Data ONTAP® 8.1
Physical Storage Management Guide
For Cluster-Mode

NetApp, Inc.
495 East Java Drive
Sunnyvale, CA 94089 U.S.A.
Telephone: +1 (408) 822-6000
Fax: +1 (408) 822-4501
Support telephone: +1 (888) 4-NETAPP
Documentation comments: doccomments@netapp.com
Information Web: www.netapp.com

Part number: 210-05789_B0
Updated for Data ONTAP 8.1.2 on 24 February 2014

Contents

Managing disks using Data ONTAP

Disks provide the basic unit of storage for storage systems running Data ONTAP that use native disk shelves. Understanding how Data ONTAP uses and classifies disks will help you manage your storage more effectively.

How Data ONTAP reports disk types

Data ONTAP associates a type with every disk. Data ONTAP reports some disk types differently than the industry standards; you should understand how Data ONTAP disk types map to industry standards to avoid confusion.

When Data ONTAP documentation refers to a disk type, it is the type used by Data ONTAP unless otherwise specified. *RAID disk types* denote the role a specific disk plays for RAID. RAID disk types are not related to Data ONTAP disk types.

For a specific configuration, the disk types supported depend on the storage system model, the disk shelf type, and the I/O modules installed in the system. For more information about the types of disks supported by your configuration, see the *Hardware Universe* (formerly the *System Configuration Guide*) at *support.netapp.com/knowledge/docs/hardware/NetApp/syscfg/index.shtml*.

The following tables show how Data ONTAP disk types map to industry standard disk types for the SAS and FC-AL storage connection architectures, and for third-party storage:

Table 1: SAS storage connection architecture

Data ONTAP disk type	Primary disk characteristic	Industry standard disk type	Description
BSAS	Capacity	SATA	Bridged SAS–SATA disks with added hardware to enable them to be plugged into a SAS shelf.
FSAS	Capacity	NL-SAS	Near Line SAS
MSATA	Capacity	SATA	SATA disk in multi-disk carrier disk shelf
SAS	Performance	SAS	
SATA	Capacity	SATA	Available only as internal disks for FAS20xx systems.
SSD	High-performance	SSD	Solid-state disks

Table 2: FC-AL storage connection architecture

Data ONTAP disk type	Primary disk characteristic	Industry standard disk type	Description
ATA	Capacity	SATA	
FCAL	Performance	FC	

Table 3: Third-party storage

Data ONTAP disk type	Primary disk characteristic	Industry standard disk type	Description
LUN	N/A	LUN	A logical storage device backed by third-party storage and used by Data ONTAP as a disk. In this document, these LUNs are referred to as *array LUNs* to distinguish them from the LUNs that Data ONTAP serves to clients.

For information about best practices for working with different types of disks, see *Technical Report 3437: Storage Subsystem Resiliency Guide*.

Related concepts

Rules for mixing HDD types in aggregates on page 82
Storage connection architectures and topologies supported by Data ONTAP on page 10

Related references

Understanding RAID disk types on page 16

Related information

TR 3437: Storage Subsystem Resiliency Guide

Storage connection architectures and topologies supported by Data ONTAP

Data ONTAP supports two storage connection architectures: serial-attached SCSI (SAS) and Fibre Channel (FC). The FC connection architecture supports three topologies: arbitrated loop, switched, and point-to-point.

- SAS, SATA, BSAS, FSAS, SSD, and MSATA disks use the SAS connection architecture.
- FC and ATA disks use the FC connection architecture with an arbitrated-loop topology (FC-AL).
- Array LUNs use the FC connection architecture, with either the point-to-point or switched topology.

SAS-connected disk shelves are connected to the controller on a daisy chain called a *stack*. FC-connected disk shelves are connected to the controller on a loop. You cannot combine different connection architectures in the same loop or stack.

Related concepts

How you can use disks with mixed speeds in the same aggregate on page 81

How disks can be combined for the SAS disk connection type

You can combine SAS disk shelves and SATA disk shelves within the same stack, although this configuration is not recommended.

Each SAS-connected disk shelf can contain only one type of disk (SAS or SATA). The only exception to this rule is if the shelf is being used for a Flash Pool aggregate. In that case, for some SSD sizes and shelf models, you can combine SSDs and HDDs in the same shelf. For more information, see the *Hardware Universe*.

How disks can be combined for the FC-AL disk connection type

You cannot combine disk shelves containing FC disks and disk shelves containing ATA disks in the same loop.

Usable and physical disk capacity by disk size

You cannot use the nominal size of a disk in your aggregate and storage system capacity calculations. You must use either the usable capacity or the physical capacity as calculated by Data ONTAP.

The following table lists the approximate physical and usable capacities for the disk sizes currently supported by Data ONTAP. The numbers shown are in Mebibytes (MiBs). This unit of measure is equivalent to 2 to the 20th power bytes. (MBs, in contrast, are 10 to the sixth power bytes.)

You can obtain the same information for disks installed in your storage system by using the `sysconfig -r` command (available through the nodeshell). The physical capacities listed in the table are approximations; actual physical disk capacities vary by disk manufacturer. See the technical documentation for your disks for the exact physical disk capacities.

Disk size as described by manufacturer	**Physical capacity (approximate)**	**Usable capacity**
100 GB SSD (X441A-R5)	95,396 MiB	95,146 MiB
100 GB SSD(X442A-R5)	84,796 MiB	84,574 MiB
200 GB SSD	190,782 MiB	190,532 MiB
300 GB SAS/FC	280,104 MiB	272,000 MiB
450 GB SAS/FC	420,156 MiB	418,000 MiB
500 GB SATA	423,946 MiB	423,111 MiB
600 GB SAS/FC	560,208 MiB	560,000 MiB
900 GB SAS	858,483 MiB	857,000 MiB
1 TB SATA	847,884 MiB	847,555 MiB
2 TB SATA	1,695,702 MiB	1,695,466 MiB
3 TB SATA	2,543,634 MiB	2,538,546 MiB
4 TB NL-SAS	3,815,447 MiB	3,807,816 MiB

Methods of calculating aggregate and system capacity

You use the physical and usable capacity of the disks you employ in your storage systems to ensure that your storage architecture conforms to the overall system capacity limits and the size limits of your aggregates.

To maintain compatibility across different brands of disks, Data ONTAP rounds down (*right-sizes*) the amount of space available for user data. In addition, the numerical base used to calculate capacity (base 2 or base 10) also impacts sizing information. For these reasons, it is important to use the correct size measurement, depending on the task you want to accomplish:

- For calculating overall system capacity, you use the physical capacity of the disk, and count every disk that is owned by the storage system.
- For calculating how many disks you can put into an aggregate before you exceed its maximum size, you use the right-sized, or usable capacity of all data disks in that aggregate.

  Parity and dparity disks are not counted against the maximum aggregate size.

Disk speeds supported by Data ONTAP

For hard disk drives, which use rotating media, speed is measured in revolutions per minute (RPM). Faster disks provide more disk input/output operations per second (IOPS) and faster response time.

It is best to use disks of the same speed in an aggregate.

Data ONTAP supports the following rotational speeds for disks:

- SAS disks (SAS-connected)
  - 10K RPM
  - 15K RPM
- SATA, BSAS, FSAS, and MSATA disks (SAS-connected)
  - 7.2K RPM
- FCAL disks (FC-AL connected)
  - 10K RPM
  - 15K RPM
- ATA disks (FC-AL connected)
  - 5.4K RPM
  - 7.2K RPM

Solid-state disks, or SSDs, are flash memory-based devices and therefore the concept of rotational speed does not apply to them. SSDs provide more IOPS and faster response times than rotating media.

For more information about supported disk speeds, see the *Hardware Universe* (formerly the *System Configuration Guide*) at *support.netapp.com/knowledge/docs/hardware/NetApp/syscfg/index.shtml*.

How disk checksum types affect aggregate and spare management

There are two checksum types available for disks used by Data ONTAP: BCS (block) and AZCS (zoned). Understanding how the checksum types differ and how they impact storage management enables you to manage your storage more effectively.

Both checksum types provide the same resiliency capabilities; BCS optimizes data access speed and capacity for disks that use 520 byte sectors. AZCS provides enhanced storage utilization and capacity for disks that use 512 byte sectors (usually SATA disks, which emphasize capacity).

Aggregates have a checksum type, which is determined by the checksum type of the disks that compose the aggregate. The following configuration rules apply to aggregates, disks, and checksums:

- Checksum types cannot be combined within RAID groups.
  This means that you must consider checksum type when you provide hot spare disks.
- When you add storage to an aggregate, if it has a different checksum type than the storage in the RAID group to which it would normally be added, Data ONTAP creates a new RAID group.
- An aggregate can have RAID groups of both checksum types.
  These aggregates have a checksum type of `mixed`.
- Disks of a different checksum type cannot be used to replace a failed disk.
- You cannot change the checksum type of a disk.

Checksum type by Data ONTAP disk type

You should know the Data ONTAP disk type and checksum type of all of the disks you manage, because these disk characteristics impact where and when the disks can be used.

The following table shows the checksum type by Data ONTAP disk type:

Data ONTAP disk type	Checksum type
SAS or FC-AL	BCS
SATA/BSAS/FSAS/ATA	BCS
SSD	BCS
MSATA	AZCS

Disk name formats

Each disk has a name that differentiates it from all other disks. Disk names have different formats, depending on the disk connection type (FC-AL or SAS) and how the disk is attached.

Each disk has a universal unique identifier (UUID) that differentiates it from all other disks in the cluster.

The names of unowned disks (broken or unassigned disks) display the alphabetically lowest node name in the cluster that can see that disk.

The following table shows the various formats for disk names, depending on how they are connected to the storage system.

Note: For internal disks, the slot number is zero, and the internal port number depends on the system model.

Disk connection	Disk name	Example
SAS, direct-attached	<node>:<slot><port>.<shelfID>.<bay>	The internal SAS-connected disk for node node1 in bay 9 for a FAS2040 is named node1:0c.0.9. The disk in shelf 2, bay 11, connected to onboard port 0a and owned by node1 is named node1:0a.2.11. The disk in shelf 6, bay 3, connected to an HBA in slot 1, port c, and owned by node1 is named node1:1c.6.3.

Disk connection	Disk name	Example
FC-AL, direct-attached	<node>:<slot><port>.<loopID>	The disk with loop ID 19 (bay 3 of shelf 1) connected to onboard port 0a and owned by node1 is named node1:0a.19. The disk with loop ID 34 connected to an HBA in slot 8, port c and owned by node1 is named node1:8c.34.
FC-AL, switch-attached	<node>:<switch_name>.<switch_port>.<loopID>	The disk with loop ID 51 connected to port 3 of switch SW7 owned by node1 is named node1:SW7.3.51.

Related concepts

Array LUN name format on page 54

Loop IDs for FC-AL connected disks

For disks connected using Fibre Channel-Arbitrated Loop (FC-AL or FC), the loop ID is an integer between 16 and 126. The loop ID identifies the disk within its loop, and is included in the disk name, which identifies the disk uniquely for the entire system.

The loop ID corresponds to the disk shelf number and the bay in which the disk is installed. The lowest loop ID is always in the far right bay of the first disk shelf. The next higher loop ID is in the next bay to the left, and so on. You can view the device map for your disk shelves with the `fcadmin device_map` command.

For more information about the loop ID map for your disk shelf, see the hardware guide for the disk shelf.

Understanding RAID disk types

Data ONTAP classifies disks as one of four types for RAID: data, hot spare, parity, or dParity. The RAID disk type is determined by how RAID is using a disk; it is different from the Data ONTAP disk type.

Data disk	Holds data stored on behalf of clients within RAID groups (and any data generated about the state of the storage system as a result of a malfunction).
Spare disk	Does not hold usable data, but is available to be added to a RAID group in an aggregate. Any functioning disk that is not assigned to an aggregate but is assigned to a system functions as a hot spare disk.
Parity disk	Stores row parity information that is used for data reconstruction when a single disk drive fails within the RAID group.
dParity disk	Stores diagonal parity information that is used for data reconstruction when two disk drives fail within the RAID group, if RAID-DP is enabled.

Related concepts

How Data ONTAP reports disk types on page 8

How disk sanitization works

Disk sanitization is the process of physically obliterating data by overwriting disks with specified byte patterns or random data so that recovery of the original data becomes impossible. You use the sanitization process to ensure that no one can recover the data on the disks. This functionality is available through the nodeshell.

Related tasks

Using disk sanitization to remove data from disks on page 35

Disk sanitization process

Understanding the basics of the disk sanitization process helps you understand what to anticipate during the sanitization process and after it is complete.

The disk sanitization process uses three successive default or user-specified byte overwrite patterns for up to seven cycles per operation. The random overwrite pattern is repeated for each cycle.

Depending on the disk capacity, the patterns, and the number of cycles, the process can take several hours. Sanitization runs in the background. You can start, stop, and display the status of the sanitization process.

The sanitization process contains two phases:

1. Formatting phase
   - For capacity HDDs (SATA, BSAS, FSAS, MSATA, or ATA) the formatting phase is skipped.
   - For performance HDDs (SAS or FC), the formatting phase consists of a SCSI format operation.
   - For SSDs, the formatting phase consists of a SCSI sanitize operation.
2. Pattern overwrite phase
   The specified overwrite patterns are repeated for the specified number of cycles.

When the sanitization process is complete, the specified disks are in a sanitized state. They are not returned to spare status automatically. You must return the sanitized disks to the spare pool before the newly sanitized disks are available to be added to another aggregate.

When disk sanitization cannot be performed

Disk sanitization is not supported for all disk types. In addition, there are times when disk sanitization cannot be performed.

You should be aware of the following facts about the disk sanitization process:

- It is not supported on all SSD part numbers.
  For information about which SSD part numbers support disk sanitization, see the *Hardware Universe* (formerly the *System Configuration Guide*) at *support.netapp.com/knowledge/docs/hardware/NetApp/syscfg/index.shtml*.
- It is not supported in takeover mode for systems in an HA pair.
- It cannot be performed on disks that were failed due to readability or writability problems.
- It does not perform its formatting phase on ATA drives.
- If you are using the random pattern, it cannot be performed on more than 100 disks at one time.
- It is not supported on array LUNs.
- If you sanitize both SES disks in the same ESH shelf at the same time, you see errors on the console about access to that shelf, and shelf warnings are not reported for the duration of the sanitization.
  However, data access to that shelf is not interrupted.

What happens if disk sanitization is interrupted

Disk sanitization is a long-running operation. If disk sanitization is interrupted by user intervention or an unexpected event such as a power outage, Data ONTAP takes action to return the disks that were being sanitized to a known state, but you must also take action before the sanitization process can finish.

If the sanitization process is interrupted by power failure, system panic, or manual intervention, the sanitization process must be repeated from the beginning. The disk is not designated as sanitized.

If the formatting phase of disk sanitization is interrupted, Data ONTAP must recover any disks that were corrupted by the interruption. After a system reboot and once every hour, Data ONTAP checks

for any sanitization target disk that did not complete the formatting phase of its sanitization. If any such disks are found, Data ONTAP recovers them. The recovery method depends on the type of the disk. After a disk is recovered, you can rerun the sanitization process on that disk; for HDDs, you can use the `-s` option to specify that the formatting phase is not repeated again.

Tips for creating and backing up aggregates containing data to be sanitized

If you are creating or backing up aggregates to contain data that might need to be sanitized, following some simple guidelines will reduce the time it takes to sanitize your data.

- Make sure your aggregates containing sensitive data are not larger than they need to be.
  If they are larger than needed, sanitization requires more time, disk space, and bandwidth.
- When you back up aggregates containing sensitive data, avoid backing them up to aggregates that also contain large amounts of nonsensitive data.
  This reduces the resources required to move nonsensitive data before sanitizing sensitive data.

How Data ONTAP monitors disk performance and health

Data ONTAP continually monitors disks to assess their performance and health. When Data ONTAP encounters certain errors or behaviors from a disk, it takes the disk offline temporarily or takes the disk out of service to run further tests.

When Data ONTAP takes disks offline temporarily

Data ONTAP temporarily stops I/O activity to a disk and takes a disk offline when Data ONTAP is updating disk firmware in background mode or when disks become non-responsive. While the disk is offline, Data ONTAP performs a quick check on it to reduce the likelihood of forced disk failures.

While the disk is offline, Data ONTAP reads from other disks within the RAID group while writes are logged. When the offline disk is ready to come back online, Data ONTAP re-synchronizes the RAID group and brings the disk online. This process generally takes a few minutes and incurs a negligible performance impact.

Note: The disk offline feature is supported only for spares and data disks within RAID-DP aggregates. A disk can be taken offline only if its containing RAID group is in a normal state and the plex or aggregate is not offline.

How Data ONTAP reduces disk failures using Rapid RAID Recovery

When Data ONTAP determines that a disk has exceeded its error thresholds, Data ONTAP can perform Rapid RAID Recovery by removing the disk from its RAID group for testing and, if necessary, failing the disk. Spotting disk errors quickly helps prevent multiple disk failures and allows problem disks to be replaced.

By performing the Rapid RAID Recovery process on a suspect disk, Data ONTAP avoids three problems that occur during sudden disk failure and the subsequent RAID reconstruction process:

- Rebuild time
- Performance degradation
- Potential data loss due to additional disk failure during reconstruction

During Rapid RAID Recovery, Data ONTAP performs the following tasks:

1. Places the suspect disk in pre-fail mode.
2. Selects a hot spare replacement disk.

   Note: If no appropriate hot spare is available, the suspect disk remains in pre-fail mode and data continues to be served. However, a suspect disk performs less efficiently. Impact on performance ranges from negligible to worse than degraded mode. For this reason, make sure hot spares are always available.

3. Copies the suspect disk's contents to the spare disk on the storage system before an actual failure occurs.
4. After the copy is complete, attempts to put the suspect disk into the maintenance center, or else fails the disk.

Note: Tasks 2 through 4 can occur only when the RAID group is in normal (not degraded) mode.

If the suspect disk fails on its own before copying to a hot spare disk is complete, Data ONTAP starts the normal RAID reconstruction process.

A message is sent to the log file when the Rapid RAID Recovery process is started and when it is complete. The messages are tagged "raid.rg.diskcopy.start:notice" and "raid.rg.diskcopy.done:notice".

Related concepts

About degraded mode on page 68
When Data ONTAP can put a disk into the maintenance center on page 20
How Data ONTAP works with hot spare disks on page 66

How the maintenance center helps prevent drive errors

Data ONTAP provides a mechanism to test drives called the maintenance center. Sometimes Data ONTAP puts drives into the maintenance center automatically; you can also put a suspect drive into the maintenance center manually. Knowing how the maintenance center works helps you manage your storage effectively.

When a disk is in the maintenance center, it is subjected to a number of tests. If the disk passes all of the tests, it is redesignated as a spare. Otherwise, Data ONTAP fails the disk.

The maintenance center is controlled by the `disk.maint_center.enable` option. It is on by default.

Data ONTAP puts disks into the maintenance center only if there are two or more spares available for that disk.

