ONTAP® 9

Disks and Aggregates Power Guide

n NetApp®

Contents

Deciding whether to use the Disks and Aggregates Power Guide

This guide describes how to manage ONTAP physical storage. It shows you how to create, expand, and manage aggregates, how to work with Flash Pool aggregates, how to manage disks, and how to manage RAID policies.

You should use this guide under the following circumstances:

- You want to use the command-line interface (CLI), not ONTAP System Manager or an automated scripting tool.
- You want to use best practices, not explore every available option.
- You have a MetroCluster configuration and you are following the procedures in the MetroCluster documentation for initial configuration and guidelines for aggregates and disk management.

If you want to use ONTAP System Manager to manage disks and aggregates, you should choose the following documentation:

- *Cluster management using System Manager*

If you require additional configuration or conceptual information, you should choose among the following documentation:

- Conceptual background for disk and aggregate management
  ONTAP concepts
- FlexVol volumes, FlexClone technology, and storage efficiency features
  Logical storage management
- NAS file access

  - *NFS management*
  - *SMB/CIFS management*
- SAN host provisioning

  - *SAN administration*
- MetroCluster documentation

  - *Fabric-attached MetroCluster installation and configuration*
  - *Stretch MetroCluster installation and configuration*
  - *MetroCluster IP installation and configuration*
- Command reference
  ONTAP 9 commands
- Automation of management tasks
  NetApp Documentation: OnCommand Workflow Automation (current releases)

Aggregate creation workflow

Creating aggregates provides storage to volumes on your system. Beginning in ONTAP 9.2, you can let ONTAP recommend aggregate configurations for your system (auto-provision). If the auto-provision method is not available or appropriate in your environment, you can configure aggregates manually.
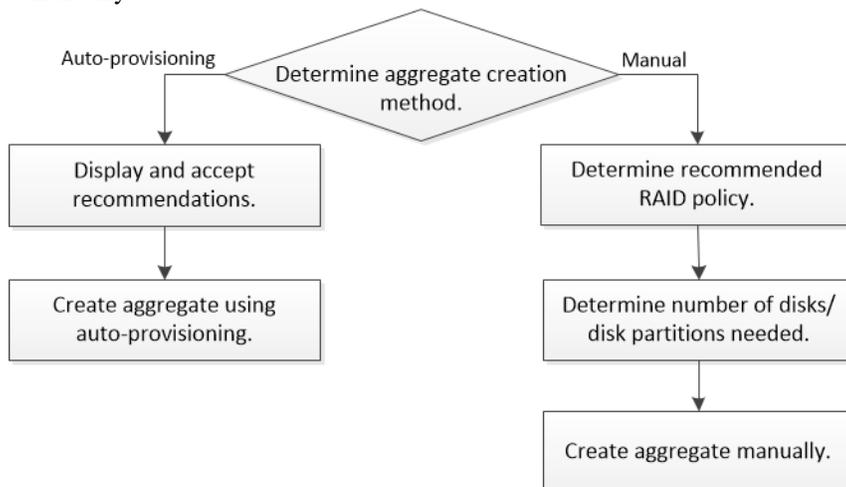
Deciding which aggregate creation method to use

Although aggregate creation with auto-provision is a best practice in ONTAP 9.2 and later, you must determine whether it is supported in your environment. If it is not, you must make decisions about RAID policy and disk configuration, and then create the aggregates manually.

When you create an aggregate using the `storage aggregate auto-provision` command, ONTAP analyzes available spare disks in the cluster and generates a recommendation about how spare disks should be used to create aggregates according to best practices. ONTAP displays the summary of recommended aggregates including their names and usable size, and then prompts you to decide whether the aggregates should be created as recommended.

In many cases, the recommended aggregate layout in the auto-provision display will be optimal for your environment. However, if your cluster is running ONTAP 9.1 or earlier, or your environment includes the following configurations, you must use the manual aggregate configuration method.

- Aggregates using third-party array LUNs
- Virtual disks with Cloud Volumes ONTAP or ONTAP Select
- MetroCluster
- SyncMirror
- MSATA disks
- FlashPool aggregates
- Multiple disk types or sizes are connected to the node

In addition, if any of the following disk conditions are present, they must be addressed before using the auto-provision method:

- Missing disks
- Fluctuation in spare disk numbers
- Unassigned disks
- Non-zeroed spares
- Disks undergoing maintenance testing

The `storage aggregate auto-provision` man page contains more information about these requirements.

Related information
[ONTAP 9 commands](#)

Creating aggregates with auto-provision

If the auto-provision method is appropriate in your environment, you run the `storage aggregate auto-provision` to generate aggregate layout recommendations. You can then create aggregates after reviewing and approving ONTAP recommendations.

Before you begin

ONTAP 9.2 or later must be running on your cluster.

About this task

The default summary generated with the `storage aggregate auto-provision` command lists the recommended aggregates to be created, including names and usable size. You can view the list and determine whether you want to create the recommended aggregates when prompted.

You can also display a detailed summary by using the `-verbose` option, which displays the following reports:

- Per node summary of new aggregates to create, discovered spares, and remaining spare disks and partitions after aggregate creation
- New data aggregates to create with counts of disks and partitions to be used
- RAID group layout showing how spare disks and partitions will be used in new data aggregates to be created
- Details about spare disks and partitions remaining after aggregate creation

If you are familiar with the auto-provision method and your environment is correctly prepared, you can use the `-skip-confirmation` option to create the recommended aggregate without display and confirmation. The `storage aggregate auto-provision` command is not affected by the CLI session `-confirmations` setting.

The `storage aggregate auto-provision` man page contains more information about the aggregate layout recommendations.

Steps

1. Run the `storage aggregate auto-provision` command with the desired display options.

   - no options: Display standard summary
   - `-verbose` option: Display detailed summary
   - `-skip-confirmation` option: Create recommended aggregates without display or confirmation

2. After reviewing the display of recommended aggregates, respond to the prompt to create the recommended aggregates.

   ```
   Do you want to create recommended aggregates? {y|n}:y

   Info: Creating node1_SSD_1 ...
         Creating node2_SSD_1 ...
   ```

Related information
[ONTAP 9 commands](#)

Default RAID policies for aggregates

Either RAID-DP or RAID-TEC is the default RAID policy for all new aggregates. The RAID policy determines the parity protection you have in the event of a disk failure.

RAID-DP provides double-parity protection in the event of a single or double disk failure. RAID-DP is the default RAID policy for the following aggregate types:

* All flash aggregates
* Flash Pool aggregates
* Performance hard disk drive (HDD) aggregates

Beginning with ONTAP 9.0, a new RAID policy called RAID-TEC is available. RAID-TEC is supported on all disk types and all platforms, including AFF. Aggregates that contain larger disks have a higher possibility of concurrent disk failures. RAID-TEC helps to mitigate this risk by providing triple-parity protection so that your data can survive up to three simultaneous disk failures. RAID-TEC is the default RAID policy for capacity HDD aggregates with disks that are 6 TB or larger.

How to determine the number of disks or disk partitions required for an aggregate

You must have enough disks or disk partitions in your aggregate to meet system and business requirements. You should also have the recommended number of hot spare disks or hot spare disk partitions to minimize the potential of data loss.

Root-data partitioning is enabled by default on certain configurations. Systems with root-data partitioning enabled use disk partitions to create aggregates. Systems that do not have root-data partitioning enabled use unpartitioned disks.

You must have enough disks or disk partitions to meet the minimum number required for your RAID policy and enough to meet your minimum capacity requirements.

> **Note:** In ONTAP, the usable space of the drive is less than the physical capacity of the drive. You can find the usable space of a specific drive and the minimum number of disks or disk partitions required for each RAID policy in *Hardware Universe*. You can also use the `storage aggregate show-spare-disks` command to find the usable space of a specific disk.

In addition to the number of disks or disk partitions necessary to create your RAID group and meet your capacity requirements, you should also have the minimum number of hot spare disks or hot spare disk partitions recommended for your aggregate:

* For all flash aggregates, you should have a minimum of one hot spare disk or disk partition.

    > **Note:** The AFF C190 defaults to no spare drive. This exception is fully supported.

* For non-flash homogenous aggregates, you should have a minimum of two hot spare disks or disk partitions.
* For Flash Pool aggregates, you should have a minimum of two hot spare disks or disk partitions for each disk type.
* To support the use of the Maintenance Center and to avoid issues caused by multiple concurrent disk failures, you should have a minimum of four hot spares in multi-disk carriers.

Related information

NetApp Hardware Universe
NetApp Technical Report 3838: Storage Subsystem Configuration Guide

Creating aggregates manually

Before you create aggregates manually, you should review disk configuration options and simulate
creation. Then you can issue the `storage aggregate create` and verify the results.

Before you begin

You must have determined the number of disks and the number of hot spare disks you need in the
aggregate.

About this task

If root-data-data partitioning is enabled and you have 24 solid state drives (SSDs) or fewer in your
configuration, it is recommended that your data partitions be assigned to different nodes.

The procedure for creating aggregates on systems with root-data partitioning and root-data-data
partitioning enabled is the same as the procedure for creating aggregates on systems using
unpartitioned disks. If root-data partitioning is enabled on your system, you should use the number
of disk partitions for the `-diskcount` option. For root-data-data partitioning, the `-diskcount`
option specifies the count of disks to use.

> **Note:** When creating multiple aggregates for use with FlexGroups, aggregates should be as
> close in size as possible.

The `storage aggregate create` man page contains more information about aggregate
creation options and requirements.

Steps

1. View the list of spare disk partitions to verify that you have enough to create your aggregate:

   `storage aggregate show-spare-disks -original-owner node_name`

   Data partitions are displayed under `Local Data Usable`. A root partition cannot be used as a
   spare.

