb1` through `diskb4` in the `/devices` directory.

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.

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.

    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.

- `COMPATIBLE.RDBMS`

    Determines the minimum software version for any database instance that uses a disk group.

- `COMPATIBLE.ADVM`

    Determines whether the disk group can contain Oracle ADVM volumes.

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 4-1   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' = '19.0',
    'compatible.rdbms' = '19.0',
    'compatible.advm' = '19.0';
```

Example 4-2   Creating the FRA disk group

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

> **ⓘ Note**
>
> Starting with Oracle AI Database 26ai, Release Update 23.4, ASM enforces higher `compatible.asm` and `compatible.rdbms` values. For newly created diskgroups in 26ai:
>
> - `compatible.asm` value must be 19.0 or above specified in `CREATE DISKGROUP` command, if not specified it will be set to 23.0.
>
> - `compatible.rdbms` value must be 19.0 or above specified in `CREATE DISKGROUP` command, if not specified it will be set to 19.0.
>
> When upgrading Grid Infrastructure to 26ai:
>
> - `compatible.asm` value must be advanced to be 19.0 or above before upgrading to 26ai.
>
> - `compatible.rdbms` value must be advanced to be 19.0 or above before upgrading to 26ai.

> **ⓘ See Also**
>
> - [About Oracle ASM Allocation Units](#) for information about allocation units
> - [Disk Group Compatibility Attributes](#) for information about `COMPATIBLE.ASM`, `COMPATIBLE.RDBMS`, and `COMPATIBLE.ADVM`

Creating Disk Groups for a New Oracle Installation

This topic describes the basic procedure for creating disk groups during a new installation of Oracle Grid Infrastructure standalone (Oracle Restart) option and Oracle AI Database.

The procedure described in this topic for a new installation of Oracle Grid Infrastructure standalone (Oracle Restart) assumes that a disk group is configured for storing the database data files and a disk group is configured for storing the fast recovery area files.

> **ⓘ Note**
>
> In Oracle Grid Infrastructure for cluster configurations, the Oracle Cluster Registry (OCR) and voting files should be stored on their own disk group. However, if only two disk groups are created for the Oracle ASM installation, then store the OCR and voting files in the disk group that contains the database data files.

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

   During the installation, create a disk group (`data`) for storing database data files and control files.

   This OUI disk group creation process is similar to creating a disk group with Oracle ASM Configuration Assistant (ASMCA). 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.

2. After the Oracle Grid Infrastructure is installed, use ASMCA to create a disk group (`fra`) for storing the fast recovery area files.

    You can also create the `fra` disk group with SQL*Plus or ASMCMD commands run from the Oracle Grid Infrastructure home.

3. Install the Oracle AI 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**

- [Managing Disk Groups with ASMCA](#) for information about using ASMCA to create or alter disk groups

- [Storing Oracle Cluster Registry and Voting Files in Oracle ASM Disk Groups](#) for information about storing Oracle Cluster Registry (OCR) and voting files

- [About Creating a Disk Group](#) for information about creating a disk group with SQL*Plus

- [mkdg](#) for information about creating a disk group with ASMCMD

- *Oracle AI Database Backup and Recovery User's Guide* for information about configuring the fast recovery area

- *Oracle AI Database Administrator's Guide* for information about specifying a fast recovery area

- *Oracle Grid Infrastructure Installation and Upgrade Guide* for your operating system for detailed information about installation of Oracle Grid Infrastructure and Oracle Restart

- *Oracle AI Database Installation Guide* for your operating system for detailed information about installation of an Oracle AI Database

---

Specifying the Logical and Physical Sector Size

You can use the optional `LOGICAL_SECTOR_SIZE` and `SECTOR_SIZE` disk group attributes with the `CREATE DISKGROUP` and `ALTER DISKGROUP` SQL statements to specify a logical sector size and physical sector size for a disk group.

ⓘ **Note**

To specify the logical and physical sector size, the disks must support sector size emulation.

The `SECTOR_SIZE` disk group attribute 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 enable the setting of a value for the sector size during disk group creation. The `COMPATIBLE.ASM` disk group attribute must be set to `12.2` or higher to alter the sector size.

The `LOGICAL_SECTOR_SIZE` disk group attribute can explicitly set the logical sector size of the disk group instead of the size that is estimated from the disks that join the disk group. The `LOGICAL_SECTOR_SIZE` disk group attribute can be set to `512`, `4096`, or `4K`. The value specifies the smallest possible I/O that the disk group can accept, and also limits what types of disks can join the disk group. The `COMPATIBLE.ASM` disk group attribute must be set to `12.2` or higher to set the logical sector size during creation of a disk group or alter the size of a disk group that has been created.

Table 4-1 lists the possible valid combinations of the logical sector size and sector size disk group attributes during disk group creation for disk sizes and types. The first column describes the user-specified values for logical sector size and sector size attributes during disk group creation. `NS` indicates that the no value was specified.

The top row of the table describes the combination of the disks used. A 512 native sector size disk has sector size equal to logical sector size equal to `512`. The 512 native and 512 emulated heading describes the combination of 512 native disks and 512 emulated disk. The 512 emulated disk has a 4 k physical sector size and a 512 logical sector size. The values in each of the table cells represents the final value of sector sizes of the disk group. `Error` indicates that the disk group creation fails and an error message is displayed.

Table 4-1   Possible combinations of sector size and logical sector size by disk size and type

SECTOR_SIZE/ LOGICAL_SECTOR_SIZE specified at disk group creation	512 native disks	512 emulated disks	4K native disks	512 native disks + 512 emulated disks	512 emulated disks + 4K native disks
4K / 4K	Error	4K / 4K	4K / 4K	Error	4K / 4K
NS / 4K	Error	4K / 4K	4K / 4K	Error	4K / 4K
4K / 512	Error	4K / 512	Error	Error	Error
512 / 512	512 / 512	512 / 512	Error	512 / 512	Error
512 / NS	512 / 512	512 / 512	Error	512 / 512	Error
4K / NS	Error	4K / 512	4K / 4K	Error	4K / 4K
NS / 512	512 / 512	512 / 512	Error	512 / 512	Error
NS / NS	512 / 512	512 / 512	4K / 4K	512 / 512	4K / 4K

You can determine the logical sector and sector size values that have either been assumed or explicitly set for a successful disk group creation by querying `V$ASM` views or running ASMCMD commands. The `LOGICAL_SECTOR_SIZE` and `SECTOR_SIZE` columns are included in the `V$ASM_DISKGROUP` and `V$ASM_ATTRIBUTE` views to display the values. These columns represent the logical sector size and sector size values of the disk group in bytes. The ASMCMD `lsdg` also displays the values of the logical sector size and sector size for the disk group. In addition, the disk group attributes are displayed with the `V$ASM_ATTRIBUTE` view and the ASMCMD `lsattr` command. For example:

```
SQL> SELECT name, value FROM V$ASM_ATTRIBUTE
     WHERE (name = 'sector_size' OR name = 'logical_sector_size') AND group_number = 1;
NAME                            VALUE
------------------------------- ---------------------------
sector_size                     4096
logical_sector_size             512
```

```
SQL> SELECT logical_sector_size, sector_size FROM V$ASM_DISKGROUP
     WHERE group_number = 1;
LOGICAL_SECTOR_SIZE SECTOR_SIZE
------------------- -----------
                512        4096
```

As shown in [Example 4-3](#), you can use the `LOGICAL_SECTOR_SIZE` and `SECTOR_SIZE` disk group attributes with the `CREATE DISKGROUP` and `ALTER DISKGROUP` SQL statements to specify the logical and physical sector size of the Oracle ASM disk group.

The following notes apply to the logical sector size and sector size disk group attributes.

- During the migration process from 512 bytes to 4096 bytes, password files stored on an Oracle ASM disk group do not need to be migrated or recreated. The password files are a special case and retain their 512 byte block size, even after the new sector sizes are 4096 bytes.

- If the value of the `LOGICAL_SECTOR_SIZE` column is 512, while the value in the `SECTOR_SIZE` column is 4096, then the disk group supports 512 sector size emulation. This means that I/O operations can be both 512 or 4096 in size. For example, we can have redo log files with a file block size of 512 bytes in the disk group that can be both read and written to, as well as creating new redo log files with a file block size of 4096 bytes.

- The physical sector size (`SECTOR_SIZE`) cannot be smaller than the logical sector size (`LOGICAL_SECTOR_SIZE`) in an Oracle ASM disk group.

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

Example 4-3   Specifying logical and sector sizes for disks in a disk group

```
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'
  QUORUM FAILGROUP quorum_failgrp DISK '/devices/diskc1'
  ATTRIBUTE 'compatible.asm' = '12.2', 'compatible.rdbms' = '12.2',
            'sector_size'='4096', 'logical_sector_size'='512';

ALTER DISKGROUP data2
  SET ATTRIBUTE 'compatible.asm' = '12.2', 'compatible.rdbms' = '12.2';

ALTER DISKGROUP data2
  SET ATTRIBUTE 'sector_size'='4096', 'logical_sector_size'='4096';
```

> ⓘ **See Also**
>
> - *Oracle AI Database SQL Language Reference* for information about the disk group attributes and the `CREATE DISKGROUP` SQL statement
>
> - *Oracle AI Database SQL Language Reference* for information about the `BLOCKSIZE` keyword
>
> - *Oracle AI Database Reference* for details about the `V$ASM_ATTRIBUTE` , `V$ASM_DISK`, and `V$ASM_DISKGROUP` views
>
> - [ASMCMD Disk Group Management Commands](#) for information about ASMCMD disk group management commands

Storing 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.

You can also store a backup of the OCR file in a disk group. Voting files and OCR are two important components of Oracle Clusterware. Voting files manage information about node membership. OCR manages cluster and Oracle Real Application Clusters (Oracle RAC) database configuration information.

A quorum failure group is a special type of failure group that does not contain user data. Quorum failure groups are used for storing Oracle ASM metadata. A quorum failure group may also contain voting files if those files are stored in a disk group that contains a quorum failure group. Additionally, Oracle ASM uses a quorum failure group to help determine if the disk group can be mounted in the event of the loss of one or more failure groups.

Because disks in quorum failure groups do not contain user data, a quorum failure group is not considered when determining redundancy requirements in respect to storing user data.

You can manage and monitor OCR and voting files with CRSCTL and `ocrconfig` command-line tools, `CREATE` and `ALTER DISKGROUP` SQL statements, and `V$ASM` views.

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.

The `QUORUM` and `REGULAR` keywords in `CREATE` and `ALTER DISKGROUP` SQL statements enable the specification of an additional qualifier for failure groups or disks when creating or altering a disk group. Note that `QUORUM` disks (or disks in `QUORUM` failure groups) cannot contain user data, while `REGULAR` disks (or disks in non-quorum failure groups) do not have this 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.

The `FAILGROUP_TYPE` column in the `V$ASM_DISK` or `V$ASM_DISK_STAT` views specifies the 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 the `V$ASM_DISK` or `V$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. The value of `USABLE_FILE_MB` in `V$ASM_DISKGROUP` or `V$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.

Example 4-4   Using the QUORUM keyword when creating a failure group

```
CREATE DISKGROUP mgmt_dg NORMAL REDUNDANCY
   FAILGROUP fg1 DISK '/devices/diskg1'
   FAILGROUP fg2 DISK '/devices/diskg2'
   QUORUM FAILGROUP fg3 DISK '/devices/diskg3'
   ATTRIBUTE 'compatible.asm' = '12.2.0.0.0';
```

> ### ⓘ See Also
>
> - [Oracle ASM Failure Groups](#) for information about failure groups
> - [Views Containing Oracle ASM Information](#) for information about `V$ASM` views
> - *Oracle Grid Infrastructure Installation and Upgrade Guide* for information about configuring voting files and Oracle Cluster Registry (OCR)
> - *Oracle Clusterware Administration and Deployment Guide* for information about managing Oracle Cluster Registry and voting files
> - *Oracle AI Database SQL Language Reference* for information about the `CREATE DISKGROUP` and `ALTER DISKGROUP` SQL statements
> - *Oracle AI Database Reference* for details about the `V$ASM`* dynamic performance views

Altering Disk Groups

You can alter a disk group with SQL*Plus, ASMCA, or ASMCMD commands.

You can use the `ALTER DISKGROUP` SQL statement to alter a disk group configuration. You can also alter a disk group configuration with Oracle ASM Configuration Assistant (ASMCA) and the ASMCMD `chdg` command.

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.

The following topics are discussed:

- [Adding Disks to a Disk Group](#)
- [Replacing Disks in Disk Groups](#)
- [Renaming Disks in Disk Groups](#)
- [Dropping Disks from Disk Groups](#)
- [Resizing Disks in Disk Groups](#)
- [Undropping Disks in Disk Groups](#)
- [Manually Rebalancing Disk Groups](#)
- [Tuning Rebalance Operations](#)
- [Scrubbing Disk Groups](#)
- [Managing Oracle ADVM Volumes in a Disk Group](#)

> ⓘ **See Also**
>
> - [Managing Disk Groups with ASMCA](#) for information about altering a disk group configuration with Oracle ASM Configuration Assistant (ASMCA)
> - [chdg](#) for information about altering a disk group configuration with ASMCMD `chdg`
> - [Manually Rebalancing Disk Groups](#) for more information about the `REBALANCE POWER` clause
> - *Oracle AI Database SQL Language Reference* for information about the `ALTER DISKGROUP` SQL statement
> - *Oracle Advanced Cluster File System Administrator's Guide* for more information about Oracle ACFS and Oracle ADVM

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 add a disk or failure group with the `ALTER DISKGROUP` SQL statement. The syntax used is similar to the `CREATE DISKGROUP` statement. After you add new disks, the new disks gradually begin to accommodate their share of the workload as rebalancing progresses.

When adding a disk, the disk must be the same size as the other disks in the disk group.

> ⓘ **See Also**
>
> - [Using the ALTER DISKGROUP SQL Statement to Add Disks to a Disk Group](#) for an example using SQL statements to add disks to a disk group
>
> - [Example 4-1](#) for an example of the `CREATE DISKGROUP` SQL statement
>
> - [Mirroring, Redundancy, and Failure Group Options](#) and [Understanding Oracle ASM Concepts](#) for information about redundancy levels and failure groups
>
> - [Managing Disk Groups with ASMCA](#) for information about adding disks to a disk group with ASMCA
>
> - *Oracle AI Database SQL Language Reference* for information about the `ALTER DISKGROUP` SQL statement

Using the ALTER DISKGROUP SQL Statement to Add Disks to a Disk Group

The SQL examples in this topic demonstrate the interactions of disk discovery with the `ADD DISK` operation. Assume that disk discovery identifies the following disks in `/devices` directory:

```
/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 Information](#).

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

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

The following statement successfully adds disks `/devices/diska5` and `/devices/diska6` 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;
```

When specifying a failure group, you can specify the type (regular or quorum) as well as the name. The failure group type value defaults to regular. You must include `QUORUM FAILGROUP` clause to specify a quorum failure group. The following statement successfully adds disks `/devices/diska7` and `/devices/diska8` to `data1` using the `FAILGROUP` clause to specify a failure group.

```
ALTER DISKGROUP data1 ADD FAILGROUP fg_diska78 DISK
     '/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;
```

Replacing Disks in Disk Groups

A disk or multiple disks in a disk group can be replaced, rather than dropped and added back.

The single replace operation is more efficient than dropping and adding disks. This operation is especially useful when disks are missing or damaged.

For example, you can issue the following statement to replace the `diskc7` disk with another disk identified by the `/devices/diskc18` path.

```
SQL> ALTER DISKGROUP data2 REPLACE DISK diskc7 WITH '/devices/diskc18' POWER 3;
```

The power option operates the same as the power option for the `ALTER DISKGROUP REBALANCE` statement, except that the power option cannot be set to `0`.

The `ALTER DISKGROUP` SQL statement with the `REPLACE` clause includes a `WAIT` or `NOWAIT` option, plus the `FORCE` option.

> ⓘ **Note**
>
> - [Manually Rebalancing Disk Groups](#) for information about the power option in rebalancing a disk group
> - [ASM_POWER_LIMIT](#) for information about the `ASM_POWER_LIMIT` initialization parameter

Renaming Disks in Disk Groups

> ⓘ **Note**
>
> Oracle ASM Filter Driver (ASMFD) is desupported on both Linux and Oracle Solaris beginning with Oracle AI Database 26ai.

You can rename a disk in a disk group with the `ALTER DISKGROUP RENAME DISK` SQL statement.

> ⓘ **Note**
>
> You can only use the `RENAME DISK` operation when the disk group that contains the disk is in the `MOUNT RESTRICTED` state. If any disks in the disk group are offline, then the `RENAME` operation fails. If the new disk name exists, then the `RENAME` operation fails. You must have `SYSASM` privileges to rename a disk.

In one SQL statement, you can rename one or multiple disks, or rename all disks in a disk group using the `RENAME DISKS ALL` clause.

For example, you can rename disks as follows:

```
SQL> ALTER DISKGROUP fra2 MOUNT RESTRICTED;

SQL> ALTER DISKGROUP fra2 RENAME DISK 'FRA1_0001' TO 'FRA2_0001',
        'FRA1_0002' TO 'FRA2_0002';
```

For `NORMAL`, `HIGH`, and `FLEX` redundancy disk groups, you can associate a site name using `ALTER DISKGROUP RENAME DISK` with the `SITE` clause. For example:

```
SQL> ALTER DISKGROUP data1 RENAME DISK 'DATA1_0001' SITE SITE1;
```

For information about data sites, extended disk groups, and preferred read for disks, refer to [About Oracle ASM Extended Disk Groups](#) and [PREFERRED_READ.ENABLED](#). For information about the ASMCMD `stamp` command, refer to [stamp](#).

For an `EXTERNAL` redundancy disk group, you can use `ALTER DISKGROUP RENAME DISK` with the `FAILGROUP` clause to specify a failure group for a disk. For example:

```
SQL> ALTER DISKGROUP external1 RENAME DISK 'EXTERNAL1_0001' FAILGROUP FG_EXT1_0001;
```

The `SITE` and `FAILGROUP` clauses cannot be used together in the same SQL `ALTER DISKGROUP RENAME` statement.

The `ALTER DISKGROUP` *diskgroupname* `RENAME DISKS ALL` statement can be run after the `renamedg` utility to change the names of the disks in the renamed disk group. For information about `renamedg`, refer to [Renaming Disks Groups](#).

When you run the `ALTER DISKGROUP` *diskgroupname* `RENAME DISKS ALL` statement, any disk name that is not in the format *diskgroupname_number* is renamed to that format. Disk names that are already in the *diskgroupname_number* format are not changed.

> ⓘ **Note**
>
> If a disk is labeled by any subsystem; such as ASMLIB, EXADATA, or ASMTOOL; then that label becomes the disk name by design. The `ALTER DISKGROUP RENAME DISKS` SQL statement on these subsystems has no effect and the disk is not renamed.

Dropping Disks from Disk Groups

You can drop disks from a disk group with 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.

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.

The statements in [Example 4-5](#) demonstrate how to drop disks from the disk group `data1`. 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 4-5   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;
```

> ⓘ **See Also**
>
> - [Managing Disk Groups with ASMCA](#) for information about dropping disks from a disk group with ASMCA
> - [Using the ALTER DISKGROUP SQL Statement to Add Disks to a Disk Group](#) for an example of the `ALTER DISKGROUP` statement
> - *Oracle AI Database Reference* for information about the `V$ASM_OPERATION` view
> - *Oracle AI Database SQL Language Reference* for information about the `ALTER DISKGROUP` SQL statement

Resizing Disks in Disk Groups

The `RESIZE` clause of `ALTER DISKGROUP` enables you to resize disks in a disk group.

When resizing disks in a disk group, all the disks must be of equal size.

If you do not specify a new size in the `SIZE` clause, then Oracle ASM uses the size of the disks as returned by the operating system. The new size is written to the Oracle ASM disk header and if the size 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 the disk group `data1`. If the new size is greater than the capacity of a disk, then the statement fails.

Example 4-6   Resizing disks in disk groups

```
ALTER DISKGROUP data1 RESIZE ALL 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 4-7   Undropping disks in disk groups

```
ALTER DISKGROUP data1 UNDROP DISKS;
```

> ⓘ **Note**
>
> - *Oracle AI Database SQL Language Reference* for information about the `DROP DISKGROUP` SQL statement
>
> - *Oracle AI Database SQL Language Reference* for information about the `ALTER DISKGROUP` SQL statement

Manually Rebalancing Disk Groups

You can manually rebalance the files in a disk group using the `REBALANCE` clause of the `ALTER DISKGROUP` statement.

Manual rebalancing would normally not be required because Oracle ASM automatically rebalances disk groups when their configuration changes, including changes to file groups. However, 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 the speed of the rebalance operation. A higher value increases the speed of the rebalance operation. It can be set to a minimum value of `0` which stops a rebalancing operation until the statement is either implicitly or explicitly rerun.

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.

The power level of an ongoing rebalance operation can be changed by entering the rebalance statement using the `MODIFY POWER` clause. For example, the following SQL statement modifies the power setting to `10` of an ongoing rebalance.

```
ALTER DISKGROUP data2 REBALANCE MODIFY POWER 10;
```

The the following SQL statement modifies the power setting to the default value.

```
ALTER DISKGROUP data2 REBALANCE MODIFY POWER;
```

When you rebalance a disk group, you have the option choosing the following phase options using the `WITH` or `WITHOUT` keywords, except where noted:

- `RESTORE`

  The `RESTORE` phase includes `RESYNC`, `RESILVER`, and `REBUILD` operations. The `RESTORE` phase is always run by default and cannot be excluded.

  - `RESYNC`

    This operation synchronizes the stale extents on the disks that are being brought online.

  - `RESILVER`

    This operation is specific only to Exadata systems. During this phase, data is copied from one mirror to the mirror with stale data.

  - `REBUILD`

    This operation restores the redundancy of forcing disks only. Forcing disks are those disks that have been dropped with the force option.

- `BALANCE`

This phase restores redundancy of all the disks in the disk group, including file groups, and also balances extents on Oracle ASM disks.

- `PREPARE`

  This phase completes the work corresponding to the prepare SQL operation. This phase is enabled only for `FLEX` or `EXTENDED` redundancy disk groups. `COMPATIBLE.ASM` must be set to `12.2` or higher.

- `COMPACT`

  This phase defragments and compacts extents across Oracle ASM disks.

The rebalance operation executes the phases in the optimal order. If none of the phases are specified with the `ALTER DISKGROUP...REBALANCE` SQL statement, then the behavior remains the same as the rebalance operation previous to the Oracle ASM 12.2 release during which all rebalance phases are run.

The following examples show the use of the phase options with `ALTER DISKGROUP...REBALANCE`:

```
ALTER DISKGROUP data2 REBALANCE WITH BALANCE COMPACT;
```

```
ALTER DISKGROUP data3 REBALANCE WITHOUT BALANCE;
```

The `ALTER DISKGROUP...REBALANCE` command returns immediately by default so that you can run other commands while the rebalance operation takes place asynchronously in the background. You can query the `V$ASM_OPERATION` view for the status of the rebalance operation. When not all rebalance phases are run, the `V$ASM_OPERATION` view displays an `ORA-59000` error at the end of the rebalance.

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 continues 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. All supported phases of the rebalance operation are executed. Note that specifying `RESTORE` is not necessary as that phase is always run during a rebalance.

Example 4-8   Manually rebalancing a disk group

```
ALTER DISKGROUP data2 REBALANCE RESTORE POWER 5 WAIT;
```

> **ⓘ See Also**
>
> - [ASM_POWER_LIMIT](#) for information about the `ASM_POWER_LIMIT` initialization parameter
> - [Tuning Rebalance Operations](#) for information about rebalancing operations
> - *Oracle AI Database SQL Language Reference* for information about the `ALTER DISKGROUP` SQL statement

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 range of values for the `POWER` clause is the same for the `ASM_POWER_LIMIT` initialization parameter.

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. Oracle ASM always adjusts the power to fit available memory.

When the `COMPATIBLE.ASM` disk group is set to `11.2.0.2` or higher, the rebalance operation may be run as one process using asynchronous I/O. You can check the Oracle ASM alert log for details on the rebalance process.

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.

You can also affect rebalance behavior with the `CONTENT.CHECK` and `THIN_PROVISIONED` disk group attributes.

The `EXPLAIN WORK` SQL statement determines the amount of work for a rebalance operation and the resulting calculations are displayed in the `V$ASM_ESTIMATE` view.

For example:

```
EXPLAIN WORK FOR ALTER DISKGROUP data DROP DISK data_0000;
Explained.

SELECT est_work FROM V$ASM_ESTIMATE;
EST_WORK
--------
    4211

EXPLAIN WORK SET STATEMENT_ID='online' FOR ALTER DISKGROUP data
  ONLINE disk data_000;
Explained.

SELECT est_work FROM V$ASM_ESTIMATE WHERE STATEMENT_ID='online';
EST_WORK
--------
     421
```

The `V$ASM_ESTIMATE` view provides information for adjusting `ASM_POWER_LIMIT` and the resulting power of rebalance operations. The `EST_WORK` column provides an estimate of the number of allocation units that have to be moved by the rebalance operation to complete.

The `PASS` column of `V$ASM_OPERATION` is updated for resync and rebalance operations. The contents of the column can be `RESYNC`, `REBALANCE`, or `COMPACT`. For example, the following SQL query shows values in the `PASS` column during a rebalance operation.

```
SQL> SELECT GROUP_NUMBER, PASS, STATE FROM V$ASM_OPERATION;

GROUP_NUMBER PASS      STAT
------------ --------- ----
           2 RESYNC    WAIT
           2 REBALANCE WAIT
           2 COMPACT   WAIT
```

> ⓘ **See Also**
>
> - [ASM_POWER_LIMIT](#) for information about the `ASM_POWER_LIMIT` initialization parameter
> - [Summary of Disk Group Attributes](#) for information about disk group attributes
> - [Manually Rebalancing Disk Groups](#) for more information about rebalancing disk groups

Scrubbing Disk Groups

Oracle ASM disk scrubbing improves availability and reliability by searching for data that may be less likely to be read. Disk scrubbing checks logical data corruptions and repairs them automatically in normal and high redundancy disks groups. The scrubbing process repairs logical corruptions using the mirror disks. Disk scrubbing can be combined with disk group rebalancing to reduce I/O resources. The disk scrubbing process has minimal impact to the regular I/O in production systems.

You can perform scrubbing on a disk group, a specified disk, or a specified file of a disk group with the `ALTER DISKGROUP` SQL statement. For example, the following SQL statements show various options used when running the `ALTER DISKGROUP` *disk_group* `SCRUB` SQL statement.

```
SQL> ALTER DISKGROUP data SCRUB POWER LOW;

SQL> ALTER DISKGROUP data SCRUB FILE '+DATA/ORCL/DATAFILE/example.266.806582193'
       REPAIR POWER HIGH FORCE;

SQL> ALTER DISKGROUP data SCRUB DISK DATA_0005 REPAIR POWER HIGH FORCE;
```

You can stop an ongoing scrub operation with the `STOP` option.

```
SQL> ALTER DISKGROUP data SCRUB STOP;
```

When using `ALTER DISKGROUP` with the `SCRUB` option, the following items apply:

- The optional `REPAIR` option automatically repairs disk corruptions. If the `REPAIR` option is not specified, then the `SCRUB` option only checks and reports logical corruptions of the specified target.

- The optional `POWER` value can be set to `AUTO`, `LOW`, `HIGH`, or `MAX`. If the `POWER` option is not specified, the power value defaults to `AUTO` and the power adjusts to the optimum level for the system.

- If the optional `WAIT` option is specified, the command returns after the scrubbing operation has completed. If the `WAIT` option is not specified, the scrubbing operation is added into the scrubbing queue and the command returns immediately.

- If the optional `FORCE` option is specified, the command is processed even if the system I/O load is high or scrubbing has been disabled internally at the system level.

The scrubbing process is visible in the fields of the `V$ASM_OPERATION` view while the scrubbing process is running.

Extent-Based Scrubbing

With the introduction of Oracle AI Database 26ai, it is now possible to scrub individual extents or a range of extents.

Previously, Oracle ASM scrubbing was only available on files and disk groups. When compared to scrubbing the entire file, scrubbing specific extent sets greatly reduces the scrubbing turn-around time, improves the data availability, and minimizes the performance impact.

If scrubbing is used without the repair option, it will just analyze but not repair corruptions and write trace output to the alert log. If the repair option is specified, trace output is written to the debug log. If any unrepairable blocks are found, trace output will be written to the alert log.

Below is an example of what would appear if corruptions were found:

```
SQL> ALTER DISKGROUP data SCRUB FILE '+data/datafile/SYSAUX.260.1076155700'
Starting background process SCR
...
NOTE: Corrupted block 12 found in group 1 file +data/datafile/SYSAUX.260.1076155700
2022-06-01-02T12:08.09874-07:00
NOTE: Corrupted block 10 found in group 1 file +data/datafile/SYSAUX.260.1076155700
...
```

Every time a block, or group of blocks, is specified, the entire extent sets that contain all the specified blocks, are scrubbed for completeness. With this information the extended `ALTER`

`DISKGROUP … SCRUB FILE` functionality can scrub just the extent where the blocks are corrupted as shown in the example below:

```
ALTER DISKGROUP data SCRUB FILE '+data/datafile/SYSAUX.260.1076155700' block 10 count 3
```

To scrub multiple groups of data blocks and the underlying extents, multiple sets of block/count pairs can be specified in the same `ALTER DISKGROUP... SCRUB FILE` SQL statement:

```
ALTER DISKGROUP data SCRUB FILE '+data/hm/datafile/SYSAUX.260.1076155700' block 50 count
10 block 1024 count 70
```

Managing Oracle ADVM 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 be used to host an Oracle ACFS file system.

The compatibility attributes `COMPATIBLE.ASM` and `COMPATIBLE.ADVM` must be set to `11.2` or higher for the disk group.

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.

If the Oracle ADVM volume is on a flex disk group, then you can change the redundancy of the Oracle ADVM volume online with the `ALTER DISKGROUP MODIFY FILEGROUP` *volume_name* SQL statement.

> ⓘ **Note**
>
> If the volume is hosting an Oracle ACFS file system, then you cannot resize that volume with the SQL `ALTER DISKGROUP` statement. You must use the `acfsutil size` command instead.

The following are examples of the `ALTER DISKGROUP VOLUME` statement.

Example 4-9   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.
```

> **ⓘ See Also**
>
> - [Disk Group Compatibility Attributes](#) for information about disk group compatibility attributes
>
> - [About Oracle ASM File Group Properties](#) for information about file group properties
>
> - *Oracle Advanced Cluster File System Administrator's Guide* for more information about managing Oracle ACFS and Oracle ADVM with ASMCMD and ASMCA tools
>
> - *Oracle Advanced Cluster File System Administrator's Guide* for information about the resizing a volume with `acfsutil size`
>
> - *Oracle AI Database SQL Language Reference* for information about the `ALTER DISKGROUP` SQL statement

Oracle ASM Disk Discovery

Disk discovery is the mechanism used to find the operating system names for disks Oracle ASM can access.

This mechanism finds 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](#)

> **ⓘ See Also**
>
> [ASM_DISKSTRING](#) for additional information about disk discovery and the `ASM_DISKSTRING` initialization parameter

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.

> **ⓘ Note**
>
> Oracle ASM Filter Driver (ASMFD) is desupported on both Linux and Oracle Solaris beginning with Oracle AI Database 26ai.

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 ASM Configuration Assistant (ASMCA) operations or 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. For example, on Linux computers, the administrator might have used Oracle 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 AI 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](#).

- 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

> ⓘ **Note**
>
> Oracle ASM Filter Driver (ASMFD) is desupported on both Linux and Oracle Solaris beginning with Oracle AI Database 26ai.

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 Oracle ASMLib or a third-party vendor, then you may 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 Also**
>
> • [Oracle ASM and Multipathing](#) and your multipathing vendor documentation for information about multipathing

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 completely missing. Other causes of reduced redundancy files are disks running out of space or an insufficient number of failure groups.

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:

> ⓘ **Note**
>
> The values in the `REQUIRED_MIRROR_FREE_MB` and `USABLE_FILE_MB` columns of the `V$ASM_DISKGROUP` view are `0` if the disk group type is `FLEX` or `EXTEND`. The `0` value in this case means no value is given, not a zero amount in the `REQUIRED_MIRROR_FREE_MB` and `USABLE_FILE_MB` columns.

• `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:

– A normal redundancy disk group with more than two REGULAR 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.

– A high redundancy disk group with more than three REGULAR 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. For information about negative values of `USABLE_FILE_MB`, refer to [Negative Values of USABLE_FILE_MB](#).

- `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`

> ⓘ **Note**
>
> On Exadata platforms, `REQUIRED_MIRROR_FREE_MB` should be computed as:
>
> - If ASM version is 12.2 OR 18.*n* or later: If less than 5 failuregroups: `REQUIRED_MIRROR_FREE_MB` = 15% of `TOTAL_MB` if >= 5 failuregroups: `REQUIRED_MIRROR_FREE_MB` = 9% of `TOTAL_MB`
>
> - For all other versions: `REQUIRED_MIRROR_FREE_MB` = 15% of `TOTAL_MB`

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

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 to normal, high, flex, and extended redundancy disk groups. You can define the failure groups for each disk group when you create or alter the disk group.

There are multiple types of disk groups based on the Oracle ASM redundancy level. Table 4-2 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 4-2   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
FLEX redundancy	Two-way, three-way, unprotected (none)	Two-way (newly-created)
EXTENDED redundancy	Two-way, three-way, unprotected (none)	Two-way

For normal and high disk group types, 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.

For the flex group type, the number of failures tolerated before dismount demands on the number of failure groups. For five or more failure groups, two disk failures are tolerated. For three or four failure groups, one disk failure is tolerated.

For the extended disk group type, each site is similar to a flex disk group. If the site has five failure groups or more, two disk failures with in a site can be tolerated before the site becomes compromised. If the site has three or four failure groups, the site can tolerate one disk failure before the site is compromised. When two sites are compromised, the disk group dismounts. An extended disk group requires a minimum of three failure groups for each data site.

For flex and extended disk groups, mirroring describes the availability of the files within a disk group, not the disk group itself. For example: If a file is unprotected in a flex disk group that has five failure groups, then after one failure the disk group is still mounted, but the file becomes unavailable.

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 in one disk, plus the failure of another disk. However, this scenario does not protect against the failure of two disks.

* HIGH redundancy

  Oracle ASM provides three-way (triple) mirroring by default. A loss of two Oracle ASM disks in different failure groups is tolerated.

* FLEX redundancy

Oracle ASM provides two-way mirroring by default for newly-created flex disk groups. For migrated flex disk groups, the default values are obtained from the template values in the normal or high redundancy disk groups before migration. For migration from normal redundancy, if the template defaults were not changed, then the flex defaults are two-way mirroring. For migration from high redundancy, if the template defaults were not changed, then the flex defaults are three-way mirroring.

- EXTENDED redundancy

  Oracle ASM provides two-way mirroring by default. The redundancy setting describes redundancy within a data site. For example: If there is a two-way mirrored file in a two-data-site extended disk group, then there are four copies of the file, two in each data site.

Oracle ASM file groups in a flex or extended disk group can have different redundancy levels.

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.

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

> ⓘ **See Also**
>
> - Managing Oracle ASM Flex Disk Groups for information about Oracle ASM flex disk groups, extended disk groups, and file groups
> - Managing Disk Group Templates for information about specifying Oracle ASM disk group templates

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 more 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.

Oracle recommends a minimum of three failure groups for normal redundancy disk groups and five failure groups for high redundancy disk groups to maintain the necessary number of copies of the Partner Status Table (PST) and to ensure robustness with respect to storage hardware failures.

In the event of a system failure, three failure groups in a normal redundancy disk group allow a comparison among three PSTs to accurately determine the most up to date and correct version of the PST, which could not be done with a comparison between only two PSTs. Similarly with a high redundancy disk group, if two failure groups are offline, then Oracle ASM would be able to make a comparison among the three remaining PSTs.

If configuring an extra failure group presents a problem with storage capacity management, then a quorum failure group can be used as the extra failure group to store a copy of the PST. A quorum failure group does not require the same capacity as the other failure groups.

Failure groups can be specified as regular or quorum failure groups. For information about quorum failure groups, see Storing Oracle Cluster Registry and Voting Files in Oracle ASM Disk Groups.

> ⓘ **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 AI 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](#).

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 disk 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.

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 12 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 and is only applicable to normal and high redundancy disk groups.

If the attribute is not set explicitly, then the default value (`12h`) 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 during which you are able 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 6 hours or 360 minutes. For example:

```
ALTER DISKGROUP data SET ATTRIBUTE 'disk_repair_time' = '6h'
ALTER DISKGROUP data SET ATTRIBUTE 'disk_repair_time' = '360m'
```

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. Examples of taking disks offline and bringing them online follow.

The following example takes disk `DATA_001` offline and drops it after fifteen minutes.

```
ALTER DISKGROUP data OFFLINE DISK DATA_001 DROP AFTER 15m;
```

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 disks 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 DISKS IN FAILGROUP FG2;
```

The next example brings all of the disks in failure group `FG2` online:

```
ALTER DISKGROUP data ONLINE DISKS 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 `ALTER DISKGROUP ONLINE` statements displays the name and state of the current operation that you are performing. For example, the following SQL query shows values in the `PASS` column during an online operation.

```
SQL> SELECT GROUP_NUMBER, PASS, STATE FROM V$ASM_OPERATION;

GROUP_NUMBER PASS      STAT
------------ --------- ----
           1 RESYNC    RUN
           1 REBALANCE WAIT
           1 COMPACT   WAIT
```

An offline operation does not generate a display in a `V$ASM_OPERATION` view query.

The resync checkpoint maintains the position in the resync process in the event of a failure. If the resync process or an instance fails, then the next resync operation continues from the position where the previous process left off, rather than starting from the beginning or the resync process.

You can set the `FAILGROUP_REPAIR_TIME` and `CONTENT.TYPE` disk group attributes. The `FAILGROUP_REPAIR_TIME` disk group attribute specifies a default repair time for the failure groups in the disk group. The `CONTENT.TYPE` disk group attribute specifies the type of data expected to be stored in a disk group. You can set these attributes with ASMCA, ASMCMD `mkdg`, or SQL `CREATE` and `ALTER DISKGROUP` statements.

The ASMCMD `lsop` command shows the resync time estimate. There are separate rows in the `V$ASM_OPERATION` table for different phases of rebalance: disk resync, rebalance, and data compaction.

The ASMCMD `online` command has a `power` option to specify the power for the online operation. The SQL `ALTER DISKGROUP REPLACE DISK` statement also has the power option.

The ASMCMD `chdg` command provides the `replace` option in addition to the `add` and `drop` tags. The ASMCMD `mkdg` command has an additional time parameter (`-t`) to specify the time to offline a failure group.

