Automatic Storage Management (ASM)

Technical overview of ASM from Oracle Database 11g Release 2 to the present

August, 2024, Version [1.0]

ORACLE

Purpose statement

This document provides an overview of ASM features and enhancements starting with 11g Release 2 to the latest version of Oracle ASM. It is intended solely to help you assess the business benefits of utilizing ASM to manage Oracle Database.

Disclaimer

ORACLE

Table of Contents

Introduction

Automatic Storage Management (ASM) is one of the most successful and widely adopted Oracle database features. ASM, introduced in Oracle Database 10g and through its many releases, has evolved to meet the changing needs of the Oracle Database itself as well as the hardware environments in which the Oracle database operates. This paper presents ASM features by reviewing the evolution of features that began in Oracle Database 10g and continues to new ASM features in the current release. If the reader's interest is primarily with a particular release, they can skip to the section titled "Current ASM releases and beyond."

Automatic Storage Management Overview and Background

Automatic Storage Management (ASM) is a purpose-built file system and volume manager for the Oracle database. ASM was first released with Oracle Database 10g. For Oracle databases, ASM simplified both the file system and volume management aspects of database management. In addition to simplifying storage management, ASM improved file system scalability, performance, and database availability. These benefits hold for both single-instance and Oracle Real Application Cluster (Oracle RAC) databases.

Before ASM was introduced, customers typically had to deploy third-party file systems and volume management products. With ASM, storage resources are allocated to ASM for management and database data are kept in ASM Disk Groups

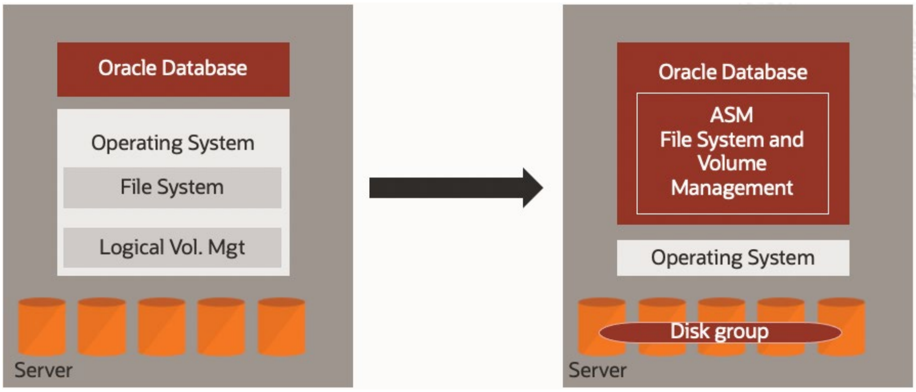

Figure 1. ASM replacing Operating System Logical Volume Manager and File System

Because of its innovative rebalancing of data in Disk Groups, ASM provides best-in-class performance by distributing data evenly across all storage resources. Whenever the physical storage configuration changes, data is redistributed to maintain an even distribution. This redistribution is called rebalancing and maintains an even distribution of IO for optimal performance. Lastly, the ASM architecture scales to very large configurations in terms of both databases and storage, all without compromising functionality or performance.

ASM is built to maximize database availability. For example, it provides self-healing automatic mirror reconstruction and resynchronization and supports dynamic and on-line storage reconfigurations. Features such as just-in-time provisioning and clustered pooling of storage help customers realize significant cost savings and achieve lower total cost of ownership, making it ideal for database consolidation. ASM provides all these capabilities without additional licensing fees.

In summary, ASM is a file system and volume manager optimized for Oracle database files providing:

- Simplified and automated storage management
- Increased storage utilization, uptime, and agility
- Predictable performance and availability service levels

Total Storage Management with Oracle Database 11g Release 2

With the release of Oracle Database 11g Release 2, Oracle added the ASM Cluster File System (ACFS) to complement ASM's file management for the database. ACFS provides the same level of storage management provided by ASM. Specifically, ACFS simplifies and automates storage management functions, increases storage utilization, uptime and agility to deliver predictable performance and availability for conventional file data stored outside an Oracle Database.