2. Simulate the creation of the aggregate:

   `storage aggregate create -aggregate aggregate_name -node node_name -raidtype raid_dp -diskcount number_of_disks_or_partitions -simulate true`

3. If any warnings are displayed from the simulated command, adjust the command and repeat
   the simulation.

4. Create the aggregate:

   `storage aggregate create -aggregate aggr_name -node node_name -raidtype raid_dp -diskcount number_of_disks_or_partitions`

5. Display the aggregate to verify that it was created:

   `storage aggregate show-status aggregate_name`

   Related information
   ONTAP 9 commands

Aggregate expansion workflow

Expanding an aggregate involves identifying the aggregate to expand, determining how much new storage is needed, installing new disks, assigning disk ownership, and creating new a RAID group if needed.

```
          Identify aggregate
             to expand
                 │
                 ▼
          Identify how much
           new storage is
          needed, compute
          number of disks
                 │
                 ▼
            ◇ Space                    ┌──────────────┐
          available for ──No──────────▶│ Add a disk   │
             disks? ◇                   │    shelf     │
                 │                      └──────┬───────┘
                Yes                            │
                 │                             │
                 ▼                             │
          Add new disks                        │
            to node                            │
                 │                             │
                 ▼                             │
            ◇ Is auto-                 ┌──────────────┐
          assignment of ──No──────────▶│  Manually    │
          disk ownership                │ assign disks │
           enabled? ◇                   └──────┬───────┘
                 │                             │
                Yes                            │
                 │                             │
                 ▼                             │
          Expand aggregate ◀─────────────────┘
```
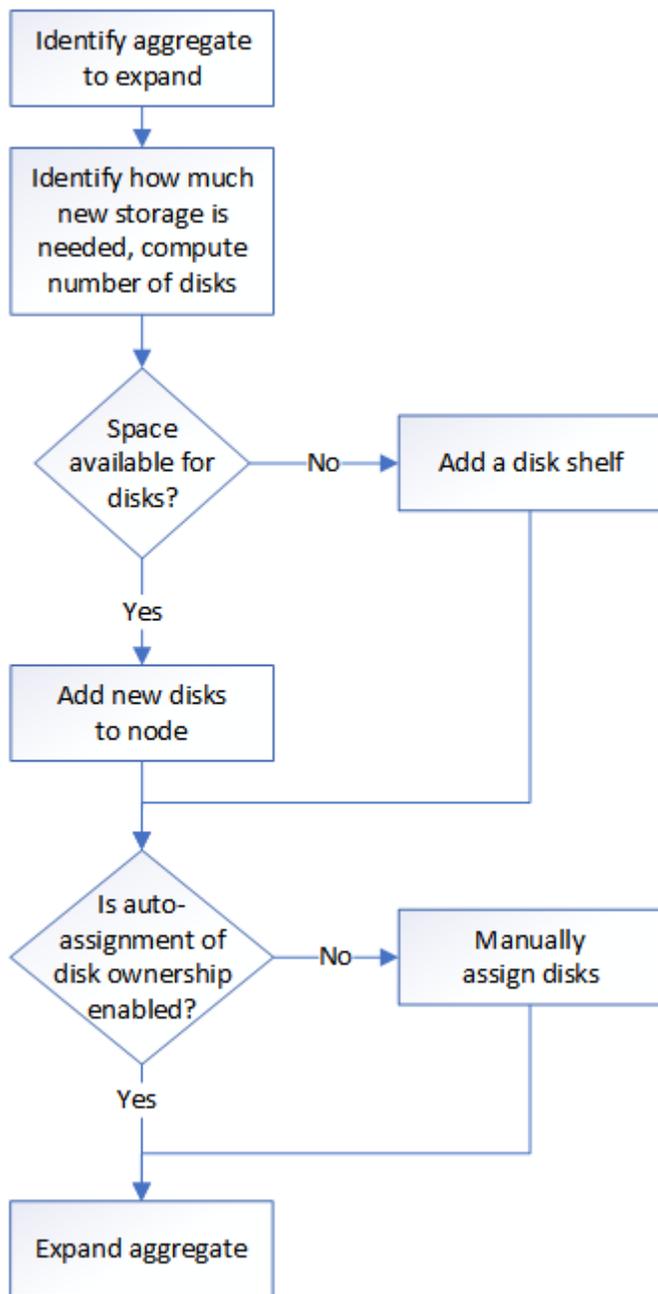
Adding drives to a node or shelf

You add drives to a node or shelf to increase the number of hot spares or to add space to an aggregate.

Before you begin

The drive you want to add must be supported by your platform.

NetApp Hardware Universe

About this task

The minimum number of drives you should add in a single procedure is six. Adding a single drive might reduce performance.

Steps

1. Check the NetApp Support Site for newer drive and shelf firmware and Disk Qualification Package files.

   If your node or shelf does not have the latest versions, update them before installing the new drive.

   Drive firmware is automatically updated (nondisruptively) on new drives that do not have current firmware versions.

2. Properly ground yourself.

3. Gently remove the bezel from the front of the platform.

4. Identify the correct slot for the new drive.

   > **Note:** The correct slots for adding drives vary depending on the platform model and ONTAP version. In some cases you need to add drives to specific slots in sequence. For example, in an AFF A800 you add the drives at specific intervals leaving clusters of empty slots. Whereas, in an AFF A220 you add new drives to the next empty slots running from the outside towards the middle of the shelf.

   See the *NetApp Hardware Universe* to identify the correct slots for your configuration.

5. Insert the new drive:
   a. With the cam handle in the open position, use both hands to insert the new drive.
   b. Push until the drive stops.
   c. Close the cam handle so that the drive is fully seated into the mid plane and the handle clicks into place. Be sure to close the cam handle slowly so that it aligns correctly with the face of the drive.

6. Verify that the drive's activity LED (green) is illuminated.

   When the drive's activity LED is solid, it means that the drive has power. When the drive's activity LED is blinking, it means that the drive has power and I/O is in progress. If the drive firmware is automatically updating, the LED blinks.

7. To add another drive, repeat Steps 4 through 6.

   The new drives are not recognized until they are assigned to a node. You can assign the new drives manually, or you can wait for ONTAP to automatically assign the new drives if your node follows the rules for drive autoassignment.

8. After the new drives have all been recognized, verify their addition and their ownership information:

   `storage aggregate show-spare-disks`
   You should see the new drives, owned by the correct node.

9. Optional: Zero the newly added drives:

storage disk zerospares

Drives that have been used previously in an ONTAP aggregate must be zeroed before they can be added to another aggregate. Zeroing the drives now can prevent delays in case you need to quickly increase the size of an aggregate. The drive zeroing command runs in the background and can take hours to complete, depending on the size of the non-zeroed drives in the node.

Result

The new drives are ready to be added to an aggregate, placed onto the list of hot spares, or you can create a new aggregate.

Related concepts

When you need to update the Disk Qualification Package on page 34
The Disk Qualification Package (DQP) adds full support for newly qualified drives. Before you update drive firmware or add new drive types or sizes to a cluster, you must update the DQP. A best practice is to also update the DQP regularly; for example, every quarter or semi-annually.

How hot spare disks work on page 34
A hot spare disk is a disk that is assigned to a storage system and is ready for use, but is not in use by a RAID group and does not hold any data.

Aggregate creation workflow on page 5
Creating aggregates provides storage to volumes on your system. Beginning in ONTAP 9.2, you can let ONTAP recommend aggregate configurations for your system (auto-provision). If the auto-provision method is not available or appropriate in your environment, you can configure aggregates manually.

When you need to update the Disk Qualification Package on page 34
The Disk Qualification Package (DQP) adds full support for newly qualified drives. Before you update drive firmware or add new drive types or sizes to a cluster, you must update the DQP. A best practice is to also update the DQP regularly; for example, every quarter or semi-annually.

Related tasks

Expanding aggregates on page 12
You can add disks to an aggregate so that it can provide more storage to its associated volumes. The procedure for adding partitioned disks to an aggregate is similar to the procedure for adding unpartitioned disks.

Manually assigning disk ownership

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

Steps

1. Display all unowned disks:

   storage disk show -container-type unassigned

2. Assign each disk:

   storage disk assign -disk *disk_name* -owner *owner_name*

   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

Fast zeroing of drives

Beginning in ONTAP 9.4, you can automatically and quickly zeros drives (both SSDs and HDDs) before provisioning without experiencing long wait times..

For systems that are freshly installed with ONTAP 9.4 or later or systems that are reinitialized with ONTAP 9.4 or later, drive zeroing takes place automatically and is complete in seconds.

If you need to manually zero a drive, you can use one of the following methods:

- Use the `storage disk zerospares` command.

  Admin privileges are required to use this command.
- From the boot menu select one of the following options:

  ◦ `(4) Clean configuration and initialize all disks`
  ◦ `(9a) Unpartition all disks and remove their ownership information`
  ◦ `(9b) Clean configuration and initialize node with whole disks`

The fast zeroing enhancement does not support systems upgraded from a release earlier than ONTAP 9.4.

If any node on the cluster contains an aggregate with fast-zeroed drives, then you cannot revert the cluster to ONTAP 9.2 or earlier.

Expanding aggregates

You can add disks to an aggregate so that it can provide more storage to its associated volumes. The procedure for adding partitioned disks to an aggregate is similar to the procedure for adding unpartitioned disks.

Before you begin

You must know what the RAID group size is for the aggregate you are adding the storage to.

About this task

When you expand an aggregate, you should be aware of whether you are adding partition or unpartitioned disks to the aggregate. When you add unpartitioned drives to an existing aggregate, the size of the existing RAID groups is inherited by the new RAID group, which can affect the number of parity disks required. If an unpartitioned disk is added to a RAID group composed of partitioned disks, the new disk is partitioned, leaving an unused spare partition.

When you provision partitions, you must ensure that you do not leave the node without a drive with both partitions as spare. If you do, and the node experiences a controller disruption, valuable information about the problem (the core file) might not be available to provide to the technical support.

> **Note:** Do not use the `disklist` command to expand your aggregates. This could cause partition misalignment.

Steps

1. Show the available spare storage on the system that owns the aggregate:

   storage aggregate show-spare-disks -original-owner *node_name*

   You can use the `-is-disk-shared` parameter to show only partitioned drives or only unpartitioned drives.