> ⓘ **See Also**
>
> - [Disk Group Compatibility](#) for information about disk group compatibility
> - [Managing Disk Group Attributes](#) for information about disk group attributes
> - *Oracle Grid Infrastructure Installation and Upgrade Guide* and [Upgrading and Patching Oracle ASM](#) for information about upgrading Oracle ASM
> - *Oracle AI Database SQL Language Reference* for information about `ALTER DISKGROUP`
> - *Oracle AI Database SQL Language Reference* for information about `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.

> ⓘ **Note**
>
> In an Oracle extended cluster, which contains nodes that span multiple physically separated sites, the `PREFERRED_READ.ENABLED` disk group attribute controls whether preferred read functionality is enabled for a disk group. If preferred read functionality is enabled, then this functionality enables an instance to determine and read from disks at the same site as itself, which can improve performance. Whether or not `PREFERRED_READ.ENABLED` has been enabled, preferred read can be set at the failure group level on an Oracle ASM instance or a client instance in a cluster with the `ASM_PREFERRED_READ_FAILURE_GROUPS` initialization parameter, which is available for backward compatibility.

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.

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.

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.

> (i) **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**
>
> - [PREFERRED_READ.ENABLED](#) for information about the `PREFERRED_READ.ENABLED` disk group attribute
>
> - [ASM_PREFERRED_READ_FAILURE_GROUPS](#) for more information about the `ASM_PREFERRED_READ_FAILURE_GROUPS` initialization parameter
>
> - *Oracle Real Application Clusters Administration and Deployment Guide* for information about configuring preferred read disks in extended clusters
>
> - *Oracle AI 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](#).

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.

Oracle ASM has the following limits on the number of disk groups, disks, and files:

- 511 disk groups in a storage system for Oracle Database 12*c* Release 1 or later
- 10,000 Oracle ASM disks in a disk group
- 65,530 Oracle ASM disks in a storage system
- 1 million files for each disk group

Without any Oracle Exadata Storage, Oracle ASM has the following storage limits if the `COMPATIBLE.ASM` or `COMPATIBLE.RDBMS` disk group attribute is set to less than `12.1`:

- 2 terabytes (TB) maximum storage for each Oracle ASM disk
- 20 petabytes (PB) maximum for the storage system

Without any Oracle Exadata Storage, Oracle ASM has the following storage limits if the `COMPATIBLE.ASM` and `COMPATIBLE.RDBMS` disk group attributes are set to `12.1` or greater:

- 4 PB maximum storage for each Oracle ASM disk with the allocation unit (AU) size equal to 1 MB
- 8 PB maximum storage for each Oracle ASM disk with the AU size equal to 2 MB
- 16 PB maximum storage for each Oracle ASM disk with the AU size equal to 4 MB
- 32 PB maximum storage for each Oracle ASM disk with the AU size equal to 8 MB
- 320 exabytes (EB) maximum for the storage system

With all Oracle Exadata Storage, Oracle ASM has the following storage limits:

- 4 PB maximum storage for each Oracle ASM disk with the AU size equal to 1 MB
- 8 PB maximum storage for each Oracle ASM disk with the AU size equal to 2 MB
- 16 PB maximum storage for each Oracle ASM disk with the AU size equal to 4 MB
- 32 PB maximum storage for each Oracle ASM disk with the AU size equal to 8 MB
- 320 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 65,530 disks. However, the total number of disks that can be in one disk group is limited to 10,000 disks. This is a limitation on the number of Oracle ASM disks, not necessarily the number of physical storage devices. A storage array could group multiple physical storage devices into a LUN that is used as a single Oracle ASM disk.

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 AI Database supports data file sizes up to 128 TB depending on the file system. In addition, Oracle AI Database has a file size limit that is dependent on the `DB_BLOCK_SIZE` initialization parameter. In non-Exadata environments ASM per disk storage restrictions are removed when using ASM and rdbms compatibility above 12.1.

> ⓘ **See Also**
>
> - [Disk Group Compatibility](#) for information about Oracle ASM disk group compatibility attributes
> - [Considerations When Setting Disk Group Compatibility](#) for additional information about Oracle ASM file size limits
> - *Oracle Advanced Cluster File System Administrator's Guide* for information about Oracle ACFS space usage

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](#)
- [Features Enabled By Disk Group Compatibility Attribute Settings](#)
- [Considerations When Setting Disk Group Compatibility](#)

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 `19.0` can take advantage of Oracle ASM 19c features. For information about the features enabled for combinations of compatibility attribute settings, refer to [Table 4-4](#).

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 AI 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 `19.0`, and RDBMS compatibility is set to `12.2.0.1`, then the Oracle ASM software version must be at least `19.0`, and the Oracle Database client software version must be at least `12.2.0.1`. The `COMPATIBLE.ADVM` attribute determines whether the Oracle ASM Dynamic Volume Manager feature can create a 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. For examples of valid combinations of compatible attribute settings, refer to [Table 4-3](#).

You can also set and update disk group attributes with the ASMCMD `setattr` command. For information about the ASMCMD `setattr` command, refer to [setattr](#).

> ⓘ **Note**
>
> - 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. For more information, refer to [Reverting Disk Group Compatibility](#).
>
> - 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.

- COMPATIBLE.ASM

- COMPATIBLE.RDBMS

- COMPATIBLE.ADVM

> ⓘ **Note**
>
> Starting with Oracle ASM version 12.2.0.1, the minimum and default settings for Oracle ASM disk group attributes are:
>
> - `COMPATIBLE.ASM` = `11.2.0.2` and `COMPATIBLE.RDBMS` = `10.1` for `EXTERNAL`, `NORMAL`, and `HIGH` redundancy disk groups
>
> - `COMPATIBLE.ASM` = `12.2.0.1` and `COMPATIBLE.RDBMS` = `12.2.0.1` for `FLEX` and `EXTENDED` disk groups
>
> The default and minimum values for Oracle ASM disk group attributes are present when using SQL*Plus to create a disk group. Other utilities, such as ASMCA, may use a different default value when creating a disk group; however, the minimum value for Oracle ASM disk group attributes should be the same.

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.

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 parameter of the databases is set to either `12.2`, then `COMPATIBLE.RDBMS` can be set to any value between `10.1` and `12.2` inclusively. If the `COMPATIBLE` initialization parameter of the databases is set to `19.0`, then `COMPATIBLE.RDBMS` can be set to any value between `10.1` and `19.0` inclusively.

> ⓘ **Note**
>
> The database initialization parameter `COMPATIBLE` enables you to use a new release of Oracle AI Database, while at the same time guaranteeing backward compatibility with an earlier release. For more information about the `COMPATIBLE` initialization parameter, refer to *Oracle AI Database Reference*.

COMPATIBLE.ADVM

The value for the disk group `COMPATIBLE.ADVM` attribute determines whether the disk group can contain Oracle ADVM 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.

The default value of the `COMPATIBLE.ADVM` attribute depends on the level of the Oracle ASM release.

> ⓘ **See Also**
>
> *Oracle Advanced Cluster File System Administrator's Guide* for more information about Oracle ACFS and Oracle ADVM

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](#)
- [Reverting Disk Group Compatibility](#)

You can also set the disk group compatibility settings withOracle ASM command-line utility (ASMCMD) and Oracle ASM Configuration Assistant (ASMCA). Refer to [Managing Oracle ASM With ASMCA](#) and [Managing Oracle ASM with ASMCMD](#) .

> ⓘ **Note**
>
> Advancing the values for disk group compatibility attributes is an irreversible operation. See Reverting Disk Group Compatibility.

> ⓘ **See Also**
>
> *Oracle AI 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 AI Database release number. For example, you can specify compatibility as `12.2` or `19.0`; Oracle assumes that any missing version number digits are zeros.

Oracle ASM features enabled for combinations of attribute settings are shown in Features Enabled By Disk Group Compatibility Attribute Settings.

Table 4-3 shows some valid combinations of the disk group compatibility attributes and the valid Oracle ASM and database instance versions for each combination.

Table 4-3   Examples of disk group compatibility attribute settings

COMPATIBLE.ASM	COMPATIBLE.RDBMS	COMPATIBLE.ADVM	Oracle ASM Instance Version	COMPATIBLE Setting for RDBMS Instance
12.1	12.1	12.1	>= 12.1	>= 12.1
12.2	12.1	12.2	>= 12.2	>= 12.1
18.0	18.0	18.0	>= 18.0	>= 18.0
19.0	19.0	19.0	>=19.0	>=19.0
21.0	21.0	21.0	>=21.0	>=21.0

These are some possible combinations of Oracle ASM and database releases:

- The database release is Oracle Database 11*g* Release 2 (11.2) (database initialization parameter `COMPATIBLE` is set to `11.2`) and the Oracle ASM release is Oracle ASM 11*g* Release 2 (11.2). The `COMPATIBLE.ASM` and `COMPATIBLE.RDBMS` disk group attributes are set to `10.2` for a disk group.

  The Oracle ASM disk group functionality remains at Oracle Database 10*g* Release 2 (10.2).

- The database release is Oracle Database 10*g* Release 1 (10.1) 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 database release is Oracle Database 12*c* Release 2 (12.2) (database initialization parameter `COMPATIBLE` is set to `12.2`) and the Oracle ASM release is Oracle ASM 12*c* Release 2 (12.2). All the disk group compatibility attributes are set to `12.2` for a disk group.

Oracle ASM in a Domain Services Cluster and Oracle AI Database in a Member Cluster Configuration

> ⓘ **Note**
>
> Starting with Oracle Grid Infrastructure 21c, Member Clusters, which are part of the Oracle Cluster Domain architecture, are desupported. However, Domain Services Clusters continues to support Members Clusters in releases previous to Oracle Grid Infrastructure 21c.

> ⓘ **Note**
>
> Starting with Oracle Grid Infrastructure 21c, Domain Services Cluster (DSC), which is part of the Oracle Cluster Domain architecture, is deprecated and can be desupported in a future release.

Oracle ASM or Oracle IOServer running in a Domain Services Cluster (DSC) version 19.1 support Oracle Database and Oracle Grid Infrastructure running in Member Cluster versions 12.2 and 19.1.

Oracle ASM or Oracle IOServer running in a Domain Services Cluster (DSC) version 12.2 support Oracle Database and Oracle Grid Infrastructure running in Member Cluster versions 12.2 and 19.1.

An Oracle AI Database instance must be a version lower than or equal to the Oracle ASM version in the same cluster. However, this requirement is not strictly enforced when the database instance is connecting from a different cluster.

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 `19.0` and the RDBMS compatibility set to the default (the `COMPATIBLE.RDBMS` default is less than or equal to `19.0`):

```
CREATE DISKGROUP data1 DISK '/dev/sd*'
      ATTRIBUTE 'compatible.asm' = '19.0';
```

The following example creates a normal redundancy disk group `data2` with the ASM, RDBMS, and ADVM compatibility set to `19.0`:

```
CREATE DISKGROUP data2 DISK '/dev/sd*'
      ATTRIBUTE 'compatible.asm' = '19.0', 'compatible.rdbms' = '19.0',
                'compatible.advm' = '19.0';
```

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 . 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 `19.0`. An Oracle ASM instance must be at Oracle ASM 19c or higher to access the `data3` disk group.

```
ALTER DISKGROUP data3 SET ATTRIBUTE 'compatible.asm' = '19.0';
```

The following example advances the `COMPATIBLE.RDBMS` and `COMPATIBLE.ADVM` compatibility attributes of the disk group `data3` to `10.0`. This example assumes that the value of `COMPATIBLE.ASM` is set to `19.0`.

```
ALTER DISKGROUP data3 SET ATTRIBUTE 'compatible.rdbms' = '19.0',
ALTER DISKGROUP data3 SET ATTRIBUTE 'compatible.advm' = '19.0';
```

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.

You can also display the disk group compatibility attributes with the ASMCMD command `lsattr`.

> ⓘ **See Also**
>
> - Example 6-1 for an example querying the `V$ASM_ATTRIBUTE` view
> - lsattr for information about the ASMCMD `lsattr` command
> - *Oracle AI Database SQL Language Reference* for information about disk group attributes
> - *Oracle AI Database Reference* for details on the `V$ASM_DISKGROUP` and `V$ASM_ATTRIBUTE` dynamic performance views

Reverting Disk Group Compatibility

You cannot revert disk group compatibility to a previous values. Instead you must create a new disk group.

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.

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.

5. Move or restore the database files into the newly created disk group using either the `ALTER DATABASE MOVE DATAFILE` SQL statement or Recovery Manager (RMAN).

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.

> ⓘ **See Also**
>
> - [About Creating a Disk Group](#) for information about creating a disk group
> - [Moving Data Files Between Disk Groups With ALTER DATABASE](#) for information about moving data files between disk groups
> - [Moving Data Files Between Oracle ASM Disk Groups Using RMAN](#) for information about moving database files with RMAN
> - [Dropping Disk Groups](#) for information about dropping a disk group
> - *Oracle Clusterware Administration and Deployment Guide* for information about administering OCR and voting files
> - *Oracle AI Database Administrator's Guide* for information about using the `ALTER DATABASE MOVE DATAFILE` SQL statement to move data files between Oracle ASM disk groups

Features Enabled By Disk Group Compatibility Attribute Settings

This topic describes the Oracle ASM features enabled by valid combinations of the disk group compatibility attribute settings.

The following list applies to Oracle ASM features enabled by valid combinations of 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`.

- Starting with Oracle Grid Infrastructure 12.2.0.1 software, the minimum setting for `COMPATIBLE.ASM` is `11.2.0.2`.

- A value of not applicable (n/a) means that the setting of the attribute has no effect on the feature.

- Oracle ASM features not explicitly listed in the following table do not require advancing the disk group compatibility attribute settings.

- Oracle ASM features explicitly identified by an operating system in the following table are available on that operating system starting with the associated disk group attribute settings.

- If encryption is configured for the first time on Oracle ASM 11*g* Release 2 (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 (11.2.0.3) on Linux, then the disk group compatibility attributes for `ASM` and `ADVM` must be set to `11.2.0.3` or higher.

- Oracle ACFS does not support encryption or replication with the following: Oracle AI Database data files, control files, online redo logs, archived redo log files, flashback logs, RMAN backups, and Oracle Data Pump dump file sets.

- Oracle ACFS on Oracle Exadata storage is supported starting with Oracle Grid Infrastructure 12.1.0.2 on Linux.

- There may also be minimum requirements for the database `COMPATIBLE` initialization parameter.

Table 4-4   Oracle ASM 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
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
`ASM_POWER_LIMIT` value up to `1024`	>= 11.2.0.2	n/a	n/a
Content type of a disk group	>= 11.2.0.3	n/a	n/a
Appliance mode for Oracle Exadata (no fixed partnering)	>= 11.2.0.4	n/a	n/a
Replication status of a disk group	>= 12.1	n/a	n/a
Managing a shared password file in a disk group	>= 12.1	n/a	n/a
Greater than 2 TB Oracle ASM disks without Oracle Exadata storage	>= 12.1	>= 12.1	n/a
Fixed partnering for Oracle Exadata	>= 12.1.0.2	n/a	n/a
Appliance mode for Oracle Data Appliance (ODA)	>= 12.1.0.2	n/a	n/a

Table 4-4   (Cont.) Oracle ASM features enabled by disk group compatibility attribute settings

Disk Group Features Enabled	COMPATIBLE.ASM	COMPATIBLE.RDBMS	COMPATIBLE.ADVM
Support for Oracle Exadata sparse disk groups *(sparse disk groups not supported in Oracle ACFS)*	>= 12.1.0.2	>= 12.1.0.2	n/a
Support for resync checkpoint	>= 12.1.0.2	>= 12.1.0.2	n/a
`LOGICAL_SECTOR_SIZE`	>= 12.2	n/a	n/a
Altering sector size	>= 12.2	n/a	n/a
Oracle ASM flex and extended disk groups	>= 12.2	>= 12.2	n/a
`SCRUB_ASYNC_LIMIT`	>= 12.2	n/a	n/a
`PREFERRED_READ.ENABLED` Oracle Database 12c Release 2 (12.2) is required.	>=12.2	n/a	n/a
Converting normal or high redundancy disk groups to flex disk groups without restricted mount	>=18.0	>=12.2	n/a
Oracle ASM flex disk group support for multitenant cloning	>=18.0	>=18.0	n/a
Storage conversion for member clusters	>=18.0 and <21.0	n/a	n/a
Virtual Allocation Metadata (VAM) enabled on non-sparse normal and high redundancy disk groups	>=18.0	n/a	n/a
Support for single parity protection in Oracle ASM flex disk groups The database `COMPATIBLE` initialization parameter must be set to 19.1 or greater.	>=19.1	n/a	n/a
Support for double parity protection in Oracle ASM flex disk groups The database `COMPATIBLE` initialization parameter must be set to 19.5 or greater.	>=19.5	n/a	n/a

---

ⓘ **See Also**

- *Oracle Advanced Cluster File System Administrator's Guide* for information about Oracle ACFS features enabled by disk group compatibility settings

- *Oracle Advanced Cluster File System Administrator's Guide* for more information about Oracle ACFS and Oracle ADVM, including any limitations or restrictions, about storing data files in an Oracle ACFS file system

- ASM_POWER_LIMIT for information about the rebalancing process and the `ASM_POWER_LIMIT` initialization parameter

---

Considerations When Setting Disk Group Compatibility

If you advance disk group compatibility, then you could enable the creation of files that are too large to be managed by a previous release of Oracle AI Database.

For example, 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 4-5 shows the maximum Oracle ASM file sizes supported for multiple `COMPATIBLE.RDBMS` settings with the `AU_SIZE` disk group attribute set to 1 MB. This table 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.

Table 4-5   Maximum Oracle ASM file sizes for disk groups with AU_SIZE set to 1 MB

Redundancy	COMPATIBLE.RDBMS >= 12.2
External	255 TB
Normal	93 TB
High	62 TB

Table 4-6 shows the maximum Oracle ASM file sizes supported for multiple `COMPATIBLE.RDBMS` settings with the `AU_SIZE` disk group attribute set to 4 MB.

Table 4-6   Maximum Oracle ASM file sizes for disk groups with AU_SIZE set to 4 MB

Redundancy	COMPATIBLE.RDBMS >= 12.2
External	1022 TB
Normal	1022 TB
High	998 TB

> ⓘ **See Also**
>
> - Oracle ASM Storage Limits for information about Oracle ASM storage sizes
> - *Oracle AI Database Upgrade Guide* for information about database compatibility
> - *Oracle AI Database Administrator's Guide* for information about the `COMPATIBLE` initialization parameter and irreversible compatibility
> - *Oracle AI 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 group. You can manage Oracle ASM file access control with ASMCMD commands 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](#)

> ⓘ **See Also**
>
> - [Using Separate Operating System Groups for Oracle ASM Users](#) for information about designating separate operating system groups
>
> - [ASMCMD File Access Control Commands](#) for information about managing Oracle ASM File Access Control with ASMCMD commands
>
> - [Viewing Oracle ASM File Access Control Information](#) for information about views that provide details about Oracle ASM file access control
>
> - [Authentication for Accessing Oracle ASM Instances](#) for information about controlling accessing to Oracle ASM instances

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](#) 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 you create a file, a group is not automatically associated with the file. You must use the `asmcmd chgrp` command to set the group manually because Oracle ASM File Access Control does not have the concept of the primary group of a user.

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.

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.

3. Optionally, you can create user groups that are groups of database users that share the same access permissions to Oracle ASM files.

> ⓘ **See Also**
>
> - [Setting Disk Group Attributes for Oracle ASM File Access Control](#) for information about setting disk group attributes
> - [Using SQL Statements to Manage Oracle ASM File Access Control](#) for information about using `ALTER DISKGROUP SET PERMISSION` and `ALTER DISKGROUP SET OWNERSHIP` for managing Oracle ASM File Access Control with SQL statements
> - [Using SQL Statements to Manage Oracle ASM File Access Control](#) for information about using `ALTER DISKGROUP ADD USERGROUP`

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.

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.

The `COMPATIBLE.ASM` and `COMPATIBLE.RDBMS` disk group attributes must be set to `11.2` or higher to enable Oracle ASM File Access Control.

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 `umask` permissions setting is `026` which enables read-write access for the owner (masks out nothing with `0`), read access for users in the group (masks out write permission with `2`), and no access to others (masks out all access with `6`) 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';
```

> ⓘ **See Also**
>
> - setattr for information about the ASMCMD `setattr` command
> - chmod for information about the ASMCMD `chmod` command
> - Disk Group Compatibility for information about disk group compatibility attributes

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.

When administering users with SQL*Plus, the users must be existing operating system users and their user names must have corresponding operating system user IDs. However, only users in the same cluster as the Oracle ASM instance can be validated.

> ⓘ **Note**
>
> When modifying ownership of files, the previous owner of a file cannot be dropped while the file is still open even after the ownership of the file has changed.

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 maximum length of a user group name is 63 bytes. If you specify the user name, then it must be in the OS password file and its length cannot exceed 32 characters.

  The operating system users of 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. 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 the operating system user associated with that database to the disk group. Also, you can use this clause to add the operating system user associated with a database 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 REPLACE USER

  Replaces one operating system user with another in an Oracle ASM disk group. The replacement user must not be a user currently in the disk group user list.

If the command succeeds, all files that were previously owned by current user are now owned by the replacement user. The current user is automatically removed from the user list of the disk group. Adding the current user back to the disk group is possible, but this user does not own any files.

When changing the owner of an open file, the new owner cannot be dropped with the owner's files until all instances within a cluster are restarted.

Only an Oracle ASM administrator can run this command.

```
SQL> ALTER DISKGROUP data REPLACE USER 'oracle1' WITH 'oracle2';
```

- `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.

  ```
  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.

  ```
  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 AI Database SQL Language Reference*

Maintaining Disk Groups

This section discusses maintenance operations performed on disk groups.

This section contains the following topics:

- [Mounting and Dismounting Disk Groups](#)
- [Mounting Disk Groups Using the FORCE Option](#)
- [Checking the Internal Consistency of Disk Group Metadata](#)
- [Dropping Disk Groups](#)
- [Renaming Disks Groups](#)

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 process makes the disk groups 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`. You can also specify the optional `FORCE` option. For information about using the `FORCE` option, refer to [Mounting Disk Groups Using the FORCE Option](#).

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.

> ⓘ **Note**
>
> An Oracle ASM instance mounts an incomplete disk group differently depending on the specified compatibility.

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:

```
SQL> ALTER DISKGROUP data1 MOUNT FORCE;
```

> ⓘ **See Also**
>
> - [Oracle ASM Fast Mirror Resync](#) for more information about setting the `DISK_REPAIR_TIME` disk group attribute
>
> - [Features Enabled By Disk Group Compatibility Attribute Settings](#) for information features enabled by disk group compatibility attributes
>
> - *Oracle AI 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 AI 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 Advanced 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.

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:

```
SQL> 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 disks from a disk group with ASMCA.

> ⓘ **See Also**
>
> - *Oracle Advanced Cluster File System Administrator's Guide* for information about dismounting an Oracle ACFS file system
> - [Managing Disk Groups with ASMCA](#) for information about ASMCA

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 rerun phase `two` using the generated configuration file.

- `dgname=diskgroup`

  Specifies the name of the disk group 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 clean 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 `fra1` disk group to `fra2` 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`.

After renaming a disk group, you can rename the disks in the disk group to match the new disk group name. For example:

```
SQL> ALTER DISKGROUP fra2 RENAME DISKS ALL;
```

For information about renaming disks, refer to Renaming Disks in Disk Groups.

Example 4-12   Using renamedg

```
$ renamedg dgname=fra1 newdgname=fra2 asm_diskstring='/devices/disk*'
     verbose=true

$ renamedg phase=one dgname=fra1 newdgname=fra2
     asm_diskstring='/devices/disk*' config=/tmp/fra2.conf verbose=true
```

```
$ renamedg phase=two dgname=fra1 newdgname=fra2 config=/tmp/fra2.conf
      verbose=true
```

Managing Oracle ASM Flex Disk Groups

Oracle ASM provides database-oriented storage management with flex and extended disk groups.

An Oracle ASM flex disk group is a disk group type that supports Oracle ASM file groups. An Oracle ASM file group describes a group of files that belong to a database, and enables storage management to be performed at the file group, or database, level. An Oracle ASM extended disk group has all the features of a flex disk group, with a specific design to be highly available in an extended (or stretch) cluster environment.

This section discusses the following topics:

- About Oracle ASM Flex Disk Groups
- About Oracle ASM File Groups
- About Oracle ASM File Group Properties
- About Quota Groups for Oracle ASM File Groups
- About Oracle ASM Extended Disk Groups
- About Point-In-Time Database Cloning With Oracle ASM Flex Disk Groups

About Oracle ASM Flex Disk Groups

An Oracle ASM flex disk group is a disk group type that supports Oracle ASM file groups and quota groups.

In general, a flex disk group enables users to manage storage at the granularity of the database, in addition to at the disk group level.

The following are characteristics of a flex disk group:

- The redundancy of files in a flex disk group is flexible and enables storage management at the database level.
- File groups of flex disk groups are used to describe database files.

  Each database has its own file group, and storage management can be done at the file group level, in addition to the disk group level. For example, you can specify different redundancy and rebalance settings for different file groups. File groups are associated with a quota group, enabling easy quota management.

- The redundancy setting of a flex disk group is set to `FLEX REDUNDANCY` and each file group of the flex disk group has its own redundancy property setting.
- A flex disk group generally tolerates two failures, the same as a `HIGH` redundancy disk group.

  However, if the disk group has fewer than five failure groups, then it can only tolerate one failure. This restriction is the result of the quorum requirement of the disk group.

- The failure tolerated by a database is dependent on the redundancy setting of its associated file group in the flex disk group.
- A flex disk group requires a minimum of three failure groups.

- If a disk group with `FLEX REDUNDANCY` has only two regular failure groups and one quorum failure group, then only two copies of the data are stored, even if `HIGH REDUNDANCY` is selected for a database file in the flex disk group. This behavior is a consequence of having only two regular failure groups. However, if an additional regular failure group is added later, then a third copy of the data is created automatically. After the addition of the regular failure group, if one failure group is lost and a rebalance completes successfully, then the disk group can remain mounted after a second failure group goes offline.

- Migrating (converting) to a flex disk group can be performed for a `NORMAL` or `HIGH` redundancy disk group with the `CONVERT REDUNDANCY TO FLEX` clause of the `ALTER DISKGROUP` SQL statement. You cannot migrate an `EXTERNAL` redundancy disk group to a `FLEX` redundancy disk group.

> ⓘ **Note**
>
> – When migrating to a flex disk group, a disk group with `NORMAL` redundancy must have a minimum of three failure groups and a disk group with `HIGH` redundancy must have a minimum of five failure groups.
>
> – The Virtual Allocation Metadata (VAM) migration disk group attribute must be enabled and the migration completed before migrating without the use of a restrictive mount.

- A flex disk group enables the creation of point-in-time database clones.

  A point-in-time database clone is a full copy of a specific database, not the underlying storage system. Only data files are copied. Other files are created, or referenced as required to create the clone. For information about flex disk groups and point-in-time database clones, refer to About Point-In-Time Database Cloning With Oracle ASM Flex Disk Groups.

- The default size of the allocation unit (AU) is 4 M.

- The Virtually Allocated Metadata (VAM) is enabled and required for a flex disk group.

- Flex disk groups require the `COMPATIBLE.ASM` and `COMPATIBLE.RDBMS` disk group attributes to be set at `12.2` or greater.

- To create an Oracle ADVM volume on a flex disk group, the `COMPATIBLE.ADVM` disk group attribute must be set to `12.2` or greater.

> ⓘ **Note**
>
> The `REQUIRED_MIRROR_FREE_MB` and `USABLE_FILE_MB` columns in the `V$ASM_DISKGROUP` view report no values for a flex disk group.

The following examples illustrate how to use SQL statements to:

- Create a flex disk group

- Migrate a normal disk group to a flex disk group

Example 4-13   Using CREATE DISKGROUP to create a flex disk group

```
SQL> CREATE DISKGROUP flex_data FLEX REDUNDANCY DISK my_disk_discovery_path;
```

Example 4-14   Using ALTER DISKGROUP to migrate a normal disk group to a flex disk group

```
SQL> CREATE DISKGROUP normaltoflex NORMAL REDUNDANCY
  DISK '/devices/diskm*'
  ATTRIBUTE 'au_size'='4M',
    'compatible.asm' = '19.0',
    'compatible.rdbms' = '19.0',
    'compatible.advm' = '19.0';

Diskgroup created.

SQL> ALTER DISKGROUP normaltoflex CONVERT REDUNDANCY TO FLEX;

Diskgroup altered.

SQL> SELECT name, type FROM V$ASM_DISKGROUP WHERE name = 'NORMALTOFLEX';

NAME                           TYPE
------------------------------ ------
NORMALTOFLEX                   FLEX
```

About Oracle ASM File Groups

An Oracle ASM file group is a group of files that share the same set of properties and characteristics.

> ⓘ **Note**
>
> A multitenant container database is the only supported architecture in Oracle Database 21c and later releases. While the documentation is being revised, legacy terminology may persist. In most cases, "database" and "non-CDB" refer to a CDB or PDB, depending on context. In some contexts, such as upgrades, "non-CDB" refers to a non-CDB from a previous release.

A major benefit of file groups is the ability to have a different availability specification for each database that shares the same disk group.

The set of properties for an Oracle ASM file group includes redundancy, rebalance rebuild priority, rebalance power limit, client compatibility, striping, quota group, and the access control list.

The following list provides important notes about file groups.
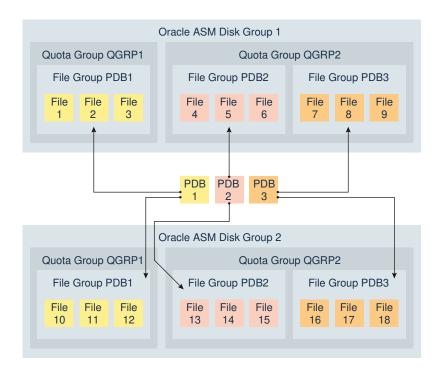
- A disk group contains at least one file group, the default file group.
- A disk group can contain multiple file groups.
- A disk group must have `FLEX` or `EXTENDED` redundancy to contain a file group.
- A disk group can store files belonging to multiple databases with each database having a separate file group.
- A database can have only one file group in a disk group.
- A database can span multiple disk groups with multiple file groups in different disk groups. The file groups belonging to a database in multiple disk groups should all have the same name for consistency and easy identification.

- A file group can belong to only one disk group.

- A file group can describe only one database, PDB, CDB, volume, or cluster.

- A file group of an Oracle ADVM volume in a flex disk group always has the name of the associated volume. There can be only one volume in a file group.

- A file group can belong to only one quota group.

- Automatically created file groups are associated with the generic quota group.

- When a database, PDB, or CDB is created, if an existing file group has a client ID or name that matches the ID or name of the database, PDB, or CDB, then that file group is used to describe its files. Otherwise, a new file group is created for the database, PDB, or CDB.

- A file group that is automatically created as a result of a create database operation is identified as system-created. This system-created file group is dropped when the database is dropped. However, if you manually create a file group, then that file group is not automatically dropped when you drop the related database. You must explicitly drop the manually-created file group.

The following figure illustrates the use of Oracle ASM file groups and quota groups in a multitenant environment. In this illustration, PDB3 could be a clone of PDB2. File groups and quota groups in different disk groups can have the same name.

Figure 4-1   Oracle ASM File Groups

In the figure, file groups named PDB1 in disk group 1 and disk group 2 are dedicated to pluggable database PDB1, file groups named PDB2 in disk group 1 and disk group 2 are dedicated to pluggable database PDB2, and file groups named PDB3 in disk group 1 and disk group 2 are dedicated to pluggable database PDB3.

File group PDB1 in disk group 1 belongs to quota group QGRP1 in disk group 1. File group PDB1 in disk group 2 belongs to quota group QGRP1 in disk group 2. File groups PDB2 and PDB3 in disk group 1 belong to quota group QGRP2 in disk group 1. File groups PDB2 and PDB3 in disk group 2 belong to quota group QGRP2 in disk group 2.

You can administer file groups with SQL statements. For example, you can use the `FILEGROUP` clause of the `ALTER DISKGROUP` SQL statement to administer file groups.

Example 4-15   Adding a File Group to a Database

This example shows how SQL statements are used to add a file group for a pluggable database. The SQL statements also associate the newly added file group to an existing quota group.

```
ALTER DISKGROUP DATA ADD FILEGROUP DATAFG1_PDB1
     DATABASE ORCL_PDB1
     SET 'quota_group' = 'Quota_QGRP1';
```

Example 4-16   Modifying a File Group

This example shows how to modify the `redundancy` property for a specific file type in a file group.

```
ALTER DISKGROUP DATA MODIFY FILEGROUP DATAFG1_PDB1
    SET 'datafile.redundancy' = 'HIGH';
```

ASMCMD provides the following commands to manage and track file and quota groups.

- `chfg` and `chqg` to modify a file group or quota group

- `lsfg` and `lsqg` to list file groups or quota groups

- `mkfg` and `mkqg` to create a file group or quota group

- `rmfg` and `rmqg` to delete a file group or quota group

- `mvfg` to move a file group

You can drop a file group including its associated files (drop including content) using the `CASCADE` keyword with the `ALTER DISKGROUP ... DROP FILEGROUP` SQL statement or the `−r` option with the ASMCMD `rmfg` command.

`ALTER DISKGROUP` syntax:

```
ALTER DISKGROUP <diskgroup_name> DROP FILEGROUP <filegroup_name> [ CASCADE ]
   [ FOR [ PLUGGABLE DATABASE <pdb_name> ] DATABASE <db_unique_name> ]
```

Look here for more information regarding the ASMCMD command rmfg.

> ⓘ **Note**
>
> When a file group of type `DATABASE` is dropped using the `CASCADE` option, ASM checks the OMF (Oracle Managed File) file name of the files being deleted to verify that they belong to the database or pluggable database that is associated with the file group being dropped. This check helps prevent unintended deletion of database files when files are in the wrong file group due to a code bug or configuration issue.
> If for any reason ASM is unable to resolve the `DB_UNIQUE_NAME` or PDB name for a file group being dropped, the command will fail with ORA-59616. In such cases, if using the `ALTER DISKGROUP...DROP FILEGROUP` SQL statement, the user must specify the `FOR PLUGGABLE DATABASE` or `FOR DATABASE` clause in order to drop the file group.

To monitor Oracle ASM file groups, you can check the `V$ASM_FILEGROUP`, `V$ASM_FILEGROUP_PROPERTY`, and `V$ASM_FILEGROUP_FILE` views. You can also display information about Oracle ASM file groups with the ASMCMD `lsfg` command.

> ⓘ **See Also**
>
> - [About Quota Groups for Oracle ASM File Groups](#) for information about quota groups.
>
> - [ASMCMD File Group Management Commands](#) for information about ASMCMD commands to administer Oracle ASM File Groups
>
> - *Oracle AI Database Reference* for information about the `V$ASM*` dynamic performance views

About Oracle ASM File Group Properties

The following list summarizes the properties of file groups. You can create a customized a set of file group properties in a file group template that can be inherited by a number of databases. For more information, refer to [File Group Property Templates](#).

- `COMPATIBLE.CLIENT`

  The current value of this file group property is the last known client compatibility during a file open operation.

  The `COMPATIBLE.RDBMS` disk group attribute must be set less than or equal to:

  — `COMPATIBLE.CLIENT` of each file group (when set)

  — `COMPATIBLE` database initialization parameter (for each database that is connected to Oracle ASM)

  — `COMPATIBLE.ASM` of the disk group

  You must be an Oracle ASM administrator to modify this property.

- `DBCLONE_STATUS`

  This file group property specifies the status of the database split operation. This property cannot be set manually.

  During the split operation of the database associated with this file group, the file group property contains one of the following values.

  — `PREPARING`

    The process to prepare the database for splitting has started, but the database is not yet ready to be split.

  — `PREPARED`

    The preparation process is complete and the database is ready to be split.

  — `SPLITTING`

    The process to split the database files has started.

  — `SPLIT COMPLETED`

    The database has been split.

  — `DROPPING`

The process to drop the prepared database clone has started.

– `FAILED`

The clone operation has failed. The rebalance operation should drop the incomplete clone.

- `OWNER`

This file group property specifies the `OWNER` setting. To set the `OWNER` property, the `ACCESS_CONTROL.ENABLED` disk group attribute must be set to `TRUE` to enable Oracle ASM File Access Control. The user specified as the owner must be a member of the disk group.

Only the Oracle ASM administrator, the `OWNER`, the `OWNER` of the `USER_GROUP` associated with the file group, and a member of the `USER_GROUP` associated with the file group can modify the attributes of file groups that are labeled with `A database administrator can modify this property`. This is true only if `ACCESS_CONTROL.ENABLED` is true.

You can display the owner with the `USER_NUMBER` and `USER_INCARNATION` columns of the `V$ASM_FILEGROUP` view. The values in `USER_NUMBER` and `USER_INCARNATION` can be joined with the `USER_NUMBER` and `INCARNATION` columns of the `V$ASM_USER` view to display the user names.

The default value is `NULL`.

You must be an Oracle ASM administrator to modify this property.

- `PARENT_FILEGROUP_NUMBER`

This file group property specifies the Parent File Group number. This property cannot be set manually.

When a database is prepared for split, a child file group is created for the shadow files. The child file group automatically has the `PARENT_FILEGROUP_NUMBER` property set.

- `POWER_LIMIT`

This file group property specifies the rebalance power, which determines the number of parallel relocations for disk rebalancing of this file group. The valid settings are `1` to `1024`. When rebalancing a file group, the power limit value is applied as follows:

1. If the power option has been explicitly specified in the `ALTER DISKGROUP REBALANCE` SQL statement or the ASMCMD `rebal` command, then the power value specified in the SQL rebalance statement or ASMCMD rebalance command is used for rebalancing all file groups.

2. If the power option has not been specified in the SQL rebalance statement or ASMCMD rebalance command, then the file group `POWER_LIMIT` property value, if set, applies for the file group rebalance.

3. If the power option has not been specified in the SQL rebalance statement or ASMCMD rebalance command, and the `POWER_LIMIT` property has not been set for the file group, then the power value defaults to the value of the `ASM_POWER_LIMIT` initialization parameter for the file group rebalance.

You must be a database administrator to modify this property.

- `PRIORITY`

This file group property specifies the rebalance priority and ensures that certain types of files are rebalanced before others. File groups with the highest priority are completely rebalanced first, then file groups with the next highest priority, and so on to the lowest priority.

You can set the rebalance priority setting to `HIGHEST`, `HIGH`, `MEDIUM`, `LOW`, or `LOWEST`. The default value is `MEDIUM`.

You must be a database administrator to modify this property.

- `QUOTA_GROUP`

  This file group property specifies the name of the quota group to which this file group belongs. The valid setting is an existing quota group name. The default value is `GENERIC`. Every file group in the disk group must be a member of a quota group.

  You must be an Oracle ASM administrator to modify this property.