- ACFS includes: Automatic Storage Management Dynamic Volume Manager (ADVM) as a volume manager for Automatic Storage Management Cluster File System.
- Automatic Storage Management Cluster File System (ACFS) provides advanced data services and security features for managing general purpose files.
- ASM, ACFS, and Oracle Clusterware are bundled and packaged in the Oracle Grid Infrastructure home

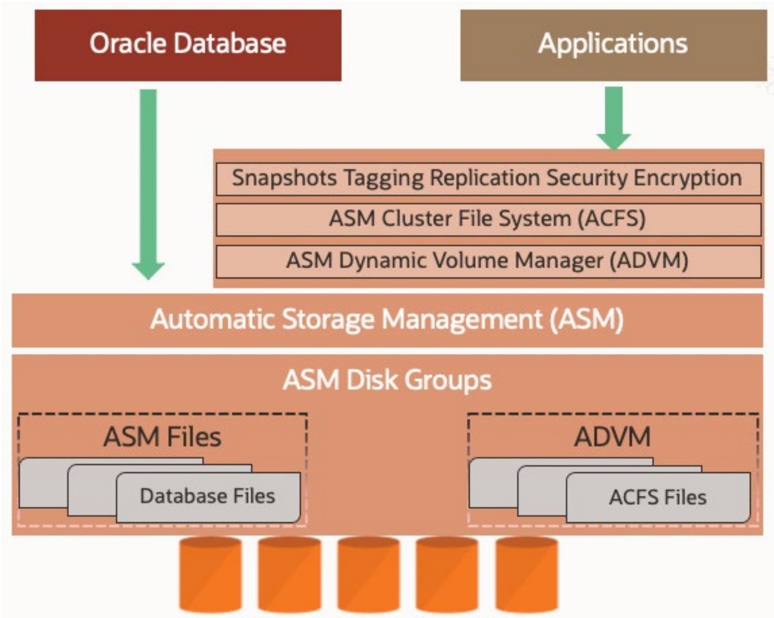

Figure 2. ASM System Layers

Oracle Grid Infrastructure provides an integrated foundation for database and general-purpose files as well as an infrastructure for clustered environments. Oracle Grid Infrastructure streamlines management of volumes, file systems and cluster configurations, therefore eliminating the need for multiple third-party software layers that add complexity and cost.

Flexible Storage Management

Flex ASM in Oracle Database 12c Release 1

ASM in Oracle Database 12c Release 1 was enhanced to address storage management needs for environments with larger cluster configurations by seamlessly adapting to frequently changing cluster configurations, which are expected in such environments. Additional features introduced in this release include enhancements for Oracle's engineered systems, such as Exadata and the Oracle Database Appliance.

The most significant enhancement for ASM in Oracle Database 12c Release 1 is a set of features collectively called Oracle Flex ASM. Oracle Flex ASM provides critical capabilities for large-scale clustering environments. These environments typically deploy database clusters of varying sizes that not only have stringent performance and reliability requirements but also must be able to rapidly adapt to changing workloads with minimal management.

Oracle Flex ASM fundamentally changed the ASM cluster architecture. Before the introduction of Oracle Flex ASM, an ASM instance had to run on every server in a cluster. Each ASM instance communicated with the other ASM instances in the cluster, and collectively they presented shared Disk Groups to database clients running in the cluster. The collection of ASM instances formed an ASM cluster. If an ASM instance were to fail before the introduction of Flex ASM, then all the database instances running on the same server as the failing ASM instance, also failed. The gray boxes in figure 3 represent ASM instances in an environment before Oracle Database 12c Release 1.