   ```
   cl1-s2::> storage aggregate show-spare-disks -original-owner cl1-s2 -is-disk-shared true

   Original Owner: cl1-s2
    Pool0
     Shared HDD Spares
                                                     Local     Local
                                                     Data    Root Physical
   ```

```
Disk                      Type   RPM Checksum        Usable   Usable    Size Status
------------------------- ----- ------ -------------- -------- -------- -------- --------
1.0.1                     BSAS   7200 block            753.8GB  73.89GB  828.0GB zeroed
1.0.2                     BSAS   7200 block            753.8GB       0B  828.0GB zeroed
1.0.3                     BSAS   7200 block            753.8GB       0B  828.0GB zeroed
1.0.4                     BSAS   7200 block            753.8GB       0B  828.0GB zeroed
1.0.8                     BSAS   7200 block            753.8GB       0B  828.0GB zeroed
1.0.9                     BSAS   7200 block            753.8GB       0B  828.0GB zeroed
1.0.10                    BSAS   7200 block                 0B  73.89GB  828.0GB zeroed
2 entries were displayed.
```

2. Show the current RAID groups for the aggregate:

storage aggregate show-status *aggr_name*

```
cl1-s2::> storage aggregate show-status -aggregate data_1

Owner Node: cl1-s2
 Aggregate: data_1 (online, raid_dp) (block checksums)
  Plex: /data_1/plex0 (online, normal, active, pool0)
   RAID Group /data_1/plex0/rg0 (normal, block checksums)
                                                  Usable Physical
    Position Disk                     Pool Type   RPM   Size    Size Status
    -------- ------------------------- ---- ----- ------ -------- -------- ----------
    shared   1.0.10                      0  BSAS   7200  753.8GB  828.0GB (normal)
    shared   1.0.5                       0  BSAS   7200  753.8GB  828.0GB (normal)
    shared   1.0.6                       0  BSAS   7200  753.8GB  828.0GB (normal)
    shared   1.0.11                      0  BSAS   7200  753.8GB  828.0GB (normal)
    shared   1.0.0                       0  BSAS   7200  753.8GB  828.0GB (normal)
5 entries were displayed.
```

3. Simulate adding the storage to the aggregate:

**storage aggregate add-disks -aggregate *aggr_name* -diskcount
number_of_disks_or_partitions -simulate true**

You can see the result of the storage addition without actually provisioning any storage. If any
warnings are displayed from the simulated command, you can adjust the command and repeat
the simulation.

```
cl1-s2::> storage aggregate add-disks data_1 -diskcount 5 -simulate true

Addition of disks would succeed for aggregate "data_1" on node "cl1-s2". The
following disks would be used to add to the aggregate: 1.0.2, 1.0.3, 1.0.4, 1.0.8, 1.0.9.
```

4. Add the storage to the aggregate:

**storage aggregate add-disks -aggregate *aggr_name* -raidgroup new -diskcount
*number_of_disks_or_partitions***

When creating a Flash Pool aggregate, if you are adding disks with a different checksum than
the aggregate, or if you are adding disks to a mixed checksum aggregate, you must use the -
checksumstyle parameter.

If you are adding disks to a Flash Pool aggregate, you must use the -disktype parameter to
specify the disk type.

You can use the -disksize parameter to specify a size of the disks to add. Only disks with
approximately the specified size are selected for addition to the aggregate.

```
cl1-s2::> storage aggregate add-disks -aggregate data_1 -raidgroup new -diskcount 5
```

5. Verify that the storage was added successfully:

storage aggregate show-status -aggregate *aggr_name*

```
cl1-s2::> storage aggregate show-status -aggregate data_1

Owner Node: cl1-s2
 Aggregate: data_1 (online, raid_dp) (block checksums)
  Plex: /data_1/plex0 (online, normal, active, pool0)
   RAID Group /data_1/plex0/rg0 (normal, block checksums)
                                                  Usable Physical
    Position Disk                     Pool Type   RPM   Size    Size Status
    -------- ------------------------- ---- ----- ------ -------- -------- ----------
    shared   1.0.10                      0  BSAS   7200  753.8GB  828.0GB (normal)
    shared   1.0.5                       0  BSAS   7200  753.8GB  828.0GB (normal)
    shared   1.0.6                       0  BSAS   7200  753.8GB  828.0GB (normal)
    shared   1.0.11                      0  BSAS   7200  753.8GB  828.0GB (normal)
    shared   1.0.0                       0  BSAS   7200  753.8GB  828.0GB (normal)
    shared   1.0.2                       0  BSAS   7200  753.8GB  828.0GB (normal)
    shared   1.0.3                       0  BSAS   7200  753.8GB  828.0GB (normal)
    shared   1.0.4                       0  BSAS   7200  753.8GB  828.0GB (normal)
    shared   1.0.8                       0  BSAS   7200  753.8GB  828.0GB (normal)
    shared   1.0.9                       0  BSAS   7200  753.8GB  828.0GB (normal)
10 entries were displayed.
```

6. Verify that the node still has at least one drive with both the root partition and the data partition as spare:

`storage aggregate show-spare-disks -original-owner *node_name*`

```
cl1-s2::> storage aggregate show-spare-disks -original-owner cl1-s2 -is-disk-shared true

Original Owner: cl1-s2
 Pool0
  Shared HDD Spares
                                                     Local    Local
                                                      Data     Root Physical
 Disk                      Type    RPM Checksum      Usable   Usable     Size Status
 ------------------------- ----- ------ -------------- -------- -------- -------- --------
 1.0.1                     BSAS   7200 block          753.8GB  73.89GB  828.0GB zeroed
 1.0.10                    BSAS   7200 block              0B  73.89GB  828.0GB zeroed
 2 entries were displayed.
```

Managing aggregates

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

RAID protection levels for disks

ONTAP supports three levels of RAID protection for aggregates. Your level of RAID protection determines the number of parity disks available for data recovery in the event of disk failures.

With RAID protection, if there is a data disk failure in a RAID group, ONTAP can replace the failed disk with a spare disk and use parity data to reconstruct the data of the failed disk.

- RAID4
  With RAID4 protection, ONTAP can use one spare disk to replace and reconstruct the data from one failed disk within the RAID group.
- RAID-DP
  With RAID-DP protection, ONTAP can use up to two spare disks to replace and reconstruct the data from up to two simultaneously failed disks within the RAID group.
- RAID-TEC
  With RAID-TEC protection, ONTAP can use up to three spare disks to replace and reconstruct the data from up to three simultaneously failed disks within the RAID group.

Related information

NetApp Technical Report 3437: Storage Subsystem Resiliency Guide

How to determine the number of disks or disk partitions required for an aggregate

You must have enough disks or disk partitions in your aggregate to meet system and business requirements. You should also have the recommended number of hot spare disks or hot spare disk partitions to minimize the potential of data loss.

Root-data partitioning is enabled by default on certain configurations. Systems with root-data partitioning enabled use disk partitions to create aggregates. Systems that do not have root-data partitioning enabled use unpartitioned disks.

You must have enough disks or disk partitions to meet the minimum number required for your RAID policy and enough to meet your minimum capacity requirements.

Note: In ONTAP, the usable space of the drive is less than the physical capacity of the drive. You can find the usable space of a specific drive and the minimum number of disks or disk partitions required for each RAID policy in *Hardware Universe*. You can also use the `storage aggregate show-spare-disks` command to find the usable space of a specific disk.

In addition to the number of disks or disk partitions necessary to create your RAID group and meet your capacity requirements, you should also have the minimum number of hot spare disks or hot spare disk partitions recommended for your aggregate:

- For all flash aggregates, you should have a minimum of one hot spare disk or disk partition.

  Note: The AFF C190 defaults to no spare drive. This exception is fully supported.

- For non-flash homogenous aggregates, you should have a minimum of two hot spare disks or disk partitions.
- For Flash Pool aggregates, you should have a minimum of two hot spare disks or disk partitions for each disk type.

- To support the use of the Maintenance Center and to avoid issues caused by multiple concurrent disk failures, you should have a minimum of four hot spares in multi-disk carriers.

    Related information

    NetApp Hardware Universe
    NetApp Technical Report 3838: Storage Subsystem Configuration Guide

Correcting misaligned spare partitions

When you add partitioned disks to an aggregate, you must leave a disk with both the root and data partition available as spare for every node. If you do not and your node experiences a disruption, ONTAP cannot dump the core to the spare data partition.

Before you begin

You must have both a spare data partition and a spare root partition on the same type of disk owned by the same node.

Steps

1. Display the spare partitions for the node:

    `storage aggregate show-spare-disks -original-owner node_name`

    Note which disk has a spare data partition (spare_data) and which disk has a spare root partition (spare_root). The spare partition will show a non-zero value under the `Local Data Usable` or `Local Root Usable` column.

2. Replace the disk with a spare data partition with the disk with the spare root partition:

    `storage disk replace -disk spare_data -replacement spare_root -action start`

    You can copy the data in either direction; however, copying the root partition takes less time to complete.

3. Monitor the progress of the disk replacement:

    `storage aggregate show-status -aggregate aggr_name`

4. After the replacement operation is complete, display the spares again to confirm that you have a full spare disk:

    `storage aggregate show-spare-disks -original-owner node_name`
    You should see a spare disk with usable space under both `Local Data Usable` and `Local Root Usable`.