- `REDUNDANCY`

  This file type property specifies the redundancy for the file group. The default redundancy setting for each file type is derived from the system templates. The valid settings are:

  – `HIGH`

    `HIGH` redundancy provides three-way mirroring for all files.

  – `MIRROR`

    `MIRROR` redundancy provides two-way mirroring for all files.

  – `PARITY`

    To reduce the overhead of data protection associated with Oracle ASM, parity protection is provided for FLEX redundancy disk groups. Parity redundancy is restricted to data not requiring ongoing updating to reduce the execution impact with parity protection. The `PARITY` setting is intended for write-once files, such as archive logs and backup sets and is not supported on data files and read/write files.

  – `DOUBLE`

    `DOUBLE` specifies double parity and tolerates double failures for FLEX redundancy disk groups.

  – `UNPROTECTED`

    `UNPROTECTED` does not provide mirroring by Oracle ASM.

  If a file is created as `UNPROTECTED`, `MIRROR`, or `HIGH` redundancy, its redundancy can change to `UNPROTECTED`, `MIRROR`, or `HIGH`. If redundancy has been changed, then the `REMIRROR` column of `V$ASM_FILE` contains `Y` to indicate that the file needs new mirroring, initiating a rebalance to put the new redundancy into effect. After the rebalance completes, the value in the `REMIRROR` column contains `N` .

  `PARITY` and `DOUBLE` redundancy are supported only on FLEX redundancy disk groups (not EXTENDED disk groups).

  When a file is created with `PARITY` or `DOUBLE` redundancy, that file can never change redundancy.

  `PARITY` requires a minimum of three regular (not quorum) failure groups in the flex disk group. `DOUBLE` requires a minimum of five regular (not quorum) failure groups in the flex disk group.

  – If there are three or more failure groups when the parity file is created, then each parity extent set has 2+1 single parity. This scenario incurs 50% redundancy overhead rather than 100% redundancy overhead for NORMAL redundancy (two-way mirror files).

  – If there are five or more failure groups when the parity file is created, then each parity extent set has 2+2 double parity. Double parity requires at least 5 failure groups. This

scenario incurs 100% redundancy overhead rather than 200% redundancy overhead for HIGH redundancy (three-way mirror files).

When the file group redundancy property is modified from a `HIGH`, `MIRROR`, or `UNPROTECTED` setting to a `PARITY` or `DOUBLE` setting, the redundancy of the existing files in the file group does not change. This behavior also applies to a change from `PARITY` or `DOUBLE` to a `HIGH`, `MIRROR`, or `UNPROTECTED` setting. However, any files created in the future adopt the new redundancy setting.

Changing the file group redundancy may not modify the redundancy of files that are being created concurrently when the `ALTER DISKGROUP MODIFY FILEGROUP` SQL command is issued. To complete the redundancy change, issue the `ALTER DISKGROUP MODIFY FILEGROUP` SQL command again after those files have been created.

If the redundancy in the file group is increased, then the value of the `REDUNDANCY` column of `V$ASM_FILE` is increased after rebalance completes. However, if the redundancy in the file group is decreased, then the value of the `REDUNDANCY` column of `V$ASM_FILE` is lowered immediately before rebalance is run.

Changing the redundancy without specifying a file type affects all file types.

You must be a database administrator to modify this property.

- `STRIPING`

This file type property specifies the `STRIPING` setting for each file type. Usually the default value for each file type is sufficient and is not changed.

Most files have a `COARSE` stripe, except for the control file which has a `FINE` stripe by default. The valid settings are `COARSE` and `FINE`. The default value is `COARSE` or derived from the file type template.

`STRIPING` affects only newly-created files and cannot be changed after the file is created.

You must be a database administrator to set this property.

- `USER_GROUP`

This file group property specifies the `USER_GROUP` setting. To set the `USER_GROUP` property, the `ACCESS_CONTROL.ENABLED` disk group attribute must be set to `TRUE` to enable Oracle ASM File Access Control. The user group specified must be an existing user group.

Only the Oracle ASM administrator, the `OWNER`, the `OWNER` of the `USER_GROUP` associated with the file group, and a member of the `USER_GROUP` associated with the file group can modify the attributes of file groups that are labeled with `A database administrator can modify this property`. This is true only if `ACCESS_CONTROL.ENABLED` is true.

You can display the user group information with the `USERGROUP_NUMBER` and `USERGROUP_INCARNATION` columns of the `V$ASM_FILEGROUP` view. The values in `USERGROUP_NUMBER` and `USERGROUP_INCARNATION` can be joined with the `USERGROUP_NUMBER` and `INCARNATION` columns of the `V$ASM_USERGROUP` view to display the user group names.

Files created in a file group inherit the `USER_GROUP` property setting.

Clones inherit the `USER_GROUP` property setting.

The default value is `NULL`.

You must be an Oracle ASM administrator to modify this property.

Example 4-17   Setting a file group property for file types

The following SQL statements set the `REDUNDANCY` file group property for the `datafile` and `archivelog` file types.

```
ALTER DISKGROUP DATA MODIFY FILEGROUP DATAFG1_PDB1
          SET 'datafile.redundancy' = 'HIGH';

ALTER DISKGROUP DATA MODIFY FILEGROUP DATAFG1_PDB1
        SET 'archivelog.redundancy' = 'MIRROR';
```

Example 4-18   Setting the file group redundancy property for an ADVM volume

The following SQL statement sets the `REDUNDANCY` file group property for the Oracle ADVM volume. To enable Oracle ADVM volumes to change redundancy, both `COMPATIBLE.ASM` and `COMPATIBLE.ADVM` must be set to `18.1` or greater. For Oracle ADVM volumes, you cannot change the redundancy setting of individual file types.

```
SQL> ALTER DISKGROUP DATA MODIFY FILEGROUP volume1 SET 'redundancy'='high';
```

File Group Property Templates

You can create a file group template with the `ADD FILEGROUP` clause of `ALTER DISKGROUP` to customize a set of file group properties that can be inherited by multiple databases.

```
SQL> ALTER DISKGROUP data2 ADD FILEGROUP fg_template1 TEMPLATE SET
'datafile.redundancy'='unprotected';
```

You can customize a file group template similar to modifying a regular file group.

```
SQL> ALTER DISKGROUP data2 MODIFY FILEGROUP fg_template1 SET
'controlfile.redundancy'='high';
```

Multiple file groups can be created from a file group template to inherit the properties in the template. Using a file group template automatically creates a new file group for the database if one is not already created, and the new file group inherits the properties of the file group template.

```
SQL> ALTER DISKGROUP data2 ADD FILEGROUP fg_db DATABASE NONE FROM TEMPLATE fg_template2;
```

> ⓘ **Note**
>
> You can specify the name of an Oracle ASM file group as part of a file name. For example:
>
> +*diskgroup_name*(fg$*file_group_name*)/*file_name*
>
> Similar syntax is used for indicating the name of a file group rather than a template.

Multiple tablespaces or databases can be created to inherit customized properties from a file group template. In the example, the `fg$` prefix in `fg$fg_template2` is a fixed string that indicates the name of a file group rather than the name of a template.

```
SQL> CREATE TABLESPACE tbs1 datafile '+data2(fg$fg_template2)/dbs/tbs1.f' size 1M;
```

> **ⓘ See Also**
>
> - [What Types of Files Does Oracle ASM Support?](#) for a list of Oracle ASM file types
>
> - [Managing Oracle ASM File Access Control for Disk Groups](#) for information about Oracle ASM File Access Control
>
> - *Oracle AI Database Reference* for information about file group properties displayed in the V$ASM_FILEGROUP_PROPERTY and V$ASM_FILEGROUP dynamic performance views

About Quota Groups for Oracle ASM File Groups

A quota group defines the quota allocated to a group of Oracle ASM File Groups. A file group belongs to only one quota group. For information about file groups, refer to [About Oracle ASM File Groups](#).

A quota can describe an aggregate of space used by different databases.

The following list provides important notes about quota groups for file groups.

- A file group can belong to only one quota group.

- A quota group cannot span multiple disk groups.

- A quota group describes an aggregate of space used by one file group or multiple file groups in the same disk group.

  A file group for a PDB and a file group for a clone of that PDB could be a scenario where file groups would share a common quota group.

  A file group clone by default inherits the parent's quota group. However, the clone could belong to a different quota group.

- A quota is enforced when a file is created and resized.

- A quota is a physical space.

  As a consequence, if the quota limit has been set to 10 M, then a two-way mirrored 6 M file exceeds the quota.

- Each quota group has two values: the limit and the current used space.

  The limit value can be set below the current used space. This action prevents any additional space being allocated for files described by file groups associated with this quota group.

- A file group can be moved from one quota group to another, regardless whether or not the target quota group has enough space for the file group.

  This capability enables an Oracle ASM administrator to stop any files described by a specific file group from allocating additional space.

You can administer quota groups with SQL and ASMCMD commands. The following examples illustrate how SQL statements are used to add a quota group to a disk group and how to move a file group from its current quota group to a new quota group.

Example 4-19   Adding a Quota Group to a Disk Group

```
ALTER DISKGROUP DATA ADD QUOTAGROUP Quota_QGRP1
                SET 'quota'= 10m;
```

Example 4-20   Moving a File Group to a new Quota Group

```
ALTER DISKGROUP DATA MODIFY FILEGROUP DATAFG1_PDB1
               SET 'quota_group' = 'Quota_QGRP2';
```

About Oracle ASM Extended Disk Groups

An Oracle ASM extended disk group has all the features of a flex disk group, with a specific design to be highly available in an extended (stretch) cluster environment, which contains nodes that span multiple physically separated sites.

The following are characteristics of an extended disk group:

- The redundancy setting of an extended disk group is set to `EXTENDED REDUNDANCY` and each file group of the extended disk group has its own redundancy property setting.

- The redundancy of files and file groups in an extended disk group is specified for each site, rather than for each disk group.

  This means that the number of copies determined by the `REDUNDANCY` setting is multiplied by the number of data sites. If the `REDUNDANCY` is set to `MIRROR`, then that specifies two copies, resulting in four copies total for two data sites.

- An extended disk group can tolerate the loss of an entire site, plus the loss of up to two failure groups in another site.

  This means that an extended disk group can tolerate failures at the site level in addition to the failure group level.

- All data sites must have the same number of failure groups during disk group creation.

- A quota group is specified for each disk group, rather than for each site.

- The quota group limit is the physical space required for all copies across all sites.

  For example, with two data sites, a 6 M file with redundancy set to `MIRROR` uses 24 M of the quota limit.

- There must be a total of three sites, two data sites and one quorum site, to create an extended disk group. Each data site should have three failure groups, and the quorum site should have one failure group.

- For database files, the redundancy setting is dictated by the redundancy setting of the file groups of the extended disk group.

- The `COMPATIBLE.ASM` and `COMPATIBLE.RDBMS` disk group attributes must be set to `12.2` or greater.

- The minimum allocation unit (AU) size is 4 M.

- Converting from another disk group type to an extended disk group is not supported.

- The `SITE` specification for failure groups can be inferred from cells if Oracle Exadata has been configured. Otherwise, the site specification can be specified with the SQL statement when creating a disk group.

> ⓘ **Note**
>
> The `REQUIRED_MIRROR_FREE_MB` and `USABLE_FILE_MB` columns in the `V$ASM_DISKGROUP` view report no values for an extended disk group.

The following example illustrates how to create an extended disk group with a SQL statement.

Example 4-21   Using CREATE DISKGROUP with extended disk groups

```
SQL> CREATE DISKGROUP extended_site_data EXTENDED REDUNDANCY
        SITE NY FAILGROUP fg1 DISK '/devices/disks/disk01'
                FAILGROUP fg2 DISK '/devices/disks/disk02'
                FAILGROUP fg3 DISK '/devices/disks/disk03'
        SITE NJ FAILGROUP fg4 DISK '/devices/disks/disk04'
                FAILGROUP fg5 DISK '/devices/disks/disk05'
                FAILGROUP fg6 DISK '/devices/disks/disk06'
        SITE QM QUORUM
                FAILGROUP fg7 DISK '/devices/disks/disk07';
```

About Point-In-Time Database Cloning With Oracle ASM Flex Disk Groups

Before creating a point-in-time database clone, review the following requirements.

> ⓘ **Note**
>
> - Point-in-time database clones are supported only in Oracle ASM flex and extended disk groups.
>
> - Point-in-time database clones are supported only with Oracle Database 18c, version 18.1 or higher.
>
> - Point-in-time database clones require disk group compatibility attributes `COMPATIBLE.ASM` and `COMPATIBLE.RDBMS` be set to `18.0` or higher.
>
> - The source database (parent) must be a pluggable database. The database clone (child) created is a pluggable database under the same container database or in a different container database.
>
> - If Point-in-time clone is done using mirror copy requires database clone to have access to the ASM flex diskgroup. Cloning between two different ASM clusters (e.g using database link) is not supported.

Oracle ASM provides support for creating a point-in-time database clone which is a mirror copy of a pluggable database. All the data files of the database being cloned should be present in a single disk group. The cloning operation can be used for rapid configuration of identical copies. You can specify the redundancy setting of the database data files in the database clone, or the setting can default to the redundancy of the data files in the source database that is being the cloned.

The copy of data files created during the cloning process is initially identical to the source. However, after the cloning process has completed, updates to the source do not result in any change to the child that has been cloned. The `V$ASM_DBCLONE_INFO` view shows the relationship between the source database, the cloned database, and their file groups. The `V$ASM_DBCLONE_INFO` should be run on the Oracle ASM instance.

In the examples shown in this topic, the source database is present on the `DATA` flex disk group.

Database cloning has these steps:

1. Preparing the Mirrored Copy

   During this step, Oracle ASM allocates space for the additional copies of data. This process involves creating the cloned files and linking them with the source files. However,

the data is not copied at this time; the copying is done during remirroring. Remirroring occurs during the prepare phase of rebalance, which is initiated as part of this step. Only one mirror copy can be in existence at any time. All the data files of the given database or the pluggable database should reside on a single disk group. The mirror copy must be used to create a database clone before another mirror copy can be created.

A file group is created and all the cloned files are made part of this file group. The redundancy of the database data files in the file group is set to the value that you specify, or the value defaults to the redundancy setting of the source file group. The quota group associated with the clone is the same as that of the source database.

Oracle ASM File Access Control properties associated with the clone are the same as those associated with the source database. For example, the file access control user group property for the database clone is same as that of the source database.

At the end of this step and before you start the next step, you can drop the prepared mirror copy, as described in <u>Dropping the Prepared Mirror Copy</u>.

2. Splitting the Mirrored Copy and Creating the Database Clone

The prepare and copy step must complete before starting this step. You can display the `V$ASM_DBCLONE_INFO` view on the Oracle ASM instance to query the status of the ongoing prepare operation. The `DBCLONE_STATUS` column of the `V$ASM_DBCLONE_INFO` view indicates `PREPARED` when the file group is ready to be split.

The process initiated with `CREATE PLUGGABLE DATABASE` breaks the relationship between the source database files and the cloned database files.

Preparing the Mirrored Copy to Create a PDB in the Same Container Database

The following example prepares the mirrored copy and creates a file group on the `DATA` disk group. Before you issue the `ALTER PLUGGABLE DATABASE PREPARE` SQL command, you must connect to the source pluggable database (`pdb1` in the examples) that is being prepared for cloning.

```
SQL>  SHOW CON_NAME
CON_NAME
------------------------------
CDB$ROOT

SQL>  ALTER SESSION SET CONTAINER = pdb1;
Session altered.

SQL> SHOW CON_NAME
CON_NAME
------------------------------
PDB1

SQL> ALTER PLUGGABLE DATABASE PREPARE MIRROR COPY pdb1_mirror;
Pluggable database altered.
```

You can query the `V$ASM_DBCLONE_INFO` view on the Oracle ASM instance to check the status of the process.

```
SQL> SELECT * FROM V$ASM_DBCLONE_INFO;
GROUP_NUMBER DBCLONE_NAME MIRRORCOPY_NAME DBCLONE_STATUS PARENT_DBNAME PARENT_FILEGROUP_NAME CON_ID
------------ ------------ --------------- -------------- ------------- --------------------- ------
           1 DB_UNKNOWN   PDB1_MIRROR     PREPARED       ORCL_PDB1     ORCL_PDB1                  0
```

Preparing the Mirrored Copy to Create a PDB in a Different Container Database

The following example prepares the mirrored copy and creates a file group on the `DATA` disk group, to be used by a different container database. Before you issue the `ALTER PLUGGABLE DATABASE PREPARE` SQL command, you must connect to the source pluggable database (`pdb1` in the examples) that is being prepared for cloning.

```
SQL>  SHOW CON_NAME
CON_NAME
------------------------------
CDB$ROOT

SQL>  ALTER SESSION SET CONTAINER = pdb1;
Session altered.

SQL> SHOW CON_NAME
CON_NAME
------------------------------
PDB1

SQL> ALTER PLUGGABLE DATABASE PREPARE MIRROR COPY pdb2_mirror FOR DATABASE new_cdb;
Pluggable database altered.
```

You can query the `V$ASM_DBCLONE_INFO` view on the Oracle ASM instance to check the status of the process.

```
SQL> SELECT * FROM V$ASM_DBCLONE_INFO;
GROUP_NUMBER DBCLONE_NAME    MIRRORCOPY_NAME DBCLONE_STATUS PARENT_DBNAME PARENT_FILEGROUP_NAME CON_ID
------------ -------------- --------------- -------------- ------------- -------------------- ------
           1 FOR$DB$NEW_CDB PDB1_MIRROR     PREPARED       ORCL_PDB1     ORCL_PDB1                 0
```

Splitting the Mirrored Copy and Creating the Database Clone in the Same Container Database

The following example creates the `newpdb1` database clone using the mirror copy `pdb1_mirror`. `newpdb1` is a clone of `pdb1` pluggable database and is plugged into the CDB. The `CREATE PLUGGABLE DATABASE` SQL command must be run in the root container of the CDB.

```
SQL>  SHOW CON_NAME
CON_NAME
------------------------------
CDB$ROOT

SQL> CREATE PLUGGABLE DATABASE newpdb1 FROM pdb1 USING MIRROR COPY pdb1_mirror;
Pluggable database created.

SQL>  ALTER SESSION SET CONTAINER = newpdb1;
Session altered.

SQL> SHOW CON_NAME
CON_NAME
------------------------------
NEWPDB1
```

You can query the `V$ASM_DBCLONE_INFO`, `V$ASM_FILEGROUP`, and `V$ASM_FILEGROUP_PROPERTY` views on the Oracle ASM instance for additional information.

```
SQL> SELECT * FROM V$ASM_DBCLONE_INFO;
GROUP_NUMBER DBCLONE_NAME   MIRRORCOPY_NAME DBCLONE_STATUS  PARENT_DBNAME PARENT_FILEGROUP_NAME CON_ID
------------ ------------- --------------- --------------- ------------- -------------------- ------
           1 ORCL_NEWPDB1  PDB1_MIRROR     SPLIT COMPLETED ORCL_PDB1     ORCL_PDB1                 0
```

```
SQL> SELECT NAME, GROUP_NUMBER, FILEGROUP_NUMBER FROM V$ASM_FILEGROUP;
NAME                                 GROUP_NUMBER FILEGROUP_NUMBER
------------------------------------ ------------ ----------------
DEFAULT_FILEGROUP                               1                0
ORCL_CDB$ROOT                                   1                1
ORCL_PDB$SEED                                   1                2
ORCL_PDB1                                       1                3
PDB1_MIRROR                                     1                4


SQL> SELECT NAME, VALUE FROM V$ASM_FILEGROUP_PROPERTY WHERE FILEGROUP_NUMBER = 4;
NAME                                 VALUE
------------------------------------ ------------------------------------
PRIORITY                             MEDIUM
PARENT_FILEGROUP_NUMBER              3
DBCLONE_STATUS                       SPLIT COMPLETED
COMPATIBLE.CLIENT                    18.0.0.0.0
REDUNDANCY                           MIRROR
STRIPING                             FINE
...
```

Splitting the Mirrored Copy and Creating the Database Clone in a Different Container Database

The following example creates the `newpdb2` database clone using the mirror copy `pdb2_mirror`. `newpdb2` is a clone of `pdb1` pluggable database and is plugged into a different CDB `new_cdb`. The `CREATE PLUGGABLE DATABASE` SQL command must be run in the root container of `new_cdb`.

Before running the `CREATE PLUGGABLE DATABASE` command, you must create a database link that can connect to the source CDB, from where the prepare command was issued. The `CREATE PLUGGABLE DATABASE` SQL command must be run in `CDB$ROOT`. The database user that is used by the database link to connect to the source database must have the `CREATE PLUGGABLE DATABASE` system privilege, either granted commonly or granted locally in that application container, to run the command.

The following commands must be run in the source CDB.

```
SQL> SELECT NAME FROM V$DATABASE;
NAME
---------
SRC_CDB

SQL> SHOW CON_NAME
CON_NAME
------------------------------
CDB$ROOT

SQL> CREATE USER c##dbl_user IDENTIFIED BY password;
User created.

SQL> GRANT CREATE SESSION, CREATE PLUGGABLE DATABASE TO c##dbl_user CONTAINER=all ;
Grant succeeded.
```

The following commands must be run in the target CDB.

```
SQL> SELECT NAME FROM V$DATABASE;
NAME
---------
NEW_CDB

SQL> SHOW CON_NAME
```

```
                         CON_NAME
                         ------------------------------
                         CDB$ROOT

                         SQL> CREATE DATABASE LINK cdb1_dblink CONNECT TO c##dbl_user IDENTIFIED BY password
                         USING tns_alias_to_src_cdb;
                         Database link created.

                         SQL> CREATE PLUGGABLE DATABASE newpdb2 FROM pdb1@cdb1_dblink USING MIRROR COPY
                         pdb2_mirror;
                         Pluggable database created.

                         SQL> ALTER SESSION SET CONTAINER = newpdb2;
                         Session altered.

                         SQL> SHOW CON_NAME
                         CON_NAME
                         ------------------------------
                         NEWPDB2
```

You can query the `V$ASM_DBCLONE_INFO`, `V$ASM_FILEGROUP`, and `V$ASM_FILEGROUP_PROPERTY` views on the Oracle ASM instance for additional information.

```
SQL> SELECT * FROM V$ASM_DBCLONE_INFO;
GROUP_NUMBER DBCLONE_NAME  MIRRORCOPY_NAME DBCLONE_STATUS  PARENT_DBNAME PARENT_FILEGROUP_NAME CON_ID
------------ ------------- --------------- --------------- ------------- --------------------- ------
           1 ORCL_NEWPDB2  PDB2_MIRROR     SPLIT COMPLETED ORCL_PDB1     ORCL_PDB1                  0

SQL> SELECT NAME, GROUP_NUMBER, FILEGROUP_NUMBER FROM V$ASM_FILEGROUP;
NAME                                GROUP_NUMBER FILEGROUP_NUMBER
----------------------------------- ------------ ----------------
DEFAULT_FILEGROUP                              1                0
ORCL_CDB$ROOT                                  1                1
ORCL_PDB$SEED                                  1                2
ORCL_PDB1                                      1                3
PDB2_MIRROR                                    1                4


SQL> SELECT NAME, VALUE FROM V$ASM_FILEGROUP_PROPERTY WHERE FILEGROUP_NUMBER = 4;
NAME                                VALUE
----------------------------------- ------------------------------------
PRIORITY                            MEDIUM
PARENT_FILEGROUP_NUMBER             3
DBCLONE_STATUS                      SPLIT COMPLETED
COMPATIBLE.CLIENT                   18.0.0.0.0
REDUNDANCY                          MIRROR
STRIPING                            FINE
...
```

Dropping the Prepared Mirror Copy

After preparing the mirrored copy, and before splitting the mirrored copy and creating the database clone, you can drop the prepared mirror copy.

The drop action discards all the changes performed by the `ALTER PLUGGABLE DATABASE PREPARE` SQL command. Similar to the `PREPARE` SQL operation, the drop action also triggers a rebalance on the appropriate disk group. You can use the `V$ASM_DBCLONE_INFO` view to query the status of the ongoing drop operation. The `DBCLONE_STATUS` column of the `V$ASM_DBCLONE_INFO` view indicates `DROPPING` when the file group is being dropped and the entry is removed after all changes are discarded.

The following example illustrates how to drop the prepared mirror copy when you are connected to the source pluggable database (`pdb1` in the example).

```
SQL> ALTER SESSION SET CONTAINER = pdb1;
Session altered.

SQL> ALTER PLUGGABLE DATABASE DROP MIRROR COPY pdb1_mirror;
Pluggable database altered.
```

Cleaning Up After Failed Attempts

The `DBCLONE_STATUS` column of the `V$ASM_DBCLONE_INFO` view indicates `FAILED` when the point-in-time clone process is unsuccessful. If you experience a failed attempt to create a point-in-time clone, then you can resolve the problem by issuing a rebalance on the disk group to clean up the file group. For example, after connecting to the Oracle ASM instance, you can run the following:

```
SQL> ALTER DISKGROUP data REBALANCE WAIT;
```

> ⓘ **See Also**
>
> - *Oracle Multitenant Administrator's Guide* for information about multitenant databases
> - *Oracle AI Database SQL Language Reference* for information about the `CREATE PLUGGABLE DATABASE` SQL statement
> - *Oracle AI Database SQL Language Reference* for information about the `CREATE DATABASE LINK` SQL statement
> - *Oracle AI Database Reference* for information about the `V$ASM_DBCLONE_INFO` view

5
Administering Oracle ASM Files, Directories, and Templates

Administration of Oracle ASM includes the management of 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 File Names](#)
- [Creating and Referencing Oracle ASM Files in the Database](#)
- [Managing Alias Names for Oracle ASM File Names](#)
- [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 5-1 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 5-1   File types supported by Oracle ASM

File Type	Default Templates
Archive log files	`ARCHIVELOG`
Oracle ASM Persistent initialization parameter file (SPFILE)	`ASMPARAMETERFILE`
Oracle ASM Persistent initialization parameter file (SPFILE) backup	`ASMPARAMETERFILEBACKUP`
Audit spill files	`AUDIT_SPILLFILES`
Automatically generated control file backup	`AUTOBACKUP`
Auto-login software keystore	`AUTOLOGIN_KEY_STORE`
Archive log backup piece	`BACKUPSET`
Data file backup pieces	`BACKUPSET`
Data file incremental backup pieces	`BACKUPSET`
Change tracking file	`CHANGETRACKING`
Control files	`CONTROLFILE`

Table 5-1   (Cont.) File types supported by Oracle ASM

File Type	Default Templates
Data file copy	`DATAFILE`
Data files	`DATAFILE`
Data Guard configuration file	`DATAGUARDCONFIG`
Data Pump dumpset	`DUMPSET`
Flashback logs	`FLASHBACK`
Flash file	`FLASHFILE`
Cross-platform transportable data files incremental backup	`INCR XTRANSPORT BACKUPSET`
Software keystore	`KEY_STORE`
Oracle Cluster Registry file	`OCRFILE`
Redo log files	`ONLINELOG`
Persistent initialization parameter file (SPFILE)	`PARAMETERFILE`
Temporary files	`TEMPFILE`
Voting files	`VOTINGFILE`
Cross-platform transportable data files	`XTRANSPORT`
Cross-platform transportable data files backup	`XTRANSPORT BACKUPSET`
Oracle ASM Dynamic Volume Manager volumes	n/a
PASSWORD file	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 Advanced Cluster File System (Oracle ACFS) and Oracle ASM Dynamic Volume Manager (Oracle ADVM) extend Oracle ASM support to include database and application executables, database data files, 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 Also**
>
> - Managing Disk Group Templates for a description of the system default templates
> - lstmpl for an example of the output of the ASMCMD `lstmpl` command
> - *Oracle Advanced Cluster File System Administrator's Guide* for more information about Oracle ACFS and Oracle ADVM, including files supported by Oracle ACFS

About Oracle ASM File Names

Every file created in Oracle ASM is provided a file name generated by Oracle Managed Files, known as a fully qualified file name (or system-generated file name). The fully qualified file name represents a complete path name in the Oracle ASM file system. An example of a fully qualified file name is:

```
+data/orcl/controlfile/Current.256.541956473
```

You can use the fully qualified file name to reference (read or retrieve) an Oracle ASM file. You can also use other abbreviated file name formats, such as an alias Oracle ASM file name, to reference an Oracle ASM file.

Oracle ASM generates a fully qualified file name upon any request to create a file. A creation request does not or cannot specify a fully qualified file name. 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 file name. If you specify an alias in the creation request, Oracle ASM also creates the alias so that it references the fully qualified file name.

Oracle ASM file creation requests are either single file creation requests or multiple file creation request.

> ⓘ **Note**
>
> - 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 AI Database file names, Oracle ASM file names are kept in the control file and the RMAN catalog.
>
> - You can use fully qualified and numeric file names in single-file creation if you specify the `REUSE` keyword, as described in Using Oracle ASM File Names in SQL Statements.
>
> - Forward slash (/) and backslash (\) are interchangeable in file names. File names are case insensitive, but case retentive.

Table 5-2 specifies the valid usage for each file name form. If the form is used for file creation, the table specifies whether the created file is an Oracle Managed Files (OMF).

Table 5-2   Valid usage for Oracle ASM file name forms

File Name 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

This section contains the following topics:

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 file name 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 file name 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 file name 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 file name 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 file name (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 file name 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.

> ⓘ **Note**
>
> A multitenant container database is the only supported architecture in Oracle Database 21c and later releases. While the documentation is being revised, legacy terminology may persist. In most cases, "database" and "non-CDB" refer to a CDB or PDB, depending on context. In some contexts, such as upgrades, "non-CDB" refers to a non-CDB from a previous release.

A fully qualified file name has the following form in a non-multitenant environment:

```
+diskgroup/dbname/filetype/filetypetag.file.incarnation
```

With multitenant databases, a fully qualified file name has the following form:

`+diskgroup/cdb/pdb_guid/filetype/filetypetag.file.incarnation`

The fields of the fully qualified file name are:

- `+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 5-3.

- `filetypetag` is type-specific information about the file and can be one of the tags shown in Table 5-3.

- `file.incarnation` is the file/incarnation pair, used to ensure uniqueness.

- `cdb` is the name of the CDB

- `pdb_guid` is the PDB GUID value.

Examples of fully qualified Oracle ASM file names are:

```
SQL> SELECT NAME FROM V$DATAFILE;

NAME
--------------------------------------------------------------------------
+DATA/ORCL/DATAFILE/system.257.967544915
+DATA/ORCL/DATAFILE/sysaux.258.967544961
+DATA/ORCL/DATAFILE/undotbs1.259.967544985
+DATA/ORCL/64A52F53A7693286E053CDA9E80AED76/DATAFILE/system.267.967545119
+DATA/ORCL/64A52F53A7693286E053CDA9E80AED76/DATAFILE/sysaux.266.967545119
+DATA/ORCL/DATAFILE/users.260.967544987
...
```

Table 5-3   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
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

Table 5-3   (Cont.) Oracle file types and Oracle ASM file type tags

Oracle ASM File Type	Description	Oracle ASM File Type Tag	Comments
`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 File Name Forms

You can use Alias Oracle ASM file names 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 file names 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 file name, the file is created with a fully qualified name, and the alias file name is additionally created. You can then access the file with either name.

Alias Oracle ASM file names are distinguished from fully qualified file names or numeric file names 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 file names are:

```
+data/orcl/control_file1
+fra/recover/second.dbf
```

Oracle AI Database references database files by their alias file names, 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 file names. The following are examples of how the database uses alias file names:

- Alias file names appear in V$ views. For example, if you create a tablespace and use an alias file name for the data file, the `V$DATAFILE` view shows the alias file name.

- When a control file points to data files and online redo log files, it can use alias file names.

- The `CONTROL_FILES` initialization parameter can use the alias file names of the control files. Database Configuration Assistant (DBCA) creates control files with alias file names.

> ⓘ **Note**
>
> Files created using an alias file name 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 file names, see [Managing Alias Names for Oracle ASM File Names](#).

Creating a Tablespace in Oracle ASM: Using a Data File with an Alias Name

[Example 5-1](#) 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.

Because an alias is used when creating 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';
```

Example 5-1   Creating a tablespace using an alias name

```
CREATE UNDO TABLESPACE myundo
    DATAFILE '+data(my_undo_template)/orcl/my_undo_ts' SIZE 200M;
```

Alias Oracle ASM File Name with Template Form

An alias Oracle ASM file name with template is used only for Oracle ASM file creation operations. It has the following format:

```
+diskgroup(templatename)/alias
```

Alias file names with template behave identically to alias file names except that a file created with an alias file name 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.

An example of an alias Oracle ASM file name 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 file name with template are not considered Oracle
> Managed Files and might require manual deletion in the future if they are no longer
> needed.

> ⓘ **See Also**
>
> - [Example 5-1](#) for information about creating objects using an alias name
> - [Managing Disk Group Templates](#) for information about the creation and
>   maintenance of Oracle ASM templates

Incomplete Oracle ASM File Name Form

Incomplete Oracle ASM file names 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 file name 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](#) for more information.

Incomplete Oracle ASM File Name with Template Form

Incomplete Oracle ASM file names 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 file name 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 File Names in SQL Statements](#)
- [Moving Data Files Between Disk Groups With ALTER DATABASE](#)
- [Oracle AI Database File Mapping for Oracle ASM Files](#)

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 file name, 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 file names 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 file names 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 AI Database Reference* for information about initialization parameters
> - *Oracle AI Database Administrator's Guide* for information about initialization parameter files

Using Oracle ASM File Names in SQL Statements

You can specify Oracle ASM file names 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 file name. If the Oracle ASM file exists, then you must use the reference form of the file name, 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.

If a reference form is used with the `REUSE` keyword and the file does not exist, an error results.

Example 5-2 is an example of specifying an Oracle ASM file name in a SQL statement. In this case, it is used in the file creation form.

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 file name in SQL statements, see Creating Database Files in Oracle ASM Using SQL*Plus.

Example 5-2   Using an Oracle ASM File name in a SQL statement

```
CREATE TABLESPACE mytblspace DATAFILE '+data' SIZE 200M AUTOEXTEND ON;
```

Moving Data Files Between Disk Groups With ALTER DATABASE

You can use the `ALTER DATABASE MOVE DATAFILE` SQL statement to move data files between disk groups while the database is open and users are accessing the data files.

Example 5-3 shows how you can move a data file from the `DATA` disk group to the `DATA2` disk group.

Example 5-3   Moving data files online with ALTER DATABASE MOVE DATAFILE

```
/* Create a new disk group DATA2 using ASMCA, ASMCMD, or SQL */
/* Then create appropriate directories in the DATA2 disk group */

ALTER DISKGROUP data2 ADD DIRECTORY '+DATA2/ORCL';

ALTER DISKGROUP data2 ADD DIRECTORY '+DATA2/ORCL/DATAFILE';

/* Move the EXAMPLE data file in DATA to EXAMPLE_STORAGE in DATA2 */
ALTER DATABASE MOVE DATAFILE '+DATA/ORCL/DATAFILE/EXAMPLE.266.798707687'
     TO '+DATA2/ORCL/DATAFILE/EXAMPLE_STORAGE';
```

```
SELECT SUBSTR(name,1,64) FROM V$DATAFILE;

SUBSTR(NAME,1,64)
----------------------------------------------------------------
+DATA/ORCL/DATAFILE/system.258.798707555
+DATA2/ORCL/DATAFILE/example_storage
+DATA/ORCL/DATAFILE/sysaux.257.798707491
+DATA/ORCL/DATAFILE/undotbs1.260.798707621
+DATA/ORCL/DATAFILE/users.259.798707621
```

> ⓘ **See Also**
>
> - *Oracle AI Database Administrator's Guide* for more information about using the `ALTER DATABASE MOVE DATAFILE` SQL statement to move data files
>
> - *Oracle AI Database SQL Language Reference* for information about using the `ALTER DATABASE` SQL statement.
>
> - Moving Data Files Between Oracle ASM Disk Groups Using RMAN for information about using RMAN to move data files between disk groups

Oracle AI Database File Mapping for Oracle ASM Files

For an understanding of I/O performance, you must have detailed knowledge of the storage hierarchy in which files reside.

Oracle AI Database provides a set of dynamic performance views to show a complete mapping of a file to intermediate layers of logical volumes to actual physical devices. Using these dynamic views, you can locate the exact disk on which any block of a file resides.

Oracle AI Database communicates with a background process called `FMON` to manage the mapping information. Oracle provides the PL/SQL `DBMS_STORAGE_MAP` package to invoke mapping operations that populate the mapping views.

Oracle AI Database file mapping does not require third party libraries when mapping Oracle ASM files. In addition, Oracle AI Database supports Oracle ASM file mapping on all operating system platforms.

All the examples in this section are run on the Oracle AI Database instance.

This section contains the following topics:

- Enabling File Mapping With Oracle ASM Files

- Using File Mapping Views With Oracle ASM Files

- Using the DBMS_STORAGE_MAP Package With Oracle ASM Files

> ⓘ **See Also**
>
> - *Oracle Advanced Cluster File System Administrator's Guide* for more information about Oracle ACFS and Oracle ADVM
>
> - *Oracle AI Database Administrator's Guide* for additional information about Oracle AI Database file mapping

Enabling File Mapping With Oracle ASM Files

To enable file mapping, set the `FILE_MAPPING` initialization parameter to `TRUE`. The database instance does not have to be shut down to set this parameter. You can set the parameter using the following `ALTER SYSTEM` statement:

```
SQL> ALTER SYSTEM SET FILE_MAPPING=TRUE;
```

Run the appropriate `DBMS_STORAGE_MAP` mapping procedure.

- In a cold startup scenario, the Oracle AI Database has just started and no mapping operation has been invoked yet. You can run the `DBMS_STORAGE_MAP.MAP_ALL` procedure to build the mapping information for the entire I/O subsystem associated with the database.

  For example, the following command builds mapping information and provides for 10000 extents:

  ```
  SQL> EXECUTE DBMS_STORAGE_MAP.MAP_ALL(10000);
  ```

- In a warm start scenario where the mapping information has already been built, you have the option to invoke the `DBMS_STORAGE_MAP.MAP_SAVE` procedure to save the mapping information in the data dictionary.

  This procedure is invoked in `DBMS_STORAGE_MAP.MAP_ALL()` by default. This forces all of the mapping information in the SGA to be flushed to disk. The `DBMS_STORAGE_MAP.MAP_SAVE` procedure is invoked in `DBMS_STORAGE_MAP.MAP_ALL()` by default.

After you restart the database, use `DBMS_STORAGE_MAP.RESTORE()` to restore the mapping information into the SGA. If needed, `DBMS_STORAGE_MAP.MAP_ALL()` can be called to refresh the mapping information.

Using File Mapping Views With Oracle ASM Files

Mapping information generated by `DBMS_STORAGE_MAP` package is captured in dynamic performance views. These views include `V$MAP_COMP_LIST`, `V$MAP_ELEMENT`, `V$MAP_EXT_ELEMENT`, `V$MAP_FILE`, `V$MAP_FILE_EXTENT`, `V$MAP_FILE_IO_STACK`, `V$MAP_LIBRARY`, and `V$MAP_SUBELEMENT`.

For example, you can view file mappings with `V$MAP_FILE`.