You can control the number of times a disk is allowed to go to the maintenance center by using the `disk.maint_center.allowed_entries` option. The default value for this option is `1`, which means that if the disk is ever sent back to the maintenance center, it is automatically failed.

You can also put a disk into the maintenance center manually by using the `disk maint start` command. If the target disk is in use, it does not enter the maintenance center until its contents have been copied to another disk (unless you include the `-i` option).

Data ONTAP informs you of these activities by sending messages to the following destinations:

- The console
- A log file at `/etc/log/maintenance.log`

When Data ONTAP puts a disk into the maintenance center and that disk is housed in a storage shelf that supports automatic power cycling, power to that disk might be turned off for a short period of time. If the disk returns to a ready state after the power cycle, the maintenance center tests the disk. Otherwise, the maintenance center fails the disk immediately.

You can see the power-cycle status for ESH4 storage shelves by using the `environment shelf_power_status` command.

You can access the options and commands to control the maintenance center by using the nodeshell. For more information about the nodeshell, see the man page for the `system node run` command.

For information about best practices for working with the maintenance center, see *Technical Report 3437: Storage Best Practices and Resiliency Guide*.

Related information

TR 3437: Storage Best Practices and Resiliency Guide

When Data ONTAP can put a disk into the maintenance center

When Data ONTAP detects certain disk errors, it tries to put the disk into the maintenance center for testing. Certain requirements must be met for the disk to be put into the maintenance center.

If a disk experiences more errors than are allowed for that disk type, Data ONTAP takes one of the following actions:

- If the `disk.maint_center.spares_check` option is set to `on` (the default) and two or more spares are available (four for multi-disk carriers), Data ONTAP takes the disk out of service and assigns it to the maintenance center for data management operations and further testing.
- If the `disk.maint_center.spares_check` option is set to `on` and fewer than two spares are available (four for multi-disk carriers), Data ONTAP does not assign the disk to the maintenance center.

  It fails the disk and designates the disk as a broken disk.
- If the `disk.maint_center.spares_check` option is set to `off`, Data ONTAP assigns the disk to the maintenance center without checking the number of available spares.

Note: The `disk.maint_center.spares_check` option has no effect on putting disks into the maintenance center from the command-line interface.

Data ONTAP does not put SSDs into the maintenance center.

How Data ONTAP uses continuous media scrubbing to prevent media errors

The purpose of the continuous media scrub is to detect and correct media errors to minimize the chance of storage system disruption due to a media error while a storage system is in degraded or reconstruction mode.

By default, Data ONTAP runs continuous background media scrubbing for media errors on all storage system disks. If a media error is found, Data ONTAP uses RAID to reconstruct the data and repairs the error.

Media scrubbing is a continuous background process. Therefore, you might observe disk LEDs blinking on an apparently idle storage system. You might also observe some CPU activity even when no user workload is present.

How continuous media scrubbing impacts system performance

Because continuous media scrubbing searches only for media errors, its impact on system performance is negligible. In addition, the media scrub attempts to exploit idle disk bandwidth and free CPU cycles to make faster progress. However, any client workload results in aggressive throttling of the media scrub resource.

If needed, you can further decrease the CPU resources consumed by a continuous media scrub under a heavy client workload by increasing the maximum time allowed for a media scrub cycle to complete. You can do this by using the `raid.media_scrub.rate` option.

Why continuous media scrubbing should not replace scheduled RAID-level disk scrubs

Because the continuous media scrub process scrubs only media errors, you should continue to run the storage system's scheduled complete RAID-level scrub operation. The RAID-level scrub finds and corrects parity and checksum errors as well as media errors.

Related concepts

How you schedule automatic RAID-level scrubs on page 70

Increasing storage availability by using ACP

ACP, or Alternate Control Path, is a protocol that enables Data ONTAP to manage and control a SAS disk shelf storage subsystem. It uses a separate network (alternate path) from the data path, so management communication is not dependent on the data path being intact and available.

You do not need to actively manage the SAS disk shelf storage subsystem. Data ONTAP automatically monitors and manages the subsystem without operator intervention. However, you must provide the required physical connectivity and configuration parameters to enable the ACP functionality.

Note: You can install SAS disk shelves without configuring ACP. However, for maximum storage availability and stability, you should always have ACP configured and enabled.

After you enable ACP, you can use the `storage show acp` and `acpadmin list_all` commands, available through the nodeshell, to display information about your ACP subsystem.

Because ACP communication is on a separate network, it does not affect data access in any way.

Enabling ACP

ACP can increase your storage availability when you use SAS disk shelves. If your storage system model has a dedicated port for ACP, then ACP is enabled by default, and you do not need to explicitly enable ACP.

Before you begin

- Is the ACP subnet cabled on an isolated network, with no switches or hubs?
  For more information, see the *Installation and Service Guide* for your disk shelf.
- Have you identified a port that is not in use by any other subsystem?
- If you are configuring ACP for disk shelves attached to an HA pair, have you recorded the domain name and network mask to ensure that they are the same for both nodes?

About this task

The ACP subnet is a private Ethernet network that enables the ACP processor in the SAS module to communicate both with Data ONTAP and with the SAS IOMs in the disk shelves.

The ACP subnet is separate from the I/O data path that connects the disk shelves to the HBA on the storage controller. When you configure ACP on one of the system's network interfaces, you must supply a private domain name that conforms to the standard for private internet addresses (RFC1918). You can use the system default domain or another network name (that is, an IP address ending in 0) that conforms to the standard.

Some of the commands used in this procedure are available only through the nodeshell.

Steps

1. If your system does not have a dedicated port for ACP (e0p), ensure that the port you are assigning to ACP has no LIFs homed or hosted on it by completing the following steps:
   a) Determine if a LIF is currently hosted on the target port:

      network interface show -curr-node *node* -curr-port *port*

   b) If any LIFs are hosted on the target port, migrate them away:

      network interface migrate -vserver *Vserver* -lif *lif* -dest-node *dest_node* -dest-port *dest_port*

   c) Determine if a LIF is currently homed on the target port:

      network interface show -home-node *node* -home-port *port*

   d) If any LIFs are homed on the target port, modify their home port to another port:

      network interface modify -vserver *Vserver* -lif *lif* -home-port *new_home_port*

   e) Determine if any failover groups are configured to use the target port:

      network interface failover-group show -node *node* -port *port*

   f) If the port is used by a failover group, delete the port from the failover group:

      network interface failover-group delete -failover-group *failover_group* -node *node* -port *port*

   g) Determine if any failover rules are configured to use the target port:

      network interface failover show

   h) If the port is used in a failover rule, delete the port from the failover rule:

      network interface failover delete -server *Vserver* -lif *lif* -priority *priority_number*

2. At the Data ONTAP command line, enter the following command:

   acpadmin configure

   If you have not previously configured the networking information for ACP, you are prompted for that information. When you select a domain name and network mask for the ACP interface, Data ONTAP automatically assigns IP addresses for the ACP interface on the storage controller and both I/O modules on each disk shelf on the ACP subnet.

3. If you configured ACP to use a non-dedicated port, complete the following steps:
   a) Reboot the node.
   b) Enter the advanced privilege mode:

      set advanced

   c) Delete the port you configured for use by ACP from the resource database:

      network port delete -node *node* -port *port*

   d) Return to administrative privilege mode:

      set admin

4. You can verify your ACP connectivity by entering the following command:

   `storage show acp`

   The ACP Connectivity Status should show "Full Connectivity".

Example

For example, if you select e0P as the interface for ACP traffic, 192.168.0.0 as the ACP domain, and 255.255.252.0 as the network mask for the ACP subnet, the `storage show acp` command output looks similar to the following example:

```
my-sys-1> storage show acp

 Alternate Control Path:  enabled
 Ethernet Interface:      e0P
 ACP Status:              Active
 ACP IP address:          192.168.2.61
 ACP domain:              192.168.0.0
 ACP netmask:             255.255.252.0
 ACP Connectivity Status: Full Connectivity

Shelf Module      Reset Cnt    IP address      FW Version    Module
Type   Status
------------------------------------------------------------------------
-------------
7a.001.A          002          192.168.0.145    01.05
IOM6          active
7a.001.B          003          192.168.0.146    01.05
IOM6          active
7c.002.A          000          192.168.0.206    01.05
IOM6          active
7c.002.B          001          192.168.0.204    01.05
IOM6          active
```

How you use SSDs to increase storage performance

Solid-state drives (SSDs) are flash media-based storage devices that provide better overall performance than hard disk drives (HDDs), which are mechanical devices using rotating media. You should understand how Data ONTAP manages SSDs and the capability differences between SSDs and HDDs.

Depending on your storage system model, you can use SSDs in two ways:

- You can create Flash Pool aggregates—aggregates composed mostly of HDDs, but with some SSDs that function as a high-performance cache for your working data set.
- You can create aggregates composed entirely of SSDs, where the SSDs function as the persistent storage for all data in the aggregate.

You manage Flash Pool aggregates and aggregates composed entirely of SSDs the same way you manage aggregates composed entirely of HDDs. However, there are some differences in the way you manage SSDs from the way you manage disks. In addition, some Data ONTAP capabilities are not available on SSDs and Flash Pool aggregates.

SSDs are not supported on all storage system models. For information about which models support SSDs, see the *Hardware Universe* (formerly the *System Configuration Guide*) at *support.netapp.com/knowledge/docs/hardware/NetApp/syscfg/index.shtml*.

Related concepts

How Flash Pool aggregates work on page 78
When you cannot use aggregates composed of SSDs on page 81

How Data ONTAP manages SSD wear life

Solid-state disks (SSDs) have a different end-of-life behavior than rotating media (hard disk drives, or HDDs). Data ONTAP monitors and manages SSDs to maximize storage performance and availability.

In the absence of a mechanical failure, rotating media can serve data almost indefinitely. This is not true for SSDs, which can accept only a finite (though very large) number of write operations. SSDs provide a set of internal spare capacity, called *spare blocks*, that can be used to replace blocks that have reached their write operation limit. After all of the spare blocks have been used, the next block that reaches its limit causes the disk to fail.

Because a drive failure is an undesirable occurrence, Data ONTAP replaces SSDs before they reach their limit. When a predetermined percentage of the spare blocks have been used (approximately 90%), Data ONTAP performs the following actions:

1. Sends an AutoSupport message.
2. If a spare SSD is available, starts a disk copy to that spare.
3. If no spare is available, starts a periodic check for a spare so that the disk copy can be started when a spare becomes available.
4. When the disk copy finishes, fails the disk.

Note: You do not need to replace SSDs before they are failed by Data ONTAP. However, when you use SSDs in your storage system (as for all disk types), it is important to ensure that you have sufficient hot spares available at all times.

Capability differences between SSDs and HDDs

Usually, you manage SSDs the same as HDDs, including firmware updates, scrubs, and zeroing. However, some Data ONTAP capabilities do not make sense for SSDs, and SSDs are not supported on all hardware models.

SSDs cannot be combined with HDDs within the same RAID group. When you replace an SSD in an aggregate, you must replace it with another SSD. Similarly, when you physically replace an SSD within a shelf, you must replace it with another SSD.

The following capabilities of Data ONTAP are not available for SSDs:

- Disk sanitization is not supported for all SSD part numbers.
  For information about which SSD part numbers support sanitization, see the *Hardware Universe*.
- The maintenance center
- FlexShare

SSDs are not supported on all storage system models. For information about which models support SSDs, see the *Hardware Universe* (formerly the *System Configuration Guide*) at *support.netapp.com/knowledge/docs/hardware/NetApp/syscfg/index.shtml*.

Guidelines and requirements for using multi-disk carrier storage shelves

Data ONTAP automatically handles most of the extra steps required to manage disks in multi-disk carriers. However, there are some extra management and configuration requirements that you must understand before incorporating multi-disk carrier disk shelves in your storage architecture.

When using storage from multi-disk carrier disk shelves such as the DS4486, you must familiarize yourself with the guidelines and requirements governing the following topics:

- The process that Data ONTAP uses to avoid impacting any RAID groups when a multi-disk carrier needs to be removed
- When it is safe to remove a multi-disk carrier after a disk failure
- The minimum required number of spares for multi-disk carrier disks
- Multi-disk carrier disk shelf configuration
- Aggregate configuration requirements when using multi-disk carrier disk shelves
- Guidelines and best practices for using disks from a multi-disk carrier disk shelf in an aggregate

How Data ONTAP avoids RAID impact when a multi-disk carrier must be removed

Data ONTAP takes extra steps to ensure that both disks in a carrier can be replaced without impacting any RAID group. Understanding this process helps you know what to expect when a disk from a multi-disk carrier disk shelf fails.

A multi-disk carrier disk shelf, such as the DS4486, has double the disk density of other SAS disk shelves. It accomplishes this by housing two disks per disk carrier. When two disks share the same disk carrier, they must be removed and inserted together. This means that when one of the disks in a carrier needs to be replaced, the other disk in the carrier must also be replaced, even if it was not experiencing any issues.

Removing two data or parity disks from an aggregate at the same time is undesirable, because it could leave two RAID groups degraded, or one RAID group double-degraded. To avoid this situation, Data ONTAP initiates a disk evacuation operation for the carrier mate of the failed disk, as well as the usual reconstruction to replace the failed disk. The disk evacuation operation copies the contents of the carrier mate to a disk in a different carrier so the data on that disk remains available when you remove the carrier. During the evacuation operation, the status for the disk being evacuated shows as `evacuating`.

In addition, Data ONTAP tries to create an optimal layout that avoids having two carrier mates in the same RAID group. Depending on how the other disks are laid out, achieving the optimal layout can require as many as three consecutive disk evacuation operations. Depending on the size of the disks and the storage system load, each disk evacuation operation could take several hours, so the entire swapping process could take an entire day or more.

If insufficient spares are available to support the swapping operation, Data ONTAP issues a warning and waits to perform the swap until you provide enough spares.

How to determine when it is safe to remove a multi-disk carrier

Removing a multi-disk carrier before it is safe to do so can result in one or more RAID groups becoming degraded, or possibly even a storage disruption. Data ONTAP provides several indications of when it is safe to remove a multi-disk carrier.

When a multi-disk carrier needs to be replaced, the following events must have occurred before you can remove the carrier safely:

- An AutoSupport message must have been logged indicating that the carrier is ready to be removed.
- An EMS message must have been logged indicating that the carrier is ready to be removed.
- Both disks in the carrier must be displayed in the list of broken disks.

  You can see the list of broken disks by using the `storage disk show -broken` command.

  The disk that was evacuated to allow the carrier to be removed shows the outage reason of `evacuated`.
- The amber LED on the carrier must be lit continuously.

- The green LED on the carrier must show no activity.

  Attention: You cannot reuse the carrier mate of a failed disk. When you remove a multi-disk carrier that contains a failed disk, you must replace and return the entire carrier.

Spare requirements for multi-disk carrier disks

Maintaining the proper number of spares for disks in multi-disk carriers is critical for optimizing storage redundancy and minimizing the amount of time Data ONTAP must spend copying disks to achieve an optimal disk layout.

You must maintain a minimum of two hot spares for multi-disk carrier disks at all times. To support the use of the Maintenance Center, and to avoid issues caused by multiple concurrent disk failures, you should maintain at least four hot spares for steady state operation, and replace failed disks promptly.

If two disks fail at the same time with only two available hot spares, Data ONTAP might not be able to swap the contents of both the failed disk and its carrier mate to the spare disks. This scenario is called a *stalemate*. If this happens, you are notified through EMS messages and AutoSupport messages. When the replacement carriers become available, you must follow the instructions provided by the EMS messages or contact technical support to recover from the stalemate.

Shelf configuration requirements for multi-disk carrier disk shelves

You can combine multi-disk carrier disk shelves with single-disk carrier disk shelves (standard disk shelves) on the same storage system. However, you cannot combine the two disk shelf types in the same stack.

Aggregate requirements for disks in multi-disk carrier storage shelves

Aggregates composed of disks in multi-disk carrier disk shelves must conform to some configuration requirements.

The following configuration requirements apply to aggregates composed of disks in multi-disk carrier disk shelves:

- The RAID type must be RAID-DP.
- The format must be 64-bit.
- All HDDs in the aggregate must be the same Data ONTAP disk type.
  The aggregate can be a Flash Pool aggregate.

Related concepts

How Flash Pool aggregates work on page 78

Considerations for using disks from a multi-disk carrier storage shelf in an aggregate

Observing the requirements and best practices for using disks from a multi-disk carrier disk shelf in an aggregate enables you to maximize storage redundancy and minimize the impact of disk failures.

Disks in multi-disk carriers always have the Data ONTAP disk type of MSATA. MSATA disks cannot be mixed with HDDs from a single-carrier disk shelf in the same aggregate.

The following disk layout requirements apply when you are creating or increasing the size of an aggregate composed of MSATA disks:

- Data ONTAP prevents you from putting two disks in the same carrier into the same RAID group.
- Do not put two disks in the same carrier into different pools, even if the shelf is supplying disks to both pools.
- Do not assign disks in the same carrier to different nodes.
- For the best layout, do not name specific disks; allow Data ONTAP to select the disks to be used or added.

  If the operation cannot result in an optimal layout due to the placement of the disks and available spares, Data ONTAP automatically swaps disk contents until an optimal layout is achieved. If there are not enough available spares to support the swaps, Data ONTAP issues a warning and waits to perform the disk swaps until you provide the necessary number of hot spares. If you name disks and an optimal layout cannot be achieved, you must explicitly force the operation; otherwise, the operation fails.

Aggregate creation example

To create an aggregate using MSATA disks, you can specify the disk type and size but leave the disk selection and layout to Data ONTAP by using a command like this:

```
storage aggregate create -aggregate c1n1_aggr1 -node node1 -
disktype MSATA -diskcount 14
```

Adding disks to a storage system

You add disks to a storage system to increase the number of hot spares, to add space to an aggregate, or to replace disks.

Before you begin

You must have confirmed that your storage system supports the type of disk you want to add. For information about supported disk drives, see the *Hardware Universe* (formerly the *System Configuration Guide*) at *support.netapp.com/knowledge/docs/hardware/NetApp/syscfg/index.shtml*.

Steps

1. Check the NetApp Support Site for newer disk and shelf firmware and Disk Qualification Package files. If your system does not have the latest versions, update them before installing the new disk.
2. Install one or more disks according to the hardware guide for your disk shelf or the hardware and service guide for your storage system.

   The new disks are not recognized until they are assigned to a system. You can assign the new disks manually, or you can wait for Data ONTAP to automatically assign the new disks if your system follows the rules for disk autoassignment.
3. After the new disks have all been recognized, verify their addition and their ownership information by entering the following command:

   `storage disk show -spare`

   You should see the new disks, owned by the correct system, listed as hot spare disks.
4. You can zero the newly added disks now, if needed, by entering the following command:

   `storage disk zerospares`

   Note: Disks that have been used previously in a Data ONTAP aggregate must be zeroed before they can be added to another aggregate. Zeroing the disks now can prevent delays in case you need to quickly increase the size of an aggregate. The disk zeroing command runs in the background and can take hours to complete, depending on the size of the non-zeroed disks in the system.