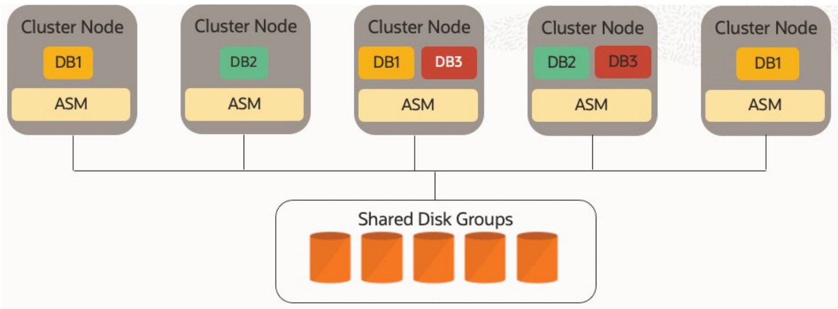

Figure 3. Oracle RAC deployment on ASM

With Oracle Database 12c Release 1, a smaller number of ASM instances run on a subset of servers in the cluster. The number of ASM instances running is called the ASM cardinality. The default ASM cardinality is 3, but that can be easily changed with a Clusterware command. If one server fails that is running with an ASM instance, then Oracle Clusterware starts a replacement ASM instance on a different server to maintain the pre-defined ASM cardinality.

If an ASM instance fails for whatever reason, any active Oracle Database 12c or later database instance relying on that ASM instance simply reconnects to a surviving ASM instance on a different server. Whenever a database instance needs to form a connection to an ASM instance for service, Oracle Clusterware attempts to load balance the new connection, ensuring that the total load of connections is evenly distributed across all active ASM instances.

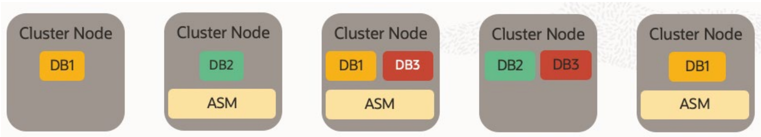

Figure 4a. Flex ASM

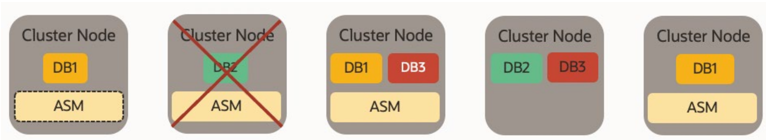

Figure 4b. Flex ASM: Automatic ASM Instance failure handling

Figures 4a and 4b illustrate the ASM architecture with Oracle Flex ASM following a failure. With Flex ASM, there are a reduced number of ASM instances required in the cluster, and Oracle Database 12c clients can connect across the network to an ASM instance on any server. Furthermore, Oracle Database 12c clients can failover to a surviving server with an ASM instance if a server with an ASM instance fails, all without disruption to the database client. Flex ASM also reduces the resource consumption of ASM instances such as CPU, Memory and inter-instance messages between ASM instances.

Flex Disk Groups and Database-Oriented Storage Management in 19c

Before ASM, a significant challenge for large database environments was achieving optimal performance and the best storage configurations for databases with differing objectives. For example, files belonging to one database might be best placed on a particular file system and storage configuration providing the needed performance for a specific application. However, as the database grows in size and demand, the DBA had to make ongoing changes in the storage configuration to keep pace with the changing database workload.

ASM simplified this effort by allowing a small number of Disk Groups for containing all the databases, without regard to the underlying storage configuration. This meant that a particular Disk Group was the home for many different databases. This organization of keeping databases in a small number of Disk Groups is called Disk Group-oriented storage management. See Figure 5 below, with many different databases and their files, sharing a common Disk Group.

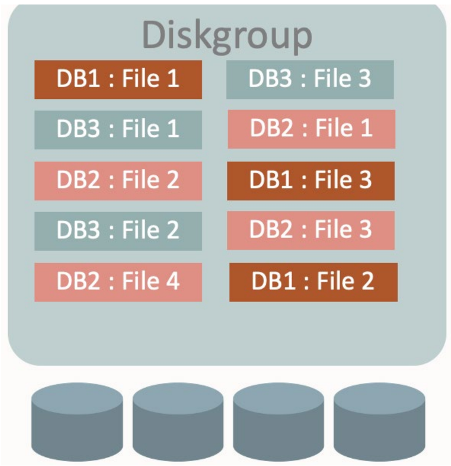

Figure 5: Disk Group oriented Storage Management

Managing storage by allocating storage resources to a few Disk Groups greatly helps with the manageability for organizations deploying many databases. It eliminated the piecemeal mode of the past where administrators had to manage the relationship between database objects and storage with fine granularity.