```
SQL> SELECT FILE_MAP_IDX, SUBSTR(FILE_NAME,1,45), FILE_TYPE, FILE_STRUCTURE FROM V$MAP_FILE;

FILE_MAP_IDX SUBSTR(FILE_NAME,1,45)                        FILE_TYPE   FILE_STRU
------------ --------------------------------------------- ----------- ---------
           0 +DATA/ORCL/DATAFILE/system.258.841314095      DATAFILE    ASMFILE
           1 +DATA/ORCL/DATAFILE/sysaux.257.841314021      DATAFILE    ASMFILE
           2 +DATA/ORCL/DATAFILE/undotbs1.260.841314161    DATAFILE    ASMFILE
           3 +DATA/ORCL/DATAFILE/example.266.841314219     DATAFILE    ASMFILE
           4 +DATA/ORCL/DATAFILE/users.259.841314161       DATAFILE    ASMFILE
           5 +DATA/ORCL/ONLINELOG/group_3.264.841314209    LOGFILE     ASMFILE
           6 +FRA/ORCL/ONLINELOG/group_3.259.841314211     LOGFILE     ASMFILE
           7 +DATA/ORCL/ONLINELOG/group_2.263.841314207    LOGFILE     ASMFILE
           8 +FRA/ORCL/ONLINELOG/group_2.258.841314207     LOGFILE     ASMFILE
           9 +DATA/ORCL/ONLINELOG/group_1.262.841314203    LOGFILE     ASMFILE
          10 +FRA/ORCL/ONLINELOG/group_1.257.841314203     LOGFILE     ASMFILE
          11 +DATA/ORCL/TEMPFILE/temp.265.841314217        TEMPFILE    ASMFILE
          12 +DATA/ORCL/CONTROLFILE/current.261.841314199  CONTROLFILE ASMFILE
          13 +FRA/ORCL/CONTROLFILE/current.256.841314201   CONTROLFILE ASMFILE
```

> **ⓘ See Also**
>
> *Oracle AI Database Reference* for information about dynamic views

Using the DBMS_STORAGE_MAP Package With Oracle ASM Files

You can use the procedures in the `DBMS_STORAGE_MAP` PL/SQL package to control mapping operations. For example, you can use the `DBMS_STORAGE_MAP.MAP_OBJECT` procedure to build mapping information for the database object that is specified by object name, owner, and type. After the `DBMS_STORAGE_MAP.MAP_OBJECT` procedure is run, then you can create a query to display the mapping information contained in the `MAP_OBJECT` view.

```
SQL> EXECUTE DBMS_STORAGE_MAP.MAP_OBJECT('EMPLOYEES','HR','TABLE');

SQL> SELECT io.OBJECT_NAME o_name, io.OBJECT_OWNER o_owner, io.OBJECT_TYPE o_type,
       mf.FILE_NAME, me.ELEM_NAME, io.DEPTH,
      (SUM(io.CU_SIZE * (io.NUM_CU - DECODE(io.PARITY_PERIOD, 0, 0,
                         TRUNC(io.NUM_CU / io.PARITY_PERIOD)))) / 2) o_size
   FROM MAP_OBJECT io, V$MAP_ELEMENT me, V$MAP_FILE mf
   WHERE io.OBJECT_NAME =  'EMPLOYEES'
   AND   io.OBJECT_OWNER = 'HR'
   AND   io.OBJECT_TYPE =  'TABLE'
   AND   me.ELEM_IDX = io.ELEM_IDX
   AND   mf.FILE_MAP_IDX = io.FILE_MAP_IDX
   GROUP BY io.ELEM_IDX, io.FILE_MAP_IDX, me.ELEM_NAME, mf.FILE_NAME, io.DEPTH,
         io.OBJECT_NAME, io.OBJECT_OWNER, io.OBJECT_TYPE
   ORDER BY io.DEPTH;

O_NAME     O_OWNER O_TYPE FILE_NAME                                ELEM_NAME       DEPTH O_SIZE
--------- ------- ------ ---------------------------------------- ---------------- ----- ------
EMPLOYEES HR      TABLE  +DATA/ORCL/DATAFILE/example.266.841314219 +/devices/diskd2    0     64
EMPLOYEES HR      TABLE  +DATA/ORCL/DATAFILE/example.266.841314219 +/devices/diske2    0     64
```

> **ⓘ See Also**
>
> *Oracle AI Database PL/SQL Packages and Types Reference* for information about the `DBMS_STORAGE_MAP` package

Managing Alias Names for Oracle ASM File Names

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 file names.

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 (`+`).

This section contains the following sections:

- [Adding an Alias Name for an Oracle ASM File Name](#)
- [Renaming an Alias Name for an Oracle ASM File Name](#)

- [Dropping an Alias Name for an Oracle ASM File Name](#)
- [Dropping Files and Associated Aliases from a Disk Group](#)

> ⓘ **See Also**
>
> - *Oracle AI Database SQL Language Reference* for information about the `ALTER DISKGROUP` statement
> - [mkalias](#) for information about creating aliases with ASMCMD

Adding an Alias Name for an Oracle ASM File Name

Use the `ADD ALIAS` clause of the `ALTER DISKGROUP` statement to create an alias name for an Oracle ASM file name. The alias name must consist of the full directory path and the alias itself.

[Example 5-4](#) adds a new alias name for a system-generated file name.

Example 5-4   Adding an alias name for an Oracle ASM file name

```
ALTER DISKGROUP data ADD ALIAS '+data/orcl/second.dbf'
     FOR '+data/orcl/datafile/mytable.342.123456789';
```

Renaming an Alias Name for an Oracle ASM File Name

Use the `RENAME ALIAS` clause of the `ALTER DISKGROUP` statement to rename an alias for an Oracle ASM file name. The old and the new alias names must consist of the full directory paths of the alias names.

[Example 5-5](#) renames an alias.

Renaming an alias does not update the database reference to the alias.

Example 5-5   Renaming an alias name for an Oracle ASM file name

```
ALTER DISKGROUP data RENAME ALIAS '+data/orcl/datafile.dbf'
     TO '+data/payroll/compensation.dbf';
```

Dropping an Alias Name for an Oracle ASM File Name

Use the `DROP ALIAS` clause of the `ALTER DISKGROUP` statement to drop an alias for an Oracle ASM file name. 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 5-6](#) drops an alias.

[Example 5-7](#) fails because it attempts to drop a system-generated file name and this syntax is not allowed.

Example 5-6   Dropping an alias name for an Oracle ASM file name

```
ALTER DISKGROUP data DROP ALIAS '+data/payroll/compensation.dbf';
```

Example 5-7   Dropping an alias name for an Oracle ASM file name

```
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 file name, 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 5-8, the alias name for the file is used when deleting both the file and the alias from a disk group.

In Example 5-9, the Oracle Managed Files (system-generated) file name is used when dropping the file and any associated alias.

Example 5-8   Dropping a file and alias from a disk group using the alias name

```
ALTER DISKGROUP data DROP FILE '+data/payroll/compensation.dbf';
```

Example 5-9   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 file name that Oracle ASM assigns to each file represents a path in this directory hierarchy. The following is an example of a system-generated file name:

```
+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 5-10 creates a hierarchical directory for disk group `data`, which can contain, for example, the alias name `+data/orcl/control_file1`.

Assuming no subdirectory exists under the directory `+data/orcl`, the SQL statement fails in Example 5-11.

Example 5-10   Creating a new directory

```
ALTER DISKGROUP data ADD DIRECTORY '+data/orcl';
```

Example 5-11   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 5-12 renames a directory.

Example 5-12   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 5-13 deletes a directory along with its contents.

Example 5-13   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.

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**
>
> - `cp` for information about the ASMCMD `cp` command.
> - Inside /sys/asm for additional information about the `/sys/asm` folder
> - *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 AI 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 5-1 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.
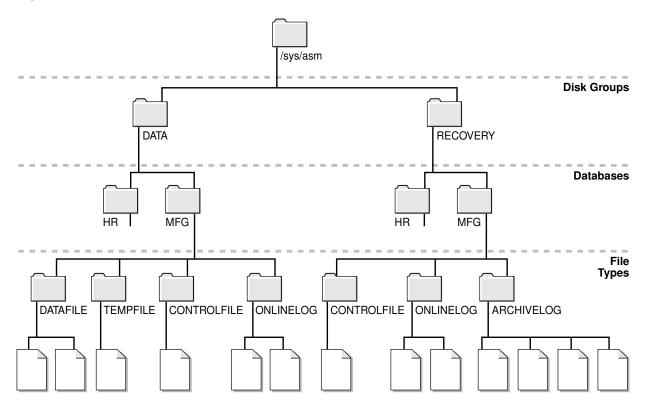
Figure 5-1   Hierarchical Structure of Oracle ASM Folders in an XML DB Installation

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 AI Database PL/SQL Packages and Types Reference* for more information
> about the `DBMS_FILE_TRANSFER` package

Managing Disk Group 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](#), 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 5-6](#) 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` SQL 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.

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](#)

> **ⓘ See Also**
>
> - *Oracle AI Database SQL Language Reference* for information about the `ALTER DISKGROUP` statement
> - *Oracle AI Database Reference* for information about the `V$ASM_TEMPLATE` view which lists all of the templates known to the Oracle ASM instance

Template Attributes

Table 5-4 shows the permitted striping attribute values and allocation unit size chunks. These values correspond to the `STRIPE` column of `V$ASM_TEMPLATE`.

Table 5-4   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 5-5 shows the permitted redundancy values for Oracle ASM templates. These values correspond to the `REDUND` column of `V$ASM_TEMPLATE`.

Table 5-5   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 5-6 shows the initial attribute settings for some default templates. The type of mirroring associated with the **Mirroring** column for normal, high, and external redundancy disk groups is specified in Table 5-5. For example, the **Mirror** column for high redundancy disk groups displays `MIRROR` in Table 5-6. In Table 5-5, the `MIRROR` value for high redundancy disk groups corresponds to `Three-way mirroring`.

Table 5-6   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
ARCHIVELOG	COARSE	MIRROR	MIRROR	UNPROTECTED	COLD	MIRRORCOLD
ASMPARAMETERFILE	COARSE	MIRROR	MIRROR	UNPROTECTED	COLD	MIRRORCOLD
AUDIT_SPILLFILES	COARSE	MIRROR	MIRROR	UNPROTECTED	COLD	MIRRORCOLD
AUTOBACKUP	COARSE	MIRROR	MIRROR	UNPROTECTED	COLD	MIRRORCOLD
BACKUPSET	COARSE	MIRROR	MIRROR	UNPROTECTED	COLD	MIRRORCOLD
CHANGETRACKING	COARSE	MIRROR	MIRROR	UNPROTECTED	COLD	MIRRORCOLD
CONTROLFILE	FINE	HIGH	MIRROR	UNPROTECTED	COLD	MIRRORCOLD
DATAFILE	COARSE	MIRROR	MIRROR	UNPROTECTED	COLD	MIRRORCOLD
DATAGUARDCONFIG	COARSE	MIRROR	MIRROR	UNPROTECTED	COLD	MIRRORCOLD
DUMPSET	COARSE	MIRROR	MIRROR	UNPROTECTED	COLD	MIRRORCOLD
FLASHBACK	COARSE	MIRROR	MIRROR	UNPROTECTED	COLD	MIRRORCOLD
FLASHFILE	COARSE	MIRROR	MIRROR	UNPROTECTED	COLD	MIRRORCOLD

Table 5-6   (Cont.) 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
KEY_STORE	COARSE	MIRROR	MIRROR	UNPROTECTED	COLD	MIRRORCOLD
OCRFILE	COARSE	MIRROR	MIRROR	UNPROTECTED	COLD	MIRRORCOLD
ONLINELOG	COARSE	MIRROR	MIRROR	UNPROTECTED	COLD	MIRRORCOLD
PARAMETERFILE	COARSE	MIRROR	MIRROR	UNPROTECTED	COLD	MIRRORCOLD
TEMPFILE	COARSE	MIRROR	MIRROR	UNPROTECTED	COLD	MIRRORCOLD
VOTINGFILE	COARSE	MIRROR	MIRROR	UNPROTECTED	COLD	MIRRORCOLD
XTRANSPORT	COARSE	MIRROR	MIRROR	UNPROTECTED	COLD	MIRRORCOLD

COLD and MIRRORCOLD are the default values and the only permitted values for the PRIMARY_REGION and MIRROR_REGION attributes. These values specify the use of the inner most tracks (closest to spindle) on the disk drive.

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 5-6, 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 5-14 creates a new template named reliable for the normal redundancy disk group data.

Example 5-15 creates a new template named unreliable that specifies UNPROTECTED (no mirroring) for files.

> ⓘ **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.

Example 5-14   Adding a template with high redundancy to a disk group

```
ALTER DISKGROUP data ADD TEMPLATE reliable ATTRIBUTES (HIGH FINE);
```

Example 5-15   Adding a template with external redundancy to a disk group

```
ALTER DISKGROUP data ADD TEMPLATE unreliable ATTRIBUTES (UNPROTECTED);
```

> ⓘ **See Also**
>
> *Oracle AI 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 5-16 changes the striping attribute specification of the `reliable` template for disk group `data`.

Example 5-16   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 5-17 drops the previously created template `unreliable` from `data`:

Example 5-17   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 5-18 assumes that the template (`mytemplate`) has been defined.

Example 5-18   Using a user-defined template to specify attributes

```
ALTER SYSTEM SET DB_CREATE_FILE_DEST = '+data(mytemplate)';

CREATE TABLESPACE mytblspace;
```

6

Using Views to Display Oracle ASM Information

Dynamic views display important information about Oracle ASM.

This chapter contains information about using dynamic views to display Oracle Automatic Storage Management (Oracle ASM) information.

- Views Containing Oracle ASM Information
- Viewing Oracle ASM File Access Control Information

> ⓘ **See Also**
>
> - *Oracle AI Database Reference* for information about all dynamic performance views
> - Managing Oracle ASM with ASMCMD for ASMCMD `ls*` commands that display Oracle ASM information.

Views Containing Oracle ASM Information

The views listed in Table 6-1 provide information about Oracle ASM.

For information about all dynamic performance views that display information about Oracle ASM, refer to *Oracle AI Database Reference*

> ⓘ **Note**
>
> The `REDUNDANCY_LOWERED` column in the `V$ASM_FILE` view no longer provides information about files with reduced redundancy. This column is deprecated, and it always displays a value of `U`.

Table 6-1   Oracle ASM dynamic views

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$ASM_AUDIT_CLEANUP_JOBS	Displays information about the configured audit trail purge jobs.
V$ASM_AUDIT_CONFIG_PARAMS	Displays information about the currently configured audit trail properties.

Table 6-1   (Cont.) Oracle ASM dynamic views

View	Description
`V$ASM_AUDIT_LAST_ARCH_TS`	Displays information about the last archive timestamps set for audit trail cleanup or purges.
`V$ASM_CLIENT`	In an Oracle ASM instance, identifies databases using disk groups managed by the Oracle ASM instance.
	In an Oracle AI Database instance, contains information about the Oracle ASM instance if the database has any open Oracle ASM files.
`V$ASM_DBCLONE_INFO`	In an Oracle ASM instance, shows the relationship between the source database, the cloned database, and their file groups.
`V$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. Because performing a discovery is very resource intensive, this view is not recommended for monitoring scripts.
`V$ASM_DISK_IOSTAT`	Displays information about disk I/O statistics for each Oracle ASM client.
	In an Oracle AI Database instance, only the rows for that instance are shown.
`V$ASM_DISK_STAT`	Contains the same columns as `V$ASM_DISK`, but to reduce overhead, does not perform a discovery when it is queried. `V$ASM_DISK_STAT` only returns information about any disks that are part of mounted disk groups in the storage system.
	The `V$ASM_DISK_STAT` view is recommended for monitoring scripts.
	To display information about all disks, use `V$ASM_DISK` instead.
`V$ASM_DISKGROUP`	Describes a disk group (number, name, size related info, state, and redundancy type).
	This view performs disk discovery every time it is queried. Because performing a discovery is very resource intensive, this view is not recommended for monitoring scripts.
`V$ASM_DISKGROUP_STAT`	Contains the same columns as `V$ASM_DISKGROUP`, but to reduce overhead, `V$ASM_DISKGROUP_STAT` does not perform a discovery when it is queried.
	The `V$ASM_DISKGROUP_STAT` view is recommended for monitoring scripts.
	To display more complete information about all disks, use `V$ASM_DISKGROUP`.
`V$ASM_ESTIMATE`	Displays an estimate of the work involved in execution plans for Oracle ASM disk group rebalance and resync operations.
`V$ASM_FILE`	Contains one row for every Oracle ASM file in every disk group mounted by the Oracle ASM instance.
`V$ASM_FILEGROUP`	Contains one row for every file group in every disk group mounted by the Oracle ASM instance.
`V$ASM_FILEGROUP_FILE`	Contains one row for each file associated with a file group in every disk group mounted by the Oracle ASM instance.
`V$ASM_FILEGROUP_PROPERTY`	Contains one row for every property associated with the file group or each file type of every file group in every disk group mounted by the Oracle ASM instance.
`V$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 AI Database instance, contains no rows.

Table 6-1   (Cont.) Oracle ASM dynamic views

View	Description
V$ASM_QUOTAGROUP	Contains one row for every Oracle Automatic Storage Management (Oracle ASM) quota group discovered by the Oracle ASM instance on the node.
V$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$ASM views, the value of the disk group number is not a static value. When a disk group is mounted, a disk group number is chosen. This number may change across disk group mounts. A disk group number is not recorded in any persistent structure, but the current value can be viewed in the GROUP_NUMBER column of the V$ASM views.

Example 6-1   Viewing disk group attributes with V$ASM_ATTRIBUTE

This is an example of the use of the V$ASM_ATTRIBUTE and V$ASM_DISKGROUP views. 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.

```
SELECT SUBSTR(dg.name,1,12) AS diskgroup, SUBSTR(a.name,1,24) 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
     AND a.name NOT LIKE '%template%';


DISKGROUP    NAME                     VALUE                    READ_ON
------------ ------------------------ ------------------------ -------
DATA         idp.type                 dynamic                  N
DATA         idp.boundary             auto                     N
DATA         vam_migration_done       false                    Y
DATA         scrub_metadata.enabled   TRUE                     N
DATA         scrub_async_limit        1                        N
DATA         content_hardcheck.enable FALSE                    N
DATA         access_control.umask     066                      N
DATA         access_control.enabled   FALSE                    N
DATA         cell.sparse_dg           allnonsparse             N
DATA         cell.smart_scan_capable  FALSE                    N
DATA         compatible.advm          19.0.0.0.0               N
DATA         compatible.rdbms         19.0.0.0.0               N
DATA         compatible.asm           19.0.0.0.0               N
DATA         appliance._partnering_ty GENERIC                  Y
DATA         au_size                  1048576                  Y
DATA         content.check            FALSE                    N
DATA         content.type             data                     N
DATA         logical_sector_size      512                      N
DATA         sector_size              512                      N
DATA         ate_conversion_done      true                     Y
DATA         preferred_read.enabled   FALSE                    N
DATA         thin_provisioned         FALSE                    N
DATA         failgroup_repair_time    24.0h                    N
DATA         phys_meta_replicated     true                     Y
DATA         disk_repair_time         12.0h                    N
```

Example 6-2   Viewing the compatibility of a disk group with V$ASM_DISKGROUP

This is an example of displaying the compatibility for a disk group with the `V$ASM_DISKGROUP` view.

```
SELECT name AS diskgroup, substr(compatibility,1,12) AS asm_compat,
      substr(database_compatibility,1,12) AS db_compat FROM V$ASM_DISKGROUP;


DISKGROUP                    ASM_COMPAT   DB_COMPAT
---------------------------- ------------ ------------
DATA                         19.0.0.0.0   19.0.0.0.0
FRA                          19.0.0.0.0   19.0.0.0.0
```

Example 6-3   Viewing disks in disk groups with V$ASM_DISK

This is an example of the use of the `V$ASM_DISK` and `V$ASM_DISKGROUP` views run on an Oracle ASM instance. The example displays the disks associated with a disk group, plus the mount status, state, and failure group of the disks.

```
SELECT SUBSTR(dg.name,1,16) AS diskgroup, SUBSTR(d.name,1,16) AS asmdisk,
      d.mount_status, d.state, SUBSTR(d.failgroup,1,16) AS failgroup
      FROM V$ASM_DISKGROUP dg, V$ASM_DISK d WHERE dg.group_number = d.group_number;


DISKGROUP        ASMDISK          MOUNT_S STATE    FAILGROUP
---------------- ---------------- ------- -------- ----------------
DATA             DATA_0008        CACHED  NORMAL   DATA_0008
DATA             DATA_0000        CACHED  NORMAL   DATA_0000
DATA             DATA_0004        CACHED  NORMAL   DATA_0004
DATA             DATA_0015        CACHED  NORMAL   DATA_0015
DATA             DATA_0003        CACHED  NORMAL   DATA_0003
DATA             DATA_0012        CACHED  NORMAL   DATA_0012
DATA             DATA_0017        CACHED  NORMAL   DATA_0017
DATA             DATA_0013        CACHED  NORMAL   DATA_0013
DATA             DATA_0007        CACHED  NORMAL   DATA_0007
DATA             DATA_0014        CACHED  NORMAL   DATA_0014
DATA             DATA_0009        CACHED  NORMAL   DATA_0009
DATA             DATA_0001        CACHED  NORMAL   DATA_0001
DATA             DATA_0016        CACHED  NORMAL   DATA_0016
DATA             DATA_0011        CACHED  NORMAL   DATA_0011
DATA             DATA_0005        CACHED  NORMAL   DATA_0005
DATA             DATA_0010        CACHED  NORMAL   DATA_0010
DATA             DATA_0002        CACHED  NORMAL   DATA_0002
DATA             DATA_0006        CACHED  NORMAL   DATA_0006
FRA              FRA_0012         CACHED  NORMAL   FRA_0012
FRA              FRA_0013         CACHED  NORMAL   FRA_0013
FRA              FRA_0007         CACHED  NORMAL   FRA_0007
FRA              FRA_0006         CACHED  NORMAL   FRA_0006
FRA              FRA_0010         CACHED  NORMAL   FRA_0010
FRA              FRA_0000         CACHED  NORMAL   FRA_0000
FRA              FRA_0003         CACHED  NORMAL   FRA_0003
...
```

Example 6-4   Viewing disks in disk groups with V$ASM_DISK_STAT

This is an example of the use of the `V$ASM_DISK_STAT` and `V$ASM_DISKGROUP_STAT` views run on an Oracle ASM instance. The example displays the disks associated with a specific disk group along with the mount status, state, and various read and write statistics.

```
SELECT SUBSTR(dgs.name,1,10) AS diskgroup, SUBSTR(ds.name,1,10) AS asmdisk,
      ds.mount_status, ds.state, ds.reads, ds.writes, ds.read_time, ds.write_time,
      bytes_read, bytes_written
      FROM V$ASM_DISKGROUP_STAT dgs, V$ASM_DISK_STAT ds
      WHERE dgs.group_number = ds.group_number AND dgs.name = 'DATA';
```

```
DISKGROUP   ASMDISK    MOUNT_S STATE    READS  WRITES READ_TIME WRITE_TIME BYTES_READ BYTES_WRITTEN
---------- ---------- ------- ------- ------ ------ --------- ---------- ---------- -------------
DATA       DATA_0000  CACHED  NORMAL     841  10407 1.212218   3.511977   23818240   178369024
DATA       DATA_0008  CACHED  NORMAL   26065   1319 1.592524    .297728  436203520    38358528
DATA       DATA_0010  CACHED  NORMAL     561    868  .794849    .337575   18631680    22584320
DATA       DATA_0004  CACHED  NORMAL     695  10512 1.282711   3.351801   23240704   177246208
DATA       DATA_0006  CACHED  NORMAL     484   1642 1.506733    .45724    19857408    30191616
DATA       DATA_0016  CACHED  NORMAL     583   1028 2.283268    .263629   21012480    17682432
DATA       DATA_0007  CACHED  NORMAL     724   2316 1.259379    .546318   26017792    42283008
DATA       DATA_0009  CACHED  NORMAL     537    757 1.146663    .241434   19893248    20633088
DATA       DATA_0014  CACHED  NORMAL    1049   1464 7.346259    .677313   25378816    27578368
DATA       DATA_0017  CACHED  NORMAL    1440   1326 1.132886   2.541013   25899008    26537984
DATA       DATA_0013  CACHED  NORMAL     714   1391 1.527926    .371432   18169856    22814720
DATA       DATA_0001  CACHED  NORMAL     713    807  .790505    .219565   20406272    28561408
DATA       DATA_0012  CACHED  NORMAL     617   1206 1.016893   3.60425    24477696    25391104
DATA       DATA_0003  CACHED  NORMAL   15567  11500 5.642053   3.328861  266956800   183625728
DATA       DATA_0015  CACHED  NORMAL     642   1357 2.545441    .403455   22179840    24973312
DATA       DATA_0011  CACHED  NORMAL    7585   1685 1.121678    .359123  135217152    37572608
DATA       DATA_0005  CACHED  NORMAL     513   1431 1.007476    .47202    26427392    21344256
DATA       DATA_0002  CACHED  NORMAL   11368   2196 1.209433    .861601  199213056    32090624
```

Example 6-5   Viewing disk group clients with V$ASM_CLIENT

This is an example of the use of the V$ASM_CLIENT and V$ASM_DISKGROUP views on an Oracle ASM instance. The example displays disk groups with information about the connected database client instances.

```
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     19.0.0.0.0   19.0.0.0.0
DATA                            orcl         orcl     19.0.0.0.0   19.0.0.0.0
DATA                            +ASM         asmvol   19.0.0.0.0   19.0.0.0.0
FRA                             orcl         orcl     19.0.0.0.0   19.0.0.0.0
...
```

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.

For more information about Oracle ASM File Access Control, see Managing Oracle ASM File Access Control for Disk Groups.

Example 6-6   Viewing Oracle ASM File Access Control information with V$ASM_USER

This example shows information about Oracle ASM File Access Control users displayed in the V$ASM_USER view.

```
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 6-7   Viewing File Access Control information with V$ASM_USERGROUP

This example shows information about Oracle ASM File Access Control user groups displayed in the V$ASM_USERGROUP view.

```
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 6-8   Viewing File Access Control information with V$ASM_USERGROUP_MEMBER

This example shows information about Oracle ASM File Access Control user groups and members displayed in the V$ASM_USERGROUP_MEMBER view.

```
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 6-9   Viewing Oracle ASM File Access Control information with V$ASM_FILE

This example shows information about Oracle ASM File Access Control file permissions displayed in the V$ASM_FILE view.

```
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.name = 'FRA' AND dg.group_number = u.group_number AND
      u.group_number = ug.group_number AND ug.group_number = f.group_number AND
      f.group_number = a.group_number 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
...
```

7

Administering Oracle ASM with Oracle Enterprise Manager

Oracle Enterprise Manager Cloud Control provides tools for administering Oracle ASM.

This chapter describes how to administer Oracle Automatic Storage Management (Oracle ASM) with Oracle Enterprise Manager Cloud Control.

This chapter contains the following topics:

- [Accessing the 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](#)

- [Monitoring Alerts and Incidents With Oracle ASM Support Workbench](#)

- [Monitoring Oracle Flex ASM](#)

> ⓘ **See Also**
>
> *Oracle Enterprise Manager Cloud Control Administrator's Guide*

Accessing the Oracle Automatic Storage Management Home Page

All Oracle ASM administration tasks begin with the Oracle Automatic Storage Management home page in Oracle Enterprise Manager Cloud Control. 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 Advanced 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.

To access the Oracle Automatic Storage Management home page on a single-instance system:

1. Log in to Oracle Enterprise Manager Cloud Control.

2. Under the **Targets** drop down menu at the top of the page, select **All Targets**.

3. In the lists of targets on the **All Targets** page, click the target for the Oracle ASM instance.

4. 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 `SYSASM`. The Oracle Automatic Storage Management home page displays.

For more information about authentication, refer to [Authentication for Accessing Oracle ASM Instances](#).

Configuring Oracle ASM Initialization Parameters with Oracle Enterprise Manager

You can configure Oracle ASM with the Oracle Enterprise Manager Configuration Parameters page.

To configure the Oracle ASM instance:

1. Access the Oracle Automatic Storage Management home page.

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.

   - Auto Mount Disk Groups (`ASM_DISKGROUPS` initialization parameter)

     Enter a string value.

   - Rebalance Power (`ASM_POWER_LIMIT` initialization parameter)

     Enter an integer value.

   - 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.

4. Click **Apply** to save your changes or click **Revert** to discard any changes made on the page.

> ⓘ **See Also**
>
> - [Accessing the Oracle Automatic Storage Management Home Page](#) for instructions on accessing the Oracle Automatic Storage Management home page
> - [ASM_DISKSTRING](#) for information about the `ASM_DISKSTRING` initialization parameter
> - [ASM_DISKGROUPS](#) for information about the `ASM_DISKGROUPS` initialization parameter
> - [ASM_POWER_LIMIT](#) for information about the `ASM_POWER_LIMIT` initialization parameter
> - [ASM_PREFERRED_READ_FAILURE_GROUPS](#) for information about the `ASM_PREFERRED_READ_FAILURE_GROUPS` initialization parameter

Managing Oracle ASM Users with Oracle Enterprise Manager

You can manage Oracle ASM users with the Oracle Enterprise Manager Users page

. Access this page by clicking the Users tab that is near the top center of the Oracle Automatic Storage Management home page.

To manage Oracle ASM users:

1. Access the Oracle Automatic Storage Management home page.

2. Click the **Users** link at the top to view the Users property page.

   `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.

   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.

3. 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.

4. 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
>
> - [Accessing the Oracle Automatic Storage Management Home Page](#) for instructions on accessing the Oracle Automatic Storage Management home page
>
> - [Creating Oracle ASM Instances with ASMCA](#) for information about creating an Oracle ASM instance with ASMCA
>
> - [Authentication for Accessing Oracle ASM Instances](#) for information about authentication for Oracle ASM instances
>
> - *Oracle Grid Infrastructure Installation and Upgrade Guide* for information about installing Oracle ASM
>
> - *Oracle AI Database Installation Guide* for information about reviewing database accounts and passwords

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](#)

- [Specifying a Power Limit For Disk Online](#)

- [Replacing An Existing Disk With a New Disk](#)

- [Setting and Modifying Content Type for the Disk Group](#)

- [Setting and Modifying the Failure Group Repair Time](#)

- [Running Rebalance on Multiple Disk Groups](#)

- [Viewing Rebalance Process Related Data](#)

- [Setting and Modifying the Thin Provisioned Disk Group Attribute](#)

- [Running the On Demand Scrubbing Process](#)

- [Determining Whether Scrubbing Process Is Running](#)

For more information about managing Oracle ASM disk groups, refer to [Administering Oracle ASM Disk Groups](#)

Creating Disk Groups

You can create disk groups with Oracle Enterprise Manager.

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.

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. 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.

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.

8. Specify a **Disk Compatibility** value for the disk group to enable your system to use the latest Oracle ASM features.

   > ⓘ **Note**
   >
   > You cannot reverse the setting of a disk group compatibility value.

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**
>
> - [Accessing the Oracle Automatic Storage Management Home Page](#) for instructions on accessing the Oracle Automatic Storage Management home page
> - [Creating Disk Groups](#) for information about using the `FORCE` option when creating disk groups
> - [Configuring Disk Group Compatibility Attributes](#) for information about configuring disk group compatibility attributes
> - [Disk Group Compatibility](#) for information about disk group compatibility attributes
> - [Creating Disk Groups](#) for more information about creating disk groups

Adding Disks to Disk Groups

You can add disks to disk groups with Oracle Enterprise Manager.

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.

To add one or more disks to a disk group:

1. Access the Oracle Automatic Storage Management home page.

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 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.

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.

   - 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**
>
> - Accessing the Oracle Automatic Storage Management Home Page for instructions on accessing the Oracle Automatic Storage Management home page
>
> - Manually Rebalancing Disk Groups and Tuning Rebalance Operations for information about controlling rebalance operations
>
> - ASM_POWER_LIMIT for information about rebalance power settings and the `ASM_POWER_LIMIT` initialization parameter
>
> - Adding Disks to a Disk Group for an example of using the `FORCE` option
>
> - Altering Disk Groups 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 Accessing the Oracle Automatic Storage Management Home Page for instructions.

2. Click the **Disk Groups** link at the top of the page to view the Disk Groups page.

   Click the name of a disk in the Name column and Oracle Enterprise Manager displays the General page for the disk group to modify.

3. 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.

4. 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".

- 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.

> ⓘ **Note**
>
> For information on using the SQL `ALTER DISKGROUP` command to take disks offline/online, see the Oracle AI Database SQL Language Reference.

Dropping Disks from Disk Groups

You can drop disks from disk groups with Oracle Enterprise Manager.

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.

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.

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.

> ℹ️ **See Also**
>
> - [Accessing the Oracle Automatic Storage Management Home Page](#) for instructions about accessing the Oracle Automatic Storage Management home page
> - [Dropping Disks from Disk Groups](#) for information about using the `FORCE` option when dropping disk groups
> - [Dropping Disks from Disk Groups](#) for more information about 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 "[Accessing the Oracle Automatic Storage Management Home Page](#)" for instructions.

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**.

   For more information about dropping disks, see "Dropping Disk Groups".

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 "Accessing the Oracle Automatic Storage Management Home Page" for instructions.

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 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.

For more information, see "Managing Capacity in Disk Groups" and "Views Containing Oracle ASM Disk Group Information"

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, 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.

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](#)

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 the setting of a disk group 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` compatibility of the disk group 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".

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".

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".

Specifying a Power Limit For Disk Online

To specify a power limit for disk online, follow these steps:

1. Access the Oracle Automatic Storage Management home page.

2. Select the **Disk Groups** tab. The Disk Groups page is displayed.

3. Select a disk group. The Disk Group page appears and shows all Member Disks that belong to the disk group in the Member Disks table.

4. Select a disk, then click **Online**. The online confirmation page is displayed with the advance option to specify a power limit.

5. Specify the **Power Limit**, then click **Submit**. The disk is set to online with the specified power limit.

> ⓘ **See Also**
>
> - [Accessing the Oracle Automatic Storage Management Home Page](#) for instructions about accessing the Oracle ASM home page
>
> - [Manually Rebalancing Disk Groups](#) for information about the power option in rebalancing a disk group
>
> - [ASM_POWER_LIMIT](#) for information about the `ASM_POWER_LIMIT` initialization parameter

Replacing An Existing Disk With a New Disk

To replace an existing disk with a new disk, follow these steps:

1. Access the Oracle Automatic Storage Management home page.

    See "[Accessing the Oracle Automatic Storage Management Home Page](#)" for instructions.

2. Click the Disk Group tab. The Disk Group page displays.

3. Select a Disk Group. The Disk Group page appears which shows all Member Disks that belong to the disk group.

4. Select an offline disk to replace. The Replace Disks page is displayed.

    Select a new disk of size equal to or greater than the original disk being replaced to ensure the same level of data protection.

5. Select a new disk and then specify the options, such as Rebalance Power, and then click **OK**. The disk is then replaced. Rebalance operations redistribute data evenly across all drives. Oracle ASM automatically rebalances a disk group whenever disks are added or dropped. To manually rebalance all disk groups, you must specify the Rebalance Power. Higher values use more I/O bandwidth and complete rebalance more quickly. Lower values cause rebalance to take longer, but use less I/O bandwidth. Values range from 0 to 1024.

For additional information about replacing a disk, refer to "[Replacing Disks in Disk Groups](#)".

Setting and Modifying Content Type for the Disk Group

To set and modify the content type for a disk group, follow these steps:

1. Access the Oracle Automatic Storage Management home page.

    See "[Accessing the Oracle Automatic Storage Management Home Page](#)" for instructions.

2. Click the Disk Groups tab. The Disk Group page displays.

3. Click **Create**. In the Disk Resync Attributes section of the Create Disk Group page, you can select the value **Recovery**, **Data** or **System** from the **Content Type** drop-down list. The attribute can be set only if the `ASM` compatibility is greater than or equal to 12.1.0.0.

4. To edit the content type for an existing disk group, navigate to the Disk Group home page by clicking the disk group in the Disk Groups table. The Disk Group home page displays.

5. Click **Edit** in the Advanced Attribute Section. The Edit Advanced Attributes for Disk Group page appears. In the Disk Resync Attributes section, select from the values **Recovery**, **Data**, or **System** from the **Content Type** drop-down list.

For information about the `CONTENT.TYPE` disk group attribute, refer to "[CONTENT.TYPE](#)".

Setting and Modifying the Failure Group Repair Time

To set and modify the Failure Group Repair Time, follow these steps:

1. Access the Oracle Automatic Storage Management home page.

   See "Accessing the Oracle Automatic Storage Management Home Page" for instructions.

2. Click the Disk Groups tab. The Disk Groups page displays.

3. Click **Create**. Specify the repair time for the **Failure Group** in the Candidate Member Disks table.

4. To edit an existing Failure Group repair time, from the Disk Group page choose the **Disk Group** for the repair time you want to modify. The Disk Group page appears.

5. In the Advanced Attributes section, click **Edit**. The Edit Advanced Attributes for Disk Group page displays.

6. In the Disk Resync Attributes section, modify the **Failure Group Repair Time**, then click **OK**. The failure Group Repair Time applies to all the failure groups in the disk group except those containing only one disk. This value overrides the disk repair time and is overridden by the actual value provided when a disk is taken offline.

For additional information about setting and modifying the failure group repair time, refer to "Oracle ASM Fast Mirror Resync".

Running Rebalance on Multiple Disk Groups

Enterprise Manager enables you to rebalance on multiple disk groups and distribute the data copy across other member instances in the cluster. To run rebalance on multiple disk groups, follow these steps:

1. Access the Oracle Automatic Storage Management home page.

   See "Accessing the Oracle Automatic Storage Management Home Page" for instructions.

2. Click the Disk Group tab. The Disk Group page displays.

3. Select multiple disk groups then click **Rebalance**. The Rebalance Disk Groups Confirmation page appears where you can set the Rebalance Power by displaying Advanced Options.

4. Set the Rebalance Power value, then click **OK**. The command to rebalance is submitted.

For additional information about rebalancing disk groups, refer to "Manually Rebalancing Disk Groups".

Viewing Rebalance Process Related Data

To view rebalance process-related data, including the number of participating instances on the Pending Operations page, follow these steps:

1. Access the Oracle Automatic Storage Management home page.

   See "Accessing the Oracle Automatic Storage Management Home Page" for instructions.