Result

The new disks are ready to be added to an aggregate, used to replace an existing disk, or placed onto the list of hot spares.

Related concepts

How automatic ownership assignment works for disks on page 41
Guidelines for assigning ownership for disks on page 41

Related information

Disk Qualification Package Instructions: support.netapp.com/NOW/download/tools/diskqual/
Disk Drive & Firmware Matrix: support.netapp.com/NOW/download/tools/diskfw/

Replacing disks that are currently being used in an aggregate

You can use the `storage disk replace` command to replace disks that are part of an aggregate without disrupting data service. You do this to swap out mismatched disks from a RAID group. Keeping your RAID groups homogeneous helps optimize storage system performance.

Before you begin

You should already have an appropriate hot spare disk of the correct type, size, speed, and checksum type installed in your storage system. This spare disk must be assigned to the same system and pool as the disk it will replace. For multi-disk carrier disks, you should have at least two hot spare disks available, to enable Data ONTAP to provide an optimal disk layout.

About this task

If you need to replace a disk—for example a mismatched data disk in a RAID group—you can replace the disk. This operation uses Rapid RAID Recovery to copy data from the specified old disk in a RAID group to the specified spare disk in the storage system. At the end of the process, the spare disk replaces the old disk as the new data disk, and the old disk becomes a spare disk in the storage system.

Note: If you replace a smaller disk with a larger disk, the capacity of the larger disk is downsized to match that of the smaller disk; the usable capacity of the aggregate is not increased.

Step

1. Enter the following command:

   `storage disk replace -disk` ***`old_disk_name`*** **`-replacement`** ***`new_spare_disk_name`*** **`-action start`**

   If you need to stop the disk replace operation, you can use the `-action stop` option. If you halt a disk replace operation, the target spare disk needs to be zeroed before it can be used in another aggregate.

Result

The old disk is converted to a spare disk, and the new disk is now used in the aggregate.

Converting a data disk to a hot spare

Data disks can be converted to hot spares by destroying the aggregate that contains them.

Before you begin

The aggregate to be destroyed cannot contain volumes.

About this task

Converting a data disk to a hot spare does not change the ownership information for that disk. You must remove ownership information from a disk before moving it to another storage system.

Step

1. Destroy the aggregate that contains the disk by entering the following command:
   storage aggregate delete -aggregate *aggr_name*
   All disks in use by that aggregate are converted to hot spare disks.

Removing disks from a storage system

How you remove a disk from your storage system depends how the disk is being used. By using the correct procedure, you can prevent unwanted AutoSupport notifications from being generated and ensure that the disk functions correctly if it is reused in another storage system.

You cannot reduce the number of disks in an aggregate by removing data disks. The only way to reduce the number of data disks in an aggregate is to copy the data and transfer it to a new aggregate that has fewer data disks.

Removing a failed disk

A disk that is completely failed is no longer counted by Data ONTAP as a usable disk, and you can immediately disconnect the disk from the disk shelf. However, you should leave a partially failed disk connected long enough for the Rapid RAID Recovery process to complete.

About this task

If you are removing a disk because it has failed or because it is producing excessive error messages, you should not use the disk again in this or any other storage system.

Steps

1. Find the disk ID of the failed disk by entering the following command:
   storage disk show -broken

If the disk does not appear in the list of failed disks, it might be partially failed, with a Rapid RAID Recovery in process. In this case, you should wait until the disk is present in the list of failed disks (which means that the Rapid RAID Recovery process is complete) before removing the disk.

2. Determine the physical location of the disk you want to remove by entering the following command:

   storage disk set-led -disk *disk_name* 2

   The fault LED on the face of the disk is lit for 2 minutes.

3. Remove the disk from the disk shelf, following the instructions in the hardware guide for your disk shelf model.

Removing a hot spare disk

Removing a hot spare disk requires you to remove ownership information from the disk. This prevents the disk from causing problems when it is inserted into another storage system, and notifies Data ONTAP that you are removing the disk to avoid unwanted AutoSupport messages.

About this task

Removing a hot spare disk does not make the contents of that disk inaccessible. If you need absolute assurance that the data contained by this disk is irretrievable, you should sanitize the disk instead of completing this procedure.

Steps

1. Find the disk name of the hot spare disk you want to remove:

   storage disk show -spare

2. Determine the physical location of the disk you want to remove:

   storage disk set-led -disk *disk_name*

   The fault LED on the face of the disk is lit.

3. If disk ownership automatic assignment is on, turn it off:

   storage disk option modify -node *node_name* -autoassign off

4. Repeat the previous step for the node's HA partner.

5. Remove the software ownership information from the disk:

   storage disk removeowner *disk_name*

6. Remove the disk from the disk shelf, following the instructions in the hardware guide for your disk shelf model.

7. If you turned off disk ownership automatic assignment previously, turn it on now:

   storage disk option modify -node *node_name* -autoassign on

Related concepts

How to determine when it is safe to remove a multi-disk carrier on page 27

Related tasks

Using disk sanitization to remove data from disks on page 35

Removing a data disk

The only time that you should remove a data disk from a storage system is if the disk is not functioning correctly. If you want to remove a data disk so that it can be used in another system, you must convert it to a hot spare disk first.

About this task

You can cause Data ONTAP to fail the disk immediately or allow a disk copy to finish before the disk is failed. If you do not fail the disk immediately, you must wait for the disk copy to finish before physically removing the disk. This operation might take several hours, depending on the size of the disk and the load on the storage system.

Do not immediately fail a disk unless it is causing immediate performance or availability issues for your storage system. Depending on your storage system configuration, additional disk failures could result in data loss.

Steps

1. Determine the name of the disk you want to remove.
2. Determine the physical location of the disk you want to remove by entering the following command:

   storage disk set-led -disk *disk_name* 2

   The red LED on the face of the disk is lit for 2 minutes.
3. Take the appropriate action based on whether you need to fail the disk immediately or not.

If you...	Then...
Can wait for the copy operation to finish (recommended)	Enter the following command to pre-fail the disk: **storage disk fail *disk_name*** Data ONTAP pre-fails the specified disk and attempts to create a replacement disk by copying the contents of the pre-failed disk to a spare disk. If the copy operation is successful, then Data ONTAP fails the disk and the new replacement disk takes its place. If the copy operation fails, the pre-failed disk fails and the storage system operates in degraded mode until the RAID system reconstructs a replacement disk.

If you...	Then...
Need to remove the disk immediately	Enter the following command to cause the disk to fail immediately: **`storage disk fail -disk disk_name -immediate`** The disk fails and the storage system operates in degraded mode until the RAID system reconstructs a replacement disk.

4. Remove the failed disk from the disk shelf, following the instructions in the hardware guide for your disk shelf model.

Related concepts

About degraded mode on page 68
How to determine when it is safe to remove a multi-disk carrier on page 27

Using disk sanitization to remove data from disks

Disk sanitization enables you to remove data from a disk or set of disks so that the data can never be recovered.

Before you begin

The disks that you want to sanitize must be spare disks; they must be owned but not used in an aggregate.

About this task

When disk sanitization is enabled on a storage system, it cannot be disabled again.

Steps

1. Enter the nodeshell for the system that owns the disks you want to sanitize by entering the following command:

   `system node run -node node_name`

2. Enable the disk sanitization option to be modified by entering the following command:

   `options nodescope.reenabledoptions licensed_feature.disk_sanitization.enable`

3. Enable disk sanitization by entering the following command:

   `options licensed_feature.disk_sanitization.enable on`

   You are asked to confirm the command, because it is irreversible.

4. Sanitize the specified disks by entering the following command:

disk sanitize start [-p *pattern1*|-r [-p *pattern2*|-r [-p *pattern3*|-r]]] [-c *cycle_count*] *disk_list*

Attention: Do not turn off the storage system, disrupt the storage connectivity, or remove target disks while sanitizing. If sanitizing is interrupted during the formatting phase, the formatting phase must be restarted and allowed to finish before the disks are sanitized and ready to be returned to the spare pool.

If you need to abort the sanitization process, you can do so by using the `disk sanitize abort` command. If the specified disks are undergoing the formatting phase of sanitization, the abort does not occur until the phase is complete. At that time, Data ONTAP displays a message telling you that the sanitization process was stopped.

`-p` *`pattern1`* `-p` *`pattern2`* `-p` *`pattern3`* specifies a cycle of one to three user-defined hex byte overwrite patterns that can be applied in succession to the disks being sanitized. The default pattern is three passes, using 0x55 for the first pass, 0xaa for the second pass, and 0x3c for the third pass.

`-r` replaces a patterned overwrite with a random overwrite for any or all of the passes.

`-c` *`cycle_count`* specifies the number of times that the specified overwrite patterns are applied. The default value is one cycle. The maximum value is seven cycles.

`disk_list` specifies a space-separated list of the IDs of the spare disks to be sanitized.

5. If you want to check the status of the disk sanitization process, enter the following command:

   disk sanitize status [*disk_list*]

6. After the sanitization process is complete, return the disks to spare status by entering the following command for each disk:

   disk sanitize release *disk_name*

7. Return to the clustered Data ONTAP CLI by entering the following command:

   exit

8. Determine whether all of the disks were returned to spare status by entering the following command:

   storage disk show -spare

If...	Then...
All of the sanitized disks are listed as spares	You are done. The disks are sanitized and in spare status.

If...	Then...

Some of the sanitized disks are not listed as spares	Complete the following steps:
	a. Enter advanced privilege mode:
	`set -privilege advanced`
	b. Assign the unassigned sanitized disks to the appropriate node by entering the following command for each disk:
	`storage disk assign -disk disk_name -owner system_name`
	c. Return the disks to spare status by entering the following command for each disk:
	`storage disk unfail -disk disk_name -s`
	d. Return to administrative mode:
	`set -privilege admin`

Result

The specified disks are sanitized and designated as hot spares.

Related concepts

How disk sanitization works on page 16

Commands for managing disks

Data ONTAP provides the `storage disk` command for managing disks.

If you want to...	Use this command...
Display a list of failed disks	`storage disk show -broken`
Display a list of spare disks	`storage disk show -spare`
Display a list of disks in the maintenance center	`storage disk show -maintenance`
Display the RAID type of each disk in an aggregate	`storage disk show -aggregate aggr_name -raid`
Display the RAID type, current usage, aggregate and RAID group for disks	`storage disk show -raid`
Display the checksum type for a specific disk	`storage disk show -fields checksum-compatibility`

If you want to...	Use this command...
Display the checksum type for all spare disks	`storage disk show -fields checksum-compatibility -container-type spare`
Display disk connectivity and placement information	`storage disk show -fields disk,primary-port,secondary-name,secondary-port,shelf,bay`
Zero all non-zeroed disks	`storage disk zerospares`

See the man page for each command for more information.

Commands for displaying disk space information

You can see how disk space is being used in your aggregates and volumes and their Snapshot copies.

To display information about...	Use this command...
Aggregates, including details about used and available space percentages, Snapshot reserve size, and other space usage information	`storage aggregate show -aggregate`
How disks and RAID groups are used in an aggregate and RAID status	`system node run -node <nodename> aggr status -r`
The amount of disk space that would be reclaimed if you deleted a specific Snapshot copy	`volume snapshot compute-reclaimable` (advanced)
The amount of disk space used by a volume	`volume show -fields size,used,available,percent-used`

For detailed information about these commands, see the appropriate man page.

Managing ownership for disks

Disk ownership determines which node owns a disk. Data ONTAP stores ownership information directly on the disk.

Reasons to assign ownership of disks and array LUNs

Storage system ownership must be assigned for disks and array LUNs before they become an effective part of your system. You must explicitly assign ownership for array LUNs. Disks can be automatically or manually assigned.

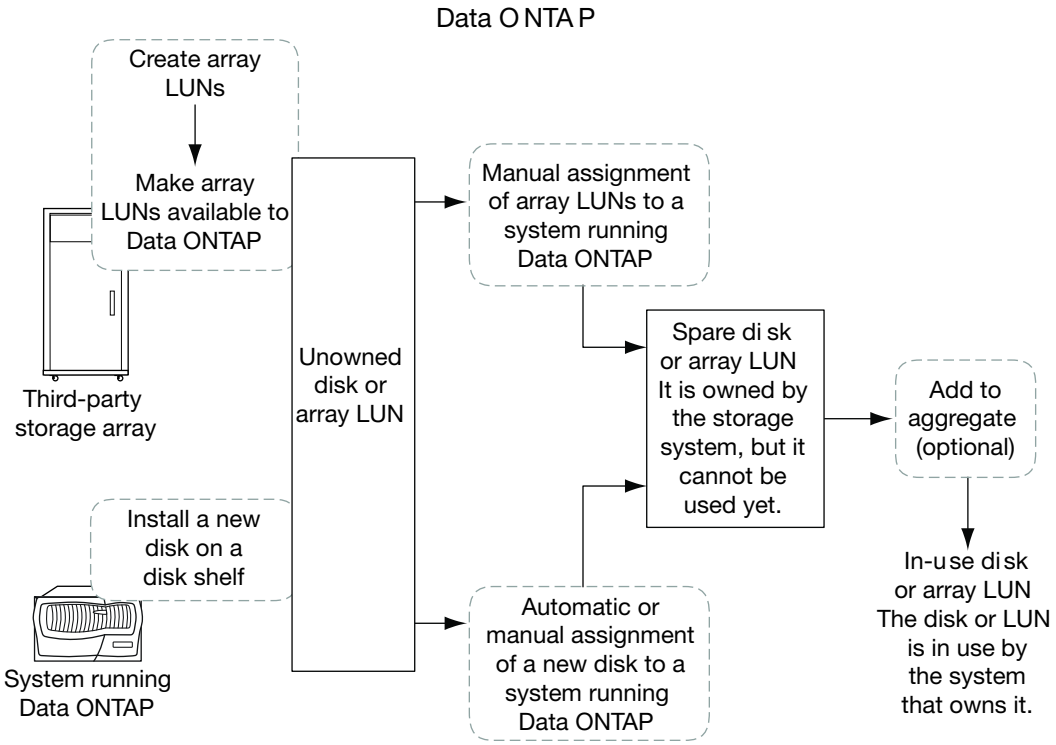
You assign ownership of a disk or array LUN to accomplish the following actions:

- Associate the disk or array LUN with a specific storage system.
  For a stand-alone system, all disks and array LUNs are owned by that system. In an HA configuration, the disks and array LUNs can be owned by either system.
- Enable the disk or array LUN to be used and managed by the system that owns it.
  Unowned disks cannot be used as spares and do not receive the automatic firmware updates that owned disks do.

How disks and array LUNs become available for use

When you add a disk or array LUN to a system running Data ONTAP, the disk or array LUN goes through several stages before it can be used by Data ONTAP to store data or parity information.

The process for making a disk available for use differs slightly from the process for making an array LUN available for use. Both processes are shown in the following diagram.



The process for disks includes the following actions:

1. The administrator physically installs the disk into a disk shelf.
   Data ONTAP can see the disk but the disk is still unowned.
2. If the system is configured to support disk autoassignment, Data ONTAP assigns ownership for the disk. Otherwise, the administrator must assign ownership of the disk manually.
   The disk is now a spare disk.
3. The administrator or Data ONTAP adds the disk to an aggregate.
   The disk is now in use by that aggregate. It could contain data or parity information.

The process for array LUNs includes the following actions:

1. The storage array administrator creates the array LUN and makes it available to Data ONTAP.
   Data ONTAP can see the array LUN but the array LUN is still unowned.
2. The Data ONTAP administrator assigns ownership for the array LUN to a V-Series system.
   The array LUN is now a spare array LUN.
3. The Data ONTAP administrator adds the array LUN to an aggregate.
   The array LUN is now in use by that aggregate and is used to contain data.

How automatic ownership assignment works for disks

If your configuration follows some basic rules to avoid ambiguity, Data ONTAP can automatically assign ownership for disks. Automatic ownership assignment is not available for array LUNs.

If you decide to change the way Data ONTAP has assigned the disks, you can do so at any time.

If you need to temporarily remove disk ownership for a disk while you perform an administrative task, you must disable automatic disk ownership first to prevent Data ONTAP from immediately reassigning ownership for that disk.

You can disable disk automatic disk ownership assignment by using the `storage disk option modify` command.

What automatic ownership assignment does

When automatic disk ownership assignment runs, Data ONTAP looks for any unassigned disks and assigns them to the same system as all other disks on their loop or stack.

Note: If a single loop or stack has disks assigned to multiple systems, Data ONTAP does not perform automatic ownership assignment on that loop or stack. To avoid this issue, always follow the disk assignment guidelines.

When automatic ownership assignment is invoked

Automatic disk ownership assignment does not happen immediately after disks are introduced into the storage system.

Automatic ownership assignment is invoked at the following times:

- Every five minutes during normal system operation
- Ten minutes after the initial system initialization

  This delay enables the person configuring the system enough time to finish the initial disk assignments so that the results of the automatic ownership assignment are correct.
- Whenever you enable automatic ownership assignment.

Guidelines for assigning ownership for disks

When you assign ownership for disks, you need to follow certain guidelines to keep automatic ownership assignment working and to maximize fault isolation.

- Always assign all disks on the same loop or stack to the same system.
- Always assign disks in the same multi-disk carrier to the same system.

Assigning ownership for disks

Disks must be owned by a node before they can be used in an aggregate. If your cluster is not configured to use automatic disk ownership assignment, you must assign ownership manually.

About this task

You can use the wildcard character to assign more than one disk at once.

If you are reassigning a spare disk that is already owned by a different node, you must use the `-force` option for the `storage disk assign` command.

You cannot reassign a disk that is in use in an aggregate.

Steps

1. Display all unowned disks by entering the following command:
   `storage disk show -container-type unassigned`
2. Assign each disk by entering the following command:
   `storage disk assign -disk` ***`disk_name`*** **`-owner`** ***`owner_name`***

Related concepts

How automatic ownership assignment works for disks on page 41

How you use the wildcard character with the disk ownership commands

You can use the wildcard character ("`*`") with some commands, including commands to manage disk ownership. However, you should understand how Data ONTAP expands the wildcard character.

You can use the wildcard character with the following disk ownership commands:

- storage disk modify
- storage disk assign
- storage disk show
- storage disk removeowner

When you use the wildcard character with these commands, Data ONTAP expands it with zero or more characters to create a list of disk names that will be operated on by the command. This can be very useful when you want to assign all of the disks attached to a particular port or switch, for example.

Note: Be careful when you use the wildcard character. It is accepted anywhere in the disk name string, and is a simple string substitution. Therefore, you might get unexpected results.

For example, to assign all disks on port 1 of the switch brocade23 to node03, you would use the following command:

```
storage disk assign -disk brocade23:1.* -owner node03
```

However, if you left off the second ".", as in the following command, you would assign all disks attached to ports 1, 10, 11, 12, and so on:

```
storage disk assign -disk brocade23:1* -owner node03
```

Managing array LUNs using Data ONTAP

For Data ONTAP to be able to use storage on a third-party storage array, some tasks must be done on the storage array and some tasks must be done in Data ONTAP.

For example, the storage array administrator must create array LUNs for Data ONTAP use and map them to Data ONTAP. You can then assign them to nodes running Data ONTAP.