One challenge with course-grain management of Disk Group-oriented management is when it comes to consolidating a large number of databases, having different requirements, into just a few simple Disk Groups. It was sometimes observed that customers would deploy many Disk Groups for storing databases that have different requirements. For example, production databases might be separated from test and development databases, because the latter did not have the same performance or reliability requirements as the production databases. While separating databases into different Disk Groups provides finer granularity, with respect to control, it works against the objective of reducing management overhead through consolidation.

File Groups

For these reasons, Oracle introduced the concept of "Database-Oriented Storage Management." This concept introduces a new Disk Group type, called a Flex Disk Group. With Flex Disk Groups, all files belonging to a specific database are collectively identified with a new ASM object called a File Group.

A File Group logically contains the files associated with a single database. A File Group provides the means for identifying all the files that are part of a database and that reside within a single Disk Group. An individual database may have multiple File Groups residing in different Disk Groups. Figure 6 below shows this logical grouping.

When a database is first created, a File Group is also created with the same name of the database. However, if a File Group with that name already exits, then the existing File Group is used. Note that File Group names are unique within a Disk Group meaning different Disk Groups can have File Groups with the same name. This provides for a database to span multiple Disk Groups without a naming conflict.

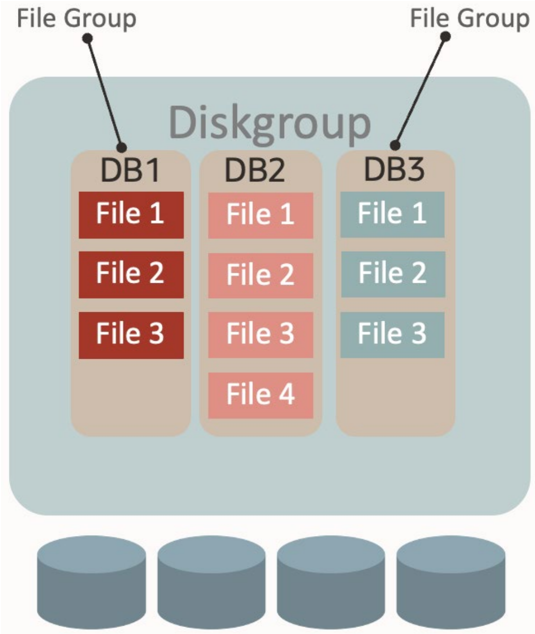

Figure 6 Database Oriented Storage Management

Flexible Disk Group Redundancy

Different File Groups within a single Flex Disk Group can have different redundancies. Additionally, File Group redundancy can be changed as a situation requires. For example, a production database can use High redundancy, which is three-way mirrored, while a test database, in the same Disk Group, can have a File Group with Mirror redundancy, which is two-way mirrored. If desired, the redundancy of a database, i.e., the File Group, can be changed online. When a File Group's redundancy is changed, ASM invokes a rebalance operation causing the redundancy change at the storage level. File Group names are unique within a Disk Group. However, different Disk Groups can have File Groups with the same name. This provides for a database to span Disk Groups without creating a naming conflict.

Enabling File Groups to have different redundancies within a Disk Group is very useful in reducing storage costs. This feature allows individual databases to be stored with the least redundancy it needs rather than requiring all databases in a Disk Group to be stored at the highest redundancy required of the most critical database in the Disk Group. For example, a Disk Group may have ten databases and perhaps only one of the databases require three-way mirroring. The other nine might get by with two-way mirroring. Without File Groups, administrators would need to either managed multiple Disk Groups of different redundancies, or to use the highest redundancy of the most critical database, for the databases.

Quota Groups

An important feature required for consolidating databases in a Disk Group is storage quota management. Without the means of providing quota management, a single database could consume all the space is a Disk Group. Flex Disk Groups provide a new feature called Quota Groups.