2. Click the Disk Group tab. The Disk Groups page displays.

3. Click a Disk Group link. The Disk Group home page General section displays the number of **Pending Operations** as a link to the Pending Operations page.

4. Click the link. Information on the rebalance process, along with details on the number of participating instances, are displayed.

For additional information about rebalancing disk groups, refer to "Tuning Rebalance Operations".

Setting and Modifying the Thin Provisioned Disk Group Attribute

To set and modify the thin-provisioned disk group attribute, follow these steps:

1. Access the Oracle Automatic Storage Management home page.

   See "Accessing the Oracle Automatic Storage Management Home Page" for instructions.

2. Click the Disk Groups tab. The Disk Groups page is displayed.

3. Click the Disk Group that contains the Thin Provisioned attribute you want to change. The Disk Group page displays.

4. In the Advance Attributes section, click **Edit**. The Edit Advanced Attributes for Disk Group page displays.

5. In the Thin Provisioning section, select **True** or **False** from the Thin Provisioned field drop-down list. If thin provisioning is enabled, the disk space not in use can be returned to the array. This attribute can be set only if the `ASM` compatibility is greater than or equal to 12.1.0.0.

Running the On Demand Scrubbing Process

To run the On Demand scrubbing process, follow these steps:

1. Access the Oracle Automatic Storage Management home page.

   See "Accessing the Oracle Automatic Storage Management Home Page" for instructions.

2. Click the Disk Group tab. The Disk Group page is displayed.

3. Click **Scrub**. The Scrub Disk Group page is displayed where you can use advanced options such as repair, power level, wait and force. You can also use the scrub action on disks and files.

4. Provide the advance option and click **OK**. A background process is initiated that scrubs the disk group.

For additional information about scrubbing disk groups, refer to "Scrubbing Disk Groups".

Determining Whether Scrubbing Process Is Running

To determine whether the scrubbing process is running, follow these steps:

1. Access the Oracle Automatic Storage Management home page.

   See "Accessing the Oracle Automatic Storage Management Home Page" for instructions.

2. Click the Disk Group tab. The Disk Group page displays.

3. Click a Disk Group link. The Disk Group General Section displays a Pending Operation link if a scrubbing process is ongoing.

4. Click the link. Details on the scrubbing process are displayed.

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.

To manage Oracle ASM File Access Control:

1. Access the Oracle Automatic Storage Management home page.

   See "Accessing the Oracle Automatic Storage Management Home Page" 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, remove, or replace users from a user group and create or delete user groups. You can also modify the ownership or permission of files.

   To access files in a disk group, operating system users must be granted access privileges to the disk group. After 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.

   Oracle ASM File Access Control operations on this page are available when the files are open.

   For more information about Oracle ASM File Access Control, see "Managing Oracle ASM File Access Control for Disk Groups".

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.

To manage Oracle ASM files:

1. Access the Oracle Automatic Storage Management home page.

   See "Accessing the Oracle Automatic Storage Management Home Page" 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 icons to left on the folders to expand or contract the folders. When you expand the folders, you can drill down to individual files.

6. 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.

For more information about managing Oracle ASM files, directories, and aliases, see 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.

To manage Oracle ASM templates:

1. Access the Oracle Automatic Storage Management home page.

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 or striped granularity.

> ⓘ **see Also**
>
> • [Accessing the Oracle Automatic Storage Management Home Page](#) for instructions about accessing the Oracle ASM home page
>
> • [Managing Disk Group Templates](#) for more information about disk group templates

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.

To monitor Oracle ASM performance:

1. Access the Oracle Automatic Storage Management home page.

   See "[Accessing the Oracle Automatic Storage Management Home Page](#)" for instructions.

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

For more information, refer to "Performance and Scalability Considerations for Disk Groups".

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** page, select a disk group in the list and click **Check**.

   A confirmation dialog for the check disk group operation displays.

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. To schedule a job, click **Schedule Job**. 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. 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.

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 directory.

3. Delete the alias and directory.

4. Rename the alias and directory.

> ⓘ **See Also**
>
> - Performing Oracle ASM Data Migration with RMAN for information about migrating data with RMAN
> - *Oracle AI Database 2 Day DBA* for information about the procedures for using RMAN
> - *Oracle AI Database Backup and Recovery User's Guide* for additional information about RMAN

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. Access the Oracle Automatic Storage Management home page.

    See "Accessing the Oracle Automatic Storage Management Home Page" for instructions.

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 **Administration** pull down list and select **Migrate to Oracle ASM**.

2. You may need to set credentials for the database, Oracle ASM, and host.

    If prompted for database login credentials, then enter the user SYS, provide the SYS password that was set for the Oracle AI Database instance during installation, and connect as SYSDBA.

    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 SYSASM. For more information about authentication, see "Authentication for Accessing Oracle ASM Instances".

3. 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.

4. 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.

5. 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.

6. 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.

7. 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.

> ⓘ **Note**
>
> If your migration job fails, then do not start your database until you complete a recovery operation. Otherwise, you may corrupt your database.

8. 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 may need to wait for the links to display after a subsequent refresh.

Monitoring Alerts and Incidents With Oracle ASM Support Workbench

Oracle Enterprise Manager provides Support Workbench to monitor Oracle ASM alerts and incidents.

To access Support Workbench, select **Diagnostics** > **Support Workbench** under the **Automatic Storage Management** target menu at the top of the page.

On the Oracle ASM Support Workbench page **Problems** tab you can view information about current and past problems. You also have options to view additional details and create a package that you can send to Oracle Support Services.

Support Workbench guides you through the packaging process as follows:

1. To create a package, select an incident and click **Package** on the Support Workbench Problems page. The Support Workbench **Select packaging mode** page of the wizard displays.

2. On the **Select packaging mode** page, select either the **Quick** or **Custom** packaging mode.

3. Depending on the packaging mode chosen, you are presented with various entry fields and options as you click **Next** through the wizard.

   The wizard guides you through the packaging process using a series of steps that include:

   • Viewing and editing the contents of the package

   • Viewing and editing the manifest of the package

   • Adding additional diagnostic details

   • Scheduling the generation and uploading of the package

On the Support Workbench **Checker Findings** tab, you can view and edit information about packages created for problems on the Support Workbench **Packages** tab. You can also check findings on the **Checker Findings** tab.

ⓘ **See Also**

Oracle AI Database Administrator's Guide for information about viewing problems with Oracle Enterprise Manager Support Workbench

Monitoring Oracle Flex ASM

Oracle Enterprise Manager provides functionality to monitor your Oracle Flex ASM configuration.

• On the Oracle Flex ASM home page, you can view and administer components of the Oracle Flex ASM configuration.

• On the Oracle Flex ASM Members page, you can view all the members of the Oracle Flex ASM configuration.

• On the Oracle Flex ASM Topology page, you can view the topology of the Oracle Flex ASM configuration.

For information about Oracle Flex ASM, refer to "Managing Oracle Flex ASM".

8

Performing Oracle ASM Data Migration with RMAN

Recovery Manager (RMAN) provides tools to migrate data to and from Oracle ASM.

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](#)

The procedures in this chapter cover an Oracle ASM standalone environment on a Linux platform. File locations and procedures may be different for different Oracle configurations and on other operating system platforms.

> ⓘ **See Also**
>
> - *Oracle AI Database Backup and Recovery User's Guide* for complete information about using RMAN
> - Operating system-specific documentation for migrating data in and out of Oracle ASM on specific platforms
> - *Oracle Clusterware Administration and Deployment Guide* for information about managing and migrating Oracle Cluster Registry and voting files
> - *Oracle Real Applications Cluster Administration and Deployment Guide* for migrating data in and out of Oracle ASM in an Oracle RAC configuration
> - Articles at [My Oracle Support](https://support.oracle.com) (`https://support.oracle.com`) for information about gathering and backing up Oracle ASM and Oracle ACFS metadata in a formatted manner, such as HTML format

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](#)

- [Basic 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.

Basic 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 AI 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](#).

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](#).

To migrate files from Oracle ASM to alternative storage, see [Migrating a Database from Oracle ASM to Alternative Storage](#).

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 `SYSBACKUP` privileges (rather than `SYSDBA` privileges) to enforce the separation of duty security model, 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. Connect with `SYSBACKUP` privileges to enforce the separation of duty security model.

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. 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](#).

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:

     ```
     SQL> CREATE SPFILE='+DATA/spfilesid.ora' FROM PFILE='?/dbs/initsid.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 "Back up the data files to the Oracle ASM disk group" in [Preparing to Migrate the Database to Oracle ASM Using RMAN](). 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 "Set Oracle Managed Files initialization parameters to Oracle ASM locations" in Migrating the Database to Oracle ASM Using RMAN are set.

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 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;
}
```

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;
/
```

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 and Migrating the Database to Oracle ASM Using RMAN 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 file name used in a SQL statement is an Oracle ASM location, then change it to a file name 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. You can use RMAN `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.

You can also use the `ALTER DATABASE MOVE DATAFILE`. SQL statement to move data files. For information about moving data files online with `ALTER DATABASE MOVE DATAFILE`, refer to Moving Data Files Between Disk Groups With ALTER DATABASE.

For this scenario using RMAN, assume that you are using disk groups `DATA` and `USERDATA` and you want to move the data file `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 using the RMAN `BACKUP AS COPY` procedure with the `SET NEWNAME` and `SWITCH` commands, 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. 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. 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
Managing Oracle ASM With ASMCA

Oracle ASM Configuration Assistant (ASMCA) provides utilities for managing Oracle ASM.

Oracle ASM Configuration Assistant (ASMCA) supports installing and configuring Oracle ASM instances, Oracle ASM disk groups, Oracle ASM Dynamic Volume Manager (Oracle ADVM) volumes, and Oracle Advanced Cluster File System (Oracle ACFS) file systems. In addition, you can use the ASMCA command-line interface.

This chapter discusses the following topics:

- [Getting Started With the ASMCA GUI Tool](#)
- [Managing Instances and Disk Groups With ASMCA](#)
- [ASMCA Command-Line Interface](#)

ⓘ **See Also**

Oracle Advanced Cluster File System Administrator's Guide for information about managing Oracle ACFS and Oracle ADVM with ASMCA

ⓘ **Note**

Some ASMCA functionality and menu options are only available in specific environments, such as an Oracle RAC configuration. For details about a specific ASMCA page, access the online help provided with the **Help** button.

Getting Started With the ASMCA GUI Tool

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.

To start the command-line ASMCA, see [ASMCA Command-Line Interface](#).

> ⓘ **Note**
>
> Some ASMCA functionality and menu options are only available in specific environments, such as an Oracle RAC configuration. For details about a specific ASMCA page, access the online help provided with the **Help** button.

Managing Instances and Disk Groups With ASMCA

This section contains the following topics:

- [Creating Oracle ASM Instances with ASMCA](#)
- [Managing Disk Groups with ASMCA](#)

> ⓘ **Note**
>
> Some ASMCA functionality and menu options are only available in specific environments, such as an Oracle RAC configuration. For details about a specific ASMCA page, access the online help provided with the **Help** button.

> ⓘ **Note**
>
> On Solaris and AIX, if you shut down the database instance using the `SHUTDOWN ABORT` command, volume device files are not removed.

Creating Oracle ASM Instances with ASMCA

Oracle ASM Configuration Assistant enables you to create an Oracle ASM instance if the Oracle ASM instance is not found on the host.

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. `ASMSNMP` is a less privileged user that is primarily used to monitor Oracle ASM instances.

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.

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.

> ⓘ **See Also**
>
> - [Initialization Parameters for Oracle ASM Instances](#) for information about Oracle ASM instance initialization parameters
>
> - *Oracle Grid Infrastructure Installation and Upgrade Guide* for information about installing Oracle ASM
>
> - *Oracle AI Database Installation Guide* for information about reviewing database accounts and passwords

Managing Disk Groups with ASMCA

Oracle ASM Configuration Assistant enables you to configure or create Oracle ASM disk groups with the **Configure ASM Disk Groups** tab.

> ⓘ **Note**
>
> Oracle ASM Filter Driver (ASMFD) is desupported on both Linux and Oracle Solaris beginning with Oracle AI Database 26ai.

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.

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 and drop disks from the disk group

- Add, drop, and list quota groups.

- Add, drop, and move file groups.

- Edit the disk group attributes

- Manage templates for the disk group

- Create an Oracle ACFS file system on the disk group for database use

- Mount and dismount the disk group

- Drop the disk group

- Drop and clear Oracle ASMLib labels

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.

On the **Create Disk Group** 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.

If Oracle ASMLib has been configured for your system, you can add Oracle ASMLib labels to disks or clear Oracle ASMLib labels from disks.

You can choose to show advanced options when creating a disk group. The advanced options section 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.

> ⓘ **Note**
>
> The values for **Free (GB)** and **Usable (GB)** space are `0` if the disk group type is `FLEX` or `EXTEND`. The `0` value in this case means no value is given, not a zero amount in for **Free (GB)** and **Usable (GB)** space.

> ⓘ **See Also**
>
> * Storing Oracle Cluster Registry and Voting Files in Oracle ASM Disk Groups for information about quorum failure groups
> * Disk Group Compatibility and Overview of Disk Group Compatibility for information about disk group compatibility attributes and advancing disk group compatibility
> * Oracle ASM Disk Discovery and ASM_DISKSTRING for information about disk discovery
> * Administering Oracle ASM Disk Groups for more information about managing disk groups
> * *Oracle AI Database Reference* for descriptions of disk group information displayed in the `V$ASM_DISKGROUP` view

ASMCA Command-Line Interface

The ASMCA command-line interface provides non-GUI support for configuring Oracle ASM instances, Oracle ASM disk groups, volumes, and Oracle ACFS.

* Running ASMCA Command-Line
* ASMCA Commands for Instances and Disk Groups

Running ASMCA Command-Line

The syntax for running the ASMCA command-line tool is:

```
asmca -silent command_name [options]
```

[Table 9-1](#) contains the options available with the `asmca` command-line tool.

> ### ⓘ See Also
>
> *Oracle Advanced Cluster File System Administrator's Guide* for information about managing Oracle ACFS and Oracle ADVM with ASMCA commands

Table 9-1   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`
	`-configureParameter`
	`-createDiskGroup`
	`-editDiskGroupAttributes`
	`-addDisk`
	`-createFileGroups`
	`-dropFileGroups`
	`—createQuotaGroups`
	`-dropQuotaGroups`
	`-createVolume`
	`-createACFS`
	`-createACFSSnapshot`
	`-dropACFSSnapshot`
	
	See [ASMCA Commands for Instances and Disk Groups](#).

`options`	Specifies the options to run for a command. The options vary by command. See the individual commands described in [ASMCA Commands for Instances and Disk Groups](#) 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 for Instances and Disk Groups

This section describes the commands that can be run with ASMCA command-line to manage Oracle ASM objects.

- [Configure an Oracle ASM Instance](#)

- [Configure Parameters for a Specified Instance](#)

- [Create a Disk Group](#)

- [Add a Disk to a Disk Group](#)

- [Edit Oracle ASM Disk Group Attributes](#)

- [Create Oracle ASM File Groups](#)

- [Drop Oracle ASM File Groups](#)

- [Create Oracle ASM Quota Groups](#)

- [Drop Oracle ASM Quota Groups](#)

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
        { -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 ] }
      { -executeRootScript {MANUAL|ROOT|SUDO} |
        [-rootPassword root_password ] |
        [-sudoUser sudo_user] [-sudoUserPassword sudo_user_password] }
      [-asmNetworks asm_network]
```

[Table 9-2](#) contains the options available with the `-configureASM` command.

Table 9-2   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.
`-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.
`-executeRootScript {`MANUAL|ROOT|SUDO`}`	Executes the root script. Specify `MANUAL` for a manual run, `ROOT` for the `root` password, or `SUDO` for the `sudo` password.
`-rootPassword` *`root_password`*	Specifies the password of the `root` user.
`-sudoUser` *`sudo_user`* `-sudoUserPassword` *`sudo_user_password`*	Runs `sudo` as the specified user with the specified password.
`-asmNetworks` *`asm_network`*	Specifies a comma separated list of Oracle ASM networks. Each network is specified in the form "*interface_name*/*subnet*". If not specified, then the existing private network is used as the Oracle ASM network.

The following are additional parameters used for an Oracle Flex ASM configuration when Oracle Cluster Registry (OCR) is in Network Attached Storage (NAS): `-executeRootScript`, `-rootPassword`, `-sudoUser -sudoUserPassword`, and `-asmNetworks`

For additional options that are common to multiple commands, see Table 9-1.

Examples

To configure an Oracle ASM instance:

Example 9-1   Using asmca -silent -configureASM

```
asmca -silent
        -configureASM
```

```
-diskString '/devices/disk*'
-sysAsmPassword my_sysasm_password
-asmsnmpPassword my_asmsnmp_passwd
```

Configure Parameters for a Specified Instance

`-configureParameter` modifies the parameters of the specified target instance.

Syntax

```
asmca -silent
        -configureParameter [-target {ASM|IOS|APX|ALL} ]
        [{-param paramname=paramvalue , ... }]
        [-sysAsmPassword sysasm_password ]
```

Table 9-3 contains the options available with the `-configureParameter` command.

Table 9-3   Options for the -configureParameter command

Option	Description
`-target {ASM|IOS|APX|ALL}`	Specifies the target instance of the command. If no target is specified, the default is the Oracle ASM instance.
`-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 9-1.

Examples

To configure parameters for a specified target instance:

Example 9-2   Using asmca -silent -configureParameter

```
asmca -silent
        -configureParameter
            -param
            asm_power_limit=3
        -sysAsmPassword my_sysasm_password
```

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 ]
        [-attribute <Tuple of the form <attribute name=attribute value> to specify a disk
group attribute.>]
```

Table 9-4 contains the options available with the -createDiskGroup command.

Table 9-4   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.
-attribute	Specifies a disk group attribute.

For additional options that are common to multiple commands, see Table 9-1.

For information about disk group compatibility attributes, see Disk Group Compatibility.

Examples

To create disk groups in an Oracle ASM instance:

Example 9-3   Using asmca -silent -createDiskGroup

```
$ asmca -silent -createDiskGroup
        -diskGroupName mynewdg
            -disk '/devices/diske*'
            -disk '/devices/diskk*'
        -redundancy NORMAL
        -au_size 64
        -compatible.asm '19.0.0.0.0'
        -compatible.rdbms '19.0.0.0.0'
        -compatible.advm '19.0.0.0.0'

DiskGroup mynewdg created successfully.
```

Edit Oracle ASM Disk Group Attributes

`-editDiskGroupAttributes` updates the Oracle ASM disk group attributes.

Syntax and Description

```
-editDiskGroupAttributes
      (-diskGroupName disk_group)
      (-attribute <Tuple of the form <attribute name=attribute value> to specify a disk
group attribute.>)
```

The following table contains the options available with the `—editDiskGroupAttributes` command.

Table 9-5   Options for the -editDiskGroupAttributes command

Option	Description
`-diskGroupName` *diskgroup*	Specifies the name of the Oracle ASM disk group to create.
`-attribute` *value*	Specifies the name and assigned value of an attribute.

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 9-6 contains the options available with the `-addDisk` command.

Table 9-6   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.

Table 9-6   (Cont.) Options for the -addDisk command

Option	Description
`-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 9-1.

Examples

To add disks to an existing disk group:

Example 9-4   Using asmca -silent -addDisk

```
asmca -silent
        -addDisk
          -diskGroupName mynewdg
              -disk '/devices/diskl1'
              -disk '/devices/diskl2'

Disks added successfully to diskgroup mynewdg.
```

Create Oracle ASM File Groups

`-createFileGroups` creates Oracle ASM file groups.

Syntax and Description

```
-createFileGroups
      (-diskGroupName disk_group)
      (-fileGroupNames file_group_name_list)
      (-fileGroupUsages file_group_usages_list)
      (-fileGroupUsageIDs file_group_usagesID_list)
```

The following table contains the options available with the `—createFileGroups` command.

Table 9-7   Options for the -createFileGroups command

Option	Description
`-diskGroupName` *`disk_group`*	Specifies the name of the Oracle ASM disk group.
`-fileGroupNames` *`file_group_name_list`*	Specifies a comma-delimited list of file group names to be added to the disk group.
`-fileGroupUsages` *`file_group_usages_list`*	Specifies a comma-delimited list of file group usage strings to be processed.
`-fileGroupUsagesIDs` *`file_group_usagesID_list`*	Specifies a comma-delimited list of file group usage IDs to be processed.

Drop Oracle ASM File Groups

`-dropFileGroups` drops Oracle ASM file groups.

Syntax and Description

```
-dropFileGroups
      (-diskGroupName disk_group)
      (-fileGroupNames file_group_name_list)
```

The following table contains the options available with the `-dropFileGroups` command.

Table 9-8   Options for the -dropFileGroups command

Option	Description
`-diskGroupName disk_group`	Specifies the name of the Oracle ASM disk group to create.
`-fileGroupNames file_group_name_list`	Specifies a comma-delimited list of file group names to be dropped.

Create Oracle ASM Quota Groups

`-createQuotaGroups` creates Oracle ASM quota groups.

Syntax and Description

```
-createQuotaGroups
      (-diskGroupName disk_group)
      (-quotaGroupNames quota_group_name_list)
      (-quotaGroupSizes quota_group_size_list)
```

The following table contains the options available with the `-createQuotaGroups` command.

Table 9-9   Options for the -createQuotaGroups command

Option	Description
`-diskGroupName disk_group`	Specifies the name of the Oracle ASM disk group.
`-quotaGroupNames quota_group_name_list`	Specifies a comma-delimited list of quota group names to be added to the disk group.
`-quotaGroupSizes quota_group_size_list`	Specifies a comma-delimited list of quota group sizes to be processed.

Drop Oracle ASM Quota Groups

`-dropQuotaGroups` drops Oracle ASM quota groups.

Syntax and Description

```
-dropQuotaGroups
      (-diskGroupName disk_group)
      (-quotaGroupNames quota_group_names_list)
```

The following table contains the options available with the `-dropQuotaGroups` command.

Table 9-10   Options for the -dropQuotaGroups command

Option	Description
-diskGroupName *disk_group*	Specifies the name of the Oracle ASM disk group.
-quotaGroupNames *quota_group_name_list*	Specifies a comma-delimited list of quota group names to be dropped.

10

Managing Oracle ASM with ASMCMD

Oracle Automatic Storage Management (Oracle ASM) Command-Line Utility (ASMCMD) provides commands for managing Oracle ASM.

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 Client Cluster Management Commands](#)
- [ASMCMD Audit Files Management Commands](#)
- [ASMCMD File Management Commands](#)
- [ASMCMD Disk Group Management Commands](#)
- [ASMCMD File Group Management Commands](#)
- [ASMCMD Template Management Commands](#)
- [ASMCMD File Access Control Commands](#)

> ⓘ **See Also**
>
> *Oracle Advanced Cluster File System Administrator's Guide* for information about ASMCMD commands to manage Oracle ADVM volumes

> ⓘ **Note**
>
> Oracle does not recommend using identifiers for Oracle AI Database object names that must be quoted. While the use of quoted identifiers may be valid as names in some command-line tools or in SQL statements, such as `CREATE DISKGROUP "1DATA"`, the names may not be valid when using other tools that manage the object. For more information about naming Oracle AI Database objects, refer to *Oracle AI Database SQL Language Reference*.

ASMCMD Audit Files Management Commands

This document describes the ASMCMD Oracle ASM audit files management commands.

The following table provides a summary of ASMCMD audit files management commands.

NOT_SUPPORTED

Command	Description
audcleanaudittrail	Deletes the audit files of the specified audit trail type.
audclearproperty	Clears the value for the specified audit property.
audcleartimestamp	Clears the last archive timestamp set by "`audsettimestamp`" command.
audcreatejob	Creates a purge job for specified audit trail type.
auddropjob	Drops the specified audit purge job.
audsetdebug	Sets the tracing to the specified debug level.
audsetjobinterval	Sets the job interval for the specified audit purge job.
audsetjobstatus	Sets the status of the specified purge job to either "`enable`" or "`disable`" state.
audsetproperty	Sets the value of specified audit property for given audit trail type.
audsettimestamp	Sets the last archive timestamp for specified audit trail type.
audshowtimestamp	Displays the last archive timestamps set by "`audsettimestamp`" command.
lsaudcleanupjobs	Displays the configured audit trail purge jobs.
lsaudconfigparams	Displays the configured audit trail properties for the given audit trail type.

audcleanaudittrail

Purpose

Deletes the audit files of the specified audit trail type. The command will delete the files based on Last Archive Timestamp set using "`audsettimestamp`" command, if `--useTimestamp` option is specified. Else, it will delete all the files that are not being actively written to.

Syntax and Description

`audcleanaudittrail {--os|--uni} [--useTimestamp] [--target <target_instance>]`

Options for the `audcleanaudittrail` command

Table 10-2   Options for the audcleanaudittrail command

Option	Description
`--os`	Cleans up audit files of OS audit trail type.
`--uni`	Cleans up audit files of Unified audit trail type.
`--useTimestamp`	If specified, uses last archive timestamp in cleaning up the audit records.
`--target`	Target instance where the command should be executed. The value could be either IOS or APX.

Examples

The following are examples of `audcleanaudittrail` command.

In the example below, the audit files of OS audit trail type that are older than the timestamp specified by "`audsettimestamp`" command will be deleted.

Example 10-1   Using the ASMCMD audcleanaudittrail command

```
ASMCMD> audcleanaudittrail --os --useTimestamp
```

In the example below, the audit files of Unified audit trail type that are older than the timestamp specified by "`audsettimestamp`" command will be deleted

Example 10-2   Using the ASMCMD audcleanaudittrail command

```
ASMCMD> audcleanaudittrail --uni --useTimestamp
```

In the example below, the audit files of Unified audit trail type in APX instance that are older than the timestamp, set using "audsettimestamp" command, will be deleted.

Example 10-3   Using the ASMCMD audcleanaudittrail command

```
ASMCMD> audcleanaudittrail --uni --useTimestamp --target APX
```

audclearproperty

Purpose

Clears the value for the specified audit property. When `--use_def` argument is used, the value of the property is set to its respective default value.

Syntax and Description

```
audclearproperty {--file_max_size|--file_max_age} [--use_def] {--os|--uni} [--target
<target_instance>]
```

Options for the `audclearproperty` command

Table 10-3   Options for the audclearproperty command

Option	Description
`--file_max_size`	Maximum audit file size ⓘ **Note** For more details, see [Administering Oracle ASM audit trail](#)
`--file_max_age`	Maximum audit file age ⓘ **Note** For more details, see [Administering Oracle ASM audit trail](#)
`--use_def`	Resets or sets the value back to default value.
`--os`	Clears property of OS audit trail type.
`--uni`	Clears property of Unified audit trail type.

Table 10-3   (Cont.) Options for the audclearproperty command

Option	Description
`--target`	Target instance where the command should be executed. The value could be either APX or IOS.

> ⓘ **Note**
>
> The default value for the age of an audit file is 5 Days and the default value for the size of an audit file is 10 MB.

Examples

The following are examples of `audclearproperty` command.

The example below sets the `file_max_size` property for the audit files of OS audit trail type to its default value.

Example 10-4   Using the ASMCMD audclearproperty command

```
ASMCMD> audclearproperty --file_max_size --use_def --os
```

The example below clears the file_max_age property for the audit files of Unified audit trail type in APX instance. In other words, it sets `file_max_age` property to zero.

Example 10-5   Using the ASMCMD audclearproperty command

```
ASMCMD> audclearproperty --file_max_age --uni --target APX
```

audcleartimestamp

Purpose

Clears the last archive timestamp set by audsettimestamp.

Syntax and Description

```
audcleartimestamp {--os  --inst <instancenumber> |--uni} [--target <target_instance>]
```

Options for the `audcleartimestamp` command

Table 10-4   Options for the audcleartimestamp command

Option	Description
`--os`	Clears the last archive timestamp set for OS audit trail type.
`--uni`	Clears the last archive timestamp set for Unified audit trail type.
`--inst <instancenumber>`	Instance number
`--target`	Target instance where the command should be executed. The value could be either APX or IOS.

Examples

The following are examples of `audcleartimestamp` command.

The example below clears the last archive timestamp set for instance number 1 of OS audit trail type.

Example 10-6   Using the ASMCMD audcleartimestamp command

```
ASMCMD> audcleartimestamp --os --inst 1
```

The example below clears the last archive timestamp set for Unified audit trail type.

Example 10-7   Using the ASMCMD audcleartimestamp command

```
ASMCMD> audcleartimestamp --uni
```

The example below clears the timestamp set for Unified audit trail type in IOS instance.

Example 10-8   Using the ASMCMD audcleartimestamp command

```
ASMCMD> audcleartimestamp --uni --target IOS
```

audcreatejob

Purpose

Creates a purge job for specified audit trail type. The purge job runs at a frequency specified by `--int` and purges the audit files.

Syntax and Description

```
audcreatejob --name <job_name> --int <purge_int> {--os | --uni} [--use_ts] [--target
<target_instance>]
```

Options for the `audcreatejob` command.

Table 10-5   Options for the audcreatejob command

Option	Description
`--name <job_name>`	Name of the purge job
`--int <purge_int>`	Purge interval in hours
`--os`	Purge job for OS audit trail type
`--uni`	Purge job for Unified audit trail type
`--use_ts`	When specified, the purge will be based on the last archive timestamp set by audsettimestamp command.
`-target`	target instance where the command should be executed. The value could be either APX or IOS.

Examples

The following are examples for `audcreatejob` command.

The example below creates a job called `OSAUD_PURGE` with purge interval 10 hours for OS audit trail type and the job will use the timestamp, set earlier using `audsettimestamp`, while purging audit files (since `--use_ts` is specified).

Example 10-9   Using the ASMCMD audcreatejob command

```
ASMCMD> audcreatejob --name OSAUD_PURGE --int 10 --os --use_ts
```

The example below creates a job called `OSAUD_NOTS` with purge interval 10 hours for OS audit trail type and the job will NOT use the timestamp, set earlier using `audsettimestamp`, while purging audit files (since `--use_ts` is NOT specified).

Example 10-10   Using the ASMCMD audcreatejob command

```
ASMCMD> audcreatejob --name OSAUD_NOTS --int 10 -os
```

The example below creates a job called `OSAUD_PURGE_APX` in APX instance with purge interval 10 hours for OS audit trail type and the job will not use the timestamp while purging audit files (since --use_ts is not specified).

Example 10-11   Using the ASMCMD audcreatejob command

```
ASMCMD> audcreatejob --name OSAUD_PURGE_APX --int 10 --os --target APX
```

auddropjob

Purpose

Drops the specified audit purge job. This command does not require any audit trail type information.

Syntax and Description

```
auddropjob --name <job_name> [--target <target_instance>]
```

Options for the `auddropjob` command.

Table 10-6   Options for the auddropjob command

Option	Description
`--name <job_name>`	Name of the purge job
`-target`	target instance where the command should be executed. The value could be either APX or IOS.

Examples

The below example drops the audit purge job `OSAUD_PURGE_APX` in APX instance.

Example 10-12   Using the ASMCMD auddropjob command

```
ASMCMD> auddropjob --name OSAUD_PURGE_APX --target APX
```

audsetdebug

Purpose

Sets the tracing to the specified debug level.

Syntax and Description

```
audsetdebug {--debug | --error} [--target <target_instance>]
```

Options for the `audsetdebug` command.

Table 10-7   Options for the audsetdebug command

Option	Description
--debug	Dumps all debug information into the traces.
--error	Dumps only error information into the traces.
-target	target instance where the command should be executed. The value could be either APX or IOS.

> ⓘ **Note**
>
> The default debug level is "error".

Examples

The following example sets the debug level to "`debug`".

Example 10-13   Using the ASMCMD audsetdebug command

```
ASMCMD> audsetdebug --debug
```

The following example sets the debug level to "debug" in IOS instance.

Example 10-14   Using the ASMCMD audsetdebug command

```
ASMCMD> audsetdebug --debug --target IOS
```

audsetjobinterval

Purpose

Sets the job interval for the specified audit purge job.

Syntax and Description

```
audsetjobinterval --name <job_name>  --int <interval> [--target <target_instance>]
```

Options for the `audsetjobinterval` command.

Table 10-8   Options for the audsetjobinterval command

Option	Description
--name *<job_name>*	Name of audit purge job
--int *<interval>*	Interval in hours
-target	target instance where the command should be executed. The value could be either APX or IOS.

Examples

The following example sets the interval of audit cleanup job `OSAUD_PURGE` to 10 hours.

Example 10-15   Using the ASMCMD audsetjobinterval command

```
ASMCMD> audsetjobinterval --name OSAUD_PURGE --int 10
```

The following example example sets the interval of audit cleanup job `OSAUD_PURGE_APX` in APX instance to 20 hours.

Example 10-16   Using the ASMCMD audsetjobinterval command

```
ASMCMD> audsetjobinterval --name OSAUD_PURGE_APX --int 20 --target APX
```

audsetjobstatus

Purpose

Sets the status of the specified purge job to enabled or disabled state.

Syntax and Description

```
audsetjobstatus --name <job_name>  {--disable | --enable} [--target <target_instance>]
```

Options for the `audsetjobstatus` command.

Table 10-9   Options for the audsetjobstatus command

Option	Description
`--name <job_name>`	Audit purge job name
`--enable/--disable`	Enables or disables the given job.
`-target`	target instance where the command should be executed. The value could be either APX or IOS.

> ⓘ **Note**
>
> Audit purge job, when disabled, will not purge audit files periodically.

Examples

The following example disables the audit job `OSAUD_PURGE`.

Example 10-17   Using the ASMCMD audsetjobstatus command

```
ASMCMD> audsetjobstatus --name OSAUD_PURGE --disable
```

The following example disables the audit job `OSAUD_PURGE_APX` in APX instance.

Example 10-18   Using the ASMCMD audsetjobstatus command

```
ASMCMD> audsetjobstatus --name OSAUD_PURGE_APX --disable --target APX
```

audsetproperty

Purpose

Sets the value of specified audit property for given audit trail type.

Syntax and Description

```
audsetproperty {--file_max_size | --file_max_age } --val <value> {--os | --uni} [--target <target_instance>]
```

Options for the `audsetproperty` command.

Table 10-10   Options for the audsetproperty command

Option	Description
`--file_max_size`	Maximum size of the audit file
	ⓘ **Note**
	The value for `file_max_size` property must be in range [1,2000000] KB.
`--file_max_age`	Maximum age of the audit file
	ⓘ **Note**
	The value for `file_max_age` property must be in range [1, 497] days.
`--val <value>`	Value of the property
`--os`	Sets property for OS audit trail type.
`--uni`	Sets property for Unified audit trail type.
`-target`	target instance where the command should be executed. The value could be either APX or IOS.

Examples

The following are examples of the `audsetproperty` command.

The example below sets the `file_max_size` property for the audit files of OS audit trail type audit file to 10 KB.

Example 10-19   Using the ASMCMD audsetproperty command

```
ASMCMD> audsetproperty --file_max_size --val 10 --os
```

The example below sets the `file_max_size property` in IOS instance for audit files of Unified audit trail type to 100 KB.

Example 10-20   Using the ASMCMD audsetproperty command

```
ASMCMD> audsetproperty --file_max_size --val 100 --uni --target IOS
```

audsettimestamp

Purpose

Sets the last archive timestamp for specified audit trail type. This timestamp will be used by the audit purge job and `audcleanaudittrail` command, if they are configured with `--useTimestamp` option. In case of OS audit trail type, instance number is required.

Syntax and Description

```
audsettimestamp {--os|--uni} [--inst <id>] <timestamp> [--target <target_instance>]
```

Options for the `audsettimestamp` command.

Table 10-11   Options for the audsettimestamp command

Option	Description
`--os`	Last archive timestamp for OS audit trail type
`--uni`	Last archive timestamp for Unified audit trail type
`--inst <id>`	Instance number
`<timestamp>`	Timestamp
`-target`	target instance where the command should be executed. The value could be either APX or IOS.

Examples

The following example sets the last archive timestamp of OS audit trail type in instance number 1.

Example 10-21   Using the ASMCMD audsettimestamp command

```
ASMCMD> audsettimestamp --os --inst 1 22-AUG-2012 07:48:53
```

The following example sets the timestamp of Unified audit trail type in IOS instance.

Example 10-22   Using the ASMCMD audsettimestamp command

```
ASMCMD> audsettimestamp --uni 01-JAN-2010 06:00:00 --target IOS
```

audshowtimestamp

Purpose

Displays the last archive timestamp set using "`audsettimestamp`" command.

Syntax and Description

```
audshowtimestamp [{--os|--uni}] [-g] [--target <target_instance>]
```

Options for the `audshowtimestamp` command.

Table 10-12   Options for audshowtimestamp command

Option	Description
`--os`	Displays last archive timestamp set for OS audit trail type.
`--uni`	Displays last archive timestamp set for Unified audit trail type.
`-g`	Displays clusterwide timestamp information.
`-target`	target instance where the command should be executed. The value could be either APX or IOS.

Examples

The following are examples of the `audshowtimestamp` command.

The example below displays the last archive timestamp set for OS audit trail type.

Example 10-23   Using the ASMCMD audshowtimestamp command

```
ASMCMD> audshowtimestamp --os
```

The example below displays the clusterwide timestamp information for OS audit trail type.

Example 10-24   Using the ASMCMD audshowtimestamp command

```
ASMCMD> audshowtimestamp --os -g
```

The example below displays the clusterwide timestamp information for OS audit trail type.

Example 10-25   Using the ASMCMD audshowtimestamp command

```
ASMCMD> audshowtimestamp --os -g
```

> ⓘ **See Also**
>
> • [Database Reference](#) for `V$ASM_AUDIT_LAST_ARCH_TS`

lsaudcleanupjobs

Purpose

Displays the configured audit trail purge jobs.

Syntax and Description

```
lsaudcleanupjobs [--os |--uni] [--target <target_instance>]
```

Options for the `lsaudcleanupjobs` command.

Table 10-13   Options for lsaudcleanupjobs command

Option	Description
`--os`	Displays only the purge job configured for OS audit trail type.
`--uni`	Displays only the purge job configured for Unified audit trail type.
`-target`	target instance where the command should be executed. The value could be either APX or IOS.

> ⓘ **Note**
>
> `lsaudcleanupjobs` displays the purge jobs configured for both OS and Unified audit trail types, when no audit trail option is provided.

Examples

The following example queries audit cleanup jobs configured for OS audit trail type.

Example 10-26   Using the ASMCMD lsaudcleanupjobs command

```
ASMCMD> lsaudcleanupjobs --os
```

The following example queries audit cleanup jobs configured for OS audit trail type in IOS instance.

Example 10-27   Using the ASMCMD lsaudcleanupjobs command

```
ASMCMD> lsaudcleanupjobs --uni --target IOS
```

> ⓘ **See Also**
>
> • [Database Reference](#) for `V$ASM_AUDIT_CLEANUP_JOBS`

lsaudconfigparams

Purpose

Displays the configured audit trail properties for the given audit trail type.