If the storage array administrator wants to make configuration changes to an array LUN after it is assigned to a node, for example, to resize it, you might need to perform some activities in Data ONTAP before it is possible to reconfigure the LUN on the storage array.

Related concepts

How ownership for disks and array LUNs works on page 46

Overview of setting up Data ONTAP to use array LUNs

You should complete basic setup of a cluster to work with native disks before setting up the cluster nodes to use third-party storage. The storage array administrator must present array LUNs to Data ONTAP and configure the required storage array parameters before you can configure the nodes to use the array LUNs.

Note: Storage array administrators can prepare storage for Data ONTAP any time before you assign the array LUNs to nodes running Data ONTAP.

Setup task	Who typically performs the task	Where to find information
1. Configure a cluster, join nodes to it, and verify basic setup.	Data ONTAP administrator	*Data ONTAP Software Setup Guide for Cluster-Mode*
2. Set up Data ONTAP features and test them.	Data ONTAP administrator	Various Data ONTAP guides
3. Create LUNs and make them available to Data ONTAP.	Storage array administrator or vendor	*V-Series Installation Requirements and Reference Guide* (Data ONTAP guidelines and requirements) Storage array documentation (how to create LUNs and make them available to hosts)

Setup task	Who typically performs the task	Where to find information
4. Configure parameters on the storage array so that it can work with Data ONTAP.	Storage array administrator or vendor	*V-Series Implementation Guide for Third-Party Storage* (for storage array parameters that must be set to work with Data ONTAP)
5. Connect the nodes to the storage array.	Data ONTAP administrator	*V-Series Installation Requirements and Reference Guide* (for connection procedures)
6. For each V-Series node that you want to use third-party storage, Install the license (v-series).	Data ONTAP administrator	*Data ONTAP Physical Storage Management Guide for Cluster-Mode*
7. Verify that there are no errors that would prevent the nodes from using the third-party storage, and that the configuration is set up as intended.	Data ONTAP administrator and the storage array administrator or vendor	*V-Series Systems Installation Requirements and Reference Guide*
8. Assign ownership of array LUNs to specific nodes.	Data ONTAP administrator	*Data ONTAP Physical Storage Management Guide for Cluster-Mode*
9. Create additional aggregates and assign additional LUNs to nodes as needed.	Data ONTAP administrator	*Data ONTAP Physical Storage Management Guide for Cluster-Mode*

See the *V-Series Systems Installation Requirements and Reference Guide* for general requirements for setting up third-party storage to work with Data ONTAP and the *V-Series Implementation Guide for Third-Party Storage* for required storage array-specific parameters to work with Data ONTAP.

Related tasks

Assigning ownership of array LUNs on page 51

Installing the license for using third-party storage (Cluster-Mode with V-Series systems)

Data ONTAP 8.1.x operating in Cluster-Mode requires that a v-series license be installed on each V-Series node in the cluster that you want to use third-party storage. Any V-Series node without this license installed shuts down after 72 hours.

Before you begin

Before starting this procedure, you must have completed the following:

- Installed the cluster
- Obtained the V-Series license code

About this task

In Cluster-Mode in Data ONTAP 8.1 and later, the only nodeshell-specific license that is required and accepted is the v-series license. Unlike other features that require a license, the v-series license must be installed on each V-Series node, not one license for the cluster.

Note: If your account team gave you other nodeshell licenses, do not add them through the nodeshell. In Data ONTAP 8.1 and later, licenses for Cluster-Mode, for example, protocol level licenses, are added through the clustershell.

Step

1. For each V-Series node in the cluster that you want to use third-party storage, enter the following command on the node through the nodeshell:

   ::>system node run -node *node name* license add *license code*

   The following messages are shown:

```
A v-series license has been installed.
V-Series Storage enabled.
```

How ownership for disks and array LUNs works

Disk and array LUN ownership determines which node owns a disk or array LUN. Understanding how ownership works enables you to maximize storage redundancy and manage your hot spares effectively.

Data ONTAP stores ownership information directly on the disk or array LUN.

Reasons to assign ownership of disks and array LUNs

Storage system ownership must be assigned for disks and array LUNs before they become an effective part of your system. You must explicitly assign ownership for array LUNs. Disks can be automatically or manually assigned.

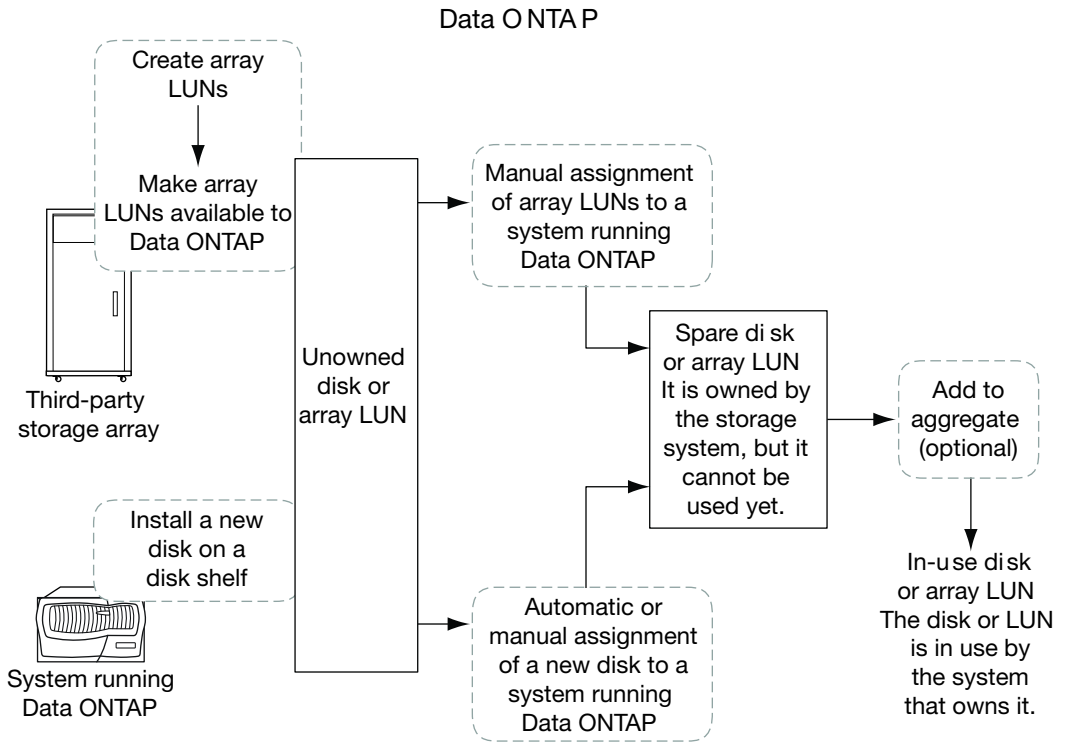
You assign ownership of a disk or array LUN to accomplish the following actions:

- Associate the disk or array LUN with a specific storage system.
  For a stand-alone system, all disks and array LUNs are owned by that system. In an HA configuration, the disks and array LUNs can be owned by either system.
- Enable the disk or array LUN to be used and managed by the system that owns it.
  Unowned disks cannot be used as spares and do not receive the automatic firmware updates that owned disks do.

How disks and array LUNs become available for use

When you add a disk or array LUN to a system running Data ONTAP, the disk or array LUN goes through several stages before it can be used by Data ONTAP to store data or parity information.

The process for making a disk available for use differs slightly from the process for making an array LUN available for use. Both processes are shown in the following diagram.



The process for disks includes the following actions:

1. The administrator physically installs the disk into a disk shelf.
   Data ONTAP can see the disk but the disk is still unowned.
2. If the system is configured to support disk autoassignment, Data ONTAP assigns ownership for the disk. Otherwise, the administrator must assign ownership of the disk manually.
   The disk is now a spare disk.
3. The administrator or Data ONTAP adds the disk to an aggregate.
   The disk is now in use by that aggregate. It could contain data or parity information.

The process for array LUNs includes the following actions:

1. The storage array administrator creates the array LUN and makes it available to Data ONTAP.
   Data ONTAP can see the array LUN but the array LUN is still unowned.
2. The Data ONTAP administrator assigns ownership for the array LUN to a V-Series system.
   The array LUN is now a spare array LUN.
3. The Data ONTAP administrator adds the array LUN to an aggregate.
   The array LUN is now in use by that aggregate and is used to contain data.

What it means for Data ONTAP to own an array LUN

Data ONTAP cannot use an array LUN presented to it by a storage array until you configure a logical relationship in Data ONTAP that identifies a specific system running Data ONTAP as the *owner* of the array LUN.

A storage array administrator creates array LUNs and makes them available to specified FC initiator ports of storage systems running Data ONTAP. (The process for how to do this varies among storage array vendors.) When you assign an array LUN to a system running Data ONTAP, Data ONTAP writes data to the array LUN to identify that system as the *owner* of the array LUN. Thereafter, Data ONTAP ensures that only the owner can write data to and read data from the array LUN.

From the perspective of Data ONTAP, this logical relationship is referred to as *disk ownership* because Data ONTAP considers an array LUN to be a virtual disk. From the perspective of Data ONTAP, you are assigning disks to a storage system.

An advantage of the disk ownership scheme is that you can make changes through the Data ONTAP software that, on typical hosts, must be done by reconfiguring hardware or LUN access controls. For example, through Data ONTAP you can balance the load of requests among a group of systems running Data ONTAP by moving data service from one system to another, and the process is transparent to most users. You do not need to reconfigure hardware or the LUN access controls on the storage array to change which system running Data ONTAP is the owner and, therefore, servicing data requests.

Attention: The Data ONTAP software-based scheme provides ownership control only for storage systems running Data ONTAP; it does not prevent a different type of host from overwriting data in an array LUN owned by a system running Data ONTAP. Therefore, if multiple hosts are accessing

array LUNs through the same storage array port, be sure to use LUN security on your storage array to prevent the systems from overwriting each other's array LUNs.

Array LUN reconfiguration, such as resizing the array LUN, must be done from the storage array. Before such activities can occur, you must release Data ONTAP ownership of the array LUN.

Why you might assign array LUN ownership after installation

For a V-Series system ordered with disk shelves, you are not required to set up third-party storage during initial installation. For a V-Series system using only third-party storage, you need to assign only two array LUNs during initial installation.

If you ordered your V-Series system with disk shelves, you do not need to assign any array LUNs initially because the factory installs the root volume on a disk for you. If you are using only third-party storage, you must configure one array LUN for the root volume and one array LUN as a spare for core dumps during initial installation. In either case, you can assign ownership of additional array LUNs to your system at any time after initial installation.

After initial configuration of your system, you might assign ownership of an array LUN in circumstances such as the following:

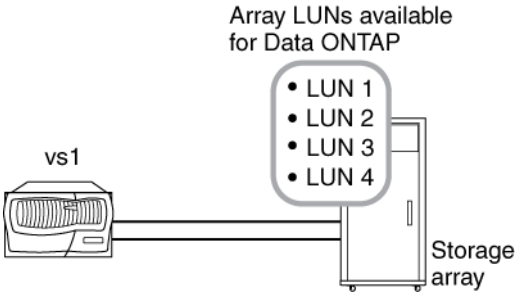
- You ordered your V-Series system with native disk shelves and you did not set up your system to work with third-party storage initially
- You left some LUNs that the storage array presented to Data ONTAP unowned and you now need to use the storage
- Another system released ownership of a particular array LUN and you want this system to be able to use the LUN
- The storage array administrator had not made the LUNs available to Data ONTAP when you initially configured your system and you now want to use the storage

Examples showing when Data ONTAP can use array LUNs

After an array LUN has been assigned to a storage system, it can be added to an aggregate and used for storage or it can remain a spare LUN until it is needed for storage.

No storage system owns the LUNs yet

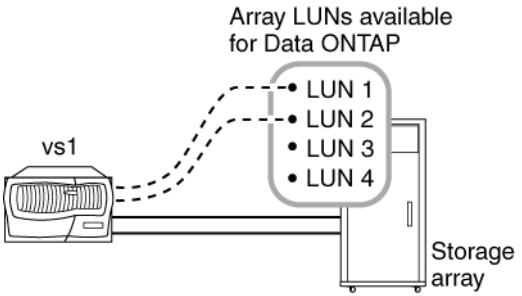
In this example, the storage array administrator made the array LUNs available to Data ONTAP. However, system vs1 has not yet been configured to "own" any of the LUNs. Therefore, it cannot read data from or write data to any array LUNs on the storage array.



Only some array LUNs are owned

In this example, vs1 was configured to own array LUNs 1 and 2, but not array LUNs 3 and 4. LUNs 3 and 4 are still available to Data ONTAP, however, and can be assigned to a storage system later.

Data ONTAP used the smallest of the two array LUNs, LUN 1, for the root volume. System vs1 can read data from and write data to LUN 1, because LUN 1 is in an aggregate. LUN 2 remains a spare LUN because it has not yet been added to an aggregate. System vs1 cannot read data from and write data to LUN 2 while it is a spare.

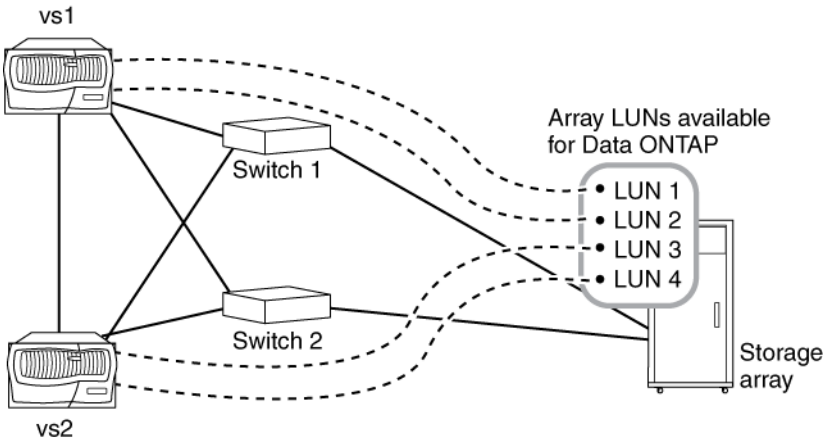


After you perform initial setup of the storage system, you could configure vs1 to also own LUN 3, LUN 4, both, or neither, depending on your storage needs.

Ownership of LUNs in an HA pair

In this example, two storage systems running Data ONTAP are configured in an HA pair. In an HA pair, only one node can be the owner of a particular LUN, but both nodes must be able to see the same LUNs so that the partner can take over if the owning node becomes unavailable.

LUN 1 through LUN 4 were created on the storage array and mapped to the ports on the storage array to which the storage systems are connected. All four LUNs are visible to each node in the HA pair.



Assume that during initial setup vs1 was assigned ownership of LUN 1 and LUN 2. LUN 1 was automatically added to the root volume, so LUN 1 is now "in use" by vs1. LUN 2 remains a spare until it is explicitly added to an aggregate on vs1. Similarly, assume that during initial setup vs2 was assigned ownership of LUN 3 and LUN 4, with LUN 3 assigned to the root volume. LUN 4 remains a spare LUN until it is explicitly added to an aggregate.

The key points of this example are as follows:

- By deploying the storage systems in an HA pair, one system can take over services for its partner if the partner becomes unavailable.
- Only one storage system can own a specific array LUN. However, all array LUNs assigned to a node in an HA pair must be visible to—but not assigned to or owned by—the other node in the HA pair.
- By deploying two switches, if one switch fails, the other switch provides the alternate path to the storage array.
- Both switches must be zoned correctly so that each storage system in the HA pair can see the array LUNs owned by its partner.

Assigning ownership of array LUNs

Array LUNs must be owned by a node before they can be added to an aggregate to be used as storage.

Before you begin

- Back-end configuration testing (testing of the connectivity and configuration of devices behind the V-Series systems) must be completed.
- Array LUNs that you want to assign must be presented to the V-Series systems.

About this task

You can assign ownership of array LUNs that have the following characteristics:

- They are unowned.
- They have no storage array configuration errors, such as the following:
  - The array LUN is smaller than or larger than the size that Data ONTAP supports.
  - The LDEV is mapped on only one port.
  - The LDEV has inconsistent LUN IDs assigned to it.
  - The LUN is available on only one path.

Data ONTAP issues an error message if you try to assign ownership of an array LUN with back-end configuration errors that would interfere with the V-Series system and the storage array operating together properly. You must fix such errors before you can proceed with array LUN assignment. See the *V-Series Installation Requirements and Reference Guide* for information about how to fix these types of errors.

Data ONTAP alerts you if you try to assign an array LUN with a redundancy error, for example, all paths to this array LUN are connected to the same controller or only one path to the array LUN. You can fix a redundancy error before or after assigning ownership of the LUN.

Steps

1. Enter the following command to see the array LUNs that have not yet been assigned to a node:

   `storage disk show -container-type unassigned`

2. Enter the following command to assign an array LUN to this node:

   `storage disk assign -disk` ***`arrayLUNname`*** **`-owner`** ***`nodename`***

   If you want to fix a redundancy error after disk assignment instead of before, you must use the `-force` parameter with the `storage disk assign` command.

   If you want the array LUN to be designated as an AZCS checksum type, you must add `-c zoned` to your command.

   The default and recommended checksum type is block. For a description of the block (BCS) and advanced checksum (AZCS) types, see the *V-Series Installation Requirements and Reference Guide*.

Related concepts

How ownership for disks and array LUNs works on page 46

Related tasks

Modifying assignment of spare array LUNs on page 53

Verifying back-end configuration

It is important to detect and resolve any configuration errors before you bring the configuration online in a production environment. You start installation verification by using `storage array config show` command.

The `storage array show config` command shows how storage arrays connect to the cluster. If Data ONTAP detects an error in the back-end configuration, the following message is displayed at the bottom of the `storage array show config` output:

```
Warning: Configuration errors were detected.  Use 'storage errors show'
for detailed information.
```

You then use the `storage errors show` output to see details of the problem, at the LUN level. You must fix any errors shown by `storage errors show`.

For detailed information about what back-end configuration you need to verify and how to do it, see the *V-Series Installation Requirements and Reference Guide*.

Modifying assignment of spare array LUNs

You can change the ownership of a *spare* array LUN to another node. You might want to do this for load balancing over the nodes.

Steps

1. At the console of the node that owns the array LUN you want to reassign, enter the following command to see a list of spare array LUNs on the node:

   `storage disk show -owner local`

   The array LUNs owned by the node, both spares and LUNs in aggregates, are listed.
2. Confirm that the LUN you want to reassign to another node is a spare LUN.
3. Enter the following command to assign ownership of the array LUN to another node:

   `storage disk assign` ***`arrayLUNname`*** **`-owner`** ***`new_owner_name`*** **`-force`**

   Note: The array LUN ownership is not changed if the `-force` option is not used or if the array LUN was already added to an aggregate.
4. Enter the following command to verify that the ownership of the spare array LUN was changed to the other node:

   `storage disk show -owner local`

   The spare array LUN that you changed to the new owner should no longer appear in the list of spares. If the array LUN still appears, repeat the command to change ownership.