Example

You display your spare partitions for node c1-01 and see that your spare partitions are not aligned:

```
c1::> storage aggregate show-spare-disks -original-owner c1-01

Original Owner: c1-01
 Pool0
  Shared HDD Spares
                                                      Local    Local
                                                       Data     Root Physical
Disk                      Type    RPM Checksum        Usable   Usable     Size
------------------------- ----- ------ -------------- -------- -------- --------
1.0.1                     BSAS   7200 block           753.8GB       0B   828.0GB
1.0.10                    BSAS   7200 block               0B   73.89GB   828.0GB
```

You start the disk replacement job:

```
c1::> storage disk replace -disk 1.0.1 -replacement 1.0.10 -action start
```

While you are waiting for the replacement operation to finish, you display the progress of the operation:

```
c1::> storage aggregate show-status -aggregate aggr0_1

Owner Node: c1-01
```

```
Aggregate: aggr0_1 (online, raid_dp) (block checksums)
 Plex: /aggr0_1/plex0 (online, normal, active, pool0)
  RAID Group /aggr0_1/plex0/rg0 (normal, block checksums)
                                              Usable Physical
    Position Disk                     Pool Type    RPM   Size     Size Status
    -------- ------------------------- ---- ----- ----- -------- -------- ----------
    shared   1.0.1                       0  BSAS  7200  73.89GB  828.0GB (replacing, copy in progress)
    shared   1.0.10                      0  BSAS  7200  73.89GB  828.0GB (copy 63% completed)
    shared   1.0.0                       0  BSAS  7200  73.89GB  828.0GB (normal)
    shared   1.0.11                      0  BSAS  7200  73.89GB  828.0GB (normal)
    shared   1.0.6                       0  BSAS  7200  73.89GB  828.0GB (normal)
    shared   1.0.5                       0  BSAS  7200  73.89GB  828.0GB (normal)
```

After the replacement operation is complete, you confirm that you have a full spare disk:

```
ie2220::> storage aggregate show-spare-disks -original-owner c1-01

Original Owner: c1-01
 Pool0
  Shared HDD Spares
                                                   Local    Local
                                                    Data    Root Physical
Disk                       Type    RPM Checksum    Usable  Usable    Size
-------------------------- ----- ------ ------------- -------- -------- --------
1.0.1                      BSAS   7200 block         753.8GB 73.89GB  828.0GB
```

Determining drive and RAID group information for an aggregate

Some aggregate administration tasks require that you know what types of drives compose the aggregate, their size, checksum, and status, whether they are shared with other aggregates, and the size and composition of the RAID groups.

Step

Show the drives for the aggregate, by RAID group:

storage aggregate show-status *aggr_name*
The drives are displayed for each RAID group in the aggregate.

You can see the RAID type of the drive (data, parity, dparity) in the `Position` column. If the `Position` column displays `shared`, then the drive is shared: if it is an HDD, it is a partitioned disk; if it is an SSD, it is part of a storage pool.

Example: A Flash Pool aggregate using an SSD storage pool and data partitions

```
cluster1::> storage aggregate show-status nodeA_fp_1

Owner Node: cluster1-a
 Aggregate: nodeA_fp_1 (online, mixed_raid_type, hybrid) (block checksums)
  Plex: /nodeA_fp_1/plex0 (online, normal, active, pool0)
   RAID Group /nodeA_fp_1/plex0/rg0 (normal, block checksums, raid_dp)
                                              Usable Physical
    Position Disk                     Pool Type    RPM    Size     Size Status
    -------- ------------------------- ---- ----- ----- -------- -------- -------
    shared   2.0.1                       0  SAS   10000 472.9GB  547.1GB (normal)
    shared   2.0.3                       0  SAS   10000 472.9GB  547.1GB (normal)
    shared   2.0.5                       0  SAS   10000 472.9GB  547.1GB (normal)
    shared   2.0.7                       0  SAS   10000 472.9GB  547.1GB (normal)
    shared   2.0.9                       0  SAS   10000 472.9GB  547.1GB (normal)
    shared   2.0.11                      0  SAS   10000 472.9GB  547.1GB (normal)

   RAID Group /nodeA_flashpool_1/plex0/rg1 (normal, block checksums, raid4) (Storage Pool: SmallSP)
                                              Usable Physical
    Position Disk                     Pool Type    RPM   Size     Size Status
    -------- ------------------------- ---- ----- ----- -------- -------- -------
    shared   2.0.13                      0  SSD     -   186.2GB  745.2GB (normal)
    shared   2.0.12                      0  SSD     -   186.2GB  745.2GB (normal)
8 entries were displayed.
```

Relocating aggregate ownership within an HA pair

You can change the ownership of aggregates among the nodes in an HA pair without interrupting service from the aggregates.

Both nodes in an HA pair are physically connected to each other's disks or array LUNs. Each disk or array LUN is owned by one of the nodes. Although ownership of disks temporarily changes when a takeover occurs, the aggregate relocation operations either permanently (for example, if

done for load balancing) or temporarily (for example, if done as part of takeover) change the ownership of all disks or array LUNs within an aggregate from one node to the other. The ownership changes without any data-copy processes or physical movement of the disks or array LUNs.

Relocating aggregate ownership

You can change the ownership of an aggregate only between the nodes within an HA pair.

About this task

- Because volume count limits are validated programmatically during aggregate relocation operations, it is not necessary to check for this manually.

  If the volume count exceeds the supported limit, the aggregate relocation operation fails with a relevant error message.

- You should not initiate aggregate relocation when system-level operations are in progress on either the source or the destination node; likewise, you should not start these operations during the aggregate relocation.

  These operations can include the following:

  - Takeover
  - Giveback
  - Shutdown
  - Another aggregate relocation operation
  - Disk ownership changes
  - Aggregate or volume configuration operations
  - Storage controller replacement
  - ONTAP upgrade
  - ONTAP revert

- If you have a MetroCluster configuration, you should not initiate aggregate relocation while disaster recovery operations (*switchover*, *healing*, or *switchback*) are in progress.

- If you have a MetroCluster configuration and initiate aggregate relocation on a switched-over aggregate, the operation might fail because it exceeds the DR partner's volume limit count.

- You should not initiate aggregate relocation on aggregates that are corrupt or undergoing maintenance.

- Before initiating the aggregate relocation, you should save any core dumps on the source and destination nodes.

Steps

1. View the aggregates on the node to confirm which aggregates to move and ensure they are online and in good condition:

   storage aggregate show -node *source-node*

   The following command shows six aggregates on the four nodes in the cluster. All aggregates are online. Node1 and Node3 form an HA pair and Node2 and Node4 form an HA pair.

   ```
   cluster::> storage aggregate show
   Aggregate     Size Available Used% State    #Vols  Nodes  RAID Status
   --------- -------- --------- ----- ------- ------ ------ -----------
   aggr_0    239.0GB   11.13GB   95% online       1 node1  raid_dp,
                                                            normal
   aggr_1    239.0GB   11.13GB   95% online       1 node1  raid_dp,
                                                            normal
   aggr_2    239.0GB   11.13GB   95% online       1 node2  raid_dp,
                                                            normal
   aggr_3    239.0GB   11.13GB   95% online       1 node2  raid_dp,
                                                            normal
   aggr_4    239.0GB   238.9GB    0% online       5 node3  raid_dp,
   ```

```
                                                         normal
aggr_5      239.0GB    239.0GB     0% online       4 node4 raid_dp,
                                                         normal
6 entries were displayed.
```

2. Issue the command to start the aggregate relocation:

   storage aggregate relocation start -aggregate-list *aggregate-1, aggregate-2... -node source-node* -destination *destination-node*

   The following command moves the aggregates aggr_1 and aggr_2 from Node1 to Node3.
   Node3 is Node1's HA partner. The aggregates can be moved only within the HA pair.

   ```
   cluster::> storage aggregate relocation start -aggregate-list aggr_1, aggr_2 -node
   node1 -destination node3
   Run the storage aggregate relocation show command to check relocation status.
   node1::storage aggregate>
   ```

3. Monitor the progress of the aggregate relocation with the `storage aggregate relocation show` command:

   storage aggregate relocation show -node *source-node*

   The following command shows the progress of the aggregates that are being moved to Node3:

   ```
   cluster::> storage aggregate relocation show -node node1
   Source  Aggregate    Destination    Relocation Status
   ------  -----------  -------------  -----------------------
   node1
           aggr_1       node3          In progress, module: wafl
           aggr_2       node3          Not attempted yet
   2 entries were displayed.
   node1::storage aggregate>
   ```

   When the relocation is complete, the output of this command shows each aggregate with a
   relocation status of `Done`.

   ### Related concepts

   Commands for aggregate relocation on page 19
   There are specific ONTAP commands for relocating aggregate ownership within an HA pair.

Commands for aggregate relocation

There are specific ONTAP commands for relocating aggregate ownership within an HA pair.

If you want to...	Use this command...
Start the aggregate relocation process	`storage aggregate relocation start`
Monitor the aggregate relocation process	`storage aggregate relocation show`

Related concepts

Relocating aggregate ownership on page 18
You can change the ownership of an aggregate only between the nodes within an HA pair.

Related information

ONTAP 9 commands

Assigning aggregates to SVMs

If you assign one or more aggregates to a storage virtual machine (SVM, formerly known as
Vserver), then you can use only those aggregates to contain volumes for that SVM. Assigning
aggregates to your SVMs is particularly important in a multi-tenancy environment.

Before you begin

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

About this task

Assigning aggregates to your SVMs helps you keep your SVMs isolated from each other; this is especially important in a multi-tenancy environment..

Steps

1. Check the list of aggregates already assigned to the SVM:

   `vserver show -fields aggr-list`
   The aggregates currently assigned to the SVM are displayed. If there are no aggregates assigned, "-" is displayed.

2. Add or remove assigned aggregates, depending on your requirements:

If you want to...	Use this command...
Assign additional aggregates	`vserver add-aggregates`
Unassign aggregates	`vserver remove-aggregates`

   The listed aggregates are assigned to or removed from the SVM. If the SVM already has volumes that use an aggregate that is not assigned to the SVM, a warning message is displayed, but the command is completed successfully. Any aggregates that were already assigned to the SVM and that were not named in the command are unaffected.

Example

In the following example, the aggregates aggr1 and aggr2 are assigned to SVM svm1:

`vserver add-aggregates -vserver svm1 -aggregates aggr1,aggr2`

How to determine space usage in an aggregate

You can view space usage by all volumes in one or more aggregates with the `aggregate show-space` command. This helps you see which volumes are consuming the most space in their containing aggregates so that you can take actions to free more space.

The used space in an aggregate is directly affected by the space used in the FlexVol volumes it contains. Measures that you take to increase space in a volume also affect space in the aggregate.

The following rows are included in the `aggregate show-space` command output:

- `Volume Footprints`

  The total of all volume footprints within the aggregate. It includes all of the space that is used or reserved by all data and metadata of all volumes in the containing aggregate.

- `Aggregate Metadata`

  The total file system metadata required by the aggregate, such as allocation bitmaps and inode files.

- `Snapshot Reserve`

  The amount of space reserved for aggregate Snapshot copies, based on volume size. It is considered used space and is not available to volume or aggregate data or metadata.