Syntax and Description

```
lsaudconfigparams [{--os|--uni}] [--target <target_instance>]
```

Options for the `lsaudconfigparams` command.

Table 10-14   Options for lsaudconfigparams command

Option	Description
`--os`	Displays only the audit trail properties of OS audit trail type.
`--uni`	Displays only the audit trail properties of unified audit trail type.
`-target`	target instance where the command should be executed. The value could be either APX or IOS.

> ⓘ **Note**
>
> `lsaudconfigparams` displays the audit configuration properties for both OS and Unified audit trail type, when no audit trail type option is provided.

Examples

The following example queries audit configuration parameters for OS audit trail type.

Example 10-28   Using the ASMCMD lsaudconfigparams command

```
ASMCMD> lsaudconfigparams --os
```

The following example queries audit configuration parameters for Unified audit trail type in IOS instance.

Example 10-29   Using the ASMCMD lsaudconfigparams command

```
ASMCMD> lsaudconfigparams --uni --target IOS
```

> ⓘ **See Also**
>
> • [Database Reference](#) for `V$ASM_AUDIT_CONFIG_PARAMS`

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, File Names, 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 10-15](#).

Table 10-15   Types of ASMCMD commands

Category	Commands
Oracle ASM instance management	For a list of commands, refer to [ASMCMD Instance Management Commands](#)
Oracle ASM client cluster	For a list of commands, refer to [ASMCMD Client Cluster Management Commands](#)
Oracle ASM Audit Files management	For a list of commands, refer to [ASMCMD Audit Files Management Commands](#)
Oracle ASM file management	For a list of commands, refer to [ASMCMD File Management Commands](#)
Oracle ASM disk group management	For a list of commands, refer to [ASMCMD Disk Group Management Commands](#)
Oracle ASM file group management	For a list of commands, refer to [ASMCMD File Group Management Commands](#)
Oracle ASM template management	For a list of commands, refer to [ASMCMD Template Management Commands](#)

Table 10-15   (Cont.) Types of ASMCMD commands

Category	Commands
Oracle ASM file access control	For a list of commands, refer to <u>ASMCMD File Access Control Commands</u>

> ### ⓘ See Also
>
> *Oracle Advanced Cluster File System Administrator's Guide* for information about ASMCMD commands to manage Oracle ADVM volumes

About Oracle ASM Files, File Names, 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:

- <u>System-Generated File Name or Fully Qualified File Name</u>
- <u>Directory</u>
- <u>Alias</u>
- <u>Absolute Path and Relative Path</u>
- <u>Wildcard Characters</u>

System-Generated File Name or Fully Qualified File Name

Every file created in Oracle ASM gets a system-generated file name, otherwise known as a fully qualified file name. This is similar to a complete path name in a local file system.

Oracle ASM generates file names according to the following scheme:

`+diskGroupName/databaseName/fileType/fileTypeTag.fileNumber.incarnation`

An example of a fully qualified file name is the following:

`+data/orcl/CONTROLFILE/Current.256.541956473`

In the previous fully qualified file name, `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. File Names 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 file names, refer to <u>Fully Qualified File Name Form</u>.

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 file name 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 file names 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 file name 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.

```
ASMCMD [+] > cd data/orcl
ASMCMD [+data/orcl] >
```

ASMCMD retains the case of the directory that you entered.

Alias

Aliases are file names that are references or pointers to system-generated file names. 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 file name 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 File Name Forms](#).

Absolute Path and Relative Path

When you run an ASMCMD command that accepts a file name 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 file name, otherwise known as the fully qualified file name, 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 file name:

```
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 file name 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, such as `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.

If you are running ASMCMD commands in noninteractive mode, then with most operating systems you must enclose the wildcard characters in quotes. For more information, refer to Running ASMCMD in Noninteractive Mode.

Example 10-30 illustrates the use of wildcards.

Example 10-30   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 should review the information in this topic before running the ASMCMD utility.

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.

- 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.

  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 `+ASMnode#`.

- 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 AI 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.

  With this connection, there is a limited set of operations that can be run.

  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.

> ⓘ **See Also**
>
> - [Authentication for Accessing Oracle ASM Instances](#) for more information about authentication for accessing Oracle ASM instances
>
> - [Specifying the Type of Connection](#) for more information about specifying connection options
>
> - [The SYSDBA Privilege for Managing Oracle ASM Components](#) for more information about permissions and connections
>
> - *Oracle AI Database Administrator's Guide* for more information about setting environment variables
>
> - *Oracle AI Database Administrator's Guide* for information about operating system authentication

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.

This section contains the following topics:

- [Syntax for Starting ASMCMD](#)
- [Running ASMCMD without Any Options](#)
- [Specifying the Type of Connection](#)
- [Specifying the Verbose Mode](#)
- [Including the Current Directory in the ASMCMD Prompt](#)
- [Displaying the ASMCMD Version Number](#)

Syntax for Starting ASMCMD

The syntax for starting ASMCMD in interactive mode is:

```
asmcmd [-V]
asmcmd [--nocp] [-v {errors|warnings|normal|info|debug}]
       [--privilege connection_type ] [-p]
       [--inst instance_name] [--discover]
```

[Table 10-16](#) summarizes the ASMCMD options.

Table 10-16   ASMCMD interactive options

Option	Description
-V	Displays the version of ASMCMD and then exits.
--nocp	Disables connection pooling. For information about connection pooling, refer to *Oracle AI Database Administrator's Guide*.
-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.

Table 10-16   (Cont.) ASMCMD interactive options

Option	Description
`--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.
`--inst` `instance_name`	Specifies to connect to a specific Oracle ASM instance.
`--discover`	Uses the discovery string obtained from listener, not from the `ORACLE_SID` environmental variable.

The `ORACLE_HOME` and `ORACLE_SID` environment variables determine the instance to which ASMCMD connects. ASMCMD establishes a bequeath connection in the same manner as `SQLPLUS / AS SYSASM` which requires the user to be a member of the `OSASM` group.

When Oracle Flex ASM is enabled, ASMCMD connects to any one of the Oracle ASM instances running in the cluster. The connection to the Oracle ASM instance does not depend on the `ORACLE_HOME` and `ORACLE_SID` environment variables. The ASMCMD alert log shows which instance ASMCMD is connected to. If you want to connect to a specific Oracle ASM instance, then use the `--inst` option.

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 warnings`, 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 `$ORACLE_BASE/diag/asmcmd/user_`*`username`*`/ `*`hostname`*`/alert/` directory. By default, the `trace.trc` file is created in the `$ORACLE_BASE/ 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 19.0.0.0.0
```

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.

On most operating systems, wildcard characters must be enclosed in quotes when running ASMCMD commands in noninteractive mode.

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 10-17   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 10-31 shows how to run ASMCMD in the noninteractive mode. The first command runs the `ls` command to list the disk groups for the Oracle ASM instance. The second command redirects the output of the `lsod` command to the `my_lsod_test` file. The third command runs `lsdsk` using a pattern with a wildcard character to list specific disks in the `data` disk group. Note that wildcard characters must be enclosed in quotes when running ASMCMD in noninteractive mode on most operating systems.

Example 10-32 raises an error with an incorrect disk group name and the return code equal (255) is displayed.

Example 10-33 shows an example of ASMCMD commands run in noninteractive mode inside a script.

Example 10-31   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 10-32   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 10-33   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 10-34   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 topic provides a summary of the ASMCMD 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.

> ⓘ **Note**
>
> After a change has been made to the location of the password file, the following SQL statement should be executed to flush the password file metadata cache.
>
> ```
> SQL> ALTER SYSTEM FLUSH PASSWORDFILE_METADATA_CACHE
> ```
>
> The location of the password file can be changed by running `orapwd`, or the ASMCMD `pwcopy`, `pwcreate`, `pwdelete`, `pwmove`, or `pwset` command.

> ⓘ **See Also**
>
> *Oracle Grid Infrastructure Installation and Upgrade Guide* for your operating system for information about installing and configuring Oracle Grid Infrastructure

Table 10-18lists the Oracle ASM instance management commands with brief descriptions.

Table 10-18   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.
pwcopy	Copies a password file to the specified location.
pwcreate	Creates a password file at the specified location.
pwdelete	Deletes a password file at the specified location.
pwget	Returns the location of the password file.
pwmove	Moves the location of the password file.
pwset	Sets the location of the password file.
showclustermode	Displays the current mode of the Oracle ASM cluster.
showclusterstate	Displays the current state of the cluster.
showpatches	Lists the patches applied to the Oracle Grid Infrastructure home.
showversion	Displays the Oracle ASM cluster release and software patch levels.
shutdown	Shuts down an instance.
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 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 10-19.

Table 10-19   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 10-35   Using the ASMCMD dsget command

```
ASMCMD [+] > dsget
profile: /devices1/disk*
parameter: /devices2/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 10-20.

Table 10-20   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.

Table 10-20   (Cont.) Options for the dsset command

Option	Description
`--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.

Example

The following example uses `dsset` to set the current value of the discovery diskstring in the GPnP profile.

Example 10-36   Using the ASMCMD dsset command

```
ASMCMD [+] > dsset /devices1/disk*,/devices2/disk*
```

lsct

Purpose

Lists information about current Oracle ASM clients from the `V$ASM_CLIENT` view. A client, such as Oracle AI 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] disk_group
```

The following table lists the options for the `lsct` command.

Table 10-21   Options for the lsct command

Option	Description
`-g`	Selects from the `GV$ASM_CLIENT` view. `GV$ASM_CLIENT.INST_ID` is included in the output.
`--suppressheader`	Suppresses column headings.
disk_group	Specifies the disk group.

Client information is only listed for the specified disk group.

Example

The following example displays information about the clients that are accessing the `data` disk group.

Example 10-37   Using the ASMCMD lsct command

```
ASMCMD [+] > lsct data
DB_Name   Status      Software_Version   Compatible_version   Instance_Name   Disk_Group
+ASM      CONNECTED          19.0.0.0.0           19.0.0.0.0   +ASM            DATA
asmvol    CONNECTED          19.0.0.0.0           19.0.0.0.0   +ASM            DATA
orcl      CONNECTED          19.0.0.0.0           19.0.0.0.0   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 10-38   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 10-22 lists the options for the `lspwusr` command.

Table 10-22   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 10-39   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 | --delete | grant {sysasm|sysdba|sysoper} |
          --revoke {sysasm|sysdba|sysoper} } user
```

Table 10-23 lists the options for the `orapwusr` command.

Table 10-23   Options for the orapwusr command

Option	Description
`--add`	Adds a user to the password file. Also prompts for a password.
`--modify`	Changes the password for the specified user.
`--delete`	Drops a user from the password file.
`--grant role`	Sets the role for the user. The options are `sysasm`, `sysdba`, and `sysoper`.
`--revoke role`	Removes the role for the user. The options are `sysasm`, `sysdba`, and `sysoper`.
`user`	Name of the user to add, drop, or modify.

`orapwusr` attempts to update passwords on all nodes in a cluster. This 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.

Example 10-40   Using the ASMCMD orapwusr command

```
ASMCMD [+] > orapwusr --add hruser
```

pwcopy

Purpose

Copy a password file from one disk group to another, from the operating system to a disk group, or from a disk group to the operating system.

Syntax and Description

```
pwcopy [--asm | --dbuniquename <db>] source destination [-f] [--local]
```

lists the options for the `pwcopy` command.

Table 10-24   Options for the pwcopy command

Option	Description
`--asm`	The `--asm` option associates the password file with the Oracle ASM instance that ASMCMD is logged into.
`--dbuniquename` *string*	The `--dbuniquename` *string* option identifies the database unique name associated with the password file.
`--local`	Copy the password file locally. This option is used when copying the file from an ASM disk group to an operating system file system, or vice versa, and when both the operating system file system and the ASM instance reside in different nodes.
source	The *source* value identifies the location where the existing password file is located.
destination	The *destination* value identifies the location where you want to copy the password file.
`-f`	The `-f` option enables the password file to be copied without any checks.

`pwcopy` copies a password file from one disk group to another, from the operating system to a disk group, or from a disk group to the operating system. If the `-f` option is specified, then the password file can be copied to the same disk group.

Either `--asm` or `--dbuniquename` is required to identify a CRSD resource. When either `--asm` or `--dbuniquename` is included in the `pwcopy` command, the target file is set to the current password file.

The `compatible.asm` disk group attribute must be set to `12.1` or higher for the disk group where the password is to be copied.

The `SYSASM` or `SYSDBA` privilege is required to manage the Oracle ASM and database password files.

Example

The following example copies an Oracle ASM password file in one disk group to a different disk group. Because the `pwcopy` command includes the `--asm` option, the target file (`+fra/orapwasm_new`) is set to the current password file.

Example 10-41   Using the ASMCMD pwcopy command

```
ASMCMD [+] > pwcopy --asm +DATA/orapwasm +FRA/orapwasm_new
copying +DATA/orapwasm -> +FRA/orapwasm_new
```

pwcreate

Purpose

> ⓘ **Note**
>
> Do not use `pwcreate` to create a new Oracle ASM password file when Oracle ASM is configured in a cluster. If the Oracle ASM password file is inaccessible in a cluster configuration, you must restore the password file using an available backup rather than create a new file. For articles about recreating a shared Oracle ASM password file in a cluster, such as document 1929673.1, refer to articles at My Oracle Support (`https://support.oracle.com`).

Creates an Oracle ASM or Oracle AI Database instance password file at the specified location.

Syntax and Description

```
pwcreate { --asm  |--dbuniquename string } [-f] [--format format] file_path
[sys_password]
```

Table 10-25 lists the options for the `pwcreate` command.

Table 10-25   Options for the pwcreate command

Option	Description
`--asm`	Associates the password file with the Oracle ASM instance that ASMCMD is logged into.
`--dbuniquename string`	Specifies the Oracle AI Database unique name associated with the password file.
`-f`	Forces a deletion of the existing password file and creates a new password file.
`--format format`	Specifies the format in which the password is created. Values are `12` and `12.2`. If not specified, then `12.2` is the default value.
`file_path`	Specifies the location where the password file is created.
`sys_password`	Specifies the initial `SYS` password. If this option is omitted, then the you are prompted for the password.

`pwcreate` creates a password file in the disk group specified by `file_path` . The initial `SYS` password is specified by `sys_password`.

Either `--asm` or `--dbuniquename` is required. The `compatible.asm` disk group attribute must be set to `12.1` or higher for the disk group where the password is to be located.

The `SYSASM` or `SYSDBA` privilege is required to manage the Oracle ASM and database password files.

For additional information, refer to Managing a Shared Password File in a Disk Group.

Example

The following example creates an Oracle AI Database password file in an Oracle ASM disk group.

Example 10-42   Using the ASMCMD pwcreate command

```
ASMCMD [+] > pwcreate --dbuniquename orcl '+data/ORCL/orapwdb'
Enter password for SYS:
```

pwdelete

Purpose

Deletes an Oracle ASM or database instance password file.

Syntax and Description

```
pwdelete {--asm | --dbuniquename string | file_path } [-f]
```

Table 10-26 lists the options for the `pwdelete` command.

Table 10-26   Options for the pwdelete command

Option	Description
`--asm`	The `--asm` option associates the password file with the Oracle ASM instance that ASMCMD is logged into.
`--dbuniquename string`	The `--dbuniquename string` option identifies the database unique name associated with the password file.
`-f`	Force option, the specified password file will be deleted without any checks.
`file_path`	The `file_path` value identifies the location where the password file is located.

`pwdelete` deletes the specified password file. Either `--asm` or `--dbuniquename` is required to identify a CRSD resource and to remove the password location from the CRSD resource.

The `SYSASM` or `SYSDBA` privilege is required to manage the Oracle ASM and database password files.

Example

The following example deletes the specified password file from a disk group.

Example 10-43   Using the ASMCMD pwdelete command

```
ASMCMD [+] > pwdelete +FRA/orapwasm_bak
```

pwget

Purpose

Returns the location of the password file for the Oracle ASM or database instance.

Syntax and Description

```
pwget { --asm | --dbuniquename string }
```

[Table 10-27](#) lists the options for the `pwget` command.

Table 10-27   Options for the pwget command

Option	Description
`--asm`	The `--asm` option associates the password file with the Oracle ASM instance that ASMCMD is logged into.
`--dbuniquename string`	The `--dbuniquename string` option identifies the database unique name associated with the password file.

`pwget` returns the location of the password file for the Oracle ASM instance identified by `--asm` or the database instance identified by `--dbuniquename`.

The `SYSASM` or `SYSDBA` privilege is required to manage the Oracle ASM and database password files.

Example

The following example returns the location of the Oracle ASM password file.

Example 10-44   Using the ASMCMD pwget command

```
ASMCMD [+] > pwget --asm
+DATA/orapwasm
```

pwmove

Purpose

Moves an Oracle ASM or database instance password file to the specified location.

Syntax and Description

```
pwmove { --asm | --dbuniquename string } source  destination [-f]
```

[Table 10-28](#) lists the options for the `pwmove` command.

Table 10-28   Options for the pwmove command

Option	Description
`--asm`	The `--asm` option associates the password file with the Oracle ASM instance that ASMCMD is logged into.
`--dbuniquename string`	The `--dbuniquename string` option identifies the database unique name associated with the password file.
`source`	The `source` value identifies the location where the existing password file is located.
`destination`	The `destination` value identifies the location where you want to move the password file.
`-f`	The `-f` option clears the password file for any associated resource and the new file is registered.

pwmove moves a password file from one disk group to another, from the operating system to a disk group, or from a disk group to the operating system. If the -f option is specified, then a password file can be moved to a file in the same disk group.

Either --asm or --dbuniquename is required to identify a CRSD resource.

The compatible.asm disk group attribute must be set to 12.1 or higher for the disk group where the password is to be moved.

The SYSASM or SYSDBA privilege is required to manage the Oracle ASM and database password files.

Example

The following example moves a password file from one disk group to another disk group.

Example 10-45   Using the ASMCMD pwmove command

```
ASMCMD [+] > pwmove --asm +FRA/orapwasm_bak +DATA/orapwasm
moving +FRA/orapwasm_bak -> +DATA/orapwasm
```

pwset

Purpose

Sets the location of the password file for an Oracle ASM or database instance.

Syntax and Description

```
pwset {--asm | --dbuniquename string} file_path [-f]
```

Table 10-29 lists the options for the pwset command.

Table 10-29   Options for the pwset command

Option	Description
--asm	The --asm option associates the password file with the Oracle ASM instance that ASMCMD is logged into.
--dbuniquename string	The --dbuniquename string option identifies the database unique name associated with the password file.
-f	Force option, the resource will be updated without any checks.
file_path	The file_path value identifies the location where the password file is located.

pwset sets the location of the password file for an Oracle ASM or database instance to the value specified by file_path. Either --dbuniquename or --asm is required to identify a CRSD resource.

The SYSASM or SYSDBA privilege is required to manage the Oracle ASM and database password files.

Example

The following example sets the location of the Oracle ASM password file in a disk group.

Example 10-46   Using the ASMCMD pwset command

```
ASMCMD [+] > pwset --asm +DATA/orapwasm
```

showclustermode

Purpose

Displays the current mode of the Oracle ASM cluster.

Syntax and Description

```
showclustermode
```

`showclustermode` displays the mode that the Oracle ASM cluster is currently in. The possible return values are **ASM cluster : Flex mode enabled** or **ASM cluster : Flex mode disabled**.

Example

The following example shows the use of the `showclustermode` command.

Example 10-47   Using the ASMCMD showclustermode command

```
ASMCMD [+] > showclustermode
ASM cluster : Flex mode disabled
```

showclusterstate

Purpose

Displays the current state of the cluster.

Syntax and Description

```
showclusterstate
```

`showclusterstate` displays the state that the Oracle ASM cluster is currently in. The possible values returned are **Normal**, **In Rolling Patch**, or **In Rolling Upgrade** mode.

Example

This example shows the use of the `showclusterstate` command.

Example 10-48   Using the ASMCMD showclusterstate command

```
ASMCMD [+] > showclusterstate
```

showpatches

Purpose

Lists the patches applied on the Oracle Grid Infrastructure home.

Syntax and Description

```
showpatches [-l]
```

The following table lists the options for the `showpatches` command.

Table 10-30   Options for the showpathches command

Option	Description
—l	Displays all details about the patches.

`showpatches` lists the patches that have been applied to the Oracle Grid Infrastructure home.

Example

This example shows the use of the `showpatches` command.

Example 10-49   Using the ASMCMD showpatches command

```
ASMCMD [+] > showpatches -l
Oracle ASM release patch level is [0] and no patches have been applied on the local
node. The release patch string is [18.1.0.0.0].
```

showversion

Purpose

Displays the patch levels of the Oracle ASM cluster release and software.

Syntax and Description

```
showversion [[[--releasepatch] [--softwarepatch]] | [--active]]
```

The following table lists the options for the `showversion` command.

Table 10-31   Options for the showversion command

Option	Description
--releasepatch	Displays the cluster level patch version.
	Use `asmcmd showversion --active` instead.
--softwarepatch	Displays the patch level on the local node.
	Use `asmcmd showpatches -l` instead.
--active	Displays the active version and active patch level

`showversion` displays the Oracle ASM cluster release and software patch levels. The release and software patch options may differ depending whether the Oracle ASM or Oracle Grid Infrastructure home is being patched.

> ⓘ **Note**
>
> The `releasepatch` and `softwarepatch` options may be different when the Oracle Grid Infrastructure or Oracle ASM home is being patched. When in normal mode, the options should be same. When there is no Oracle ASM instance connected, release patch information is not displayed.

Example

This example shows the use of the `showversion` command.

Example 10-50   Using the ASMCMD showversion command

```
ASMCMD [+] > showversion --active
Oracle ASM active version on the cluster is [19.0.0.0.0]. The cluster upgrade state is
[NORMAL]. The cluster active patch level is [0].
```

shutdown

Purpose

Shuts down an instance.

Syntax and Description

```
shutdown [--target target_instance] [--normal | --abort|--immediate ]
```

Table 10-32 lists the options for the `shutdown` command.

Table 10-32   Options for the shutdown command

Option	Description
`--target target_instance`	Specify the target instance. The value of `target_instance` could be either Oracle ASM (`ASM`), IOServer (`IOS`), or Oracle ASM proxy (`APX`) instance.
`--normal`	Shut down normal. This is the default action.
`--abort`	Shut down stopping all existing operations.
`--immediate`	Shut down immediately.

The default target instance is determined by the `ORACLE_SID` environmental variable. The default action is a normal shutdown.

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` option.

For more information about shutting down an Oracle ASM instance, see "About Shutting Down an Oracle ASM Instance".

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 stops all existing operations.

Example 10-51   Using the ASMCMD shutdown command

```
ASMCMD [+] > shutdown --target ASM --normal

ASMCMD [+] > shutdown --target ASM --immediate
```

```
ASMCMD [+] > shutdown --target ASM --abort
```

spbackup

Purpose

Backs up an Oracle ASM SPFILE to a backup file.

Syntax and Description

spbackup *source destination*

Table 10-33 lists the options for the `spbackup` command.

Table 10-33   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".
2. Use the ASMCMD spcopy command to copy the operating system file to a disk group. See "spcopy".

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 10-52   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] [--local] source destination
```

Table 10-34 lists the options for the `spcopy` command.

Table 10-34   Options for the spcopy command

Option	Description
`--local`	Copies the SPFILE from or to the local system where asmcmd is running. This option should only be used when there is not an ASM instance where asmcmd is running, being both in different nodes.
`-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.

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.

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 10-53   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 /oracle/product/11.2.0/grid/dbs/spfileTestASM.ora
                  +DATA/ASM/spfileCopyASM.ora
```

> ⓘ **See Also**
>
> - spset for information about using the ASMCMD `spset` command
>
> - About Backing Up, Copying, and Moving an Oracle ASM Initialization Parameter File for information about copying and moving an Oracle ASM instance initialization parameter file after upgrading
>
> - *Oracle AI Database SQL Language Reference* for information about creating a server parameter file with the `CREATE SPFILE` SQL statement

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 10-54   Using the ASMCMD spget command

```
ASMCMD [+] > spget
+DATA/ASM/ASMPARAMETERFILE/registry.253.813507611
```

spmove

Purpose

Moves an Oracle ASM SPFILE from source to destination and automatically updates the GPnP profile.

Syntax and Description

```
spmove source destination
```

lists the options for the `spmove` command.

Table 10-35   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 "About Backing Up, Copying, and Moving an Oracle ASM Initialization Parameter File".

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 10-55   Using the ASMCMD spmove command

```
ASMCMD> spmove +DATA/spfileASM.ora
               /oracle/product/11.2.0/grid/dbs/spfileMoveASM.ora

ASMCMD> spmove /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
```

lists the options for the `spset` command.

Table 10-36   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 10-56   Using the ASMCMD spset command

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

startup

Purpose

Starts up the default instance.

Syntax and Description

```
startup [--mount] [--nomount] [--restrict] [ --pfile pfile ]
```

Table 10-37 lists the options for the startup command.

Table 10-37   Options for the startup command

Option	Description
--mount	Specifies a mount operation. This is the default action.
--nomount	Specifies no mount operation.
--restrict	Specifies restricted mode.
--pfile *pfile*	Oracle ASM initialization parameter file.

This command starts up an instance specified by the ORACLE_SID environmental variable. The variable could be set to the Oracle ASM, IOServer, or APX proxy instance.

The default action for an Oracle ASM instance is a startup that mounts disk groups and enables Oracle ASM Dynamic Volume Manager (Oracle ADVM) volumes.

Example

The following is an example of the startup command that starts the Oracle ASM instance (ORACLE_SID=+ASM) without mounting disk groups and uses the asm_init.ora initialization parameter file.

Example 10-57   Using the ASMCMD startup command

```
ASMCMD> startup --nomount --pfile asm_init.ora
```

> ⓘ **See Also**
>
> - [About Mounting Disk Groups at Startup](#) for information about disk groups that are mounted at startup time
> - [About Starting Up an Oracle ASM Instance](#) for more information about starting up an Oracle ASM instance

ASMCMD Client Cluster Management Commands

This topic provides a summary of the ASMCMD client cluster management commands that are used with Oracle ASM remote instances.

Unless otherwise specified, the ASMCMD cluster management commands should be executed in the Oracle ASM-CSS cluster.

For information about Oracle Flex ASM, refer to "[Managing Oracle Flex ASM](#)".

[Table 10-38](#) lists the ASMCMD client cluster management commands with brief descriptions.

Table 10-38   Summary of ASMCMD client cluster management commands

Command	Description
chcc	Modifies a configuration for a client cluster.
mkcc	Creates a configuration for a client cluster.
lscc	Lists the configured client clusters.
rmcc	Removes the configuration for a client cluster.

> ⓘ **See Also**
>
> - *Oracle Grid Infrastructure Installation and Upgrade Guide* for information about installing clusters and cluster name requirements
> - *Oracle Clusterware Administration and Deployment Guide* for information about client clusters

chcc

Purpose

Modifies the configuration for a client cluster.

Syntax and Description

```
chcc cluster_name [--direct | --indirect]
     [--version client_cluster_version]
```

The following table lists the options for the `chcc` command.

Table 10-39   Options for the chcc command

Option	Description
`cluster_name`	Specifies the name of the client cluster.
`--direct`	Specifies direct storage access.
`--indirect`	Specifies indirect storage access.
`--version` `client_cluster_version`	Specifies the five digit client cluster version if it is different than the storage server version. The storage server version is used if not specified.

To modify the storage access method of a client cluster, you perform the following steps.

1. As the Oracle Grid Infrastructure user at the Domain Services Cluster (DSC), change the client cluster storage access configuration with the `chcc` command.

2. As the Oracle Grid Infrastructure user at the Domain Services Cluster (DSC), export the client cluster configuration into a new cluster manifest file with the `mkcc` command.

   You can ignore the warning messages reported when the `mkcc` command is issued.

3. After you have exported the new cluster manifest file, as the root user on the nodes of client cluster you copy the new cluster manifest file to the nodes with the `rootcrs.sh` script.

Examples

The following examples show how to modify the storage access method from direct to indirect Oracle ASM storage for the client cluster `--asm` and `--gimr` components.

Example 10-58   Using the ASMCMD chcc command

As the Oracle Grid Infrastructure user at the Domain Services Cluster (DSC), run the following commands:

```
$ asmcmd chcc clientCluster1 --indirect

$ asmcmd mkcc clientCluster1 /export/cluster_manifest_new.xml --asm --gimr
```

As the root user at the client cluster, copy the new cluster manifest file to the client cluster with the following commands.

1. Run the `rootcrs.sh` script on every node in the client cluster.

   ```
   # rootcrs.sh --convertcluster STORAGE_ACCESS -manifestfile cluster_manifest_new.xml
   ```

2. Run the `rootcrs.sh` script with the `-lastnode` option on last node in the client cluster.

   ```
   # rootcrs.sh --convertcluster STORAGE_ACCESS -manifestfile cluster_manifest_new.xml -lastnode
   ```

lscc

Purpose

Lists the configured client clusters.

Syntax and Description

```
lscc [--suppressheader] [-l] [cluster_name | --file wrap]
```

The following table lists the options for the `lscc` command.

Table 10-40   Options for the lscc command

Option	Description
-suppressheader	Suppresses column headings from the output.
-l	Displays all details.
cluster_name	Specifies the name of the client cluster.
--file	Displays information about the contents of the cluster manifest file.
wrap	Specifies the full path of the XML file to export the credentials.

Examples

The following example lists the configured client clusters.

Example 10-59   Using the ASMCMD lscc command

```
ASMCMD [+] > lscc
-------------------------------------------------------------------------------
          NAME         VERSION GUID
===============================================================================
 clientCluster1     12.2.0.0.0 78015b1e78756f88bf119fd8e5146df4
 clientCluster2     12.2.0.0.0 3de3bbd75e9a7f17bfe027b477b5bcae
===============================================================================

ASMCMD [+] > lscc -l
-------------------------------------------------------------------------------
          NAME         VERSION GUID                             ASM GIMR TFA ACFS RHP
=====================================================================================
 clientCluster1     12.2.0.0.0 78015b1e78756f88bf119fd8e5146df4 YES  YES YES  YES YES
 clientCluster2     12.2.0.0.0 3de3bbd75e9a7f17bfe027b477b5bcae YES  YES YES  YES  NO
=====================================================================================

ASMCMD [+] > lscc clientCluster1 --suppressheader
 clientCluster1     12.2.0.0.0 78015b1e78756f88bf119fd8e5146df4

ASMCMD [+] > lscc clientCluster1 --suppressheader -l
 clientCluster1     12.2.0.0.0 78015b1e78756f88bf119fd8e5146df4 ASM,GIMR,TFA,ACFS,RHP
```

mkcc

Purpose

Creates the configuration for a client cluster.

Syntax and Description

```
mkcc cluster_name manifest_file [--direct]
     [--version client_cluster_version]
     [--guid client_cluster_guid]
     [--gimr gimr_pdb/export_dir]
     [--asm] [--gimr] [--rhp] [--acfs] [--tfa]
```

The following table lists the options for the `mkcc` command.

Table 10-41   Options for the mkcc command

Option	Description
cluster_name	Specifies the name of the client cluster.
manifest_file	Specifies the full path of the XML file to export the credentials.
`--direct`	Specifies direct storage access. If not specified, the storage access is indirect.
`--version` *client_cluster_version*	Specifies the five digit client cluster version if it is different than the storage server version. The storage server version is used if not specified.
`--guid` *client_cluster_guid*	Specifies the 32-character, globally unique identifier for the client cluster. When this option is specified, `mkcc` does not generate a new GUID. If there is a cluster already configured with the specified GUID, then the command fails.
`--gimr` *gimr_pdb/ export_dir*	Specifies the directory from which to import the GIMR PDB when converting a standalone cluster to a client cluster.
`--asm`	Generate credentials and configuration for an Oracle ASM cluster.
`--gimr`	Generate credentials and configuration for an Oracle GIMR cluster.
`--rhp`	Generate credentials and configuration for an Oracle RHP cluster.
`--acfs`	Generate credentials and configuration for an Oracle ACFS cluster.
`--tfa`	Generate credentials and configuration for an Oracle TFA cluster.

The client cluster name must conform to the same rules as a cluster name. The name must be unique across your enterprise, must be at least one character long and no more than 15 characters in length, must be alphanumeric, cannot begin with a numeral, and may contain hyphens (-). Underscore characters (_) are not allowed.

If no client cluster component options are specified, the default is to configure all of the components and store their credentials in the same manifest file.

GNS credentials are not generated by the `asmcmd mkcc` command. To configure the client cluster as a GNS Client, run the following command as a privileged user to export the GNS instance client data configuration to the cluster manifest file:

```
srvctl export gns -clientdata manifest_file -role CLIENT
```

Examples

The following example creates credentials for the client cluster `--asm` and `--gimr` components and exports the credentials to a cluster manifest file.

Example 10-60   Using the ASMCMD mkcc command

```
ASMCMD [+] > mkcc clientCluster1 /tmp/client_cluster1.xml --version 12.2.0.0.0 --asm --gimr
```

rmcc

Purpose

Removes the configuration for a client cluster.

Syntax and Description

```
rmcc cluster_name [-f]
```

The following table lists the options for the `rmcc` command.

Table 10-42   Options for the rmcc command

Option	Description
`cluster_name`	Specifies the name of the client cluster.
`-f`	Specifies to force the removal of client cluster credentials.

Examples

The following example removes the configuration for a client cluster.

Example 10-61   Using the ASMCMD rmcc command

```
ASMCMD [+] > rmcc clientCluster1
```

ASMCMD File Management Commands

This topic provides a summary of the ASMCMD file management commands.

Table 10-43 lists the ASMCMD file management commands with brief descriptions.

Table 10-43   Summary of ASMCMD file management commands

Command	Description
amdu_extract	Extracts a file from a disk group.
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.
kscopy	Copies an Oracle KEYSTORE file from the source location to the destination location.
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 contents of an Oracle ASM directory, the attributes of the specified file, or the names and attributes of all disk groups.
mapblk	Displays where the specified file and block is located, including device path and offset within disk.
mkalias	Creates an alias for system-generated file names.
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.

amdu_extract

Purpose

Extracts a file from a disk group.

Syntax and Description

`amdu_extract` *`diskgroup`* `[--sys_filename]` *`file_name disk_string`*

Table 10-44   Options for the amdu_extract command

Option	Description
diskgroup	Name of the disk group where the file is located.
--sys_filename	If this option is specified, then the *file_name* should point to the system Oracle ASM file name, not the file alias.
file_name	Name of the file to extract.
	The file name must be one of the following:
	• An Oracle ASM alias name
	• A system Oracle ASM filename if the --sys_filename option is specified
	The file name can be specified as an absolute or a relative path.
disk_string	Value for the Oracle ASM discovery disk string.

`amdu_extract` calls the Oracle ASM Metadata Dump Utility (AMDU) command to extract a file using an Oracle ASM alias name.

Example

The following is an example of the `amdu_extract` command used to extract a file from the data disk group.

Example 10-62   Using the ASMCMD amdu_extract command

```
ASMCMD> amdu_extract data data/orcl/my_alias_filename /devices/disk*
```

cd

Purpose

Changes the current directory to the specified directory.

Syntax and Description

`cd [`*`dir`*`]`

Table 10-45 lists the options for the `cd` command.

Table 10-45   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".

Examples

The following are examples of the `cd` command changing into various directories.