5. On the destination node, enter the following command to verify that the spare array LUN whose ownership you changed is listed as a spare owned by the destination node:

   ```
   storage disk show -owner local
   ```

After you finish

You must add the array LUN to an aggregate before it can be used for storage.

Related concepts

How ownership for disks and array LUNs works on page 46

Related tasks

Assigning ownership of array LUNs on page 51

Array LUN name format

The array LUN name is a path-based name that includes the devices in the path between the V-Series system and the storage array, ports used, and the SCSI LUN ID on the path that the storage array presents externally for mapping to hosts.

On an 8.0.x V-Series system operating in Cluster-Mode, there are two names for each array LUN because there are two paths to each LUN.

Array LUN name format for systems operating in Cluster-Mode

Configuration	Array LUN name format	Component descriptions
Direct-attached	*node-name.adapter.idlun-id*	*node-name* is the name of the Cluster-Mode node. With Cluster-Mode, the node name is prepended to the LUN name so that the path-based name is unique within the cluster. *adapter* is the adapter number on the V-Series system. *id* is the channel adapter port on the storage array. *lun-id* is the array LUN number that the storage array presents to hosts. Example: *node1.0a.0L1*

Configuration	Array LUN name format	Component descriptions
Fabric-attached	*node-name*`:`*switch-name*`:`*port*`.`*id*`L`*lun-id*	*node-name* is the name of the Cluster-Mode node. With Cluster-Mode, the node name is prepended to the LUN name so that the path-based name is unique within the cluster. *switch-name* is the name of the switch. *port* is the switch port that is connected to the target port (the end point). *id* is the device ID. *lun-id* is the array LUN number that the storage array presents to hosts. Example: *node1*`:`*brocade3*`:`*6*`.`*126*`L`*1*

Related concepts

Disk name formats on page 14

Checking the checksum type of spare array LUNs

If you plan to add a spare array LUN to an aggregate by specifying its name, you need to make sure that the checksum type of the array LUN you want to add is the same as the aggregate checksum type.

About this task

You cannot mix array LUNs of different checksum types in an aggregate for third-party storage. The checksum type of the aggregate and the checksum type of the array LUNs added to it must be the same.

If you specify a number of spare array LUNs to be added to an aggregate, by default Data ONTAP selects array LUNs of the same checksum type as the aggregate.

Note: Data ONTAP 8.1.1 and later supports a new checksum scheme called *advanced zoned checksum* (AZCS). Existing zoned checksum aggregates are still supported. The checksum type of all newly created aggregates using zoned checksum array LUNS is AZCS, which provides more functionality than the “version 1” zoned checksum type that was supported in previous releases and continues to be supported for existing zoned aggregates. Zoned checksum spare array LUNs

added to an existing zoned checksum aggregate continue to be zoned checksum array LUNs. Zoned checksum spare array LUNs added to an AZCS checksum type aggregate use the AZCS checksum scheme for managing checksums.

Step

1. Check the checksum type of the spare array LUNs by entering the following command:

Mode	Command
Cluster-Mode	**`storage disk show -fields checksum-compatibility -container-type spare`** You can add a block checksum array LUN to a block checksum aggregate and a zoned array LUN to either a zoned checksum aggregate or an AZCS checksum aggregate.

Related tasks

Changing the checksum type of an array LUN on page 56

Changing the checksum type of an array LUN

You need to change the checksum type of an array LUN if you want to add it to an aggregate that is a different checksum type than the checksum type of the LUN.

Before you begin

Before changing the checksum type of an array LUN, you should have reviewed the tradeoffs between performance in certain types of workloads and storage capacity utilization of each checksum type. The *V-Series Installation Requirements and Reference Guide* contains information about checksum use for array LUNs. You can also contact your Sales Engineer for details about using checksums.

About this task

You need to assign a `zoned` checksum type to an array LUN that you plan to add to a zoned checksum aggregate or an advanced zoned checksum (AZCS) aggregate. When a zoned checksum array LUN is added to an AZCS aggregate, it becomes an advanced zoned checksum array LUN. Similarly, when a zoned checksum array LUN is added to a zoned aggregate, it is a zoned checksum type.

Step

1. Enter the following command to change the checksum type:

   `storage disk assign -disk` ***`LUN-path`*** **`-o`** ***`owner`*** **`-c`** ***`new_checksum_type`***

LUN-path is the <current-owner>:<LUNname> of the array LUN whose checksum type you want to change. *owner* is the current owner. *new_checksum_type* can be `block` or `zoned`.

Example

```
storage disk assign -disk system147b:vgbr300s181:5.126L2 -o system147b -
c block
```

The checksum type of the array LUN is changed to the new checksum type you specified.

Related tasks

Checking the checksum type of spare array LUNs on page 55

Prerequisites to reconfiguring an array LUN on the storage array

If an array LUN has already been assigned (through Data ONTAP) to a particular V-Series system, the information Data ONTAP wrote to the array LUN must be removed before the storage administrator attempts to reconfigure the array LUN on the storage array.

When the storage array presents an array LUN to Data ONTAP, Data ONTAP collects information about the array LUN (for example, its size) and writes that information to the array LUN. Data ONTAP cannot dynamically update information that it wrote to an array LUN. Therefore, before the storage array administrator reconfigures an array LUN, you must use Data ONTAP to change the state of the array LUN to *unused*. (The array LUN is unused from the perspective of Data ONTAP.)

While changing the state of the array LUN to unused, Data ONTAP does the following:

- Terminates I/O operations to the array LUN.
- Removes the label for RAID configuration information and the persistent reservations from the array LUN, which makes the array LUN unowned by any V-Series system.

After this process finishes, no Data ONTAP information remains in the array LUN.

You can do the following after the array LUN's state is unused:

- Remove the mapping of the array LUN to Data ONTAP and make the array LUN available to other hosts.
- Resize the array LUN or change its composition.

If you want Data ONTAP to use the array LUN again after its size or composition is changed, you must present the array LUN to Data ONTAP again and assign the array LUN to a V-Series system again. Data ONTAP is aware of the new array LUN size or composition.

Related tasks

Changing array LUN size or composition on page 58

Changing array LUN size or composition

Reconfiguring the size or composition of an array LUN must be done on the storage array. If an array LUN has already been assigned to a V-Series system, you must use Data ONTAP to change the state of the array LUN to unused before the storage array administrator can reconfigure it.

Before you begin

The array LUN must be a spare array LUN before you can change its state to unused.

Steps

1. On the V-Series system, enter the following command to access the nodeshell:

   system run -node *node_name*

   node_name is the name of this system.

2. On the V-Series system, enter the following command to remove ownership information:

   disk remove -w *LUNfullname*

3. Enter the following command to exit nodeshell:

   exit

   After the ownership information is removed, the array LUN cannot be used by any V-Series system until the array LUN is assigned again to a system. You can leave the array LUN as a spare or add it to an aggregate. You must assign the array LUN to an aggregate before the array LUN can be used for storage.

4. On the storage array, complete the following steps:

   a) Unmap (unpresent) the array LUN from the V-Series systems so that they can no longer see the array LUN.
   b) Change the size or composition of the array LUN.
   c) If you want Data ONTAP to use the array LUN again, present the array LUN to the V-Series systems again.

      At this point, the array LUN is visible to the FC initiator ports to which the array LUN was presented, but it cannot be used by any V-Series systems yet.

5. Enter the following command on the V-Series system that you want to be the owner of the array LUN:

   storage disk assign -disk *arrayLUNname* -owner *nodename*

   If you want the array LUN to be designated as an AZCS checksum type, you must add `-c zoned` to your command.

   After the ownership information is removed, the array LUN cannot be used by any V-Series system until the array LUN is assigned again to a system. You can leave the array LUN as a spare

or add it to an aggregate. You must assign the array LUN to an aggregate before the array LUN can be used for storage.

Related concepts

Prerequisites to reconfiguring an array LUN on the storage array on page 57

Removing one array LUN from use by Data ONTAP

If the storage array administrator no longer wants to use a particular array LUN for Data ONTAP, you must remove the information that Data ONTAP wrote to the LUN (for example, size and ownership) before the administrator can reconfigure the LUN for use by another host.

Before you begin

If the LUN that the storage array administrator no longer want Data ONTAP to use is in an aggregate, you must take the aggregate to which the LUN belongs offline and destroy the aggregate before starting this procedure. Taking an aggregate offline and destroying it changes the LUN from a data LUN to a spare LUN.

Steps

1. Enter the following to access the nodeshell:
   `system run -node` ***`node_name`***
   node_name is the name of this system.
2. Enter the following command:
   `disk remove -w` ***`LUNfullname`***
   LUNfullname is the full name of the array LUN.
3. Enter the following to exit nodeshell:
   `exit`

Preparing array LUNs before removing a V-Series system from service

You must release the persistent reservations on all array LUNs assigned to a V-Series system before removing the system from service.

About this task

When you assign Data ONTAP ownership of an array LUN, Data ONTAP places persistent reservations (ownership locks) on that array LUN to identify which V-Series system owns the LUN. If you want the array LUNs to be available for use by other types of hosts, you must remove the

persistent reservations that Data ONTAP put on those array LUNs. The reason is that some arrays do not allow you to destroy a reserved LUN if you do not remove the ownership and persistent reservations that Data ONTAP wrote to that LUN.

For example, the Hitachi USP storage array does not have a user command for removing persistent reservations from LUNs. If you do not remove persistent reservations through Data ONTAP before removing the V-Series system from service, you must call Hitachi technical support to remove the reservations.

Contact Technical Support for instructions about how to remove persistent reservations from LUNs before removing a V-Series system from service.

How Data ONTAP uses RAID to protect your data and data availability

Understanding how RAID protects your data and data availability can help you administer your storage systems more effectively.

For native storage, Data ONTAP uses RAID-DP (double-parity) or RAID Level 4 (RAID4) protection to ensure data integrity within a group of disks even if one or two of those disks fail. Parity disks provide redundancy for the data stored in the data disks. If a disk fails (or, for RAID-DP, up to two disks), the RAID subsystem can use the parity disks to reconstruct the data in the drive that failed.

For third-party storage, Data ONTAP stripes data across the array LUNs using RAID0. The storage arrays, not Data ONTAP, provide the RAID protection for the array LUNs that they make available to Data ONTAP.

RAID protection levels for disks

Data ONTAP supports two levels of RAID protection for aggregates composed of disks in native disk shelves: RAID-DP and RAID4. RAID-DP is the default RAID level for new aggregates.

For more information about configuring RAID, see *Technical Report 3437: Storage Subsystem Resiliency Guide*.

Related information

TR 3437: Storage Subsystem Resiliency Guide

What RAID-DP protection is

If an aggregate is configured for RAID-DP protection, Data ONTAP reconstructs the data from one or two failed disks within a RAID group and transfers that reconstructed data to one or two spare disks as necessary.

RAID-DP provides double-parity disk protection when the following conditions occur:

- There is a single-disk or double-disk failure within a RAID group.
- There are media errors on a block when Data ONTAP is attempting to reconstruct a failed disk.

The minimum number of disks in a RAID-DP group is three: at least one data disk, one regular parity disk, and one double-parity (or dParity) disk.

If there is a data-disk or parity-disk failure in a RAID-DP group, Data ONTAP replaces the failed disk in the RAID group with a spare disk and uses the parity data to reconstruct the data of the failed disk on the replacement disk. If there is a double-disk failure, Data ONTAP replaces the failed disks

in the RAID group with two spare disks and uses the double-parity data to reconstruct the data of the failed disks on the replacement disks.

RAID-DP is the default RAID type for all aggregates.

What RAID4 protection is

RAID4 provides single-parity disk protection against single-disk failure within a RAID group. If an aggregate is configured for RAID4 protection, Data ONTAP reconstructs the data from a single failed disk within a RAID group and transfers that reconstructed data to a spare disk.

The minimum number of disks in a RAID4 group is two: at least one data disk and one parity disk.

If there is a single data or parity disk failure in a RAID4 group, Data ONTAP replaces the failed disk in the RAID group with a spare disk and uses the parity data to reconstruct the failed disk's data on the replacement disk. If no spare disks are available, Data ONTAP goes into degraded mode and alerts you of this condition.

Attention: With RAID4, if there is a second disk failure before data can be reconstructed from the data on the first failed disk, there will be data loss. To avoid data loss when two disks fail, you can select RAID-DP. This provides two parity disks to protect you from data loss when two disk failures occur in the same RAID group before the first failed disk can be reconstructed.

Note: Nondisruptive upgrade is not supported for aggregates configured for RAID4. For more information about nondisruptive upgrade, see the *Data ONTAP Upgrade and Revert/Downgrade Guide for Cluster-Mode*.

RAID protection for third-party storage

Third-party storage arrays provide the RAID protection for the array LUNs that they make available to V-Series systems, not Data ONTAP.

Data ONTAP uses RAID 0 (striping) for array LUNs. Data ONTAP supports a variety of RAID types on the storage arrays, except RAID 0 because RAID 0 does not provide storage protection.

When creating *RAID groups* on storage arrays, you need to follow the best practices of the storage array vendor to ensure that there is an adequate level of protection on the storage array so that disk failure does not result in loss of data or loss of access to data.

Note: A *RAID group* on a storage array is the arrangement of disks that together form the defined RAID level. Each RAID group supports only one RAID type. The number of disks that you select for a RAID group determines the RAID type that a particular RAID group supports. Different storage array vendors use different terms to describe this entity—RAID groups, parity groups, disk groups, Parity RAID groups, and other terms.

V-Series systems support native disk shelves as well as third-party storage. Data ONTAP supports RAID4 and RAID-DP on the native disk shelves connected to a V-Series system but does not support RAID4 and RAID-DP with array LUNs.

See the *V-Series Implementation Guide for Third-Party Storage* to determine whether there are specific requirements or limitations about RAID types for your storage array.

Understanding RAID disk types

Data ONTAP classifies disks as one of four types for RAID: data, hot spare, parity, or dParity. The RAID disk type is determined by how RAID is using a disk; it is different from the Data ONTAP disk type.

Data disk	Holds data stored on behalf of clients within RAID groups (and any data generated about the state of the storage system as a result of a malfunction).
Spare disk	Does not hold usable data, but is available to be added to a RAID group in an aggregate. Any functioning disk that is not assigned to an aggregate but is assigned to a system functions as a hot spare disk.
Parity disk	Stores row parity information that is used for data reconstruction when a single disk drive fails within the RAID group.
dParity disk	Stores diagonal parity information that is used for data reconstruction when two disk drives fail within the RAID group, if RAID-DP is enabled.

How RAID groups work

A RAID group consists of one or more data disks or array LUNs, across which client data is striped and stored, and up to two parity disks, depending on the RAID level of the aggregate that contains the RAID group.

RAID-DP uses two parity disks to ensure data recoverability even if two disks within the RAID group fail.

RAID4 uses one parity disk to ensure data recoverability if one disk within the RAID group fails.

RAID0 does not use any parity disks; it does not provide data recoverability if any disks within the RAID group fail.

How RAID groups are named

Within each aggregate, RAID groups are named rg0, rg1, rg2, and so on in order of their creation. You cannot specify the names of RAID groups.

About RAID group size

A RAID group has a maximum number of disks or array LUNs that it can contain. This is called its maximum size, or its size. A RAID group can be left partially full, with fewer than its maximum

number of disks or array LUNs, but storage system performance is optimized when all RAID groups are full.

Related references

Storage limits on page 99

Considerations for sizing RAID groups for disks

Configuring an optimum RAID group size for an aggregate made up of disks requires a trade-off of factors. You must decide which factor—speed of recovery, assurance against data loss, or maximizing data storage space—is most important for the aggregate that you are configuring.

You change the size of RAID groups on a per-aggregate basis. You cannot change the size of an individual RAID group.

HDDs

You should follow these guidelines when sizing your RAID groups for HDD disks:

- All RAID groups in an aggregate should have the same number of disks.
  If this is impossible, any RAID group with fewer disks should have only one less disk than the largest RAID group.
- The recommended range of RAID group size is between 12 and 20.
  The reliability of SAS and FC disks can support a RAID group size of up to 28 if needed.
- If you can satisfy the first two guidelines with multiple RAID group sizes, you should choose the larger size.

SSDs

You should follow these guidelines when sizing your RAID groups for SSD disks:

- All RAID groups in an aggregate should have the same number of disks.
  If this is impossible, any RAID group with fewer disks should have only one less disk than the largest RAID group.
- The recommended range of RAID group size is between 20 and 28.

Flash Pools

For Flash Pools, the SSD tier must have the same RAID group size as the HDD tier. You should use the HDD guidelines to determine the RAID group size for the entire Flash Pool.

Related references

Storage limits on page 99

Considerations for Data ONTAP RAID groups for array LUNs

Setting up Data ONTAP RAID groups for array LUNs requires planning and coordination with the storage array administrator so that the administrator makes the number and size of array LUNs you need available to Data ONTAP.

For array LUNs, Data ONTAP uses RAID0 RAID groups to determine where to allocate data to the LUNs on the storage array. The RAID0 RAID groups are not used for RAID data protection. The storage arrays provide the RAID data protection.

Note: Data ONTAP RAID groups are similar in concept to what storage array vendors call RAID groups, parity groups, disk groups, Parity RAID groups, and other terms.

Follow these steps when planning your Data ONTAP RAID groups for array LUNs:

1. Plan the size of the aggregate that best meets your data needs.
2. Plan the number and size of RAID groups that you need for the size of the aggregate.

   Follow these guidelines:

   - RAID groups in the same aggregate should be the same size with the same number of LUNs in each RAID group. For example, you should create four RAID groups of 8 LUNs each, not three RAID groups of 8 LUNs and one RAID group of 6 LUNs.
   - Use the default RAID group size for array LUNs, if possible. The default RAID group size is adequate for most organizations.

     Note: The default RAID group size is different for array LUNs and disks.
3. Plan the size of the LUNs that you need in your RAID groups.

   - To avoid a performance penalty, all array LUNs in a particular RAID group should be the same size.
   - The LUNs should be the same size in all RAID groups in the aggregate.
4. Ask the storage array administrator to create the number of LUNs of the size you need for the aggregate.

   The LUNs should be optimized for performance, according to the instructions in the storage array vendor documentation.
5. Create all the RAID groups in the aggregate at the same time.

   Note: Do not mix array LUNs from storage arrays with different characteristics in the same Data ONTAP RAID group.

   Note: If you create a new RAID group for an existing aggregate, be sure that the new RAID group is the same size as the other RAID groups in the aggregate, and that the array LUNs are the same size as the LUNs in the other RAID groups in the aggregate.

How Data ONTAP works with hot spare disks

A hot spare disk is a disk that is assigned to a storage system but is not in use by a RAID group. It does not yet hold data but is ready for use. If a disk failure occurs within a RAID group, Data ONTAP automatically assigns hot spare disks to RAID groups to replace the failed disks.

How many hot spares you should have

Having insufficient spares increases the risk of a disk failure with no available spare, resulting in a degraded RAID group. The number of hot spares you should have depends on the Data ONTAP disk type.