A Quota Group is a logical container specifying the amount of Disk Group space that one or more File Groups are permitted to consume. As an example, in Figure 7 below, Quota Group A contains File Groups DB1 and DB2, whereas Quota Group B contains File Group DB3. The databases in Quota Group A are then limited by the specification of available space in that Quota Group.

Every Flex Disk Group has a default Quota Group. If a File Group i.e. database, is not assigned a Quota Group, it is then assigned to the default Quota Group. In fact, it is quite possible that the sum of space represented by all the Quota Groups may exceed the total physical space available. Consequently, Quota Groups represent a logical capacity limit of available space.

Changing Quota Groups requires ASM administrative privileges. An ASM administrator can create a set of Quota Groups in which subsequent databases are allocated. Separation of authorization to create and alter Quota

Groups to an ASM administrator prevents individual database administrators from intentionally or otherwise from changing their allocated space. Quota Groups facilitate consolidating many databases into a single Flex Disk Group by preventing any single database from consuming more than its fair share of storage and inhibiting the operation of the other databases.

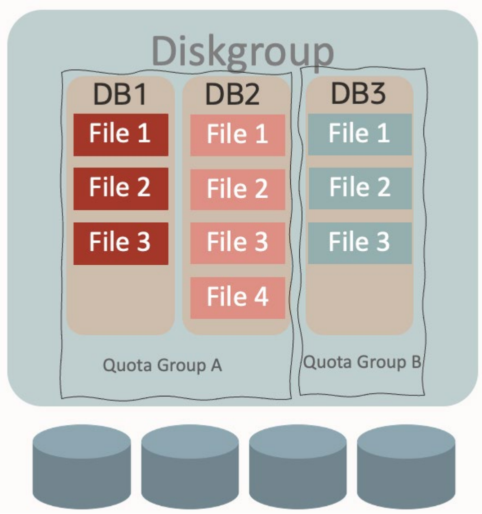

Figure 7 Quota Space Management

ASM Extended Disk Groups

With an enterprise's core operations at stake, it is vital that underlying storage for critical databases have the highest reliability and robustness. The design of ASM, from the beginning, has been to eliminate data loss after a hardware failure and ensure ongoing operation. For example, Normal and High-Redundancy Disk Groups provide data access in the face of storage failures. Early implementers of ASM saw an opportunity to extend an Oracle RAC cluster's availability by deploying RAC clusters across two closely located datacenters. This design used ASM mirroring across the datacenters so that data availability is ensured despite a complete failure of a datacenter. Figure 8a represents the architecture of what became known as Extended RAC using ASM mirroring.

The benefit of the "Extended RAC" architecture is that ASM mirrors file extents across two different Failure Groups, with each Failure Group located in a different datacenter. If one datacenter fails, then all the file extents required for the RAC instance in the surviving datacenter, remains available in that datacenter.

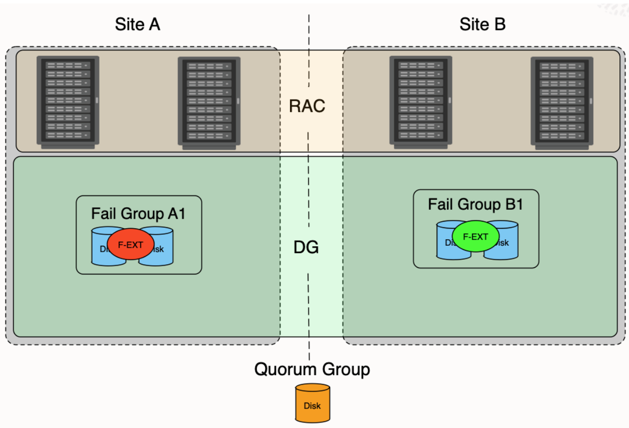

Figure 8a: Oracle Extended RAC Clusters with ASM Disk Groups

With two sites in an Extended Distance Cluster, there is a need for establishing quorum when there is a communications failure in the storage fabric and networking components connecting the two sites. Establishing quorum means the way in which Oracle Clusterware, that is operating independently on servers in each site, can agree as to which site survives after the communication failure. 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 when they are stored in a Disk Group containing a Quorum Failure Group. While Quorum Failure Groups can be used with any Disk Group, they are particularly useful for establishing quorum in an Extended Distance Cluster as they can be used to eliminate the need for a third data site.