Example 10-63   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 [connect_str:]src_file [--target target_type] [--service service_name]
[--port port_num] [--dest_dbname dbname] [--sparse] [--sparse_merge_begin]
[connect_str:]tgt_file
[--sparse_merge_end sparse_merge_end_file] [--local] [--pdbguid pdbguid]
```

The following table contains the syntax options for the `cp` command.

Table 10-46   Options for the cp command

Option	Description
--dest_dbname *dbname*	Only used if target file is in a disk group. Specified name will be used as the root directory instead of using the default (either ASM, IOS, or APX, depending which target was specified). If this option is used, the system alias root path will be +*<DG>*/*<dbname>*/.
--sparse	Indicates that it is a sparse copy of a file.
--sparse_merge_begin	Beginning depth of a sparse file to be merged.
--sparse_merge_end	Ending depth of a sparse file to be merged.
--local	Copy from/to the local file system to the ASM disk group when ASMCMD is running in a different node than the ASM instance. Not to be used to copy a file from one diskgroup to another, or when source and target file paths are both OS file systems.
--pdbguid *pdbguid*	Only used if target file is in a disk group. Specified PDB will be used as the directory for destination file system alias. If this option is used, the system alias root path will be: +*<DG>*/*<dbname>*/*<pdbguid>*/.
--target *target_type*	Specifies the target type of the instance that ASMCMD must be connected to for the copy operation. Valid options are ASM, IOS, or APX.
--service *service_name*	Specifies the Oracle ASM instance name if not the default +ASM.
--port *port_num*	Specifies the listener port number. The default is 1521.
connect_str	Specifies the connection string for a remote instance.
src_file	Name of the source file to copy.

Table 10-46   (Cont.) Options for the cp command

Option	Description
`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 the OCR, SPFILE, Password file, and Oracle Keystore file types. To back up, copy, or move an Oracle ASM SPFILE, use the `spbackup`, `spcopy`, or `spmove` commands. To create, copy, or move a Password file, use the `pwcreate`, `pwcopy`, or `pwmove` commands. To copy an OCR backup file type, the source must be from a disk group. To copy an Oracle Keystore file use the `kscopy` command.

The format of copied files is portable between Little-Endian and Big-Endian systems if the files exist in an Oracle ASM disk group. Oracle ASM automatically converts the format when the files are written. For copying a non-Oracle ASM file from or to an Oracle ASM disk group, you can copy the file to a different endian platform and then use one of the commonly used utilities to convert the file.

`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 `connect_str` is in the form:

`user@host.SID`

`user`, `host`, and `SID` are required in the `connect_str` parameter. The default port number is `1521` and can be changed with the `--port` option.

The connection privilege (`SYSASM` or `SYSDBA`) is determined by the use of the `--privilege` option when starting ASMCMD.

`src_file` must be either a fully qualified file name or an Oracle ASM alias.

During an ASMCMD copy process, the Oracle ASM server creates an Oracle Managed File (OMF) file in the following format:

`diskgroup/db_unique_name/file_type/file_name.#.#`

where `db_uniq_name` is set to `ASM` and `#` is a string of digits. In the copy process, ASMCMD `cp` creates the directory structure for the destination and creates a user alias from that location to the actual OMF file that is created.

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. You are prompted to enter a password. The fourth example shows how to copy a file in the data disk group to a remote server (operating system

file) providing a password. The fifth example shows a copy of several files in the data disk group to a directory on the operating system. The sixth example shows a copy of a file on the operating system to the data disk group. The `--dest_dbname` option was used, so the system alias for the copied file will be stored under `+DATA/DB1/DATAFILE` instead of using the default `+DATA/ASM/DATAFILE`. The seventh example shows how to copy a remote operating system file to the data disk group. The eight example shows how to copy a remote operating system file to the data disk group when listener port is not the default. The ninth example shows how to make a sparse copy of a file between sparse disk groups. The tenth example shows how to make a sparse copy of multiple sparse files.

Sparse merge: Consider a multi-level sparse file hierarchy having sparse files `SPARSE.205.14770227--SPARSE.241.14770227--SPARSE.257.71740127--SPARSE.259.91640227` where `SPARSE.205.14770227` is the base parent and `SPARSE.241.14770227` is the parent of `SPARSE.257.71740127`, which in turn is the parent of `SPARSE.259.91640227`.

The eleventh example shows how to merge multiple sparse files into a new destination file (out-of-place merge). The twelfth example shows how to merge multiple sparse files into an existing sparse file (in-place merge). The thirteenth example shows how to copy from the local operating system to an ASM disk group when there is no ASM instance in the same node as where ASMCMD is running. The fourteenth example shows how to do the reverse of the previous example, which is copying a file from an ASM disk group to an operating system file when there is no ASM instance running in the same node as where ASMCMD resides. The fifteenth example shows how to specify the PDB GUID directory for the target file system alias.

Example 10-64   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:

ASMCMD [+] > cp +data/orcl/datafile/EXAMPLE.265.691577295 sys/passwd@server.asminstance:/
scratch/file
copying +data/orcl/datafile/EXAMPLE.265.691577295 -> /scratch/file

ASMCMD [+] > cp +data/orcl/data*/orcl* /mybackups
copying +data/orcl/DATAFILE/ORCLTABLES.273.924621557 -> /mybackups/
ORCLTABLES.273.924621557
copying +data/orcl/DATAFILE/ORCLTABLES.274.924621557 -> /mybackups/
ORCLTABLES.274.924621557
copying +data/orcl/DATAFILE/ORCLTABLES.275.924621557 -> /mybackups/
ORCLTABLES.275.924621557

ASMCMD [+] > cp /mybackups/examples.bak +data/orcl/datafile/myexamples.bak --dest_dbname
DB1
copying /mybackups/examples.bak -> +data/orcl/datafile/myexamples.bak

ASMCMD [+] > cp sys@server.SID:/scratch/file +data/orcl/datafile/file
copying /scratch/file -> +data/orcl/datafile/file

ASMCMD [+] > cp --port 2500 sys@server.SID:/scratch/file +data/orcl/datafile/file
copying /scratch/file -> +data/orcl/datafile/file

ASMCMD [+] > cp --sparse +SPARSEDG/SPARSE.259.91640227 +SPARSEDG/sparsefile/sparse_cp
copying +SPARSEDG/SPARSE.259.91640227 -> +SPARSEDG/sparsefile/sparse_cp
```

```
ASMCMD [+] > cp --sparse +SPARSEDG/sparse_alias.1 +SPARSEDG/sparse_alias.2 +SPARSEDG/dir/
sp_alias.3 +SPARSEDG/sparsedir/
copying +SPARSEDG/sparse_alias.1 -> +SPARSEDG/sparsedir/sparse_alias.1
copying +SPARSEDG/sparse_alias.2 -> +SPARSEDG/sparsedir/sparse_alias.2
copying +SPARSEDG/dir/sp_alias.3 -> +SPARSEDG/sparsedir/sp_alias.3

ASMCMD [+] > cp --sparse_merge_begin +SPARSEDG/SPARSE.259.91640227 +SPARSEDG/sparsefile/
sparse_merge --sparse_merge_end +SPARSEDG/SPARSE.205.14770227
merging +SPARSEDG/SPARSE.259.91640227 through +SPARSEDG/SPARSE.205.14770227 to +SPARSEDG/
sparsefile/sparse_merge

ASMCMD [+] > cp --sparse_merge_begin +SPARSEDG/SPARSE.259.91640227 +SPARSEDG/
SPARSE.259.91640227 --sparse_merge_end +SPARSEDG/SPARSE.241.14770227
merging +SPARSEDG/SPARSE.259.91640227 through +SPARSEDG/SPARSE.241.14770227 to +SPARSEDG/
SPARSE.259.91640227

ASMCMD [+] > cp --local /mybackups/backup.bak +data/backup_dir/backup.bak
copying /mybackups/backup.bak -> +data/backup_dir/backup.bak

ASMCMD [+] > cp --local +data/hm/datafile/EXAMPLE.265.691577295 /mybackups/backup.bak
copying +data/hm/datafile/EXAMPLE.265.691577295 -> /mybackups/backup.bak

ASMCMD [+] > cp /tmp/f1.df +data/f1.df --pdbguid ABCDEFGHIJKLM1234567890
copying /tmp/f1.df -> +data/f1.df

NOTE: cp command can be used to copy files from ASM disk groups to named pipe (FIFO).
The following is an example to copy to a FIFO special file.

$bash> mkfifo /tmp/fifotest
$bash> cp /tmp/fifotest /tmp/file &
$bash> asmcmd cp +data/orcl/datafile/file /tmp/fifotest
copying +data/orcl/datafile/file -> /tmp/fifotest
```

> ⓘ **See Also**
>
> - [Specifying the Type of Connection](#) for more information about connection options with ASMCMD
> - *Oracle AI Database Net Services Administrator's Guide* for more information about connection strings

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 10-47 lists the syntax options for the du command.

Table 10-47   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".

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 10-65   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] [--filegroup filegroup] dir pattern
```

Table 10-48 lists the syntax options for the find command.

Table 10-48   Options for the find command

Option	Description
--type *type*	Type of target to find.
--filegroup	Name of the file group containing the target files.
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 file name, and can include wildcard characters. See "Wildcard Characters".

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 file names.

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 . 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. The third example returns the files contained in a file group called ORCL.

Example 10-66   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

ASMCMD [+] > find +data --filegroup ORCL *
+data/ORCL/DATAFILE/UNDOTBS1.258.691577151
+data/ORCL/CONTROLFILE/Current.260.691577263
```

kscopy

Purpose

Copies an Oracle KEYSTORE file from the source location to the destination location.

Syntax and Description

```
kscopy source_file <destination_path> [--autologin] [--local] [--dbuniquename <string>]
```

The following table contains the syntax options for the `kscopy` command.

Table 10-49   Options for the kscopy command

Option	Description
source_file	Specifies the keystore source file.
destination_path	Specifies the destination file path.
--autologin	Specifies the filetype as an autologin keystore file.
--local	Specifies that the keystore file should be copied across different nodes.
--dbuniquename	Specifies the database resource to update with the keystore file location.

Note the following about the use of `kscopy`:

- `kscopy` can copy an Oracle keystore file from an operating system path to a disk group.

- `kscopy` can copy an Oracle keystore file from a disk group to an operating system path.

- `kscopy` can copy an Oracle keystore file from a disk group to another disk group.

- `kscopy` should not be used to copy a keystore file from an operating system path to an operating system path, for this case use the appropriate operating system command.

- When the source keystore file is located in the OS file system and the file name is not the default name for keystore files `ewallet.p12` or autologin keystore files `cwallet.sso` the file type to be used is `KEYSTORE` by default. However if `--autologin` is specified the file type is set to `AUTOLOGIN KEYSTORE` instead.

> ### ⓘ See Also
>
> The `CREATE_KEYSTORE` clause of the `ADMINISTER KEY MANAGEMENT` statement in the *Oracle AI Database SQL Language Reference* for information about creating a server keystore file.

Examples

The following are examples of the `kscopy` command. The first example copies the Oracle `KEYSTORE` from the data disk group to an operating system location. The second example copies the Oracle `KEYSTORE` from an operating system location to the data diskgroup . The third example copies an Oracle `AUTOLOGIN KEYSTORE` from an operating system location to the data diskgroup. The fourth example copies an Oracle `KEYSTORE` from the data diskgroup to an operating system location in the local node. The fifth example copies an Oracle `AUTOLOGIN KEYSTORE` from an operating system location to the data diskgroup and sets "ORCL" string as the database name this keystore will be stored in.

Example 10-67   Using the ASMCMD kscopy command

```
ASMCMD> kscopy  +DATA/tde/ewallet.p12 /tmp/tde

ASMCMD> kscopy /tmp/tde/ewallet.p12 +DATA/tde

ASMCMD> kscopy /tmp/tde/cwallet.sso +DATA/tde –autologin

ASMCMD> kscopy +DATA/tde/ewallet.p12 /tmp/tde –local

ASMCMD> kscopy /tmp/tde/ewallet.p12 +DATA/tde –dbuniquename ORCL
```

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] [-lshdtLg] [--absolutepath] [--reverse]
[--permission] [--time_style style] [--fg] [pattern]
```

Table 10-50 lists the syntax options for the `ls` command.

Table 10-50   Options for the ls command

Option	Description
(none)	Displays only file names and directory names.

Table 10-50   (Cont.) Options for the ls command

Option	Description
`-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.
`-h`	With `-s`, print sizes in human readable format (e.g., 8K, 250M, 2.5G).
`-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.
`--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$ASM_DISKGROUP.INST_ID` is included in the output.
`--time_style`	Specify the format in which the date will be printed, using the format models for datetime used in the Oracle AI Database (e.g., "`DD MON YYYY`").
`--fg`	Displays the file group containing the file.
	If the `--fg` option is specified, a Filegroup column will be added that contains the name of the file group this file belongs to. If it is an alias directory, or files from a disk group with a redundancy different than FLEX or EXTENDED, the Filegroup column will be blank.
`--suppressheader`	Suppresses column headings.
`--permission`	Shows the permissions of a file (`V$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 10-50 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".

• 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 10-68   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

ASMCMD [+] > ls -l --time_style 'MON DD HHAM' +data/orcl/datafile/sy*
Type       Redund  Striped  Time         Sys  Name
DATAFILE   MIRROR  COARSE   JUL 13 05PM  Y    SYSAUX.257.691577149
DATAFILE   MIRROR  COARSE   JUL 13 02PM  Y    SYSTEM.256.691577149

ASMCMD [+] > ls -sh +data/orcl/datafile
Block_Size  Blocks      Bytes      Space  Name
       8K   12801      100M       205M  EXAMPLE.265.691577295
       8K   88321      690M       1.4G  SYSAUX.257.691577149
       8K   88321      690M       1.4G  SYSTEM.256.691577149
       8K    7681       60M       125M  UNDOTBS1.258.691577151
       8K     641        5M        12M  USERS.259.691577151

ASMCMD [+] > ls --fg +data/orcl/datafile
Filegroup  Name
```

```
ORCL       EXAMPLE.265.691577295
ORCL       SYSAUX.257.691577149
ORCL       USERS.259.691577151
ORCL       UNDOTBS1.258.691577151
ORCL       SYSTEM.256.691577149
```

lsof

Purpose

Lists the open files of the local clients.

Syntax and Description

```
lsof [--suppressheader] {-G diskgroup|--dbname db| -C instance}
```

[Table 10-51](#) lists the syntax options for the `lsof` command.

Table 10-51   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 10-69   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
```

mapblk

Purpose

Displays where the specified file and block is located, including device path and offset within disk.

Syntax and Description

```
mapblk [-l] [--suppressheader] <file> <block>
```

Table 10-52 lists the syntax options for the `mapblk` command.

Table 10-52   Options for the mapblk command

Option	Description
`(none)`	Displays `Logic_Ext`, `Block_Size`, `Offset`, `Disk_Num`, `Path`.
`-l`	Displays extended information, including `Disk_Name`, `AU_Size`, `File_Num`, `Ext_Num`, `AU_Num`.
`--suppressheader`	Suppresses column header from the output.

The `mapblk` command displays where the specified file and block is located, including device path and offset within disk. The columns for the `mapblk` command output are described below:

- `Logic_Ext` - ASM logical extent number. 0 is primary extent. 1 and 2 are secondary extents

- `Block_Size` - Logical block size (in bytes)

- `Offset` - Offset within ASM disk (in logical blocks)

- `Disk_Num` - ASM disk number

- `Path` - Device path of ASM disk

- `Disk_Name` - ASM disk name

- `AU_Size` - Allocation Unit size (in bytes)

- `File_Num` - ASM file number

- `Ext_Num` - ASM virtual Extent number

- `AU_Num` - Allocation Unit number

Example

The following example the of `mapblk` command displays datafile `+DATA/DB1/datafile/TBS1.256.863259515` block number 10 is located to offset 2058 in logical blocks(8192 bytes) within `/dev/sdd1` and offset 4618 in logical blocks(8192 bytes) within `/dev/sde1` disk group.

Example 10-70   Using the ASMCMD mapblk command

```
ASMCMD [+] > mapblk +DATA/DB1/datafile/TBS1.256.863259515 10
       Logic_Ext  Block_Size  Offset  Disk_Num  Path
               0        8192    2058         3  /dev/sdd1
               1        8192    4618         4  /dev/sde1
```

mkalias

Purpose

Creates an alias for the specified system-generated file name.

Syntax and Description

```
mkalias file alias
```

Table 10-53 lists the syntax options for the `mkalias` command.

Table 10-53   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 file name `+data/orcl/DATAFILE/SYSAUX.257.721811945`. Following the `mkalias` command, `ls --absolutepath` is run to check the results.

Example 10-71   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 10-54 lists the syntax options for the `mkdir` command.

Table 10-54   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 10-72   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 10-73   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 10-55 lists the syntax options for the `rm` command.

Table 10-55   Options for the rm command

Option	Description
`-r`	Recursively deletes files and subdirectories.
`-f`	Deletes files and subdirectories without prompting for confirmation.
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](#)".

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 10-74   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 10-56 lists the syntax options for the `rmalias` command.

Table 10-56   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 10-75   Using the ASMCMD rmalias command

```
ASMCMD [+data/orcl/datafile] > rmalias sysaux.f
```

ASMCMD Disk Group Management Commands

This topic provides a summary of the ASMCMD disk group management commands.

Table 10-57 lists the ASMCMD disk group management commands with brief descriptions.

Table 10-57   Summary of ASMCMD Disk group management commands

Command	Description
chdg	Changes a disk group (add, drop, rebalance, or migrate on the Exadata appliance).
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.
offline	Offlines a disk or a failure group.

Table 10-57   (Cont.) Summary of ASMCMD Disk group management commands

Command	Description
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.
setsparseparent	Sets the parent for a sparse child file.
setattr	Sets attributes in a disk group.
stamp	Stamps the disk, site, and failure group labels in disk headers.
stamplist	Displays the disk, site, and failure group labels in disk headers.
umount	Dismounts a disk group.

chdg

Purpose

Modifies a disk group based on an XML configuration file.

Syntax and Description

```
chdg { config_file.xml | 'contents_of_xml_file' }
```

Table 10-58 lists the syntax options for the `chdg` command.

Table 10-58   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 10-76 and Example 10-77.
contents_of_xml_file	The XML script enclosed in single quotations.

The modifications include the following:

*   adding, dropping, and resizing disks

*   rebalancing or replacing the power level of a disk group

    The power level can be set to the same values as the `ASM_POWER_LIMIT` initialization parameter

*   migrating an Exadata appliance from one fixed configuration to another

    The migration adds more disks (an equal number of disks) to each of the existing failure groups (or cells) in the disk group.

The modification includes adding, replacing, or deleting disks from an existing disk group, and the setting rebalance or replace power level. The power level can be set to the same values as the `ASM_POWER_LIMIT` initialization parameter. The modification also includes the migration from one fixed appliance configuration to another fixed appliance configuration on the Exadata

appliance. The migration adds more disks (and equal number of disks) to each of the existing failure groups (or cells) in the disk group.

When adding disks to a disk group, the diskstring must be specified in a format similar to the `ASM_DISKSTRING` initialization parameter.

The failure groups are optional parameters. The default causes every disk to belong to a its own failure group.

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.

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.

Example 10-76 shows the basic structure and the valid tags with their respective attributes for the `chdg` XML configuration file.

Example 10-76   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 or replace

<add>  items to add are placed here
</add>

<replace>  items to replace are placed here
</replace>

<drop> items to drop are placed here
</drop>

<migrate> items to be migrated to another appliance configuration on EXADATA
</migrate>

<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>
```

Example 1

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 10-77   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>
```

Example 2

The following is another example of an XML configuration file for `chdg`. This XML file migrates the failure group named `FG_01`.

Example 10-78   chdg sample XML configuration file

```
<chdg name="DATAC1 >
        <migrate>
          <add>
            <fg name = "EXA04CEL01">
                <dsk string="o/192.168.10.18/DATAC1_CD_10_exa04cel01" name=
"DATAC1_CD_10_exa04cel01" />
                <dsk string="o/192.168.10.18/DATAC1_CD_11_exa04cel01" name=
"DATAC1_CD_11_exa04cel01" />
            </fg>
            <fg name = "EXA04CEL02">
                <dsk string="o/192.168.10.20/DATAC1_CD_10_exa04cel02" name=
"DATAC1_CD_10_exa04cel02" />
                <dsk string="o/192.168.10.20/DATAC1_CD_11_exa04cel02" name=
"DATAC1_CD_11_exa04cel02" />
            </fg>
            <fg name = "EXA04CEL03"">
                <dsk string="o/192.168.10.22/DATAC1_CD_10_exa04cel03" name=
"DATAC1_CD_10_exa04cel03" />
                <dsk string="o/192.168.10.22/DATAC1_CD_11_exa04cel03" name=
"DATAC1_CD_11_exa04cel03" />
            </fg>
          </add>
        </migrate>
</chdg>
```

Example 3

The following are examples of the `chdg` command with the configuration file or configuration information on the command line.

Example 10-79   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>'
```

> ⓘ **See Also**
>
> - [ASM_POWER_LIMIT](#) for information about the `ASM_POWER_LIMIT` initialization parameter
> - [ASM_DISKSTRING](#) for information about the `ASM_DISKSTRING` initialization parameter
> - [Oracle ASM Failure Groups](#) for information about failure groups
> - [Dropping Disks from Disk Groups](#) for information about dropping disks
> - [Resizing Disks in Disk Groups](#) for information about resizing disks,
> - [mkdg](#) for information about creating a disk group with ASMCMD `mkdg`
> - [Altering Disk Groups](#) for information about altering disk groups
> - Enabling the Oracle ASM appliance.mode Attribute

chkdg

Purpose

Checks or repairs the metadata of a disk group.

Syntax and Description

```
chkdg [--repair] diskgroup
```

[Table 10-59](#) lists the syntax options for the `chkdg` command.

Table 10-59   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 10-80   Using the ASMCMD chkdg command

```
ASMCMD [+] > chkdg --repair data
```

dropdg

Purpose

Drops a disk group.

Syntax and Description

```
dropdg [-r [-f]] diskgroup
```

Table 10-60 lists the syntax options for the `dropdg` command.

Table 10-60   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 10-81   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_STAT` view.

Table 10-61 lists the syntax options for the `iostat` command.

Table 10-61   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).

Table 10-61   (Cont.) Options for the iostat command

Option	Description
`--region`	Displays information for cold and hot disk regions (Cold_Reads, Cold_Writes, Hot_Reads, Hot_Writes).
	ⓘ **Note** Disk region attributes default to cold and cold is the only permitted setting. This means that values for Hot_Reads and Hot_Writes are zero.
interval	Refreshes the statistics display based on the interval value (seconds). Use CTRL+C to stop the interval display.

Table 10-62 shows the statistics for a disk group. To view the complete set of statistics for a disk group, use the `V$ASM_DISK_STAT` and `V$ASM_DISK_IOSTAT` views.

Table 10-62   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.
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.

> ⓘ **Note**
>
> Disk region attributes default to cold and cold is the only permitted setting. This means that values for Hot_Reads and Hot_Writes are zero.

Table 10-62   (Cont.) Attribute descriptions for iostat command output

Attribute Name	Description
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.

> ⓘ **Note**
>
> Disk region attributes default to cold and cold is the only permitted setting. This means that values for Hot_Reads and Hot_Writes are zero.

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 seconds) 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 seconds) 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 10-82   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 10-63 lists the syntax options for the `lsattr` command.

Table 10-63   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.

> ⓘ **Note**
>
> Attributes are only displayed for disk groups where `COMPATIBLE.ASM` is set to 11.1 or higher.

Information about disk group attributes is retrieved from the `V$ASM_ATTRIBUTE` view. For information about disk group attributes, refer to Managing Disk Group Attributes.

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, refer to lstmpl.

To set disk group attributes, refer to setattr.

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 10-83   Using the ASMCMD lsattr command

```
ASMCMD [+] > lsattr -lm -G data
Group_Name  Name                      Value        RO  Sys
DATA        access_control.enabled    FALSE        N   Y
```

```
DATA           access_control.umask   066          N   Y
DATA           appliance._partnering_type GENERIC  Y   Y
DATA           ate_conversion_done    true         Y   Y
DATA           au_size                1048576      Y   Y
DATA           cell.smart_scan_capable FALSE       N   N
DATA           cell.sparse_dg         allnonsparse N   N
DATA           compatible.advm        19.0.0.0.0   N   Y
DATA           compatible.asm         19.0.0.0.0   N   Y
DATA           compatible.rdbms       19.0.0.0.0   N   Y
DATA           content.check          FALSE        N   Y
DATA           content.type           data         N   Y
DATA           content_hardcheck.enabled FALSE     N   Y
DATA           disk_repair_time       12.0h        N   Y
DATA           failgroup_repair_time  24.0h        N   Y
DATA           idp.boundary           auto         N   Y
DATA           idp.type               dynamic      N   Y
DATA           logical_sector_size    512          N   Y
DATA           phys_meta_replicated   true         Y   Y
DATA           preferred_read.enabled FALSE        N   Y
DATA           scrub_async_limit      1            N   Y
DATA           scrub_metadata.enabled TRUE         N   Y
DATA           sector_size            512          N   Y
DATA           thin_provisioned       FALSE        N   Y
DATA           vam_migration_done     false        Y   Y

ASMCMD [+] > lsattr -G fra -l %compat*
Name             Value
compatible.advm  19.0.0.0.0
compatible.asm   19.0.0.0.0
compatible.rdbms 19.0.0.0.0
```

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 10-64 lists the options for the `lsdg` command.

Table 10-64   Options for the lsdg command

Option	Description
(none)	Displays the disk group attributes listed in Table 10-65.
--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.

Table 10-64 (Cont.) Options for the lsdg command

Option	Description
`-g`	Selects from `GV$ASM_DISKGROUP_STAT`, or from `GV$ASM_DISKGROUP` if the `--discovery` flag is also specified. `GV$ASM_DISKGROUP.INST_ID` is included in the output.
	The Rebal attribute value is derived from the value of the OPERATION column in the `GV$ASM_OPERATION` view. The PASS column in `GV$ASM_OPERATION` also indicates if a rebalance operation is running.
`--suppressheader`	Suppresses column headings.
pattern	Returns only information about the specified disk group or disk groups that match the supplied pattern. See "Wildcard Characters".

Table 10-65 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 AI Database Reference* for descriptions of disk group information displayed in the `V$ASM_DISKGROUP` view

Table 10-65 Attribute descriptions for lsdg command output

Attribute Name	Description
State	State of the disk group. For example, `BROKEN`, `CONNECTED`, `DISMOUNTED`, `MOUNTED`, `QUIESCING`, or `UNKNOWN`.
Type	Disk group redundancy. For example, `NORMAL`, `HIGH`, `FLEX`, or `EXTERNAL`.
Rebal	`Y` indicates that a rebalance operation is in progress.
Sector	Sector size in bytes.
Block	Block size in bytes.
AU	Allocation unit size in bytes.
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 10-84   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 10-66 lists the options for the `lsdsk` command.

Table 10-66   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$ASM_DISK` view.
-p	Displays the `GROUP_NUMBER`, `DISK_NUMBER`, `INCARNATION`, `MOUNT_STATUS`, `HEADER_STATUS`, `MODE_STATUS`, `STATE`, and the `PATH` columns of the `V$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.

Table 10-66   (Cont.) Options for the lsdsk command

Option	Description
`--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".

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 10-85   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 10-67 lists the syntax options for the `lsod` command.

Table 10-67   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 10-86   Using the ASMCMD lsod command

```
ASMCMD [+] > lsod -G data --process *LGWR*
Instance Process                 OSPID Path
1        oracle@myserver02 (LGWR) 26593 /devices/diska1
1        oracle@myserver02 (LGWR) 26593 /devices/diska2
1        oracle@myserver02 (LGWR) 26593 /devices/diska3
1        oracle@myserver02 (LGWR) 26593 /devices/diskb1
1        oracle@myserver02 (LGWR) 26593 /devices/diskb2
1        oracle@myserver02 (LGWR) 26593 /devices/diskb3
1        oracle@myserver02 (LGWR) 26593 /devices/diskd1

ASMCMD [+] > lsod --process *LGWR* *diska*
Instance Process                 OSPID Path
1        oracle@myserver02 (LGWR) 26593 /devices/diska1
```

```
1          oracle@myserver02 (LGWR) 26593 /devices/diska2
1          oracle@myserver02 (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,...]'] --acfs_sec_encr
```

Table 10-68 describes the options for the `md_backup` command.

Table 10-68   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
--acfs_sec_encr	Specifies that security, encryption, or audit metadata information from Oracle ACFS is 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.

The `md_backup` command backs up Oracle Advanced Cluster File System (Oracle ACFS) metadata information for the contained Oracle ASM Dynamic Volume Manager(Oracle ADVM) volumes, including stripe size, redundancy, and other items. This command also backs up Oracle ACFS file system configuration metadata, such as compression and resize attributes, block size, and created snapshots. Additional metadata and configuration attributes may be backed up; however, Oracle ACFS file system data and Oracle ACFS CRS Resource 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 10-87   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 [,...]']
   [--acfs_sec_encr user:group |
    --acfs_sec_encr user:group --acfs_audit user:mgr_group:auditor_group ]
     [-S sql_script_file] [-G 'diskgroup [,diskgroup...]']
```

Table 10-69 describes the options for the `md_restore` command.

Table 10-69   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`.

Table 10-69   (Cont.) Options for the md_restore command

Option	Description
`--acfs_sec_encr user:group`	Restores the metadata of security or encryption or both. The colon separated user and group is used for security metadata restoration. During metadata restore, security or encryption or both must be initialized. If security or encryption or both are not pre-initialized, the colon separated values passed to the `--acfs_sec_encr` option are used to initialize security. Encryption is initialized with the SSO wallet.
`--acfs_audit user:mgr_group:auditor_group`	Restores audit metadata information on an Oracle ACFS file system. The colon separated values are used to initialize and execute audit commands. During metadata restore, audit must be initialized. If audit is not pre-initialized, the audit layer is initialized using the colon separated values given to the `--acfs_audit` option. The `--acfs_audit` option requires the `--acfs_sec_encr` option.
`-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.

Note the following when restoring Oracle ACFS information:

- When restoring the metadata of a file system, the user id is set to the user id that owned the file system when the metadata backup was created. This user id should exist on the restore cluster.

- `md_restore` restores Oracle ACFS metadata information. To run the metadata restore for Oracle ACFS file systems, a script is created. This script must be run as `root` to format file systems. Metadata for the contained Oracle ADVM volumes, such as stripe size and redundancy, is restored. Oracle ACFS file system configuration metadata, such as compression and resize attributes, block size, and created snapshots, is also restored. Oracle ACFS file System data and ACFS CRS Resource information is not restored.

- If using the `--silent` option while specifying a single disk group, and there is a dependency due to Oracle ADVM metadata or Oracle ACFS accelerators on another disk group, that dependency is dropped, and you must manually recreate that dependency later.

- When restoring Oracle ACFS encryption or security, file system extended attributes (xattrs) must be restored with the user data. It is very important to document that xattrs need to be stored and restored with the file system application data. Without the xattrs, there is no security or encryption. xattrs describe which security realms the file is a part of. With no xattr, the file ends up in no realm. The converse is that a file with xattrs, but no security initialized, ends up in no realm.

- During metadata restore, a file with a list of previously configured snapshots is created. Snapshot data is not restored, as the point in time sparse snapshot views cannot be recreated.

- If Oracle ACFS security is in use, and the metadata security information for the file system has not been backed up, this information may be copied by the security administrator to the metadata backup file location, and then restored with the rest of the disk group information.

- If Oracle ACFS security and encryption is being restored, the user and password may be pre-initialized. If it is not, the username provided is used to initialize Oracle ACFS security and encryption, with a default password which is displayed.

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 10-88   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 10-70 lists the syntax options for the `mkdg` command.

Table 10-70   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 10-89 and Example 10-90.
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.

The `mkdg` command 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.

Example 10-89 shows the basic structure and the valid tags with their respective attributes for the `mkdg` XML configuration file.

Example 10-89   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>
```

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 `19.0`.

Example 10-90   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="19.0"/>
  <a name="compatible.rdbms" value="19.0"/>
  <a name="compatible.advm" value="19.0"/>
</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 10-91   Using the ASMCMD mkdg command

```
ASMCMD [+] > mkdg data_config.xml

ASMCMD [+] > mkdg '<dg name="data"><dsk string="/dev/disk*"/></dg>'
```

> ⓘ **See Also**
>
> - [Managing Disk Group Attributes](#) for more information about disk groups attributes
> - [Disk Group Compatibility](#) for information about disk group compatibility attributes
> - [chdg](#) for information about altering a disk group with ASMCMD `chdg`
> - [Creating Disk Groups](#) for information about creating a disk group

mount

Purpose

Mounts a disk group.

Syntax and Description

```
mount [--restrict] { [-a] | [-f] diskgroup[ diskgroup ...] }
```

[Table 10-71](#) lists the syntax options for the `mount` command.

Table 10-71   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](#)".

Example

The following are examples of the `mount` command showing the use of the force, restrict, and all options.

Example 10-92   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 10-72 lists the syntax options for the `offline` command.

Table 10-72   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 `nm` or `nh`, 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 10-93   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}
        [--power n] [-w]
```

Table 10-73 lists the syntax options for the `online` command.

Table 10-73   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.

Table 10-73   (Cont.) Options for the online command

Option	Description
`--power n`	Specifies the power option (`1` to `1024`) which determines the level of resources for the disk resync and disk group rebalance. Larger values result in faster disk resync and disk group rebalance. The default is `1`.
`-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 10-94   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 [--default | --modify power] [--with phase,... | --without phase,...]
                                   [--power power] [-w] diskgroup
```

Table 10-74 lists the syntax options for the `rebal` command.

Table 10-74   Options for the rebal command

Option	Description
`-default`	Sets the rebalance power to the default, which is the value of the `ASM_POWER_LIMIT` initialization parameter. This option does not restart the rebalance operation.
`--modify power`	Modifies the rebalance power without restarting the rebalance operation. Accepted values are the same as `ASM_POWER_LIMIT` (`0` to `1024`). A value of `0` disables rebalancing.
`--with phase`	Runs rebalance only on the specified phases. Available phases are `prepare`, `balance`, and `compact`. The phases must be separated by a single comma. If this option is specified, at least one phase must be specified.
`--without phase`	Does not run rebalance on the specified phases. Phases are `prepare`, `balance`, and `compact`. The phases must be separated by a single comma. If this option is specified, at least one phase must be specified.

Table 10-74   (Cont.) Options for the rebal command

Option	Description
`--power power`	Specifies the power setting. Accepted values are the same as `ASM_POWER_LIMIT` (`0` to `1024`). A value of `0` disables rebalancing.
`-w`	Specifies the wait option. ASMCMD waits for the disk group to be rebalanced before returning control to the user. The default is not waiting.
`diskgroup`	Specifies the disk group name.

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.

You can determine if a rebalance operation is occurring with the ASMCMD `lsop` command.

Example

In the following example, the first `rebal` command rebalances the `FRA` disk group with a power level set to `4`, performing only the `balance` and `compact` phases. The second `rebal` command rebalances the `DATA` disk group with the rebalance power set to `1`, but does not restart the rebalance operation.

Example 10-95   Using the ASMCMD rebal command

```
ASMCMD [+] > rebal --with balance,compact --power 4 FRA

ASMCMD [+] > lsop
Group_Name  Operation  State  Power
FRA         REBAL      RUN    4

ASMCMD [+] > rebal --modify 1 DATA
```

> ⓘ **See Also**
>
> • [ASM_POWER_LIMIT](#) and [Tuning Rebalance Operations](#) for information about the power level
>
> • [lsop](#) for information about the ASMCMD `lsop` command
>
> • [Manually Rebalancing Disk Groups](#) for more information about rebalancing a disk group

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 10-75 lists the syntax options for the `remap` command.

Table 10-75   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 10-96   Using the ASMCMD remap command

```
ASMCMD [+] > remap DATA DATA_0001 5000-5999

ASMCMD [+] > remap FRA FRA_0002 6230-6339
```

setsparseparent

Purpose

Sets the parent for a sparse child file.

Syntax and Description

```
setsparseparent child_file1 [child_file2 ...] parent_file
```

The following table lists the syntax options for the `setsparseparent` command.

Table 10-76   Options for the setsparseparent command

Option	Description
`child_fileN`	Name of the child file.
`parent_file`	Name of the parent file.

The following are the examples of `setsparseparent` command. The first example sets a parent file in the data disk group to a sparse file in the sparse disk group. The second example sets a parent file in the data disk group to multiple sparse files in the sparse disk group.

Example 10-97   Using the ASMCMD setsparseparent command

```
ASMCMD [+] > setsparseparent +SPARSEDG/child.1.10 +DATA/parent.1.f
setting parent of +SPARSEDG/child.1.10 to +DATA/parent.1.f

ASMCMD [+] > setsparseparent +SPARSEDG/child.1a.10 +SPARSEDG/child.1b.11 +SPARSEDG/
child.1c.12 +DATA/parent.1.f
```

```
setting parent of +SPARSEDG/child.1a.10 to +DATA/parent.1.f
setting parent of +SPARSEDG/child.1b.11 to +DATA/parent.1.f
setting parent of +SPARSEDG/child.1c.12 to +DATA/parent.1.f
```

setattr

Purpose

Sets the attributes for an Oracle ASM disk group.

Syntax and Description

```
setattr -G diskgroup attribute_name attribute_value
```

Table 10-77 lists the syntax options for the `setattr` command.

Table 10-77   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 "Managing Disk Group Attributes".

Example

The following are examples of the `setattr` command. The first example sets the disk group attribute `COMPATIBLE.ASM` for the `data` disk group. The second example sets the disk group attribute `COMPATIBLE.RDBMS` for the `data` disk group.

Example 10-98   Using the ASMCMD setattr command

```
ASMCMD [+] > setattr -G data compatible.asm 19.0
```

```
ASMCMD [+] > setattr -G data compatible.rdbms 19.0
```

stamp

Purpose

Stamps the disk, site, and failure group labels in disk headers.

Syntax and Description

```
stamp --dscstr disk_discovery_string
      [--site site_name --failgroup failgroup_name]
      [--disk disk_label] [-f]
```

The following table describes the options for the `stamp` command.

Table 10-78   Options for the stamp command

Option	Description
`--dscstr`*`disk_discovery_str ing`*	Specifies the disk discovery string that identifies the disks on which to stamp labels.
`--site` *`site_name`*	Specifies the site label name.
`--failgroup` *`failgroup_name`*	Specifies the failure group label. Required when the site label is specified.
`--disk` *`disk_label`*	Specifies the disk label.
`-f`	Specifies to force the action.

ASMCMD `stamp` enables the users to stamp the disk, site, and fail group labels on disk headers. Using `stamp` is more efficient than individually specifying label information during installation and creation of disk groups.

Site label and fail groups labels are used for the site names and fail group names when creating and altering a disk group. Disk labels are only intended for use with Oracle ASMLib.

At least one label specification (`-—site` with `--failgroup` or `--disk`) must be provided.

Examples

The following example shows the use of the ASMCMD `stamp` command on the disks identified by the `/dev/sd*` disk discovery string.

Example 10-99   Using ASMCMD stamp

```
ASMCMD [+] > stamp --dscstr "/dev/sd*" --site "SALTLAKE-CENTER" --failgroup "fg1" --disk
"DISK1"
```

stamplist

Purpose

Displays the disk, site and failure group labels in disk headers.

Syntax and Description

```
stamp --dscstr disk_discovery_string  {[--site] [--failgroup] [--disk]}
```

The following table describes the options for the `stamplist` command.

Table 10-79   Options for the stamplist command

Option	Description
`--dscstr`*`disk_discovery_str ing`*	Specifies the disk discovery string that identifies the disks for which to display labels.
`--site`	Displays the site label name.
`--failgroup`	Displays the failure group label.

Table 10-79   (Cont.) Options for the stamplist command

Option	Description
`--disk`	Displays the disk label.

You must specify at least one of the `-—site`, `--failgroup`, or `--disk` options.

Examples

The following example shows the use of the ASMCMD `stamplist` command to display site, failure group, and disk labels of the disks identified by the `/dev/sd*` disk discovery string.

Example 10-100   Using ASMCMD stamplist

```
ASMCMD [+] > stamplist --dscstr "/dev/sd*" --site --failgroup --disk
```

umount

Purpose

Dismounts a disk group.

Syntax and Description

```
umount { -a | [-f] diskgroup }
```

Table 10-80 lists the syntax options for the `umount` command.

Table 10-80   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 10-101   Using the ASMCMD umount command

```
ASMCMD [+] > umount -a

ASMCMD [+] > umount -f data
```

ASMCMD File Group Management Commands

This topic provides a summary of the ASMCMD commands that enable you to manage Oracle ASM file groups and quota groups.

The following table lists the ASMCMD file group and quota group commands with brief descriptions.

Table 10-81   Summary of ASMCMD file group commands

Command	Description
chfg	Modifies a file group in a disk group based on an XML configuration file or script.
chqg	Modifies a quota group in the disk group.
lsfg	List file groups in a disk group.
lsqg	Lists the quota groups in a disk group.
mkfg	Creates a file group in a disk group based on an XML configuration file or script.
mkqg	Adds a quota group to the disk group.
mvfg	Moves a file group in a disk group to the specified quota group.
mvfile	Moves a file to the specified file group in the same disk group where the file is stored.
rmfg	Removes an existing file group from a disk group.
rmqg	Drops a quota group from a disk group.

> ⓘ **See Also**
>
> - [Managing Oracle ASM Flex Disk Groups](#) for more information about Oracle file groups
> - *Oracle AI Database Reference* for information about the V$ASM views that contain file group and quota group information.
> - *Oracle AI Database SQL Language Reference* for information about administering Oracle ASM file groups and quota groups with SQL statements

chfg

Purpose

Changes the attributes of a file group in a disk group based on an XML configuration file or script.

Syntax and Description

```
chfg { config_file.xml | 'contents_of_xml_script' }
```

The following table lists the options for the chfg command.

Table 10-82   Options for the chfg command

Option	Description
config_file.xml	Name of the XML file that contains the changes for the file group. chfg searches for the XML file in the current working directory of ASMCMD unless a path is specified

Table 10-82   (Cont.) Options for the chfg command

Option	Description
`contents_of_xml_script.xml`	The contents of an XML script enclosed in single quotes.