MSATA disks, or disks in a multi-disk carrier, should have four hot spares during steady state operation, and you should never allow the number of MSATA hot spares to dip below two.

For RAID groups composed of SSDs, you should have at least one spare disk.

For all other Data ONTAP disk types, you should have at least one matching or appropriate hot spare available for each kind of disk installed in your storage system. However, having two available hot spares for all disks provides the best protection against disk failure. Having at least two available hot spares provides the following benefits:

- When you have two or more hot spares for a data disk, Data ONTAP can put that disk into the maintenance center if needed.
  Data ONTAP uses the maintenance center to test suspect disks and take offline any disk that shows problems.
- Having two hot spares means that when a disk fails, you still have a spare available if another disk fails before you replace the first failed disk.

A single spare disk can serve as a hot spare for multiple RAID groups.

Related concepts

Spare requirements for multi-disk carrier disks on page 28

What disks can be used as hot spares

A disk must conform to certain criteria to be used as a hot spare for a particular data disk.

For a disk to be used as a hot spare for another disk, it must conform to the following criteria:

- It must be either an exact match for the disk it is replacing or an appropriate alternative.
- The spare must be owned by the same system as the disk it is replacing.

What a matching spare is

A matching hot spare exactly matches several characteristics of a designated data disk. Understanding what a matching spare is, and how Data ONTAP selects spares, enables you to optimize your spares allocation for your environment.

A matching spare is a disk that exactly matches a data disk for all of the following criteria:

- Effective Data ONTAP disk type
  The effective disk type can be affected by the value of the `raid.disktype.enable` option, which affects which disk types are considered to be equivalent.
- Size
- Speed (RPM)
- Checksum type (BCS or AZCS)

Related concepts

How Data ONTAP reports disk types on page 8

What an appropriate hot spare is

If a disk fails and no hot spare disk that exactly matches the failed disk is available, Data ONTAP uses the best available spare. Understanding how Data ONTAP chooses an appropriate spare when there is no matching spare enables you to optimize your spare allocation for your environment.

Data ONTAP picks a non-matching hot spare based on the following criteria:

- If the available hot spares are not the correct size, Data ONTAP uses one that is the next size up, if there is one.
  The replacement disk is downsized to match the size of the disk it is replacing; the extra capacity is not available.
- If the available hot spares are not the correct speed, Data ONTAP uses one that is a different speed.
  Using drives with different speeds within the same aggregate is not optimal. Replacing a disk with a slower disk can cause performance degradation, and replacing a disk with a faster disk is not cost-effective.

If no spare exists with an equivalent disk type or checksum type, the RAID group that contains the failed disk goes into degraded mode; Data ONTAP does not combine effective disk types or checksum types within a RAID group.

Related concepts

How Data ONTAP reports disk types on page 8

About degraded mode

When a disk fails, Data ONTAP can continue to serve data, but it must reconstruct the data from the failed disk using RAID parity. When this happens, the affected RAID group is said to be in *degraded mode*. The performance of a storage system with one or more RAID groups in degraded mode is decreased.

A RAID group goes into degraded mode in the following scenarios:

- A single disk fails in a RAID4 group.
  After the failed disk is reconstructed to a spare, the RAID group returns to normal mode.
- One or two disks fail in a RAID-DP group.
  If two disks have failed in a RAID-DP group, the RAID group goes into *double-degraded mode*.
- A disk is taken offline by Data ONTAP.
  After the offline disk is brought back online, the RAID group returns to normal mode.

Note: If another disk fails in a RAID-DP group in double-degraded mode or a RAID4 group in degraded mode, data loss could occur (unless the data is mirrored). For this reason, always minimize the amount of time a RAID group is in degraded mode by ensuring that appropriate hot spares are available.

Related concepts

How Data ONTAP handles a failed disk that has no available hot spare on page 69

About low spare warnings

By default, Data ONTAP issues warnings to the console and logs if you have fewer than one hot spare disk that matches the attributes of each disk in your storage system. You can change the threshold value for these warning messages by using the `raid.min_spare_count` option.

To make sure that you always have two hot spares for every disk (a best practice), you can use the nodeshell to set the `raid.min_spare_count` option to 2.

Setting the `raid.min_spare_count` option to 0 disables low spare warnings. You might want to do this if you do not have enough disks to provide hot spares (for example if your storage system does not support external disk shelves). You can disable the warnings only if the following requirements are met:

- Your system has 16 or fewer disks.
- You have no RAID groups that use RAID4.

  Note: You cannot create aggregates that use RAID4 protection while the `raid.min_spare_count` option is set to 0. If either of these requirements is no longer met after this option has been set to 0, the option is automatically set back to 1.

How Data ONTAP handles a failed disk with a hot spare

Using an available matching hot spare, Data ONTAP can use RAID to reconstruct the missing data from the failed disk onto the hot spare disk with no data service interruption.

If a disk fails and a matching or appropriate spare is available, Data ONTAP performs the following tasks:

- Replaces the failed disk with a hot spare disk.
  If RAID-DP is enabled and double-disk failure occurs in the RAID group, Data ONTAP replaces each failed disk with a separate spare disk.
- In the background, reconstructs the missing data onto the hot spare disk or disks.

  Note: During reconstruction, the system is in degraded mode, and file service might slow down.
- Logs the activity to the event log, which you can view by using the `event log show` command.
- Sends an AutoSupport message.

Attention: Replace the failed disk or disks with new hot spare disks as soon as possible, so that hot spare disks are always available in the storage system.

Note: If the available spare disks are not the correct size, Data ONTAP chooses a disk of the next larger size and restricts its capacity to match the size of the disk it is replacing.

Related concepts

How Data ONTAP handles a failed disk that has no available hot spare on page 69

How Data ONTAP handles a failed disk that has no available hot spare

When a failed disk has no appropriate hot spare available, Data ONTAP puts the affected RAID group into degraded mode indefinitely and the storage system automatically shuts down within a specified time period.

If the maximum number of disks have failed in a RAID group (two for RAID-DP, one for RAID4), the storage system automatically shuts down in the period of time specified by the `raid.timeout` option. The default timeout value is 24 hours.

To ensure that you are aware of the situation, Data ONTAP sends an AutoSupport message whenever a disk fails. In addition, it logs a warning message to the event log once per hour after a disk fails.

Attention: If a disk fails and no hot spare disk is available, contact technical support.

Related concepts

How Data ONTAP handles a failed disk with a hot spare on page 69
About degraded mode on page 68

Considerations for changing the timeout RAID option

The `raid.timeout` option controls how long a storage system runs after a RAID group goes into degraded mode or the NVRAM battery malfunctions or loses power. You can change the value of this option, but you should understand the implications of doing so.

The purpose for the system shutdown is to avoid data loss, which can happen if an additional disk failure occurs in a RAID group that is already running in degraded mode, or if a stand-alone system encounters a catastrophic error and has to shut down without NVRAM. You can extend the number of hours the system operates in these conditions by increasing the value of this option (the default value is `24`). You can even disable the shutdown by setting the option to `0`, but the longer the system operates with one or both of these conditions, the greater the chance of incurring data loss.

How RAID-level disk scrubs verify data integrity

RAID-level scrubbing means checking the disk blocks of all disks in use in aggregates (or in a particular aggregate, plex, or RAID group) for media errors and parity consistency. If Data ONTAP finds media errors or inconsistencies, it uses RAID to reconstruct the data from other disks and rewrites the data.

RAID-level scrubs help improve data availability by uncovering and fixing media and checksum errors while the RAID group is in a normal state (for RAID-DP, RAID-level scrubs can also be performed when the RAID group has a single-disk failure).

RAID-level scrubs can be scheduled or run manually.

How you schedule automatic RAID-level scrubs

By default, Data ONTAP performs a weekly RAID-level scrub starting on Sunday at 1:00 a.m. for a duration of six hours. You can change the start time and duration of the weekly scrub, or add more automatic scrubs.

To schedule an automatic RAID-level scrub, you use the `raid.scrub.schedule` option.

To change the duration of automatic RAID-level scrubbing without changing the start time, you use the `raid.scrub.duration` option, specifying the number of minutes you want automatic RAID-level scrubs to run. If you set this option to `-1`, all automatic RAID-level scrubs run to completion.

Note: If you specify a duration using the `raid.scrub.schedule` option, that value overrides the value you specify with the `raid.scrub.duration` option.

Scheduling example

The following command schedules two weekly RAID scrubs. The first scrub is for 240 minutes (four hours) every Tuesday starting at 2 a.m. The second scrub is for eight hours every Saturday starting at 10 p.m.

```
storage raid-options modify -node nodename -name raid.scrub.schedule -
value 240m@tue@2,8h@sat@22
```

Verification example

The following command displays your current RAID-level automatic scrub schedule.

```
storage raid-options show raid.scrub.schedule
```

Reverting to the default schedule example

The following command reverts your automatic RAID-level scrub schedule to the default (Sunday at 1:00 a.m., for six hours):

```
storage raid-options modify -node nodename -name raid.scrub.schedule -
value ""
```

How you run a manual RAID-level scrub

You can manually run a RAID-level scrub on individual RAID groups, plexes, aggregates, or all aggregates using the `storage aggregate scrub` command. You can also stop, suspend, and resume manual RAID-level scrubs.

If you try to run a RAID-level scrub on a RAID group that is not in a normal state (for example, a group that is reconstructing or degraded), the scrub returns errors and does not check that RAID group. You can run a RAID-level scrub on a RAID-DP group with one failed disk.

Customizing the size of your RAID groups

You can customize the size of your RAID groups based on your requirements for data availability, performance, and disk utilization.

About this task

You change the size of RAID groups on a per-aggregate basis. You cannot change the size of individual RAID groups.

The following list outlines some facts about changing the RAID group size for an aggregate:

- If you increase the RAID group size, more disks or array LUNs will be added to the most recently created RAID group until it reaches the new size.

- All other existing RAID groups in that aggregate remain the same size, unless you explicitly add disks to them.
- You cannot decrease the size of already created RAID groups.
- The new size applies to all subsequently created RAID groups in that aggregate.

Step

1. Enter the following command:
   storage aggregate modify -aggregate *aggr_name* -maxraidsize *size*

Controlling the impact of RAID operations on system performance

You can reduce the impact of RAID operations on system performance by decreasing the speed of the RAID operations.

You can control the speed of the following RAID operations with RAID options:

- RAID data reconstruction
- Disk scrubbing

The speed that you select for each of these operations might affect the overall performance of the storage system. However, if the operation is already running at the maximum speed possible and it is fully utilizing one of the three system resources (the CPU, disks, or the disk-to-controller connection bandwidth), changing the speed of the operation has no effect on the performance of the operation or the storage system.

If the operation is not yet running, you can set a speed that minimally slows storage system network operations or a speed that severely slows storage system network operations. For each operation, use the following guidelines:

- If you want to reduce the performance impact on client access to the storage system, change the specific RAID option from medium to low. Doing so also causes the operation to slow down.
- If you want to speed up the operation, change the RAID option from medium to high. Doing so might decrease the performance of the storage system in response to client access.

Controlling the performance impact of RAID data reconstruction

Because RAID data reconstruction consumes CPU resources, increasing the speed of data reconstruction sometimes slows storage system network and disk operations. You can control the speed of data reconstruction with the `raid.reconstruc.perf_impact` option.

About this task

When RAID data reconstruction and plex resynchronization are running at the same time, Data ONTAP limits the combined resource utilization to the greater impact set by either operation. For example, if `raid.resync.perf_impact` is set to `medium` and

`raid.reconstruct.perf_impact` is set to `low`, the resource utilization of both operations has a medium impact.

The setting for this option also controls the speed of Rapid RAID Recovery.

Step

1. Enter the following command:

   storage raid-options modify -node *node_name* raid.reconstruct.perf_impact *impact*

   impact can be `high`, `medium`, or `low`.

   `high` means that the storage system uses most of the system resources available for RAID data reconstruction; this setting can heavily affect storage system performance, but reconstruction finishes sooner, reducing the time that the RAID group is in degraded mode.

   `low` means that the storage system uses very little of the system resources; this setting lightly affects storage system performance. However, reconstruction takes longer to complete, increasing the time that the storage system is running in degraded mode.

   The default impact is `medium`.

Controlling the performance impact of RAID-level scrubbing

When Data ONTAP performs a RAID-level scrub, it checks the disk blocks of all disks on the storage system for media errors and parity consistency. You can control the impact this operation has on system performance with the `raid.verify.perf_impact` option.

About this task

When RAID-level scrubbing and mirror verification are running at the same time, Data ONTAP limits the combined resource utilization to the greater impact set by either operation. For example, if `raid.verify.perf_impact` is set to `medium` and `raid.scrub.perf_impact` is set to `low`, the resource utilization by both operations has a medium impact.

If there are times during the day when the load on your storage system is decreased, you can also limit the performance impact of the automatic RAID-level scrub by changing the start time or duration of the automatic scrub.

Step

1. Enter the following command:

   storage raid-options modify -node *node_name* raid.scrub.perf_impact *impact*

   impact can be `high`, `medium`, or `low`.

   `high` means that the storage system uses most of the system resources available for scrubbing; this setting can heavily affect storage system performance, but the scrub finishes sooner.

`low` means that the storage system uses very little of the system resources; this setting lightly affects storage system performance, but the scrub takes longer to complete.

The default impact is `low`.

How you use aggregates to provide storage to your volumes

To support the differing security, backup, performance, and data sharing needs of your users, you can group the physical data storage resources on your storage system into one or more aggregates. You can then design and configure these aggregates to provide the appropriate level of performance and redundancy.

Each aggregate has its own RAID configuration, plex structure, and set of assigned disks or array LUNs. The aggregate provides storage, based on its configuration, to its associated FlexVol volumes or Infinite Volume.

Aggregates have the following characteristics:

- They can be composed of disks or array LUNs.
- They can be in 64-bit or 32-bit format.
- If they are composed of disks, they can be single-tier (composed of only HDDs or only SSDs) or they can be Flash Pools, which include both of those storage types in two separate tiers.

The cluster administrator can assign one or more aggregates to a Vserver, in which case you can use only those aggregates to contain volumes for that Vserver.

For information about best practices for working with aggregates, see *Technical Report 3437: Storage Subsystem Resiliency Guide*.

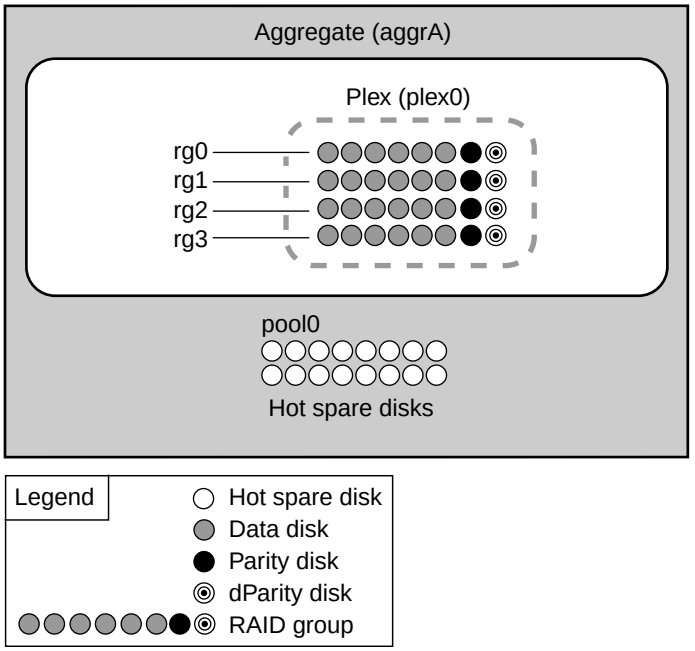
Related information

Technical Report 3437: Storage Subsystem Resiliency Guide

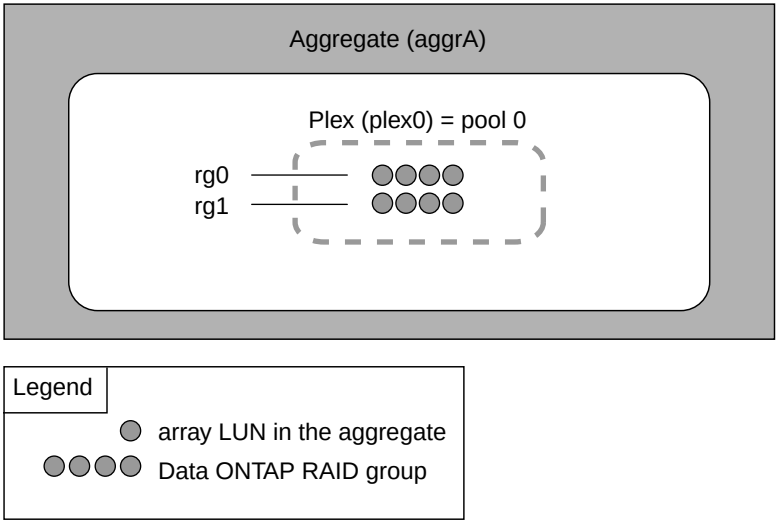
How aggregates work

Aggregates have a single copy of their data, or *plex*, which contains all of the RAID groups belonging to that aggregate. Mirrored aggregates are not currently supported in Data ONTAP Cluster-Mode.

The following diagram shows an unmirrored aggregate with disks, with its one plex.



The following diagram shows an unmirrored aggregate with array LUNs, with its one plex.



Introduction to 64-bit and 32-bit aggregate formats

Aggregates are either 64-bit or 32-bit format. 64-bit aggregates have much larger size limits than 32-bit aggregates. 64-bit and 32-bit aggregates can coexist on the same storage system or cluster.

32-bit aggregates have a maximum size of 16 TB; 64-bit aggregates' maximum size depends on the storage system model. For the maximum 64-bit aggregate size of your storage system model, see the *Hardware Universe* (formerly the *System Configuration Guide*) at *support.netapp.com/knowledge/docs/hardware/NetApp/syscfg/index.shtml*.

You decide the format of an aggregate when you create it. By default, newly created aggregates are 64-bit.

You can expand 32-bit aggregates to 64-bit aggregates by increasing their size beyond 16 TB. 64-bit aggregates, including aggregates that were previously expanded, cannot be converted to 32-bit aggregates.

You can see whether an aggregate is a 32-bit aggregate or a 64-bit aggregate by using the `storage aggregate show -fields block-type` command.

Related tasks

Increasing the size of an aggregate on page 93

Best practices for expanding a 32-bit aggregate to 64-bit

You should be aware of certain best practices before expanding an aggregate from 32-bit to 64-bit format.

Following these suggestions ensures a smooth expansion operation:

- If you are expanding aggregates that contain volumes in a SnapMirror relationship, expand the aggregate containing the source volume first whenever possible.

  Otherwise, expand the source aggregate as soon as possible after expanding the destination aggregate.
- If you are creating a FlexClone volume from a SnapMirror destination volume and you are expanding the aggregates containing the source and destination volumes, expand both source and destination, and use a base Snapshot copy that was created after the source volume was expanded.
- When you add storage to any aggregate, add an entire RAID group at a time to keep the size of your RAID groups homogeneous.