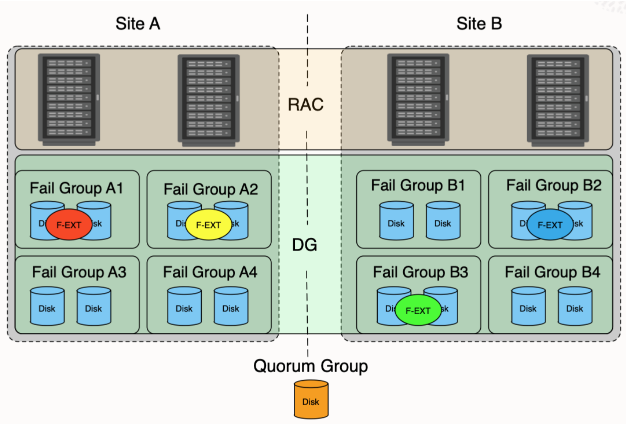

Figure 8b: Oracle Extended RAC with Extended Disk Groups

Current ASM Releases and Beyond

ASM Database Cloning in Oracle Database 19c

A storage system's ability to rapidly replicate data for deploying test and development systems is a well understood and appreciated capability. Most storage-based replication technologies utilize either mirror splitting, within storage arrays or snapshot replication in file systems. ASM, with Flex Disk Groups, now provide the ability for creating near instantaneous copies of complete databases. These database copies can be used for test and development, or when used with an Exadata system, provide a read-only master for an Exadata snapshot copies.

The advantage of ASM database clones, when compared with storage array-based replication, is that ASM database clones replicate databases rather than generic files or blocks of physical storage. Storage array or file system-based replication, in a database environment, requires coordination between database objects being replicated with the underlying technology doing the replication. With ASM Database Clones, the administrator does not need to worry about the physical storage layout. This mode of replication is another aspect of database-oriented storage management provided with ASM Flex Disk Groups.

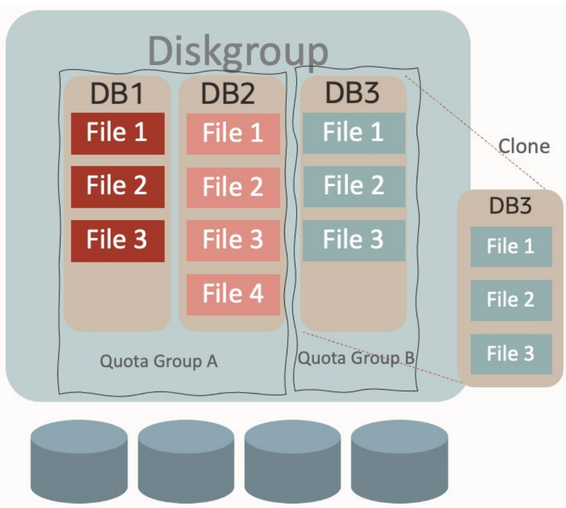

Figure 9 Oracle Database Clones using ASM

ASM database cloning works by leveraging ASM redundancy. Previously, as a protection against data loss from a hardware failure, ASM provided up to two additional redundant copies of a file's extents. Flex Disk Groups now can provide up to six redundant (Only first three copies are guaranteed to be in different failure groups. Thus, even though we have six copies, a failure of a disk may take out more than one copy. The disk group can, therefore, only tolerate two failures, the additional copies are used for splitting), copies in which one or more of the copies can be split off to provide a near-instantaneous replica. The process involves two phases, the first phase is the prepare phase and the second is the near instantaneous splitting of the file extent copies providing a distinct replica that is independent of the original. When an ASM database clone is made, all the files associated with the database are split together and provide an independent database. Figure 9 represents the splitting of the files for DB3 providing a separate and independent database DB3a. ASM database cloning is provided for Oracle Multitenant PDBs.