- `Snapshot Reserve Unusable`

  The amount of space originally allocated for aggregate Snapshot reserve that is unavailable for aggregate Snapshot copies because it is being used by volumes associated with the aggregate. Can occur only for aggregates with a non-zero aggregate Snapshot reserve.

- `Total Used`

  The sum of all space used or reserved in the aggregate by volumes, metadata, or Snapshot copies.

- `Total Physical Used`

  The amount of space being used for data now (rather than being reserved for future use).
  Includes space used by aggregate Snapshot copies.

The following example shows the `aggregate show-space` command output for an aggregate
whose Snapshot reserve is 5%. If the Snapshot reserve was 0, the row would not be displayed.

```
cluster1::> storage aggregate show-space

                      Aggregate : wqa_gx106_aggr1

    Feature                                   Used       Used%
    ------------------------------         ----------    ------
    Volume Footprints                         101.0MB        0%
    Aggregate Metadata                          300KB        0%
    Snapshot Reserve                          5.98GB         5%

    Total Used                                6.07GB         5%
    Total Physical Used                       34.82KB        0%
```

Determining which volumes reside on an aggregate

You might need to determine which volumes reside on an aggregate before performing operations
on the aggregate, such as relocating it or taking it offline.

Step

To display the volumes that reside on an aggregate, enter

`volume show -aggregate aggregate_name`
.:
All volumes that reside on the specified aggregate are displayed.

How you can determine and control a volume's space usage in the aggregate

You can determine which FlexVol volumes are using the most space in the aggregate and
specifically which features within the volume. The `volume show-footprint` command
provides information about a volume's footprint, or its space usage within the containing
aggregate.

The `volume show-footprint` command shows details about the space usage of each volume in
an aggregate, including offline volumes. This command bridges the gap between the output of the
`volume show-space` and `aggregate show-space` commands. All percentages are calculated
as a percent of aggregate size.

The following example shows the `volume show-footprint` command output for a volume
called testvol:

```
cluster1::> volume show-footprint testvol

    Vserver : thevs
    Volume  : testvol

    Feature                                   Used       Used%
    ------------------------------         ----------    -----
    Volume Data Footprint                     120.6MB        4%
    Volume Guarantee                          1.88GB        71%
    Flexible Volume Metadata                  11.38MB        0%
    Delayed Frees                             1.36MB         0%
    Total Footprint                           2.01GB        76%
```

The following table explains some of the key rows of the output of the `volume show-footprint` command and what you can do to try to decrease space usage by that feature:

Row/feature name	Description/contents of row	Some ways to decrease
`Volume Data Footprint`	The total amount of space used in the containing aggregate by a volume's data in the active file system and the space used by the volume's Snapshot copies. This row does not include reserved space.	• Deleting data from the volume. • Deleting Snapshot copies from the volume.
`Volume Guarantee`	The amount of space reserved by the volume in the aggregate for future writes. The amount of space reserved depends on the guarantee type of the volume.	Changing the type of guarantee for the volume to **none**.
`Flexible Volume Metadata`	The total amount of space used in the aggregate by the volume's metadata files.	No direct method to control.
`Delayed Frees`	Blocks that ONTAP used for performance and cannot be immediately freed. For SnapMirror destinations, this row has a value of **0** and is not displayed.	No direct method to control.
`File Operation Metadata`	The total amount of space reserved for file operation metadata.	No direct method to control.
`Total Footprint`	The total amount of space that the volume uses in the aggregate. It is the sum of all of the rows.	Any of the methods used to decrease space used by a volume.

Related information

NetApp Technical Report 3483: Thin Provisioning in a NetApp SAN or IP SAN Enterprise Environment

Methods to create space in an aggregate

If an aggregate runs out of free space, various problems can result that range from loss of data to disabling a volume's guarantee. There are multiple ways to make more space in an aggregate.

All of the methods have various consequences. Prior to taking any action, you should read the relevant section in the documentation.

The following are some common ways to make space in an aggregate, in order of least to most consequences:

- Add disks to the aggregate.
- Move some volumes to another aggregate with available space.
- Shrink the size of volume-guaranteed volumes in the aggregate.
  You can do this manually or with the `autoshrink` option of the autosize capability.
- Change volume guarantee types to **none** on volumes that are using large amounts of space (large volume-guaranteed volumes with large reserved files) so that the volumes take up less space in the aggregate.
  A volume with a guarantee type of **none** has a smaller footprint in the aggregate than a volume with a guarantee type of **volume**.
- Delete unneeded volume Snapshot copies if the volume's guarantee type is **none**.
- Delete unneeded volumes.
- Enable space-saving features, such as deduplication or compression.
- (Temporarily) disable features that are using a large amount of metadata .

Commands for managing aggregates

You use the `storage aggregate` command to manage your aggregates.

If you want to...	Use this command...
Display the size of the cache for all Flash Pool aggregates	`storage aggregate show -fields hybrid-cache-size-total -hybrid-cache-size-total >0`
Display disk information and status for an aggregate	`storage aggregate show-status`
Display spare disks by node	`storage aggregate show-spare-disks`
Display the root aggregates in the cluster	`storage aggregate show -has-mroot true`
Display basic information and status for aggregates	`storage aggregate show`
Display the type of storage used in an aggregate	`storage aggregate show -fields storage-type`
Bring an aggregate online	`storage aggregate online`
Delete an aggregate	`storage aggregate delete`
Put an aggregate into the restricted state	`storage aggregate restrict`
Rename an aggregate	`storage aggregate rename`
Take an aggregate offline	`storage aggregate offline`
Change the RAID type for an aggregate	`storage aggregate modify -raidtype`

Related information

ONTAP 9 commands

Working with Flash Pool aggregates

You configure a Flash Pool aggregate by adding SSD RAID groups to an existing HDD aggregate. The SSD RAID groups can be composed of physical SSDs or allocation units from SSD storage pools.

Flash Pool caching policies and SSD partitioning

Caching policies for the volumes in a Flash Pool aggregate let you deploy flash as high performance cache for your working data set while using lower-cost HDDs for less frequently accessed data. If you are providing cache to two or more Flash Pool aggregates, you should use Flash Pool SSD partitioning to share SSDs across the aggregates in the Flash Pool.

How Flash Pool aggregate caching policies work

Caching policies are applied to volumes that reside in Flash Pool aggregates. You should understand how caching policies work before changing them.

In most cases, the default caching policy of `auto` is the best caching policy to use. The caching policy should be changed only if a different policy provides better performance for your workload. Configuring the wrong caching policy can severely degrade volume performance; the performance degradation could increase gradually over time.

Caching policies combine a read caching policy and a write caching policy. The policy name concatenates the names of the read caching policy and the write caching policy, separated by a hyphen. If there is no hyphen in the policy name, the write caching policy is "none", except for the `auto` policy.

Read caching policies optimize for future read performance by placing a copy of the data in the cache in addition to the stored data on HDDs. For read caching policies that insert data into the cache for write operations, the cache operates as a *write-through* cache.

Data inserted into the cache by using the write caching policy exists only in cache; there is no copy in HDDs. Flash Pool cache is RAID protected. Enabling write caching makes data from write operations available for reads from cache immediately, while deferring writing the data to HDDs until it ages out of the cache.

You can change the caching policy for a volume that resides on a Flash Pool aggregate by using the -caching-policy parameter with the volume create command. When you create a volume on a Flash Pool aggregate, by default, the `auto` caching policy is assigned to the volume.

If you move a volume from a Flash Pool aggregate to a single-tier aggregate, it loses its caching policy; if you later move it back to a Flash Pool aggregate, it is assigned the default caching policy of `auto`. If you move a volume between two Flash Pool aggregates, the caching policy is preserved.

### How Flash Pool SSD partitioning works for Flash Pool aggregates using storage pools

If you are providing cache to two or more Flash Pool aggregates, you should use Flash Pool Solid-State Drive (SSD) partitioning. Flash Pool SSD partitioning allows SSDs to be shared by all the aggregates using the Flash Pool. This spreads the cost of parity over multiple aggregates, increases SSD cache allocation flexibility, and maximizes SSD performance.

For an SSD to be used in a Flash Pool aggregate, the SSD must be placed in a storage pool. You cannot use SSDs that have been partitioned for root-data partitioning in a storage pool. After the SSD is placed in the storage pool, the SSD can no longer be managed as a stand-alone disk and cannot be removed from the storage pool unless you destroy the aggregates associated with the Flash Pool and you destroy the storage pool.

SSD storage pools are divided into four equal allocation units. SSDs added to the storage pool are divided into four partitions and one partition is assigned to each of the four allocation units. The SSDs in the storage pool must be owned by the same HA pair. By default, two allocation units are assigned to each node in the HA pair. Allocation units must be owned by the node that owns the aggregate it is serving. If more Flash cache is required for aggregates on one of the nodes, the default number of allocation units can be shifted to decrease the number on one node and increase the number on the partner node.

You can use only one spare SSD for a storage pool. If the storage pool provides allocation units to Flash Pool aggregates owned by both nodes in the HA pair, then the spare SSD can be owned by either node. However, if the storage pool provides allocation units only to Flash Pool aggregates owned by one of the nodes in the HA pair, then the SSD spare must be owned by that same node.