For more information about expanding a 32-bit aggregate to 64-bit, see TR-3978, *In-Place Expansion of 32-bit Aggregates to 64-bit Overview and Best Practices*.

Related information

TR 3978: In-Place Expansion of 32-bit Aggregates to 64-bit Overview and Best Practices

How the Vserver affects which aggregates can be associated with a FlexVol volume

FlexVol volumes are always associated with one Vserver, and one aggregate that supplies its storage. The Vserver can limit which aggregates can be associated with that volume, depending on how the Vserver is configured.

When you create a FlexVol volume, you specify which Vserver the volume will be created on, and which aggregate that volume will get its storage from. All of the storage for the newly created FlexVol volume comes from that associated aggregate.

If the Vserver for that volume has aggregates assigned to it, then you can use only one of those assigned aggregates to provide storage to volumes on that Vserver. This can help you ensure that your Vservers are not sharing physical storage resources inappropriately.

If the Vserver for that volume has no aggregates assigned to it, then a cluster administrator can use any aggregate in the cluster to provide storage to the new volume. However, a Vserver administrator cannot create volumes for a Vserver with no assigned aggregates. For this reason, if you want a Vserver administrator to be able to create volumes for a specific Vserver, then you must assign aggregates to that Vserver.

For more information about configuring and managing Vservers, see the *Data ONTAP System Administration Guide for Cluster-Mode*.

Related tasks

Assigning aggregates to a Vserver on page 97

How Flash Pool aggregates work

Flash Pool aggregates enable you to add one or more RAID groups composed of SSDs to an aggregate that is otherwise composed of HDDs. The SSDs function as a high-performance cache for the working data set, increasing the performance of the aggregate without incurring the expense of using SSDs for the entire aggregate.

You create a Flash Pool aggregate by enabling the feature on an existing 64-bit aggregate composed of HDDs, and then adding SSDs to that aggregate. This results in two tiers for that aggregate: an SSD tier and an HDD tier. After you add an SSD tier to an aggregate to create a Flash Pool aggregate, you cannot remove the SSD tier to convert the aggregate back to its original configuration.

The SSD tier and the HDD tier have the same RAID type (for example, RAID-DP) and the same maximum RAID group size. The tiers can have different checksum types.

The HDD RAID groups in Flash Pool aggregates behave the same as HDD RAID groups in standard aggregates, including the rules for mixing disk types, sizes, speeds, and checksums.

The SSD tier does not contribute to the size of the aggregate as calculated against the maximum aggregate size. For example, even if an aggregate is at the maximum aggregate size, you can add an SSD tier to it.

There is a platform-dependent maximum size for the SSD tier (cache). For information about this limit for your platform, see the *Hardware Universe*.

There are two types of caching used by Flash Pool aggregates: read caching and write caching. This can affect your Flash Pool aggregate implementation in two ways:

- You can configure your read and write caching policies to ensure optimal performance.
  For most workloads, the default caching policies are optimal. However, for certain workloads, changing the caching policies can result in enhanced performance. For more information about read and write caching policies, see *Technical Report 4070: NetApp Flash Pool Design and Implementation Guide*.
- Some volumes cannot be enabled for write caching.
  When you attempt to use an aggregate associated with one or more of these volumes as a Flash Pool aggregate, you must force the operation. In this case, writes to that volume would not be cached in the SSD tier, but otherwise the Flash Pool aggregate would function normally. You can get more information about why a volume cannot be enabled for write caching by using the `volume show -instance` command.

Related information

TR 4070: NetApp Flash Pool Design and Implementation Guide

Requirements for using Flash Pool aggregates

The Flash Pool technology has some configuration requirements that you should be aware of before planning to use it in your storage architecture.

Flash Pool aggregates cannot be used in the following configurations:

- 32-bit aggregates
- Aggregates composed of array LUNs
- Aggregates that use the ZCS checksum type
- Aggregates that provide storage to Infinite Volumes
- Traditional volumes

You can use Flash Pool aggregates and the Flash Cache module (WAFL external cache) in the same system. However, data stored in a Flash Pool aggregate is not cached in the Flash Cache module. Flash Cache is reserved for data stored in aggregates composed of only HDDs. For more information about Flash Cache and WAFL external cache, see the *Data ONTAP System Administration Guide for Cluster-Mode*.

A Flash Pool aggregate can use either RAID-DP or RAID4 protection (but not both in the same aggregate).

You can use data compression on volumes associated with a Flash Pool aggregate. However, compressed blocks are not cached in the Flash Pool cache for either read or write operations.

Read-only volumes, such as SnapMirror or SnapVault destinations, are not cached in the Flash Pool cache.

If you create a Flash Pool aggregate using an aggregate that was created using Data ONTAP 7.1 or earlier, the volumes associated with that Flash Pool aggregate will not support write caching.

For more information about the types of workloads that benefit from using Flash Pool aggregates, see *Technical Report 4070: NetApp Flash Pool Design and Implementation Guide*.

For Flash Pool support information by platform model, see the *Hardware Universe* (formerly the *System Configuration Guide*) at *support.netapp.com/knowledge/docs/hardware/NetApp/syscfg/index.shtml*.

Related information

TR 4070: NetApp Flash Pool Design and Implementation Guide

How Flash Pool aggregates and Flash Cache compare

Both Flash Pool aggregates and the Flash Cache module provide a high-performance cache to increase storage performance. However, there are differences between the two technologies that you should understand before choosing between them.

You can employ both technologies on the same system. However, data stored in a Flash Pool aggregate (or an SSD aggregate) is not cached by Flash Cache.

Criteria	Flash Pool aggregates	Flash Cache
Scope	A specific aggregate	The entire system
Affected by takeover and giveback?	No. Because the data is still being served from the same aggregate, the performance of the Flash Pool aggregate is unaffected by takeover or giveback.	Yes. Previously cached data is not served from the cache during takeover. For administrator-initiated takeovers, after giveback, the cached data is again served.
Hardware slot required?	No	Yes
Supported with third-party storage?	No	Yes
Supported with Storage Encryption?	No	Yes. Data in the cache is not encrypted.

Criteria	Flash Pool aggregates	Flash Cache
Supported with SnapLock?	No	Yes

For more information about Flash Cache, see the *Data ONTAP System Administration Guide for Cluster-Mode*.

When you cannot use aggregates composed of SSDs

Aggregates composed of SSDs have some restrictions on when they can be used.

You cannot use aggregates composed of SSDs with the following configurations or technologies:

- Infinite Volumes
- Traditional volumes

How you can use disks with mixed speeds in the same aggregate

Whenever possible, you should use disks of the same speed in an aggregate. However, if needed, you can configure Data ONTAP to allow mixed speed aggregates based on the disk type.

To configure Data ONTAP to allow mixed speed aggregates, you use the following options, available through the nodeshell:

- `raid.rpm.fcal.enable`
- `raid.rpm.ata.enable`

When these options are set to `off`, Data ONTAP allows mixing speeds for the designated disk type.

By default, `raid.rpm.fcal.enable` is set to `on`, and `raid.rpm.ata.enable` is set to `off`.

Note: Even if Data ONTAP is not configured to allow mixing speeds, you can still add disks with different speeds to an aggregate setting the `-allow-mixed` parameter of the `storage aggregate add-disks` command to `true`.

For more information about the nodeshell, see the man page for the `system node run` command.

How to control disk selection from heterogeneous storage

When disks with different characteristics coexist on the same node, or when both disks and array LUNs are attached to the same node, the system has heterogeneous storage. When you create an

aggregate from heterogeneous storage, you should take steps to ensure that Data ONTAP uses the disks you expect.

If your node has heterogeneous storage and you do not explicitly specify what type of disks to use, Data ONTAP uses the disk type (including array LUNs) with the highest number of available disks. When you create or add storage to an aggregate using heterogeneous storage, you should use one of the following methods to ensure that Data ONTAP selects the correct disks or disk types:

- Through disk attributes:
  - You can specify disk size by using the `-disksize` option.
    Disks within 20% of the specified size are selected.
  - You can specify disk speed by using the `-diskrpm` option.
  - You can specify disk type by using the `-disktype` option.
- Through an explicit disk list.
  You can list the names of specific disks you want to use.

Note: For unplanned events such as disk failures, which cause Data ONTAP to add another disk to a RAID group automatically, the best way to ensure that Data ONTAP chooses the best disk for any RAID group on your system is to always have at least one spare (and preferably two) available to match all disk types and sizes in use in your system.

Rules for mixing HDD types in aggregates

You can mix disks from different loops or stacks within the same aggregate. Depending on the value of the `raid.disktype.enable` option, you can mix certain types of HDDs within the same aggregate, but some disk type combinations are more desirable than others.

When the `raid.disktype.enable` option is set to `on`, single-tier aggregates can be composed of only one Data ONTAP disk type. This setting ensures that your aggregates are homogeneous and requires that you provide sufficient spare disks for every disk type in use in your system.

The default value for the `raid.disktype.enable` option is `off`, to allow mixing disk types. For this setting, the following Data ONTAP disk types are considered to be equivalent for the purposes of creating and adding to aggregates, and spare management:

- FSAS, BSAS, SATA, and ATA
- FCAL and SAS

To maximize aggregate performance, and for easier storage administration, you should avoid mixing FC-AL-connected and SAS-connected disks in the same aggregate. This is because of the performance mismatch between FC-AL-connected disk shelves and SAS-connected disk shelves. When you mix these connection architectures in the same aggregate, the performance of the aggregate is limited by the presence of the FC-AL-connected disk shelves, even though some of the data is being served from the higher-performing SAS-connected disk shelves.

You can mix the FSAS, BSAS, and SATA disk types without affecting aggregate performance, but mixing the FCAL and SAS disk types, as well as the ATA disk type with FSAS, BSAS, or SATA, is less desirable.

MSATA disks cannot be mixed with any other disk type in the same aggregate.

Note: If you set the `raid.disktype.enable` option to `on` for a system that already contains aggregates with more than one type of HDD, those aggregates continue to function normally and accept both types of HDDs. However, no other aggregates will accept mixed HDD types as long as the `raid.disktype.enable` option is set to `on`.

For information about best practices for working with different types of disks, see *Technical Report 3437: Storage Best Practices and Resiliency Guide*.

Related concepts

How Data ONTAP reports disk types on page 8

Related information

TR 3437: Storage Best Practices and Resiliency Guide

Rules for mixing drive types in Flash Pool aggregates

By definition, Flash Pool aggregates contain more than one drive type. However, the HDD tier follows the same disk-type mixing rules as single-tier aggregates. For example, you cannot mix SAS and SATA disks in the same Flash Pool. The SSD cache can contain only SSDs.

Rules for mixing storage in aggregates for V-Series systems

When planning for aggregates, you must consider the rules for mixing storage in aggregates. You cannot mix different storage types or array LUNs from different vendors or vendor families in the same aggregate.

Adding the following to the same aggregate is not supported:

- Array LUNs and disks
- Array LUNs with different checksum types
- Array LUNs from different drive types (for example, FC and SATA) or different speeds
- Array LUNs from different storage array vendors
- Array LUNs from different storage array model families

Note: Storage arrays in the same family share the same performance and failover characteristics. For example, members of the same family all perform active-active failover, or they all perform active-passive failover. Storage arrays with 4-GB HBAs are not considered to be in the same family as storage arrays with 8-GB HBAs.

How the checksum type is determined for aggregates with array LUNs

Each Data ONTAP aggregate has a checksum type associated with it. The aggregate checksum type is determined by the checksum type of the array LUNs that are added to it.

The checksum type of an aggregate is determined by the checksum type of the first array LUN that is added to the aggregate. The checksum type applies to an entire aggregate (that is, to all volumes in the aggregate). Mixing array LUNs of different checksum types in an aggregate is not supported.

- An array LUN of type *block* must be used with block checksum type aggregates.
- An array LUN of type *zoned* must be used with advanced zoned checksum (AZCS or advanced_zoned) type aggregates.

  Note: Prior to Data ONTAP 8.1.1, zoned checksum array LUNs were used with ZCS (zoned) type aggregates. Starting in 8.1.1, any new aggregates created with zoned checksum array LUNs are AZCS aggregates. You can, however, add zoned checksum array LUNs to existing ZCS aggregates.

Before you add array LUNs to an aggregate, you must know the checksum type of the LUNs you want to add, for the following reasons:

- You cannot add array LUNs of different checksum types to the same aggregate.
- You cannot convert an aggregate from one checksum type to the other.

When you create an aggregate you can specify the number of array LUNs to be added, or you can specify the names of the LUNs to be added. If you want to specify a number of array LUNs to be added to the aggregate, the same number or more array LUNs of that checksum type must be available.

What happens when you add larger disks to an aggregate

What Data ONTAP does when you add disks to an aggregate that are larger than the existing disks depends on the version of Data ONTAP and the RAID level of the aggregate.

For Data ONTAP 8.1 and later, when you add a disk to an aggregate that is larger than the other disks in the aggregate, the new disk is capacity-restricted to be the same size as the smaller disk it replaced, and it is added to the same RAID group a disk of the same size would have been added to.

For earlier versions of Data ONTAP, when you add a larger disk to an aggregate, the result depends on the RAID level of the aggregate:

- When an existing RAID-DP group is assigned an additional disk that is larger than the group's existing dParity disk, then Data ONTAP reassigns the new disk as the regular parity disk for that RAID group and restricts its capacity to be the same size as the existing dParity disk.

Data ONTAP does *not* replace the existing dParity disk, even if the new disk is larger than the dParity disk.

Note: Because the smallest parity disk limits the effective size of disks added to a RAID-DP group, you can maximize available disk space by ensuring that the regular parity disk is as large as the dParity disk.

- When an existing RAID4 group is assigned an additional disk that is larger than the group's existing parity disk, then Data ONTAP reassigns the new disk as parity disk for that RAID group.

Note: If needed, you can replace a capacity-restricted disk with a more suitable (smaller) disk later, to avoid wasting disk space. However, replacing a disk already in use in an aggregate with a larger disk does not result in any additional usable disk space; the new disk is capacity-restricted to be the same size as the smaller disk it replaced.

Related tasks

Replacing disks that are currently being used in an aggregate on page 31

What happens when you add storage to an aggregate

By default, Data ONTAP adds new drives or array LUNs to the most recently created RAID group until it reaches its maximum size. Then Data ONTAP creates a new RAID group. Alternatively, you can specify a RAID group that you want to add storage to.

When you create an aggregate or add storage to an aggregate, Data ONTAP creates new RAID groups as each RAID group is filled with its maximum number of drives or array LUNs. The last RAID group formed might contain fewer drives or array LUNs than the maximum RAID group size for the aggregate. In that case, any storage added to the aggregate is added to the last RAID group until the specified RAID group size is reached.

If you increase the RAID group size for an aggregate, new drives or array LUNs are added only to the most recently created RAID group; the previously created RAID groups remain at their current size unless you explicitly add storage to them.

If you add a drive to a RAID group that is larger than the drives already there, the new drive is capacity-limited to be the same size as the other drives.

Note: You are advised to keep your RAID groups homogeneous when possible. If needed, you can replace a mismatched drive with a more suitable drive later.

Aggregate requirements for an Infinite Volume

A Vserver with Infinite Volume requires at least four 64-bit aggregates composed entirely of HDDs. Understanding the detailed aggregate requirements for an Infinite Volume and its containing Vserver helps you to better use aggregate space.

The following table summarizes the number and purpose of the aggregates required for an Infinite Volume and its containing Vserver:

Number of aggregates	Content	Description
1	Vserver root volume	The Vserver with Infinite Volume requires an aggregate to store the Vserver root volume.
1	Namespace constituent	The Infinite Volume requires a dedicated aggregate to store its namespace constituent.
2 or more	Data constituents	The Infinite Volume creates a data constituent on each additional aggregate that is assigned to its containing Vserver.

When you create aggregates for a Vserver with Infinite Volume, you should keep in mind that you cannot remove aggregates after they are in use.

If you create a Vserver with Infinite Volume by using the `vserver create` command, you should consider the following ways that the aggregate size affects the way the aggregate is used:

- The largest aggregate is automatically used for the namespace constituent when the Infinite Volume is created.
- You should create a small aggregate for the Vserver root volume, and assign that small aggregate to the Vserver root volume when you create the Vserver with Infinite Volume.

  The aggregate that is used for the Vserver root volume contains only the Vserver root volume and nothing else; therefore, it should be much smaller than the aggregates used by the Infinite Volume. For example, a Vserver root volume can be 20 MB.

If you use the Create Vserver wizard in System Manager, aggregates are automatically created according to the following best practices:

- The aggregates are no larger than 100 TB.
- The largest aggregate is used for the namespace constituent.
- The smallest aggregate is used for the Vserver root volume.
- The aggregates are provisioned as RAID-DP to ensure data resiliency and protection.

If you want to use RAID4 disks, you must create the aggregates before you use the Create Vserver wizard in System Manager. You can create the aggregates either by using the Create Aggregate wizard in System Manager or by using the CLI.

If you use the Create Vserver wizard in System Manager to create a Vserver with Infinite Volume that is a SnapMirror destination, aggregates are not created automatically. You must manually create the aggregates according to the aggregate requirements for Infinite Volumes in a SnapMirror relationship.

In addition, each node has an aggregate for its root volume. Node root aggregates are visible when you display a list of all the aggregates in the cluster. For information about node root volumes, see the *Data ONTAP System Administration Guide for Cluster-Mode*.

What namespace and data constituents are

Each Infinite Volume contains a number of constituents. You don't interact with the constituents; you interact with the Infinite Volume, and the Infinite Volume manages its constituents for you. Understanding what constituents exist in an Infinite Volume can help you understand how an Infinite Volume works.

Each Infinite Volume contains the following hidden constituents:

- Namespace constituent
  The namespace constituent tracks the file names for all of the files in the Infinite Volume. A junction connects the namespace constituent to the root volume of the Vserver with Infinite Volume, and the namespace constituent is the exported volume that clients mount to access the directory in an Infinite Volume. The default mount path for an Infinite Volume is `/NS`, but you can customize the mount path as required. However, the mount path must be a single element, such as `/financials`; it cannot be root (`/`) or two elements, such as `/NS/financials`. When clients store and retrieve files in the Infinite Volume, they interact with the namespace constituent inside the Infinite Volume. Each Infinite Volume contains one namespace constituent.
- Data constituents
  Data constituents store data. When clients store files in the namespace constituent, the Infinite Volume leaves the file name of each file in the namespace constituent and moves the data for each file to a data constituent. Clients never interact with data constituents. Clients interact with the namespace constituent, and the Infinite Volume manages the flow of information between the namespace constituent and its data constituents. Each Infinite Volume contains two or more data constituents.

You can view the constituents for an Infinite Volume by using the `volume show` command with the `-constituents` parameter. The storage system requires constituents to have specific names and automatically assigns the required names to constituents when you create an Infinite Volume. As a result, you cannot rename constituents.

How constituents are distributed across aggregates

When you create an Infinite Volume, the storage system uses an algorithm to automatically place the constituents for an Infinite Volume across a number of aggregates. Understanding how the algorithm

selects aggregates for constituents helps you understand how Infinite Volumes use aggregate resources in a cluster.