Parity-based Protection for Write-Once Files in Oracle Database 19c

For Oracle Database 19c, Parity Redundancy is now available as an additional protection option for write-once files. Write-once files are files such as backup sets and archived logs. Such files are typically written once and never rewritten.

Before parity protection for Flex Disk Groups, file protection could be set to unprotected, mirror, or high protection. Now parity protection is available for write-once files. Please note that parity protection is not supported on data files and other read/write files.

Parity protection requires a minimum of three regular (not quorum) Failure Groups in a Flex Disk Group. When parity protection is set, each parity extent set of the file has two data extents. That scenario incurs 50% redundancy overhead rather than 100% redundancy overhead for two-way mirrored files

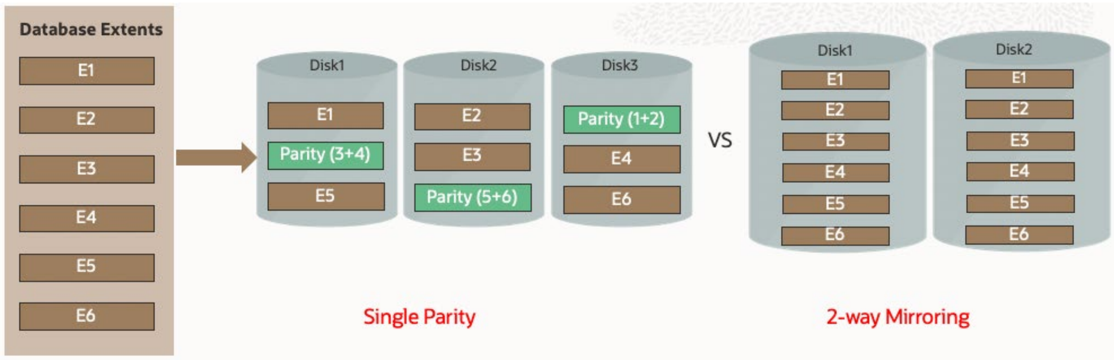

Figure 10: Single Parity

In figure 10 above, a parity extent is created for every two data extents. These extents are distributed across Failure Groups as represented as individual disks in the figure. This provides comparable protection against hardware failure as two-way mirroring provides, but at a fifty percent reduction of storage overhead. Parity protection would be slightly less robust as two-way mirroring since the possible loss of a single disk, out of three disks, is a bit greater than the possible loss of a disk out of two disks, in the case of mirroring. It should be noted that the parity extents are distributed across all Failure Groups to ensure an even IO workload balance.

When the File Group redundancy property is modified from Mirror to Parity protection, or Parity to Mirror, existing files are not reformatted. However, files created in the future adopt the new protection configuration.

Enhanced Parity Protection in Oracle Database 21c

In Oracle 21c, ASM extends parity protection benefits with the option of double parity protection for write-once files. This is intended to provide comparable protection as high redundancy provides with three-way mirroring. Subsequently, this feature was backported to Oracle 19c's latest Release update. Figure 11 below illustrates how double parity protection works.