The root element of the XML configuration file or script is the `<file_group>` tag and has these attributes:

- `name`: file group name
- `dg`: disk group name to which the file group belongs

You can set properties using these tag:

- `name`: property name
- `value`: property value
- `file_type`: property file type (optional)
- `template`: Oracle ASM template name (optional)

The following is an example of an XML configuration file for `chfg`. The configuration file alters a file group named `filegroup1` on disk group `dg_data1`. The file group is associated with quota group `quotagroup1`.

```
<filegroup name="filegroup1" dg="dg_data1">
  <p name="quota_group" value="quotagroup1"/>
</filegroup>
```

For information about Oracle ASM File Groups, refer to [Managing Oracle ASM Flex Disk Groups](#).

Example

The following examples show the use of the `chfg` command using an XML configuration file and an XML script.

Example 10-102   Using the chfg command

```
ASMCMD [+] > chfg config_file.xml

ASMCMD [+] > chfg '<filegroup name="filegroup1" dg="dg_data1">
                   <p name="redundancy" value="high"/>
                   </filegroup>'
```

chqg

Purpose

Modifies a quota group in the disk group.

Syntax and Description

```
chqg -G disk_group quota_group property value
```

The following table lists the options for the `chqg` command.

Table 10-83   Options for the chqg command

Option	Description
-G *disk_group*	Specifies the disk group.
quota_group	Specifies the quota group.
property	Specifies the name of the property.
value	Specifies the value of the property.

The chqg command modifies the specified property in the specified quota group of the specified disk group. For information about Oracle ASM File and Quota Groups, refer to "Managing Oracle ASM Flex Disk Groups".

The following are examples of the chqg command. The first example modifies the quota property of the quota group quota_grp1 in the data disk group. The second example modifies the quota property of the quota group quota_grp2 in the data disk group.

Example 10-103   Using the chqg command

```
ASMCMD [+] > chqg -G data quota_grp1 quota 200G

ASMCMD [+] > chqg -G data quota_grp2 quota UNLIMITED
```

lsfg

Purpose

List file groups in a disk group.

Syntax and Description

lsfg [-G *disk_group*] [--filegroup *file_group*] [--suppressheader]

The following table lists the options for the lsfg command.

Table 10-84   Options for the lsfg command

Option	Description
-G *disk_group*	Specifies the disk group.
—filegroup*file_group*	Specifies the file group.
—suppressheader	Suppresses the headers in the output of the command.

If the disk group option is specified, then lsfg returns only information about file groups in the specified disk group. If the file group option is specified, then lsfg returns only information about that file group.

The information about file groups is retrieved from V$ASM_FILEGROUP and V$ASM_FILEGROUP_PROPERTY views.

The following example shows the use of thelsfg command. The first command in the example displays the properties of the file group FG1 in the DATA disk group. The second command lists all the file groups and their properties in the DATA disk group.

Example 10-104   Using the lsfg command

```
ASMCMD [+] > lsfg -G DATA
File Group         Disk Group   Quota Group   Used Quota MB   Client Name   Client Type
DEFAULT_FILEGROUP  DATA         GENERIC       0
VOLUME1            DATA         GENERIC       3123            VOLUME1       VOLUME
ORCL               DATA         GENERIC       5488            ORCL          DATABASE

ASMCMD [+] > lsfg -G DATA --filegroup ORCL
File Group   Disk Group   Property     Value    File Type
ORCL         DATA         PRIORITY     MEDIUM
ORCL         DATA         STRIPING     COARSE   CONTAINER
ORCL         DATA         REDUNDANCY   HIGH     CONTROLFILE
ORCL         DATA         STRIPING     FINE     CONTROLFILE
ORCL         DATA         REDUNDANCY   MIRROR   DATAFILE
ORCL         DATA         STRIPING     COARSE   DATAFILE
...
```

lsqg

Purpose

Lists the quota groups in a disk group.

Syntax and Description

`lsqg [-G disk_group] [--quotagroup quota_group] [--suppressheader]`

The following table lists the options for the `lsqg` command.

Table 10-85   Options for the lsqg command

Option	Description
`-G disk_group`	Specifies the disk group.
`—quotagroupquota_group`	Specifies the quota group.
`—suppressheader`	Suppresses the headers in the output of the command.

If the disk group option is specified, then `lsqg` returns only information about quota groups in the specified disk group. If the quota group option is specified, then `lsqg` returns only information about that quota group.

The information about file groups is retrieved from `V$ASM_QUOTAGROUP` view.

The following example shows the use of the `lsqg` command. The first command in the example displays the properties of the file group `QG1` in the `DATA` disk group. The second command lists all the quota groups and their properties in the `DATA` disk group. The third command lists all the quota groups with name `QG1` in all the disk groups mounted by the current Oracle ASM instance. The fourth command lists all the quota groups in all the disk groups mounted by the current Oracle ASM instance.

Example 10-105   Using the lsqg command

```
ASMCMD [+] > lsqg -G DATA --quotagroup QG1
Used_Quota_MB   Quota_Limit_MB
100             200

ASMCMD [+] > lsqg -G DATA
Quotagroup_Num   Quotagroup_Name   Used_Quota_MB   Quota_Limit_MB
```

```
1                  GENERIC        12357          262143
2                  QG1            100            200
3                  QG2            400            420

ASMCMD [+] > lsqg --quotagroup QG1
Group_Num   Quotagroup_Num   Used_Quota_MB   Quota_Limit_MB
1           2                100             200
3           5                821             1024

ASMCMD [+] > lsqg
Group_Num   Quotagroup_Num   Quotagroup_Name   Used_Quota_MB   Quota_Limit_MB
1           1                GENERIC           12357           262143
1           2                QG1               100             200
1           3                QG2               400             420
2           1                GENERIC           54000           262143
3           1                GENERIC           11              262143
3           5                QG1               821             1024
```

mkfg

ⓘ **Note**

A multitenant container database is the only supported architecture in Oracle
Database 21c and later releases. While the documentation is being revised, legacy
terminology may persist. In most cases, "database" and "non-CDB" refer to a CDB or
PDB, depending on context. In some contexts, such as upgrades, "non-CDB" refers to
a non-CDB from a previous release.

Purpose

Creates a file group in a disk group based on an XML configuration file or script.

Syntax and Description

`mkfg { config_file.xml | 'contents_of_xml_script' }`

The following table lists the options for the `mkfg` command.

Table 10-86   Options for the mkfg command

Option	Description
`config_file.xml`	Name of the XML file that contains the configuration information for the file group. `mkfg` searches for the XML file in the current working directory of ASMCMD unless a path is specified
`contents_of_xml_script.xml`	The contents of an XML script enclosed in single quotes.

The root element of the XML configuration file or script is the `<file_group>` tag and has these
attributes:

- `name`: file group name

- `dg`: disk group name to which the file group belongs

- `client_type`: database, cluster, volume

- `client_name`: database, CDB, PDB, cluster or volume name

You can set properties using these tag:

- `name`: property name

- `value`: property value

- `file_type`: property file type (optional)

- `template`: Oracle ASM template name (optional)

The following is an example of an XML configuration file for `mkfg`. The configuration file creates a file group named `filegroup1` on disk group `dg_data1` for a database named `sample`. The file group is associated with quota group `quotagroup1`. The files in this file group have the redundancy set to mirror, except for the control file, which has a high redundancy. All other attributes are set to the default value.

```
<filegroup name="filegroup1" dg="dg_data1" client_type="database" client_name="sample">
  <p name="redundancy" value="mirror"/>
  <p name="redundancy" value="high" file_type="controlfile"/>
  <p name="quota_group" value="quotagroup1"/>
</filegroup>
```

For information about Oracle ASM File Groups, refer to "Managing Oracle ASM Flex Disk Groups".

Example

The following examples show the use of the `mkfg` command using an XML configuration file and an XML script.

Example 10-106   Using the mkfg command

```
ASMCMD [+] > mkfg config_file.xml

ASMCMD [+] > mkfg '<filegroup name="filegroup1" dg="dg_data1" client_type="database"
client_name="sample">
                    </filegroup>'
```

mkqg

Purpose

Adds a quota group to the disk group.

Syntax and Description

`mkqg -G disk_group quota_group [property] [value]`

The following table lists the options for the `chqg` command.

Table 10-87   Options for the mkqg command

Option	Description
`-G disk_group`	Specifies the disk group.
`quota_group`	Specifies the quota group.
`property`	Specifies the name of the property.
`value`	Specifies the value of the property.

The `mkqg` command adds the specified quota group with an optional the specified property to the specified disk group. For information about Oracle ASM file and quota groups, refer to "Managing Oracle ASM Flex Disk Groups".

The following are examples of the `mkqg` command. The first example adds the quota group `quota_grp1` to the `data` disk group without specifying a property. The second example adds the quota group `quota_grp2` to the `data` disk group while specifying the `quota` property.

Example 10-107   Using the mkqg command

```
ASMCMD [+] > mkqg -G data quota_grp1

ASMCMD [+] > mkqg -G data quota_grp2 quota 100G
```

mvfg

Purpose

Moves a file group in a disk group to the specified quota group.

Syntax and Description

```
mvfg -G disk_group --filegroup file_group quota_group
```

The following table lists the options for the `mvfg` command.

Table 10-88   Options for the mvfg command

Option	Description
`-G disk_group`	Specifies the disk group.
`--filegroup file_group`	Specifies the file group.
`quota_group`	Specifies the quota group.

The following is an example of the `mvfg` command. In the example, the file group `FG1` in the `DATA` disk group is moved to the quota group `QG1`

Example 10-108   Using the mvfg command

```
ASMCMD [+] > mvfg -G DATA --filegroup FG1 QG1
```

mvfile

Purpose

Moves a file to the specified file group in the same disk group where the file is stored.

Syntax and Description

```
mvfile file_name --filegroup file_group
```

The following table lists the options for the `mvfile` command.

Table 10-89   Options for the mvfile command

Option	Description
file_name	Specifies the file to be moved.
—filegroup *file_group*	Specifies the file group name.

The following `mvfile` commands move the specified Oracle ASM file to the named file group.

Example 10-109   Using the mvfile command

```
ASMCMD [+] > mvfile +data/orcl/controlfile/Current.260.684924747 --filegroup FG1

ASMCMD [+fra/orcl/archivelog/flashback] > mvfile log_7.264.684968167 --filegroup FG1
```

rmfg

Purpose

Removes an existing file group from a disk group.

Syntax and Description

```
rmfg [-r] disk_group file_group
```

The following table lists the options for the `rmfg` command.

Table 10-90   Options for the rmfg command

Option	Description
-r	Specifies a recursive operation.
disk_group	Specifies the disk group.
file_group	Specifies the file group.

The `-r` option specifies a recursive removal, including contents. All files in the disk group that the file group contains are removed at the same time that the file group is removed. You must specify this option to remove a file group that contains any files. If you omit this option and the file group is not empty, then ASMCMD returns an error and does not remove the file group.

The following example shows the use of the `rmfg` command. The first command in the example removes file group `FG1` from the `DATA1` disk group. The second command removes file group `FG2`, and its contents, from the `DATA2` disk group.

Example 10-110   Using the rmfg command

```
ASMCMD [+] > rmfg DATA1 FG1
ASMCMD [+] > rmfg -r DATA2 FG2
```

rmqg

Purpose

Removes a quota group from a disk group.

Syntax and Description

```
rmqg -G disk_group quota_group
```

The following table lists the options for the `rmqg` command.

Table 10-91   Options for the rmqg command

Option	Description
-G disk_group	Specifies the disk group.
quota_group	Specifies the quota group.

The following example shows the use of the `rmqg` command. The command in the example removes quota group `QG1` from the `DATA` disk group.

Example 10-111   Using the rmqg command

```
ASMCMD [+] > rmqg -G DATA QG1
```

ASMCMD Template Management Commands

This topic provides a summary of the ASMCMD template management commands.

Table 10-92 lists the ASMCMD template management commands with brief descriptions.

Table 10-92   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} ] } template
```

Table 10-93 lists the syntax options for the `chtmpl` command.

Table 10-93   Options for the chtmpl command

Option	Description
-G diskgroup	Name of the disk group.
template	Name of the template to change.

Table 10-93   (Cont.) Options for the chtmpl command

Option	Description
`--striping {coarse	fine}`	Striping specification, either `coarse` or `fine`.
`--redundancy { high	mirror	unprotected}`	Redundancy specification, either `high`, `mirror`, or `unprotected`.

At least one of the following options is required: `--striping` or `--redundancy`.

For information about updating templates, see "Modifying a Disk Group Template".

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 10-112   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 10-94 lists the syntax options for the `lstmpl` command.

Table 10-94   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".

Example

The following is an example of the `lstmpl` command. The example lists all details of the templates in the `data` disk group.

Example 10-113   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        AUDIT_SPILLFILES         COARSE  Y    MIRROR  COLD    COLD
DATA       1        AUTOBACKUP               COARSE  Y    MIRROR  COLD    COLD
DATA       1        AUTOLOGIN_KEY_STORE      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        INCR XTRANSPORT BACKUPSET COARSE Y    MIRROR  COLD    COLD
DATA       1        KEY_STORE                COARSE  Y    MIRROR  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        VOTINGFILE               COARSE  Y    MIRROR  COLD    COLD
DATA       1        XTRANSPORT               COARSE  Y    MIRROR  COLD    COLD
DATA       1        XTRANSPORT BACKUPSET     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}] template
```

Table 10-95 lists the syntax options for the `mktmpl` command.

Table 10-95   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`.

For information about adding templates, see Adding Templates to a Disk Group.

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 10-114   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 10-96 lists the syntax options for the `rmtmpl` command.

Table 10-96   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".

Example

The following is an example of the `rmtmpl` command that removes the `mytemplate` template from the `data` disk group.

Example 10-115   Using the ASMCMD rmtmpl command

```
ASMCMD [+] > rmtmpl -G data mytemplate
```

ASMCMD File Access Control Commands

This topic provides a summary of 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.

When administering users with ASMCMD, the users must be existing operating system users and their user names must have corresponding operating system user IDs. However, only users in the same cluster as the Oracle ASM instance can be validated.

> ⓘ **Note**
>
> When modifying ownership of files, the previous owner of a file cannot be dropped while the file is still open even after the ownership of the file has changed.

Table 10-97 lists the ASMCMD Oracle ASM File Access Control commands with brief descriptions.

Table 10-97   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.
rpusr	Replaces one a user with another in a disk group.

chgrp

Purpose

Changes the user group of a file or list of files.

Syntax and Description

`chgrp usergroup file [file ...]`

Table 10-98 lists the syntax options for the `chgrp` command.

Table 10-98   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.

If you change the user group settings of an open file, then the operation currently running on the file completes using the old user group settings. The new settings take effect when re-authentication is required.

Examples

The following are examples of the `chgrp` command that change the Oracle ASM user group of the specified files.

Example 10-116   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 10-99 lists the syntax options for the `chmod` command.

Table 10-99   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`
`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.

You can only set file permissions to read-write, read-only, and no permissions. You cannot set file permissions to write-only.

If you change the permission settings of an open file, then the operation currently running on the file completes using the old permission settings. The new settings take effect when re-authentication is required.

To view the permissions on a file, use the ASMCMD `ls` command with the `--permission` option. See "ls".

Examples

The following are examples of the `chmod` command that change the permissions of the specified files.

Example 10-117   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 10-100 lists the syntax options for the `chown` command.

Table 10-100   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.

If you change the ownership settings of an open file, then the following applies:

- The operation currently running on the file completes using the old ownership and user group settings. The new settings take effect when re-authentication is required.

- The new owner of the file cannot be dropped from the disk group until all instances in a cluster are restarted in a rolling manner.

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 10-118   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 10-101 lists the syntax options for the `groups` command.

Table 10-101   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 10-119   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 10-102 lists the syntax options for the `grpmod` command.

Table 10-102   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 10-120   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 10-103 lists the syntax options for the `lsgrp` command.

Table 10-103   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 10-121   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 10-104 lists the syntax options for the `lsusr` command.

Table 10-104   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 10-122   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 10-105 lists the syntax options for the `mkgrp` command.

Table 10-105   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 10-123   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 10-106 lists the syntax options for the `mkusr` command.

Table 10-106   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 10-124   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 10-107 lists the syntax options for the `passwd` command.

Table 10-107   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 10-125   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 10-108](#) lists the syntax options for the `rmgrp` command.

Table 10-108   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](#)".

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 10-126   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 10-109 lists the syntax options for the `rmusr` command.

Table 10-109   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 10-127   Using the ASMCMD rmusr command

```
ASMCMD [+] > rmusr data oracle2
```

rpusr

Purpose

Replaces one operating system (OS) user with another in a disk group.

Syntax and Description

```
rpusr diskgroup user1 user2
```

Table 10-110 lists the syntax options for the `rpusr` command.

Table 10-110   Options for the rpusr command

Option	Description
`diskgroup`	Specifies the name of the disk group.
`user1`	Name of the user to be replaced.
`user2`	Name of user that replaces the existing user in the disk group.

The `rpusr` command replaces one operating system user with another in an Oracle ASM disk group. The replacement user must not be a user currently in the disk group user list.

If the command succeeds, all files that were previously owned by current user are now owned by the replacement user. The current user is automatically removed from the user list of the disk group. Adding the current user back to the disk group is possible but this user will not own any files.

When changing the ownership of an open file, the new owner cannot be dropped with the owner's files until all instances within a cluster are restarted.

The command requires the SYSASM privilege to run.

Examples

The following is an example of the `rpusr` command that replaces the `oracle1` user with the `oracle2` user in the `data` disk group.

Example 10-128   Using the ASMCMD rpusr command

```
ASMCMD [+] > rpusr data oracle1 oracle2
```

11
Managing Oracle Flex ASM

Oracle Flex ASM enables Oracle ASM instances to run on a separate physical server from the database servers.

This section discusses Oracle Flex ASM in the following topics:

- [Overview of Oracle Flex ASM](#)
- [About Setting Up Oracle Flex ASM](#)
- [Administering Oracle Flex ASM](#)

> ⓘ **See Also**
>
> - *Oracle Clusterware Administration and Deployment Guide* for information about Oracle Flex Cluster support
> - Articles at [My Oracle Support](#) (`https://support.oracle.com`) for information about configuring Oracle Flex ASM with Oracle ASM and Oracle ACFS
> - [ASMCMD Client Cluster Management Commands](#) for information about ASMCMD commands to manage client clusters
> - *Oracle Grid Infrastructure Installation and Upgrade Guide* for information about installing and upgrading Oracle Grid Infrastructure

Overview of Oracle Flex ASM

> ⓘ **Note**
>
> Starting with Oracle Grid Infrastructure 21c, Member Clusters, which are part of the Oracle Cluster Domain architecture, are desupported. However, Domain Services Clusters continues to support Members Clusters in releases previous to Oracle Grid Infrastructure 21c.

> ⓘ **Note**
>
> Starting with Oracle Grid Infrastructure 21c, Domain Services Cluster (DSC), which is part of the Oracle Cluster Domain architecture, is deprecated and can be desupported in a future release.

Oracle Flex ASM enables an Oracle ASM instance to run on a separate physical server from the database servers. With this deployment, larger clusters of Oracle ASM instances can support more database clients while reducing the Oracle ASM footprint for the overall system.

When using Oracle Flex ASM, Oracle ASM clients are configured with direct access to storage.

With Oracle Flex ASM, you can consolidate all the storage requirements into a single set of disk groups. All these disk groups are mounted by and managed by a small set of Oracle ASM instances running in a single cluster. You can specify the number of Oracle ASM instances with a cardinality setting. The default is three instances.

A cluster is a set of nodes that provide group membership services. Each cluster has a name that is globally unique. Every cluster has one or more Hub nodes. The Hub nodes have access to Oracle ASM disks. Every cluster has at least one private network and one public network. If the cluster is going to use Oracle ASM for storage, it has at least one Oracle ASM network. A single network can be used as both a private and an Oracle ASM network. For security reasons, an Oracle ASM network should never be public. There can be only one Oracle Flex ASM configuration running within a cluster.

An Oracle ASM instance can operate in several configurations in Oracle Flex ASM:

- Local Oracle ASM clients with direct access to Oracle ASM disks (Standard Oracle ASM cluster)

- Oracle Flex ASM clients with direct access to Oracle ASM disks

- Oracle ACFS access through the Oracle ASM proxy instance

- Network-based connectivity to Oracle ASM disk groups with Oracle IOServer (IOS)

These configurations are illustrated in Figure 11-1, Figure 11-2, and Figure 11-3.

Local Oracle ASM clients with direct access to Oracle ASM disks (Standard Oracle ASM cluster)

With this mode, illustrated as **Hub Node A** in Figure 11-1, Oracle ASM continues to support existing standard architecture in which database clients are running with an Oracle ASM instance on the same host computer. The local client architecture is only supported on a Hub node.
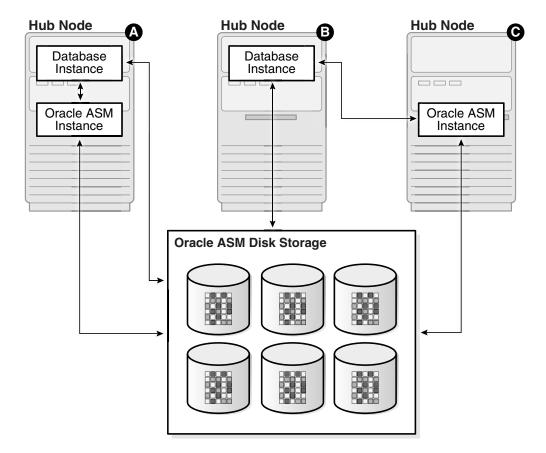
Figure 11-1   Oracle Flex ASM Client Configurations

In this configuration, the database instances are on the same Hub node as the Oracle ASM instance and are referred to as local Oracle ASM client instances. Oracle ASM metadata moves between Oracle ASM and the database instances. This client has direct I/O access to Oracle ASM disks.

Local mode does not use Oracle Flex ASM, so clusters configured with local Oracle ASM do not require an Oracle ASM network, nor do they contain other Oracle Flex ASM services.

Oracle Flex ASM clients with direct access to Oracle ASM disks

With this mode, illustrated as **Hub Node B** in Figure 11-1, database clients that are running on Hub nodes of the Oracle ASM cluster access Oracle ASM remotely for metadata, but perform block I/O operations directly to Oracle ASM disks. The hosts running the Oracle ASM server and the remote database client must both be Hub nodes. A Hub node is a node in an Oracle ASM cluster that is tightly connected with other servers and has direct access to a shared disk.

In this configuration, the database instances are on different host computers than the nearby Oracle ASM instance, as shown on **Hub Node C** in Figure 11-1, and are referred to as Oracle ASM client instances.

The databases are in the same Oracle ASM cluster as the Oracle ASM instance and the database instances are located on a Hub node. Oracle ASM metadata moves between Oracle ASM and the database instance. This client has direct I/O access to Oracle ASM disks.

Depending on the distribution of database instances and Oracle ASM instances, a database client may access Oracle ASM locally on the same node or remotely over the Oracle ASM network. This mode of operation is used by database clients on Hub nodes in the Oracle ASM

cluster. Direct access mode is also the only Oracle Flex ASM configuration supported by Oracle ASM cluster file system.

Oracle ACFS access through the Oracle ASM proxy instance

An Oracle ASM proxy instance is an Oracle instance running on a Hub node with a direct Oracle ASM client. An Oracle ASM proxy instance provides support for Oracle Advanced Cluster File System (Oracle ACFS) and Oracle ASM Dynamic Volume Manager (Oracle ADVM).

Figure 11-2 shows the configuration of Oracle ACFS and Oracle ADVM with an Oracle ASM Proxy server in Oracle Flex ASM.
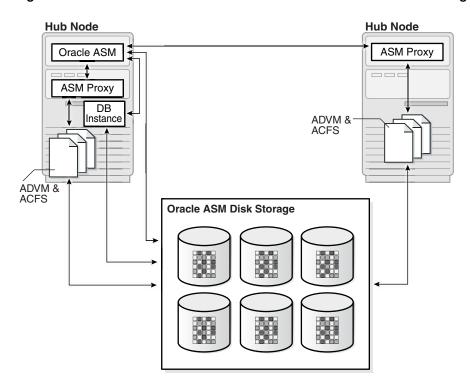
Figure 11-2   Oracle ACFS and Oracle ADVM in Oracle Flex ASM Configuration

The `INSTANCE_TYPE` initialization parameter is set to `ASMPROXY` for Oracle ASM proxy instances.

Network-based connectivity to Oracle ASM disk groups with Oracle IOServer (IOS)

An Oracle IOServer (IOS) instance provides Oracle ASM file access for Oracle AI Database instances on nodes of Oracle member clusters that do not have connectivity to Oracle ASM managed disks. For member clusters that do have connectivity to Oracle ASM managed disks, IOServer could be configured as a backup in the event of failure. An Oracle IOServer instance is used only with Oracle member clusters.

Figure 11-3 shows the relationship of Oracle IOServer and an Oracle AI Database in a member cluster.
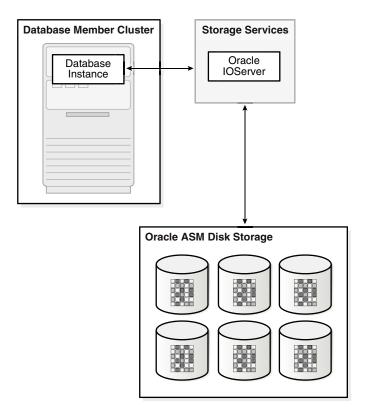
Figure 11-3   Oracle IOServer Configuration

In addition, IOS enables you to configure client clusters on such nodes. On the storage cluster, an IOServer instance opens up network ports to which clients send their IO. The IOServer instance receives data packets from the client and performs the appropriate IO to Oracle ASM disks similar to any other database client. On the client side, databases can use dNFS to communicate with an IOServer instance. However, there is no client side configuration so you are not required to provide a server IP address or any additional configuration information. On nodes and clusters that are configured to access Oracle ASM files through IOServer, the discovery of the Oracle IOS instance occurs automatically.

The IOS instance contains the following processes:

- Network processes: These processes open up network ports and receive dNFS requests from the client side. These requests are then placed in a queue for IO and Identifier processes to pick up and process. Network processes also pick up responses from those other types of processes and send the response back to the client.

- Identifier processes: These processes pick up dNFS requests to create, delete, identify, unidentify, and resize Oracle ASM files.

- IO processes: These processes perform the actual IO on files that are identified by the local IOS instance.

The `ASM_IO_PROCESSES` initialization parameter specifies the number of IO worker processes to be started in an Oracle IO server instance. For information about the `ASM_IO_PROCESSES` initialization parameter, refer to [ASM_IO_PROCESSES](#).

About Setting Up Oracle Flex ASM

To install an Oracle Flex ASM deployment, categorize the networks and choose the list of networks for use as Oracle ASM networks.

If you choose Oracle Flex ASM during a new installation, OUI requires you to choose the Oracle ASM networks.

The Oracle ASM listener resource is automatically created for each Oracle ASM network and then started on all nodes.

> ⓘ **See Also**
>
> *Oracle Grid Infrastructure Installation Guide* for information about Oracle Clusterware installation

Administering Oracle Flex ASM

Oracle Flex ASM components are administered with ASMCA, ASMCMD CRSCTL, SQL*Plus, and SRVCTL. In addition, you can set the `INSTANCE_TYPE` initialization parameter to specific instance types for use with Oracle Flex ASM.

- [Displaying Information About Oracle Flex ASM](#)
- [SRVCTL and CRSCTL Commands to Administer Oracle Flex ASM](#)
- [Relocating a Client With ALTER SYSTEM](#)
- [ASMCMD Commands to Manage Client Clusters](#)
- [Initialization Parameters for Oracle Flex ASM](#)

Displaying Information About Oracle Flex ASM

You can use the ASMCMD `showclustermode` command to determine whether Oracle Flex ASM is enabled. For example:

```
$ asmcmd showclustermode
ASM cluster : Flex mode enabled
```

You can view Oracle Flex ASM connections with SQL*Plus and ASMCMD commands. Fore example:

```
SQL> SELECT instance_name, db_name, status FROM V$ASM_CLIENT;
INSTANCE_NAME   DB_NAME  STATUS
--------------- -------- ------------
+ASM1           +ASM     CONNECTED
orcl1           orcl     CONNECTED
orcl2           orcl     CONNECTED

$ asmcmd lsct data
DB_Name  Status     Software_Version  Compatible_version  Instance_Name  Disk_Group
+ASM     CONNECTED      19.0.0.0.0           19.0.0.0.0    +ASM           DATA
orcl     CONNECTED      19.0.0.0.0           19.0.0.0.0    orcl1          DATA
orcl     CONNECTED      19.0.0.0.0           19.0.0.0.0    orcl2          DATA
```

SRVCTL and CRSCTL Commands to Administer Oracle Flex ASM

SRVCTL is extended to enable an administrator to create or change attributes of Oracle Clusterware resources. You can use SRVCTL to determine the status of the instances in an Oracle Flex ASM configuration. For example:

```
$ srvctl status asm -detail
ASM is running on mynoden02,mynoden01
ASM is enabled.
```

You can also use SRVCTL to determine whether Oracle Flex ASM is enabled. If enabled, then `srvctl config asm` displays the number of Oracle ASM instances that has been specified for use with the Oracle Flex ASM configuration. For example:

```
$ srvctl config asm
ASM instance count: 3
```

You can modify the Oracle ASM instance count, or cardinality, with the SRVCTL `modify asm` command. For example:

```
$ srvctl modify asm -count 4

$ srvctl modify asm -count ALL
```

Clients are automatically relocated to another instance if an Oracle ASM instance fails. If necessary, clients can be manually relocated.

The `SRVCTL UPDATE INSTANCE` and `SRVCTL UPDATE IOSERVER` commands can change the Oracle ASM instance for a database, or the Oracle IOServer instance for a database, or the Oracle ASM instance for an Oracle IOServer. For example:

```
$ srvctl update instance -db my_orcl_db -instance my_instance_1 -targetinstance
my_asm_instance_1

$ srvctl update ioserver -instance my_instance_2 -targetinstance my_asm_instance_2
```

You can run the `SRVCTL STATUS` commands with the `-detail` option to display which Oracle ASM instance or Oracle IOserver instance each client is connected to.

Every database user must have a wallet with credentials to connect to Oracle ASM. CRSCTL commands can be used by the database user to manage this wallet. All Oracle ASM user names and passwords are system generated.

You can run SRVCTL commands to manage Oracle ASM networks and listeners. For example, as the root user you can add an Oracle ASM network using `srvctl add asmnetwork`, with or without the optional network mask.

```
# srvctl add asmnetwork -netnum 1 -subnet nnn.nn.nn.n/255.255.255.0

# srvctl add asmnetwork -netnum 2 -subnet nnn.nn.nn.n
```

As the root or as CRS user, you can check the status or configure the Oracle ASM network with `srvctl config asmnetwork`.

```
$ srvctl config asmnetwork
ASM network 1 exists
Subnet IPv4: nnn.nn.nn.n/255.255.255.0/
Subnet IPv6:
Network is enabled
Network is individually enabled on nodes:
Network is individually disabled on nodes:
```

```
ASM network 2 exists
Subnet IPv4: nnn.nn.nn.n//
Subnet IPv6:
Network is enabled
Network is individually enabled on nodes:
Network is individually disabled on nodes:
```

As the CRS user, you can add an Oracle ASM listener with `srvctl add listener` You can use the default listener name (`LISTENER_ASM`) or specify a name for the listener.

```
$ srvctl add listener -asmlistener -netnum 1

$ srvctl add listener -asmlistener -netnum 2 -listener listener2
```

You can check the status of the Oracle ASM listener with `srvctl config listener`.

```
$ srvctl config listener -asmlistener
Name: LISTENER2_ASM
Type: ASM Listener
Owner: msmith
Subnet: nnn.nn.nn.n
Home: <CRS home>
End points: TCP:1526
Listener is enabled.
Listener is individually enabled on nodes:
Listener is individually disabled on nodes:
Name: LISTENER_ASM
Type: ASM Listener
Owner: msmith
Subnet: nnn.nn.nn.n
Home: CRS_hOME
End points: TCP:1525
Listener is enabled.
Listener is individually enabled on nodes:
Listener is individually disabled on nodes:
```

As the root user, you can run the `srvctl modify asmnetwork` command with the `-deptype` option to specify the dependency type (`mandatory` or `optional`) for Oracle ASM networks. The `-deptype` setting enables you to determine whether or not you want to tolerate the unavailability of any network.

- The `-deptype mandatory` option specifies a mandatory dependency on an Oracle ASM network. This means that the system does not tolerate the loss of that network. If only one of the Oracle ASM networks connects both the cluster nodes and Oracle ASM client clusters, then that network should be specified as `mandatory` because the Oracle ASM client clusters cannot run when that network is not available. For example:

  ```
  # srvctl modify asmnetwork -netnum 2 -deptype mandatory
  ```

- The `-deptype optional` option specifies an optional dependency on an Oracle ASM network. This means that the system can tolerate the loss of that network. If there are multiple Oracle ASM networks that only connect the cluster nodes, then you can specify them as `optional` to tolerate the failure of one of the networks. For example:

  ```
  # srvctl modify asmnetwork -netnum 2 -deptype optional
  ```

  The `optional` value is the default setting for `-deptype`.

Relocating a Client With ALTER SYSTEM

You can use the `ALTER SYSTEM RELOCATE CLIENT` command to relocate a client. For example:

```
SQL> ALTER SYSTEM RELOCATE CLIENT 'client-id';
```

In the previous SQL statement, `client-id` is of the form `instance_name:db_name`. The `INSTANCE_NAME` and `DB_NAME` columns are contained in the `V$ASM_CLIENT` view. You must connect as `SYSASM` to the Oracle ASM instance to run the SQL statement. When you issue this statement, the connection to the client is terminated and the client fails over to the least loaded instance. If the client is currently connected to the least loaded instance, then the connection to the client is terminated and the client fails over to that same instance.

ASMCMD Commands to Manage Client Clusters

You can use ASMCMD commands to administer the configuration of client clusters.

You can list client clusters with the `lscc` command and remove client clusters with the `rmcc` command.

You create a client cluster with the `mkcc` command or change a client cluster with the `chcc` command. For example, you can change the storage method from direct Oracle ASM to indirect Oracle ASM, or change from indirect Oracle ASM to direct Oracle ASM using the ASMCMD `chcc` and `mkcc` commands.

Initialization Parameters for Oracle Flex ASM

There are no new initialization parameters specifically for instances in an Oracle Flex ASM configuration; however, the settings of existing parameters should be reviewed and possibly adjusted for the Oracle Flex ASM environment.

The value of the `INSTANCE_TYPE` initialization parameter specifies the type of instance, including Oracle ASM proxy instances. In addition to `ASM` and `RDBMS` values, the `ASMPROXY` value identifies Oracle ASM proxy instances.

> ⓘ **See Also**
>
> - [Recommended Settings for Oracle ASM Initialization Parameters](#) for information about initialization parameters related to Oracle ASM
> - [ASMCMD Client Cluster Management Commands](#) for information about ASMCMD commands to manage client clusters
> - *Oracle AI Database Reference* for more information about the `INSTANCE_TYPE` initialization parameter
> - *Oracle AI Database SQL Language Reference* for information about the `ALTER SYSTEM` statement
> - *Oracle Clusterware Administration and Deployment Guide* for information about CRSCTL commands
> - *Oracle Clusterware Administration and Deployment Guide* and *Oracle Real Application Clusters Administration and Deployment Guide* for information about SRVCTL commands
> - *Oracle Grid Infrastructure Installation and Upgrade Guide* for information about installing Oracle Clusterware

12

Creating Database Files in Oracle ASM Using SQL*Plus

Oracle Automatic Storage Management (Oracle ASM) can be deployed using SQL*Plus.

This section contains 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 AI 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 example shows the use of the `CREATE DATABASE` statement, where database files are created and managed automatically by Oracle ASM. 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 can be run at the SQL prompt:

```
SQL> 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". 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:

```
SQL> 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:

```
SQL> 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 file names 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:

```
SQL> 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. These 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 can be run at the SQL prompt:

```
SQL> ALTER DATABASE ADD LOGFILE;
```

Creating Control Files in Oracle ASM

Control files can be explicitly created in multiple disk groups. The file names for control files are automatically generated.

If you must specify a control file by name, then you can use alias file names 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 file names. However, the use of the `RESETLOGS` option requires the use of a file creation form for the specification of the log files.

Example 12-1   Creating control files in Oracle ASM using NORESETLOGS

The SQL statement in the following example 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
;
```

Example 12-2   Creating control files in Oracle ASM using RESETLOGS

The following example is a `CREATE CONTROLFILE` SQL statement for a database with data files, but uses a `RESETLOGS` clause, which 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 file names are created in default subdirectories with unique names. However, all `LOG_ARCHIVE_DEST_n` parameters must have a unique string. To enter the same disk group more than once, subdirectories must be used. For example, you can set `LOG_ARCHIVE_DEST_n` as follows:

```
LOG_ARCHIVE_DEST_1 = 'LOCATION=+FRA/orcl/arc_1'
LOG_ARCHIVE_DEST_2 = 'LOCATION=+FRA/orcl/arc_2'
...
```

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 Oracle AI Database under default subdirectories. If `LOG_ARCHIVE_DEST` is set to a directory in a disk group, `LOG_ARCHIVE_FORMAT` has its normal semantics for the file alias. The underlying file is still generated in the default directories for the disk group with a system generated name.

You can also set the `DB_RECOVERY_FILE_DEST` initialization parameter to a disk group destination. In Example 12-3, archive log files are generated with `ARCHIVELOG` mode enabled, `DB_RECOVERY_FILE_DEST` set to the `+FRA` disk group, and `ORCL` is the value of the `DB_UNIQUE_NAME` parameter.

> ⓘ **See Also**
>
> *Oracle AI Database Administrator's Guide* for information about managing archived redo log files

Example 12-3   Archived log files in an Oracle ASM disk group

```
SQL> SHOW PARAMETER DB_RECOVERY_FILE_DEST

NAME                                 TYPE        VALUE
```

```
----------------------------------- ----------- ------------------------------
db_recovery_file_dest               string      +FRA

SQL> SELECT SUBSTR(name,1,64) name FROM V$ARCHIVED_LOG;

NAME
----------------------------------------------------------------
+FRA/ORCL/ARCHIVELOG/2013_06_24/thread_1_seq_112.260.818960623
+FRA/ORCL/ARCHIVELOG/2013_06_24/thread_1_seq_113.261.818978423
+FRA/ORCL/ARCHIVELOG/2013_06_25/thread_1_seq_114.262.818993031
+FRA/ORCL/ARCHIVELOG/2013_06_25/thread_1_seq_115.263.819032439
+FRA/ORCL/ARCHIVELOG/2013_06_25/thread_1_seq_116.264.819047121
+FRA/ORCL/ARCHIVELOG/2013_06_25/thread_1_seq_117.265.819064815
+FRA/ORCL/ARCHIVELOG/2013_06_26/thread_1_seq_118.266.819079417
...
```