When you create an Infinite Volume, the algorithm distributes the following constituents among the multiple aggregates available in the cluster:

- The namespace constituent is created on the largest aggregate.
- A data constituent is created on each of the remaining aggregates.

When you add an aggregate to the Vserver with Infinite Volume, a new data constituent is created on the aggregate the next time that you change the size of the Infinite Volume.

Aggregate requirements for Infinite Volumes in a SnapMirror relationship

You must have the same number and size of aggregates for the source Infinite Volume and the destination Infinite Volume for SnapMirror to work.

An Infinite Volume spans several aggregates. Before you can create a data protection mirror for the Infinite Volume, you must create a destination cluster with the following requirements:

- The destination cluster and the source cluster must have the same number of aggregates.
  You must add new aggregates to both the source and destination clusters at the same time. You can add disks to aggregates at any time to increase storage capacity. The additional capacity does not affect the SnapMirror relationship.
- The aggregates in the destination cluster must be the same size or larger than the aggregates in the source cluster.
  You should compare the sizes in KB.

Managing aggregates

You create and manage your aggregates so that they can provide storage to their associated volumes.

Creating an aggregate

You create an aggregate to provide storage to one or more FlexVol volumes, or an Infinite Volume. Aggregates are a physical storage object; they are associated with a specific node in the cluster.

Before you begin

You should know what disks or array LUNs will be used in the new aggregate.

About this task

You can specify disks by listing their IDs, or by specifying a disk characteristic such as type, size, or speed. Disks and array LUNs are owned by a specific node; when you create an aggregate, all disks in that aggregate must be owned by the same node, which becomes the home node for that aggregate.

If your storage system is attached to more than one type of disk, or to both disks and array LUNs, and you do not explicitly specify what type of disks to use, Data ONTAP creates the aggregate using the disk type (including array LUNs) with the highest number of available disks. To ensure that Data ONTAP uses the disk type that you expect, always specify the disk type when creating aggregates from heterogeneous storage.

You can display a list of the available spares by using the `storage disk show -spare` command. This command displays the spare disks for the entire cluster. If you are logged into the cluster on the cluster management interface, you can create an aggregate on any node in the cluster. To ensure that the aggregate is created on a specific node, use the `-node` option or specify disks that are owned by that node.

Aggregate names must conform to the following requirements:

- Begin with either a letter or an underscore (_)
- Contain only letters, digits, and underscores
- Contain no more than 250 characters

Steps

1. Create the aggregate by using the `storage aggregate create` command.

   You can optionally specify the following options:

   - Aggregate's home node (that is, the node on which the aggregate is located unless the aggregate fails over to the node's storage failover partner)
   - List of specific disks or array LUNs that are to be added to the aggregate

- Number of disks to include
- Checksum style to use for the aggregate
- Type of disks to use
- Size of disks to use
- Disk speed to use
- RAID type for RAID groups on the aggregate
- Maximum number of disks or array LUNs that can be included in a RAID group
- Whether disks with different RPM are allowed
- Format of the aggregate (64-bit or 32-bit)

For more information about these options, see the storage aggregate create man page.

2. Verify the RAID group and disks of your new aggregate by entering the following command:
   storage aggregate show -aggregate *aggr_name*

Examples

The following command creates a 64-bit aggregate named aggr2 on node node1 that is composed of 6 SAS disks. This example accepts the default values of 64-bit format, RAID-DP RAID, and RAID group size.

```
storage aggregate create -aggregate aggr2 -node node1 -diskcount 6 -
disktype SAS
```

Creating a Flash Pool aggregate

You create a new Flash Pool aggregate by adding SSDs to an existing 64-bit aggregate composed of HDDs.

Before you begin

- You must have identified a valid 64-bit aggregate composed of HDDs to convert into a Flash Pool aggregate.
- You must have determined write-caching eligibility for the volumes associated with the aggregate, and completed any required steps to resolve eligibility issues.
- You must have determined the SSDs you will be adding, and these SSDs must be owned by the node on which you are creating the Flash Pool aggregate.
- You must have determined the checksum types of both the SSDs you are adding and the HDDs already in the aggregate.

About this task

There are platform- and workload-specific best practices for Flash Pool aggregate SSD tier size and configuration. For information about these best practices, see *Technical Report 4070: NetApp Flash Pool Design and Implementation Guide*.

Steps

1. Mark the aggregate as hybrid-enabled by entering the following command:
   `storage aggregate modify -aggregate aggr_name -hybrid_enabled true`

   If this step does not succeed, repeat the steps to determine write-caching eligibility.
2. Add the SSDs to the aggregate by using the `storage aggregate add-disks` command.

   You can specify the SSDs by ID or by using the `diskcount` and `disktype` parameters. You do not need to specify a new RAID group; Data ONTAP automatically puts the SSDs into their own RAID group.

   If the HDDs and the SSDs do not have the same checksum type, or if the aggregate is a mixed-checksum aggregate, then you must use the `checksumstyle` parameter to specify the checksum type of the disks you are adding to the aggregate.

After you finish

You must ensure that either automatic aggregate Snapshot copy creation is disabled or that automatic aggregate Snapshot copy deletion is enabled for the Flash Pool aggregate. For information about automatic aggregate Snapshot copy creation and deletion, see the *Data ONTAP System Administration Guide for Cluster-Mode*.

Related concepts

How Flash Pool aggregates work on page 78
Requirements for using Flash Pool aggregates on page 79

Related tasks

Determining and enabling volume write-caching eligibility on page 91

Related information

TR 4070: NetApp Flash Pool Design and Implementation Guide

Determining and enabling volume write-caching eligibility

Understanding whether the FlexVol volumes associated with an aggregate are eligible for write caching can help you ensure that the volumes with high performance requirements can get the maximum performance improvement from having their associated aggregate converted to a Flash Pool aggregate.

About this task

Flash Pool aggregates employ two types of caching: *read caching* and *write caching*. Read caching is available for all volumes. Write caching is available for most volumes, but might be disabled for some volumes due to an internal ID collision. You can use this procedure to determine write caching

eligibility to help you decide which aggregates are good candidates to become Flash Pool aggregates. You do not need any SSDs to complete this procedure.

If an aggregate you want to convert to a Flash Pool aggregate is associated with volumes that are ineligible for write caching, you have two choices:

- If you do not need write caching enabled on those volumes, you can go ahead and convert the aggregate to a Flash Pool aggregate by using the force option.
  You do not get the benefit of the Flash Pool aggregate cache for write operations on those volumes.
- If you need write caching enabled on those volumes, you can temporarily move those volumes to another aggregate and move them back, which resolves the ID collision.

Steps

1. Attempt to enable the Flash Pool aggregate capability on the aggregate:
   `storage aggregate modify *aggr_name* -hybrid-enabled true`
2. Take the applicable action based on the result of Step 1:

If...	Then...
The Flash Pool aggregate capability is successfully enabled	Disable the Flash Pool aggregate capability again: **`storage aggregate modify *aggr_name* -hybrid-enabled false`** You have completed this task. All of the volumes in the aggregate are eligible for write caching.
Data ONTAP displays an error message telling you that the aggregate cannot be converted to a Flash Pool aggregate	Determine which volumes are not eligible: **`volume show -volume * -fields hybrid-cache-write-caching-ineligibility-reason -aggregate *aggr_name*`** Each volume in the aggregate is listed, along with its reason for ineligibility if it is ineligible. Eligible volumes display a hyphen (“-”).

3. Your next steps depend on your requirements for the ineligible volumes:

If you...	Then...
Do not need write caching enabled on the ineligible volumes	You have completed this task. You must use the force option when you convert the aggregate to a Flash Pool aggregate.
Need write caching enabled on the ineligible volumes	You must move (or copy and delete) all but one of each set of volumes with the same conflicting ID to another aggregate and then move them back until no more volumes show an ID conflict.

Example with ID collisions

The following example shows the system output when there are ID collisions:

```
clus1::> vol show -volume * -fields hybrid-cache-write-caching-
ineligibility-reason  -aggregate aggr1
  (volume show)
vserver volume   hybrid-cache-write-caching-ineligibility-reason
------- -------- -----------------------------------------------
vs0     root_vs0 -
vs0     vol1     -
vs0     vol2     "ID Collision(27216)"
vs0     vol3     "ID Collision(27216)"
4 entries were displayed.
```

Increasing the size of an aggregate

You can add disks or array LUNs to an aggregate so that it can provide more storage to its associated volumes. If you need to add enough storage to a 32-bit aggregate to increase its size beyond 16 TB, you can do so; this operation expands the aggregate to 64-bit format.

Before you begin

You must understand the following concepts:

- The requirement to add disks or array LUNs owned by the same system
- For aggregates composed of disks:
  - Benefits of keeping your RAID groups homogeneous for disk size and speed.
  - Which types of disks can be used together.
  - Checksum rules when disks of more than one checksum type are in use.
  - How to ensure that the correct disks are added to the aggregate (the disk addition operation cannot be undone).
  - How to add disks to aggregates from heterogeneous storage.
  - The minimum number of disks to add for best performance.
  - The number of hot spares you need to provide for protection against disk failures.
  - Requirements for adding disks from multi-disk carrier disk shelves

About this task

When you add HDDs to an aggregate, you should add a complete RAID group. For information about adding SSDs to a Flash Pool aggregate, see *Technical Report 4070: NetApp Flash Pool Design and Implementation Guide*.

Steps

1. Verify that appropriate spare disks or array LUNs are available for you to add by entering the following command:

   `storage disk show -spare`

For disks, make sure that enough of the spares listed are of the correct type, size, speed, and checksum type for the target RAID group in the aggregate to which you are adding the disks.

2. Add the disks or array LUNs by entering the following command:

   storage aggregate add-disks -aggregate *aggr_name* [-raidgroup *raid_group_name*] *disks*

   If you are adding disks with a different checksum than the aggregate, as when creating a Flash Pool aggregate, or if you are adding disks to a mixed checksum aggregate, you must either specify the disks to be added with a disk list or use the `-checksumstyle` parameter.

   If you are adding disks to a Flash Pool aggregate, you must either specify the disks to be added with a disk list or use the `-disktype` parameter to specify the disk type.

   If you specify the `-raidgroup` parameter, the storage is added to the RAID group you specify. *`raid_group_name`* is the name that Data ONTAP gave to the group—for example, rg0. If you are adding SSDs to the SSD tier of a Flash Pool aggregate, you do not need to specify the RAID group name; the SSD RAID group is selected by default based on the type of the disks you are adding.

   `disks` specifies the disks to be added in one of the following ways:

   - `-diskcount`, usually further qualified by disk type or checksum type
   - `-disklist` *`disk1`* [*`disk2`*...]

3. If the previous step was unsuccessful because you are adding disks to a 32-bit aggregate and the additional disks would cause its size to exceed 16 TB, complete the following steps to expand the aggregate to 64-bit:

   a) Repeat the `storage aggregate add-disks` command you entered before, with the `-64bit-upgrade normal` parameter added.

      Example

      For example, if you entered the **storage aggregate add-disks -diskcount 10 -disktype SAS** command, you would enter the following command:

      storage aggregate add-disks -diskcount 10 -disktype SAS -64bit-upgrade normal

      Data ONTAP checks each volume associated with the aggregate to ensure that it has enough free space to be expanded to 64-bit. If all of the volumes have enough free space, the disks are added and the aggregate is expanded to the 64-bit format. If any of the volumes are too full to be expanded, the command fails.

   b) If the previous command failed, run the command again, replacing the `-64-bit-upgrade normal` parameter with the `-64-bit-upgrade check` parameter. Follow the instructions in the output of that command.

   c) If you had to add more space to any volume, repeat the `storage aggregate -add-disks` command again, this time with the `-64bit-upgrade normal` parameter.

d) If you want to ensure that the disk usage quota accounting for this aggregate is exactly correct, reinitialize quotas on all of its volumes.

If you do not reinitialize quotas, quotas on volumes associated with this aggregate will remain active, but the disk usage accounting will be slightly lower than the actual usage until the next time quotas are reinitialized.

Related concepts

Best practices for expanding a 32-bit aggregate to 64-bit on page 77
What happens when you add larger disks to an aggregate on page 84
How to control disk selection from heterogeneous storage on page 81

Related references

Storage limits on page 99

Related information

TR 4070: NetApp Flash Pool Design and Implementation Guide

Moving an aggregate composed of array LUNs

You might want to move an aggregate composed of array LUNs to a less loaded system to balance the load processing over the systems.

Before you begin

- You should plan the number and size of your aggregates ahead of time so that you have flexibility in the amount of the workload that you can shift from one system to another.
- You should ensure that the *target* system meets the following requirements:
  - The target system must be running a version of Data ONTAP that is the same as or later than the version running on the source system.
  - The target system must support the size of the aggregate being moved.

Steps

1. Enter the following commands on the target system:
   a) Enter the following to access the nodeshell:

      `system run -node` ***`node_name`***

      node_name is the name of the target system.
   b) Obtain the system ID of the target system by entering either of the following commands:

      `disk show`
      or

sysconfig

You need to provide the target system's ID on the source system when you assign each of the array LUNs to the target system.

2. Enter the following commands on the source system:

a) Enter the following command to access the nodeshell:

system run -node *node_name*

node_name is the name of the source system.

b) Enter the following command to display the array LUNs that the aggregate contains:

aggr status *aggr_name* -r

The array LUNs that are displayed are the LUNs that you need to reassign to the target system to be able to move the aggregate.

c) Write down the names of the array LUNs in the aggregate that you want to move.

d) Enter the following command to shut down the source system:

halt

e) At the boot environment prompt, enter the following command to boot the source system:

bye

f) Interrupt the boot process by pressing Ctrl-C when you see the following message on the console:
`Press Ctrl-C for Boot menu`

g) Enter Maintenance mode.

h) When prompted whether you want to continue with booting, enter the following:

y

i) Enter the following command to take the aggregate offline:

aggr offline *aggr_name*

aggr_name is the name of the traditional volume or aggregate.

j) Enter the following and confirm that the aggregate is offline:

aggr status

k) In Maintenance mode, enter the following command *separately* for each array LUN in the aggregate that you are moving to the target system:

disk assign -s *system_id_target* *disk_id* -f

system_id_target is the system ID of the target system (the system to which you want to move the array LUN.)

disk_id is the ID of the array LUN you want to move.

Note: Entering this command automatically removes ownership of the array LUN from the source system and assigns it to the target system.

3. Enter the following commands on the target system.

a) Enter the following command to start a scan so that the target system can recognize the LUNs you moved to it as its own:

disk show

The target system should still be in the nodeshell

b) Enter the following command:

aggr status

The display shows the *foreign* aggregate as offline. (The aggregate you are moving is a foreign aggregate to the target system.) If the foreign aggregate has the same name as an existing aggregate on the system, Data ONTAP renames it *aggr_name(1)*, where *aggr_name* is the original name of the aggregate.

Attention: If the foreign aggregate is incomplete, that is, if you have not moved all the array LUNs in the aggregate, go back to the source system to add the missing array LUNs to the aggregate you moved to the target system. (Enter the following on the source system:

disk assign -s *system_id_target disk_id* -f

c) If Data ONTAP renamed the foreign aggregate because of a name conflict and you want to change the name, enter the following command to rename the aggregate :

aggr rename *aggr_name new_name*

aggr_name is the name of the aggregate you want to rename.

new_name is the new name of the aggregate.

Example

The following command renames the users(1) aggregate as newusers:

```
aggr rename users(1) newusers
```

d) Enter the following command to confirm that the aggregate you moved came online:

aggr status *aggr_name*

aggr_name is the name of the aggregate.

4. On the source system, reboot the system out of Maintenance mode.

Assigning aggregates to a Vserver

If you assign one or more aggregates to a Vserver, then you can use only those aggregates to contain volumes for that Vserver. This can help you keep your Vservers isolated from each other; this is especially important in a multi-tenancy environment.

Before you begin

The Vserver and the aggregates you want to assign to that Vserver must already exist.

Steps

1. Check the list of aggregates already assigned to this Vserver by entering the following command:

   `vserver show -fields aggr-list`

   The aggregates currently assigned to this Vserver are displayed. If there are no aggregates assigned, "-" is displayed.

2. Assign one or more aggregates to a Vserver by entering the following command:

   `vserver modify -vserver` ***`Vserver_name`*** **`-aggr-list`** ***`aggr_name`***

   To assign more than one aggregate to this Vserver, list all of the aggregate names separated by commas.

   Note: If there is already one or more aggregates assigned to this Vserver, and you want those aggregates to continue to be assigned to this Vserver, you must include their names in the list you provide. Otherwise, they will no longer be assigned to this Vserver.

   The aggregates you specified are assigned to the Vserver. If the Vserver already has volumes contained by aggregates that are not assigned to the Vserver, a warning is displayed, but the command succeeds.

Example

The following example assigned the aggregates aggr1 and aggr2 to Vserver vs1:

```
vserver modify -vserver vs1 -aggr-list aggr1,aggr2
```

Related concepts

How the Vserver affects which aggregates can be associated with a FlexVol volume on page 78

Storage limits

There are limits for storage objects that you should consider when planning and managing your storage architecture.

Limits are listed in the following sections:

Aggregate limits

Limit	Native storage	3rd-party storage	Notes
Aggregates Maximum per node	100	100	In an HA configuration, this limit applies to each node individually, so the overall limit for the pair is doubled.
Aggregates (32-bit) Maximum size	16 TB	16 TB	
Aggregates (64-bit) Maximum size	Model-dependent	Model-dependent	See the *Hardware Universe*.
Aggregates Minimum size	RAID-DP: 3 disks RAID4: 2 disks	Model-dependent	See the *V-Series Support Matrix*.
RAID groups Maximum per aggregate	150	150	

RAID group limits

Limit	Native storage	3rd-party storage	Notes
Maximum per system	400	400	
Maximum per aggregate	150	150	

RAID group sizes

RAID type	Default size	Maximum size	Minimum size
RAID-DP	SATA/BSAS/FSAS/ MSATA/ATA: 14 FC/SAS: 16 SSD: 23	SATA/BSAS/FSAS/ MSATA/ATA: 20 FC/SAS: 28 SSD: 28	3
RAID4	SATA/BSAS/FSAS/ MSATA/ATA: 7 FC/SAS/SSD: 8	SATA/BSAS/FSAS/ MSATA/ATA: 7 FC/SAS/SSD: 14	2
RAID0	8	26	1

Copyright information

Trademark information

How to send your comments

You can help us to improve the quality of our documentation by sending us your feedback.

Your feedback is important in helping us to provide the most accurate and high-quality information. If you have suggestions for improving this document, send us your comments by email to *doccomments@netapp.com*. To help us direct your comments to the correct division, include in the subject line the product name, version, and operating system.

You can also contact us in the following ways:

- NetApp, Inc., 495 East Java Drive, Sunnyvale, CA 94089 U.S.
- Telephone: +1 (408) 822-6000
- Fax: +1 (408) 822-4501
- Support telephone: +1 (888) 463-8277

Index

D

R

S

T

U

V

W

Z