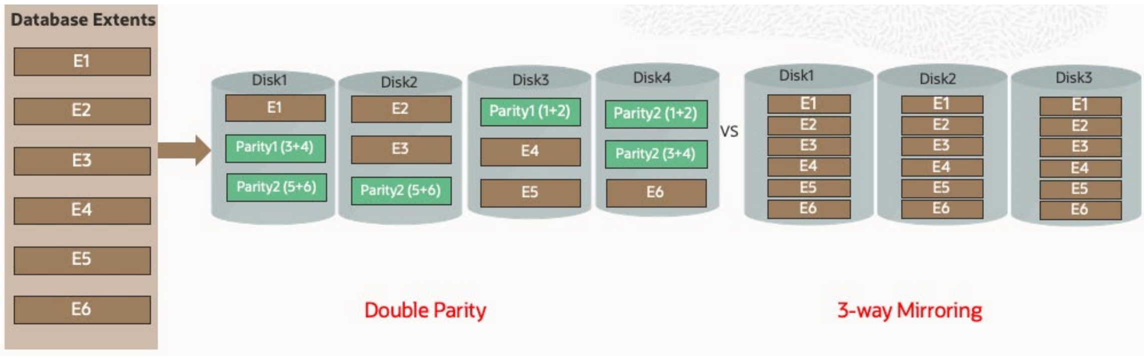

Figure 11: Double Parity

File Group Templates

Database-oriented storage management, offered in Oracle 12c Release 2, introduced File Groups. File Groups provides the functionality to identify all the files within a Disk Group belonging to a specific database, CDB, or PDB. File Groups have named properties representing the collective nature of the files in the File Group. For example, one important property is redundancy. This attribute specifies the type of redundancy for the files, e.g. High, Mirror, Parity or Unprotected. A File Group property can be modified as in this example:

ALTER DISKGROUP DG1 MODIFY FILEGROUP FileGroup_PDB1 SET 'datafile.redundancy' = 'HIGH';

As described previously, File Groups are created automatically through actions such as creating a database when the File Group does not already exist. An administrator could manually modify the properties in such cases, but that requires extra work and potentially involves data movement to change redundancy. A means for creating a File Group Template has been introduced in Oracle 21c to simplify the management of these properties. This feature has been subsequently backported to Oracle 19c latest Release Update. The template can be set beforehand to capture the desired properties. Later, the template is used whenever an action requiring the creation of a File Group is undertaken. An example of creating a File Group Template is:

ALTER DISKGROUP data ADD FILEGROUP fg_template1 TEMPLATE SET 'datafile.redundancy'='HIGH';

Or to modify a previously created template:

ALTER DISKGROUP data MODIFY FILEGROUP fg_template1 SET 'redundancy'='high';

Templates can be defined in the spfile as follows

DB_FILE_CREATE_DEST = +DATA(FG$fg_template1)

File Group Templates greatly simplify database deployment. Without File Group Templates, the user has to create the File Group manually prior to creating a database, in order to define the properties beforehand. A limitation of manually created File Groups is that they do not automatically get deleted when the database is dropped, while auto-created File Groups are cleaned up when the database is deleted.

Conclusion

Automatic Storage Management was created to address the storage management challenges in Oracle database environments. First introduced in Oracle 10g, ASM with that release, undertook perhaps the most significant need in storage management. That challenge was and is simplifying the initial storage management tasks and eliminating the need for tenuous and ongoing reconfiguration of the storage environment as database demands change.

The next phase of ASM's evolution came in Oracle Database 11g Release 2 with the introduction of ACFS. ACFS extended storage management for all the customer's data. This eliminated the need for multiple storage management frameworks. ASM managed the storage for the database environment and ACFS provided for data stored outside the database, all under the ASM management umbrella.

ASM in Oracle Database 12c Release 1 continued the tradition of storage management evolution with the objective of efficiently supporting large cluster configurations, including on-premises cloud deployments, with the introduction of Flex ASM. This evolution continued with Oracle Database 12c Release 2 enabling greater ease for consolidating many databases into a few Disk Groups through the introduction of database-oriented storage management. The key feature of database-oriented storage management is the Flex Disk Group that provides quota management and quality of service controls.

Oracle Database 19c and later releases introduce additional features facilitating database management, including ASM database cloning. ASM database cloning provides point-in-time database replicas built on database-oriented storage management delivered in Oracle Database 12c Release 2. In later releases, Oracle Database 19c and 21c introduce features that reduce overall storage costs with parity-protected data for write-once files. This feature reduces total physical storage required for files such as backup sets and archived logs.

In summary, ASM started with a simple yet powerful idea and has evolved over two decades of development. Each phase of evolution has been responsive to changing technology and customer requirements. ASM is an important piece of the database stack and is universally embraced by customers far and wide.

ORACLE

Connect with us

Call +**1.800.ORACLE1** or visit **oracle.com**. Outside North America, find your local office at: **oracle.com/contact**.

blogs.oracle.com facebook.com/oracle twitter.com/oracle