The following illustration is an example of Flash Pool SSD partitioning. The SSD storage pool provides cache to two Flash Pool aggregates:
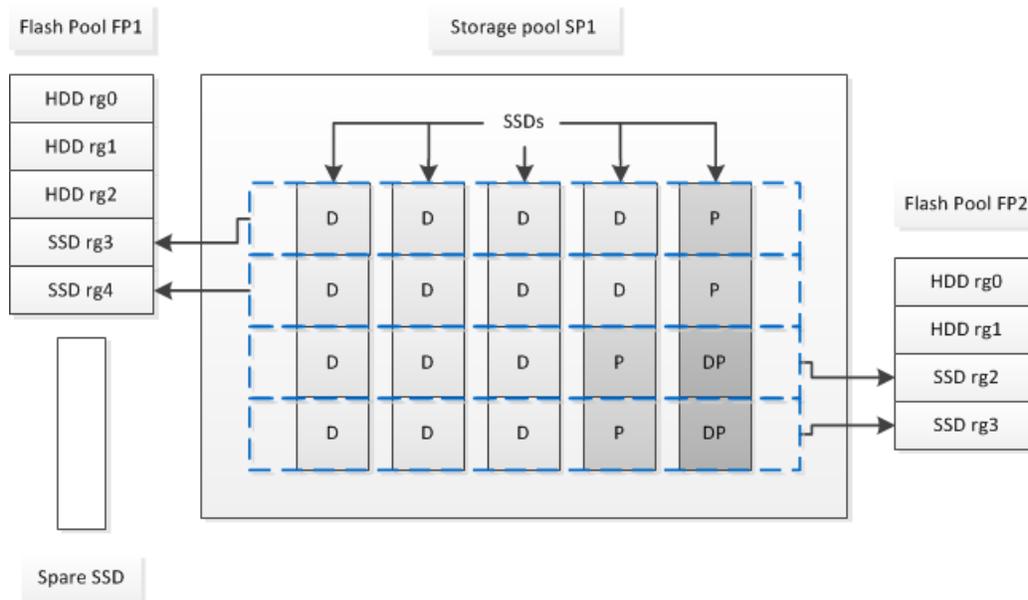
Storage pool SP1 is composed of five SSDs and a hot spare SSD. Two of the storage pool's allocation units are allocated to Flash Pool FP1, and two are allocated to Flash Pool FP2. FP1 has a cache RAID type of RAID4. Therefore, the allocation units provided to FP1 contain only one partition designated for parity. FP2 has a cache RAID type of RAID-DP. Therefore, the allocation units provided to FP2 include a parity partition and a double-parity partition.

In this example, two allocation units are allocated to each Flash Pool aggregate. However, if one Flash Pool aggregate required a larger cache, you could allocate three of the allocation units to that Flash Pool aggregate, and only one to the other.

Determining Flash Pool candidacy and optimal cache size

Before converting an existing aggregate to a Flash Pool aggregate, you can determine whether the aggregate is I/O bound, and what would be the best Flash Pool cache size for your workload and budget. You can also check whether the cache of an existing Flash Pool aggregate is sized correctly.

Before you begin

You should know approximately when the aggregate you are analyzing experiences its peak load.

Steps

1. Enter advanced mode:

   `set advanced`

2. If you need to determine whether an existing aggregate would be a good candidate for
   conversion to a Flash Pool aggregate, determine how busy the disks in the aggregate are during
   a period of peak load, and how that is affecting latency:

   `statistics show-periodic -object disk:raid_group -instance `*`raid_group_name`*` -counter`
   `disk_busy|user_read_latency -interval 1 -iterations 60`

   You can decide whether reducing latency by adding Flash Pool cache makes sense for this
   aggregate.

   The following command shows the statistics for the first RAID group of the aggregate "aggr1":

   `statistics show-periodic -object disk:raid_group -instance /aggr1/plex0/rg0 -counter`
   `disk_busy|user_read_latency -interval 1 -iterations 60`

3. Start Automated Workload Analyzer (AWA):

   **`storage automated-working-set-analyzer start -node`*`node_name`*` -aggregate `*`aggr_name`*
   AWA begins collecting workload data for the volumes associated with the specified aggregate.

4. Exit advanced mode:

   `set admin`

   Allow AWA to run until one or more intervals of peak load have occurred. AWA collects
   workload statistics for the volumes associated with the specified aggregate, and analyzes data
   for up to one rolling week in duration. Running AWA for more than one week will report only
   on data collected from the most recent week. Cache size estimates are based on the highest
   loads seen during the data collection period; the load does not need to be high for the entire
   data collection period.

5. Enter advanced mode:

   `set advanced`

6. Display the workload analysis:

   **`storage automated-working-set-analyzer show -node `*`node_name`*

7. Stop AWA:

   **`storage automated-working-set-analyzer stop `*`node_name`*
   All workload data is flushed and is no longer available for analysis.

8. Exit advanced mode:

   `set admin`

Creating a Flash Pool aggregate using physical SSDs

You create a Flash Pool aggregate by enabling the feature on an existing aggregate composed of
HDD RAID groups, and then adding one or more SSD RAID groups to that aggregate. This results
in two sets of RAID groups for that aggregate: SSD RAID groups (the SSD cache) and HDD
RAID groups.

Before you begin

- You must have identified a valid aggregate composed of HDDs to convert to a Flash Pool
  aggregate.
- You must have determined write-caching eligibility of 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.
- You must have determined the number of SSDs you are adding and the optimal RAID group size for the SSD RAID groups.

  Using fewer RAID groups in the SSD cache reduces the number of parity disks required, but larger RAID groups require RAID-DP.
- You must have determined the RAID level you want to use for the SSD cache.
- You must have determined the maximum cache size for your system and determined that adding SSD cache to your aggregate will not cause you to exceed it.
- You must have familiarized yourself with the configuration requirements for Flash Pool aggregates.

About this task

After you add an SSD cache to an aggregate to create a Flash Pool aggregate, you cannot remove the SSD cache to convert the aggregate back to its original configuration.

By default, the RAID level of the SSD cache is the same as the RAID level of the HDD RAID groups. You can override this default selection by specifying the `raidtype` option when you add the first SSD RAID groups.

Steps

1. Mark the aggregate as eligible to become a Flash Pool aggregate:

   `storage aggregate modify -aggregate aggr_name -hybrid-enabled true`

   If this step does not succeed, determine write-caching eligibility for the target aggregate.

2. Add the SSDs to the aggregate by using the `storage aggregate add` command.

   You can specify the SSDs by ID or by using the `diskcount` and `disktype` parameters.

   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.

   You can specify a different RAID type for the SSD cache by using the `raidtype` parameter.

   If you want the cache RAID group size to be different from the default for the RAID type you are using, you should change it now, by using the `-cache-raid-group-size` parameter.

Creating a Flash Pool aggregate using SSD storage pools

You can configure a Flash Pool aggregate by adding one or more allocation units from an SSD storage pool to an existing HDD aggregate.

Related concepts

How Flash Pool SSD partitioning works for Flash Pool aggregates using storage pools on page 24

If you are providing cache to two or more Flash Pool aggregates, you should use Flash Pool Solid-State Drive (SSD) partitioning. Flash Pool SSD partitioning allows SSDs to be shared by all the aggregates using the Flash Pool. This spreads the cost of parity over multiple aggregates, increases SSD cache allocation flexibility, and maximizes SSD performance.

Determining whether a Flash Pool aggregate is using an SSD storage pool

You manage Flash Pool aggregates differently when they use SSD storage pools to provide their cache than when they use discrete SSDs.

Step

Display the aggregate's drives by RAID group:

`storage aggregate show-status aggr_name`

If the aggregate is using one or more SSD storage pools, the value for the `Position` column for the SSD RAID groups is displayed as `Shared`, and the name of the storage pool is displayed next to the RAID group name.

Creating an SSD storage pool

You can create solid state drive (SSD) storage pools to provide SSD cache for two to four Flash Pool aggregates.

About this task

- You must supply a disk list when creating or adding disks to a storage pool.
  Storage pools do not support a `diskcount` parameter.
- The SSDs used in the storage pool should be the same size.

Steps

1. Determine the names of the available spare SSDs:

   `storage aggregate show-spare-disks -disk-type SSD`

   The SSDs used in a storage pool can be owned by either node of an HA pair.

2. Create the storage pool:

   `storage pool create -storage-pool sp_name -disk-list disk1,disk2,...`

3. Optional: Verify the newly created storage pool:

   `storage pool show -storage-pool sp_name`

Result

After the SSDs are placed into the storage pool, they no longer appear as spares on the cluster, even though the storage provided by the storage pool has not yet been allocated to any Flash Pool caches. You cannot add SSDs to a RAID group as discrete drives; their storage can be provisioned only by using the allocation units of the storage pool to which they belong.

Creating a Flash Pool aggregate using SSD storage pool allocation units

You can configure a Flash Pool aggregate by adding one or more allocation units from an SSD storage pool to an existing HDD aggregate.

Before you begin

- You must have identified a valid aggregate composed of HDDs to convert to a Flash Pool aggregate.
- You must have determined write-caching eligibility of the volumes associated with the aggregate, and completed any required steps to resolve eligibility issues.
- You must have created an SSD storage pool to provide the SSD cache to this Flash Pool aggregate.
  Any allocation unit from the storage pool that you want to use must be owned by the same node that owns the Flash Pool aggregate.
- You must have determined how much cache you want to add to the aggregate.

You add cache to the aggregate by allocation units. You can increase the size of the allocation units later by adding SSDs to the storage pool if there is room.

- You must have determined the RAID type you want to use for the SSD cache.

  After you add a cache to the aggregate from SSD storage pools, you cannot change the RAID type of the cache RAID groups.

- You must have determined the maximum cache size for your system and determined that adding SSD cache to your aggregate will not cause you to exceed it.

  You can see the amount of cache that will be added to the total cache size by using the `storage pool show` command.

- You must have familiarized yourself with the configuration requirements for Flash Pool aggregates.

About this task

If you want the RAID type of the cache to different from that of the HDD RAID groups, you must specify the cache RAID type when you add the SSD capacity. After you add the SSD capacity to the aggregate, you can no longer change the RAID type of the cache.

After you add an SSD cache to an aggregate to create a Flash Pool aggregate, you cannot remove the SSD cache to convert the aggregate back to its original configuration.

Steps

1. Mark the aggregate as eligible to become a Flash Pool aggregate:

   `storage aggregate modify -aggregate aggr_name -hybrid-enabled true`

   If this step does not succeed, determine write-caching eligibility for the target aggregate.

2. Show the available SSD storage pool allocation units:

   `storage pool show-available-capacity`

3. Add the SSD capacity to the aggregate:

   `storage aggregate add aggr_name -storage-pool sp_name -allocation-units number_of_units`

   If you want the RAID type of the cache to be different from that of the HDD RAID groups, you must change it when you enter this command by using the `raidtype` parameter.

   You do not need to specify a new RAID group; ONTAP automatically puts the SSD cache into separate RAID groups from the HDD RAID groups.

   You cannot set the RAID group size of the cache; it is determined by the number of SSDs in the storage pool.

   The cache is added to the aggregate and the aggregate is now a Flash Pool aggregate. Each allocation unit added to the aggregate becomes its own RAID group.

4. Optional: Confirm the presence and size of the SSD cache:

   `storage aggregate show aggr_name`
   The size of the cache is listed under `Total Hybrid Cache Size`.

   Related concepts

   How Flash Pool SSD partitioning works for Flash Pool aggregates using storage pools on page 24

If you are providing cache to two or more Flash Pool aggregates, you should use Flash Pool Solid-State Drive (SSD) partitioning. Flash Pool SSD partitioning allows SSDs to be shared by all the aggregates using the Flash Pool. This spreads the cost of parity over multiple aggregates, increases SSD cache allocation flexibility, and maximizes SSD performance.

   Related information

   NetApp Technical Report 4070: Flash Pool Design and Implementation Guide

Determining the impact to cache size of adding SSDs to an SSD storage pool

If adding SSDs to a storage pool causes your platform model's cache limit to be exceeded, ONTAP does not allocate the newly added capacity to any Flash Pool aggregates. This can result in some or all of the newly added capacity being unavailable for use.

About this task

When you add SSDs to an SSD storage pool that has allocation units already allocated to Flash Pool aggregates, you increase the cache size of each of those aggregates and the total cache on the system. If none of the storage pool's allocation units have been allocated, adding SSDs to that storage pool does not affect the SSD cache size until one or more allocation units are allocated to a cache.

Steps

1. Determine the usable size of the SSDs you are adding to the storage pool:

   `storage disk show disk_name -fields usable-size`

2. Determine how many allocation units remain unallocated for the storage pool:

   `storage pool show-available-capacity sp_name`
   All unallocated allocation units in the storage pool are displayed.

3. Calculate the amount of cache that will be added by applying the following formula:

   (4 − number of unallocated allocation units) × 25% × usable size × number of SSDs

Adding SSDs to an SSD storage pool

When you add solid state drives (SSDs) to an SSD storage pool, you increase the storage pool's physical and usable sizes and allocation unit size. The larger allocation unit size also affects allocation units that have already been allocated to Flash Pool aggregates.

Before you begin

You must have determined that this operation will not cause you to exceed the cache limit for your HA pair. ONTAP does not prevent you from exceeding the cache limit when you add SSDs to an SSD storage pool, and doing so can render the newly added storage capacity unavailable for use.

About this task

When you add SSDs to an existing SSD storage pool, the SSDs must be owned by one node or the other of the same HA pair that already owned the existing SSDs in the storage pool. You can add SSDs that are owned by either node of the HA pair.

The SSD you add to the storage pool must be the same size as disk currently used in the storage pool.

Steps

1. Optional: View the current allocation unit size and available storage for the storage pool:

   `storage pool show -instance sp_name`

2. Find available SSDs:

   `storage disk show -container-type spare -type SSD`

3. Add the SSDs to the storage pool:

   `storage pool add -storage-pool sp_name -disk-list disk1,disk2…`
   The system displays which Flash Pool aggregates will have their size increased by this operation and by how much, and prompts you to confirm the operation.

Commands for managing SSD storage pools

ONTAP provides the `storage pool` command for managing SSD storage pools.

If you want to...	Use this command...
Display how much storage a storage pool is providing to which aggregates	`storage pool show-aggregate`
Display how much cache would be added to the overall cache capacity for both RAID types (allocation unit data size)	`storage pool show -instance`
Display the disks in a storage pool	`storage pool show-disks`
Display the unallocated allocation units for a storage pool	`storage pool show-available-capacity`
Change the ownership of one or more allocation units of a storage pool from one HA partner to the other	`storage pool reassign`

Related information

[ONTAP 9 commands](#)

Determining whether to modify the caching policy of Flash Pool aggregates

You can assign cache-retention policies to volumes in Flash Pool aggregates to determine how long the volume data remains in the Flash Pool cache. However, in some cases changing the cache-retention policy might not impact the amount of time the volume's data remains in the cache.

About this task

If your data meets any of the following conditions, changing your cache-retention policy might not have an impact:

- Your workload is sequential.
- Your workload does not reread the random blocks cached in the solid state drives (SSDs).
- The cache size of the volume is too small.

The following steps check for these conditions. The task must be done in advanced privilege mode.

Steps

1. View the workload volume:

   statistics start -object workload_volume

2. Determine the workload pattern of the volume:

   statistics show -object workload_volume -instance *volume-workload* -counter sequential_reads

3. Determine the hit rate of the volume:

   statistics show -object wafl_hya_vvol -instance *volume* -counter read_ops_replaced_pwercent|wc_write_blks_overwritten_percent

4. Determine the `Cacheable Read` and `Project Cache Alloc` of the volume:

   system node run -node *node_name* wafl awa start *aggr_name*

5. Display the AWA summary:

   system node run -node *node_name* wafl awa print *aggr_name*

6. Compare the volume's hit rate to the `Cacheable Read`.

If the hit rate of the volume is greater than the `Cacheable Read`, then your workload does not reread random blocks cached in the SSDs.

7. Compare the volume's current cache size to the `Project Cache Alloc`.

If the current cache size of the volume is greater than the `Project Cache Alloc`, then the size of your volume cache is too small.

Modifying caching policies of Flash Pool aggregates

You should modify the caching policy of a volume only if a different caching policy is expected to provide better performance. You can modify the caching policy of a volume on a Flash Pool aggregate.

Before you begin

You must determine whether you want to modify your caching policy.

About this task

In most cases, the default caching policy of **auto** is the best caching policy that you can use. The caching policy should be changed only if a different policy provides better performance for your workload. Configuring the wrong caching policy can severely degrade volume performance; the performance degradation could increase gradually over time. You should use caution when modifying caching policies. If you experience performance issues with a volume for which the caching policy has been changed, you should return the caching policy to **auto**.

Step

Modify the volume's caching policy:

volume modify -volume *volume_name* -caching-policy *policy_name*

Example

The following example modifies the caching policy of a volume named "vol2" to the policy **none**:

volume modify -volume vol2 -caching-policy none

Related tasks

Determining whether to modify the caching policy of Flash Pool aggregates on page 31
You can assign cache-retention policies to volumes in Flash Pool aggregates to determine how long the volume data remains in the Flash Pool cache. However, in some cases changing the cache-retention policy might not impact the amount of time the volume's data remains in the cache.

Setting the cache-retention policy for Flash Pool aggregates

Beginning with ONTAP 9.0, you can assign cache-retention policies to volumes in Flash Pool aggregates. Data in volumes with a high cache-retention policy remains in cache longer and data in volumes with a low cache-retention policy is removed sooner. This increases performance of your critical workloads by making high priority information accessible at a faster rate for a longer period of time.

Before you begin

You should know whether your system has any conditions that might prevent the cache-retention policy from having an impact on how long your data remains in cache.

About this task

The task must be done in advanced privilege mode.

Steps

1. Change the privilege setting to advanced:

   `set -privilege advanced`

2. Verify the volume's cache-retention policy:

   By default the cache retention policy is **`normal`**.

3. Set the cache-retention policy:

ONTAP Version	Command
ONTAP 9.0, 9.1	**`priority hybrid-cache set`** *`volume_name`* **`read-cache=`***`read_cache_value`* **`write-cache=`***`write_cache_value`* **`cache-retention-priority=`***`cache_retention_policy`* Set *`cache_retention_policy`* to **`high`** for data that you want to remain in cache longer. Set *`cache_retention_policy`* to **`low`** for data that you want to remove from cache sooner.
ONTAP 9.2 or later	**`volume modify -volume`** *`volume_name`* **`-vserver`** *`vserver_name`* **`-caching-policy`** *`policy_name`***`.`**

4. Verify that the volume's cache-retention policy is changed to the option you selected.

5. Return the privilege setting to admin:

   `set -privilege admin`

Managing disks

You can perform various tasks to manage your disks, including removing a failed disk, removing data from a disk, and removing ownership of a disk. There are also tasks you can perform related to managing disks using root-data partitioning and related to self-encrypting disks.

When you need to update the Disk Qualification Package

The Disk Qualification Package (DQP) adds full support for newly qualified drives. Before you update drive firmware or add new drive types or sizes to a cluster, you must update the DQP. A best practice is to also update the DQP regularly; for example, every quarter or semi-annually.

You need to download and install the DQP in the following situations:

- Whenever you add a new drive type or size to the node
  For example, if you already have 1-TB drives and add 2-TB drives, you need to check for the latest DQP update.
- Whenever you update the disk firmware
- Whenever newer disk firmware or DQP files are available
- Whenever you upgrade to a new version of ONTAP.
  The DQP is not updated as part of an ONTAP upgrade.

Related information
[*NetApp Downloads: Disk Qualification Package*](#)
[*NetApp Downloads: Disk Drive Firmware*](#)

How hot spare disks work

A hot spare disk is a disk that is assigned to a storage system and is ready for use, but is not in use by a RAID group and does not hold any data.

If a disk failure occurs within a RAID group, the hot spare disk is automatically assigned to the RAID group to replace the failed disks. The data of the failed disk is reconstructed on the hot spare replacement disk in the background from the RAID parity disk. The reconstruction activity is logged in the `/etc/message` file and an AutoSupport message is sent.

If the available hot spare disk is not the same size as the failed disk, a disk of the next larger size is chosen and then downsized to match the size of the disk that it is replacing.

How low spare warnings can help you manage your spare disks

By default, warnings are issued to the console and logs if you have fewer than one hot spare drive that matches the attributes of each drive in your storage system. You can change the threshold value for these warning messages to ensure that your system adheres to best practices.

You should set the `min_spare_count` RAID option to **2** to ensure that you always have the minimum recommended number of spare disks.

Displaying disk and partition ownership

You can view disk ownership to determine which node controls the storage. You can also view the partition ownership on systems that use shared disks.

Steps

1. Display the ownership of physical disks using the `storage disk show -ownership` command:

```
cluster::> storage disk show -ownership
Disk      Aggregate Home     Owner    DR Home  Home ID    Owner ID    DR Home ID
Reserver     Pool
-------- --------- -------- -------- -------- ---------- ----------- -----------
----------- ------
1.0.0    aggr0_2   node2    node2    -        2014941509 2014941509  -
2014941509  Pool0
1.0.1    aggr0_2   node2    node2    -        2014941509 2014941509  -
2014941509  Pool0
1.0.2    aggr0_1   node1    node1    -        2014941219 2014941219  -
2014941219  Pool0
1.0.3    -         node1    node1    -        2014941219 2014941219  -
2014941219  Pool0
...
```

2. If you have a system that uses shared disks, display the partition ownership using the `storage disk show -partition-ownership` command:

```
cluster::> storage disk show -partition-ownership
                              Root                  Data        Container
Container
Disk      Aggregate Root Owner  Owner ID    Data Owner  Owner ID    Owner       Owner
ID
-------- --------- ----------- ----------- ----------- ----------- ----------
-----------
1.0.0    -         node1       1886742616  node1       1886742616  node1
1886742616
1.0.1    -         node1       1886742616  node1       1886742616  node1
1886742616
1.0.2    -         node2       1886742657  node2       1886742657  node2
1886742657
1.0.3    -         node2       1886742657  node2       1886742657  node2
1886742657
...
```

Manually assigning ownership of partitioned disks

You can set the ownership of the container disk or the partitions manually or by using auto-assignment—just as you do for unpartitioned disks.

About this task

Note: If a container disk fails in a half-populated shelf and is replaced, ONTAP will not auto-assign ownership. In this case, any assignment of new disks will need to be done manually. To make auto-assign work on half-populated shelves, place disks equally on lower half and 6 on far right bays to begin with. That is, 6 disks from bays 0-5 and 6 disks from bays 18-23. After the container disk is assigned in an ADP-configured system, ONTAP's software will handle any partitioning and partition assignments that are required, without user intervention.

Related tasks

Manually assigning disks with root-data partitioning on page 36
For root-data partitioning there are three owned entities (the container disk and the two partitions) collectively owned by the HA pair.

Manually assigning disks with root-data-data partitioning on page 36

For root-data-data partitioning there are four owned entities (the container disk and the three partitions) collectively owned by the HA pair.

Manually assigning disks with root-data partitioning

For root-data partitioning there are three owned entities (the container disk and the two partitions) collectively owned by the HA pair.

About this task

The container disk and the two partitions do not all need to be owned by the same node in the HA pair as long as they are all owned by one of the nodes in the HA pair. However, when you use a partition in an aggregate, it must be owned by the same node that owns the aggregate.

Steps

1. Display the current ownership for the partitioned disk:

`storage disk show -disk disk_name -partition-ownership`

2. Set the CLI privilege level to advanced:

`set -privilege advanced`

3. Enter the appropriate command, depending on which ownership entity you want to assign ownership for:

If you want to assign ownership for the...	Use this command...
Container disk	**`storage disk assign -disk disk_name -owner owner_name`**
Data partition	**`storage disk assign -disk disk_name -owner owner_name -data true`**
Root partition	**`storage disk assign -disk disk_name -owner owner_name -root true`**

If any of the ownership entities are already owned, then you must include the `-force` option.

Manually assigning disks with root-data-data partitioning

For root-data-data partitioning there are four owned entities (the container disk and the three partitions) collectively owned by the HA pair.

About this task

Root-data-data partitioning creates one small partition as the root partition and two larger, equally sized partitions for data.

Parameters must be used in the `disk assign` command to assign the proper partition of a root-data-data partitioned disk. You cannot use these parameters with disks that are part of a storage pool. The default value is false.

- The [-data1 [true]] parameter assigns the *data1* partition of a root-data1-data2 partitioned disk.
- The [-data2 [true]] parameter assigns the *data2* partition of a root-data1-data2 partitioned disk.

Steps

1. Display the current ownership for the partitioned disk:

`storage disk show -disk disk_name -partition-ownership`

2. Set the CLI privilege level to advanced:

`set -privilege advanced`

3. Enter the appropriate command, depending on which ownership entity you want to assign ownership for:

If you want to assign ownership for the...	Use this command...
Container disk	`storage disk assign -disk disk_name -owner owner_name`
Data1 partition	`storage disk assign -disk disk_name -owner owner_name -data1 true`
Data2 partition	`storage disk assign -disk disk_name -owner owner_name -data2 true`
Root partition	`storage disk assign -disk disk_name -owner owner_name -root true`

If any of the ownership entities are already owned, then you must include the `-force` option.

Additional root-data partitioning management options

Beginning with ONTAP 9.2, a new root-data partitioning option is available from the Boot Menu that provides additional management features for disks that are configured for root-data partitioning.

The following management features are available under the Boot Menu Option 9.

- Unpartition all disks and remove their ownership information
  This option is useful if your system is configured for root-data partitioning and you need to reinitialize it with a different configuration.
- Clean configuration and initialize system with partitioned disks
  This option is useful for the following:

  ◦ Your system is not configured for root-data partitioning and you would like to configure it for root-data partitioning
  ◦ Your system is incorrectly configured for root-data partitioning and you need to correct it
  ◦ You have an AFF platform or a FAS platform with only SSDs attached that is configured for the previous version of root-data partitioning and you want to upgrade it to the newer version of root-data partitioning to gain increased storage efficiency
- Clean configuration and initialize system with whole disks
  This option is useful if you need to:

  ◦ Unpartition existing partitions
  ◦ Remove local disk ownership
  ◦ Reinitialize your system with whole disks using RAID-DP

Configuring automatic assignment of disk ownership

You can configure ONTAP to automatically assign disk ownership according to a disk's stack, shelf, or bay. If configured, automatic disk ownership assignments occur 10 minutes after system initialization and every five minutes during normal system operation.

Before you begin

- Your system must adhere to the requirements for automatic disk ownership.

- If you have multiple stacks or shelves that must have different ownership, one disk must have been manually assigned on each stack or shelf so that automatic ownership assignment works on each stack or shelf.
- Use the *bay* autoassign-policy only for entry level platforms. If you try to use the *bay* autoassign-policy for a non-entry level platform, it will fail.

About this task

The behavior of the **default** automatic assignment policy depends on the system model. For entry level models, the **default** policy is equivalent to the **bay** policy. For all other systems, it is equivalent to the **stack** policy.

Steps

1. Configure automatic disk assignment:

   storage disk option modify -autoassign-policy *autoassign_policy* -node *node_name*

   - Use **stack** as the *autoassign_policy* to configure automatic ownership at the stack or loop level.
   - Use **shelf** as the *autoassign_policy* to configure automatic ownership at the shelf level.
   - Use **bay** as the *autoassign_policy* to configure automatic ownership at the bay level.

2. Verify the automatic assignment settings for the disks:

   storage disk option show

   ```
   cluster1::> storage disk option show

   Node          BKg. FW. Upd.  Auto Copy    Auto Assign   Auto
                                                           Assign Policy
   ------------- -------------  ------------ ------------- --------
   cluster1-1    on             on           on            default
   cluster1-2    on             on           on            default
   ```

Which disk autoassignment policy to use

You can typically use the default autoassignment policy, which is equivalent to the **stack** policy for most systems, and to the **bay** policy for entry-level systems (AFF A2xx, FAS2xxx). However, for some configurations, you might need to change the autoassignment policy.

You must select the appropriate autoassignment based on your configuration:

If you are using...	Then use the autoassignment policy value of...
Stand-alone entry-level system	**stack**
Entry-level systems in an HA configuration with a single, shared shelf	**bay**
Entry-level systems in an HA configuration with one stack of two or more shelves	**shelf**
MetroCluster configurations with one stack per node, two or more shelves	**shelf**
All other configurations	**stack**

Removing a failed disk

A disk that is completely failed is no longer counted by 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 ownership from a disk

ONTAP writes disk ownership information to the disk. Before you remove a spare disk or its shelf from a node, you should remove its ownership information so that it can be properly integrated into another node.

Before you begin

The disk you want to remove ownership from must meet the following requirements:

- It must be a spare disk.
  You cannot remove ownership from a disk that is being used in an aggregate.
- It cannot be in the maintenance center.
- It cannot be undergoing sanitization.
- It cannot be failed.
  It is not necessary to remove ownership from a failed disk.

About this task

If you have automatic disk assignment enabled, ONTAP could automatically reassign ownership before you remove the disk from the node. For this reason, you disable automatic ownership assignment until the disk is removed, and then reenable it.

Steps

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

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

2. If needed, repeat the previous step for the node's HA partner.

3. Remove the software ownership information from the disk:

   `storage disk removeowner disk_name`

   To remove ownership information from multiple disks, use a comma-separated list:

storage disk removeowner sys1:0a.23,sys1:0a.24,sys1:0a.25

4. If the disk is partitioned for root-data partitioning, remove ownership from the partitions by entering both of the following commands:

   storage disk removeowner *disk_name* -root true

   storage disk removeowner *disk_name* -data true
   Both partitions are no longer owned by any node.

5. If you turned off disk ownership automatic assignment previously, turn it on after the disk has been removed or reassigned:

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

6. If needed, repeat the previous step for the node's HA partner.

Sanitizing a disk

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 disk must be a spare, and must be owned by a node but not used in an aggregate.
  If the disk is partitioned, neither partition can be used in an aggregate.
- The disk cannot be a self-encrypting disk (SEDs) or a Federal Information Processing Standard (FIPS) disk.
  You must use the storage encryption disk sanitize command to sanitize an SED or FIPS disk.
  Encryption of data at rest
- The disk cannot be part of a storage pool.

About this task

When disk sanitization is enabled, it disables some ONTAP commands.

- dd (to copy blocks of data)
- dumpblock (to print dumps of disk blocks)
- setflag wafl_metadata_visible (to allow access to internal WAFL files)

After disk sanitization is enabled on a node, it cannot be disabled.

Steps

1. Enter the nodeshell for the node that owns the disks you want to sanitize:

   system node run -node *node_name*

2. Enable disk sanitization:

   options licensed_feature.disk_sanitization.enable on
   You are asked to confirm the command because it is irreversible.

3. If the disks you want to sanitize are partitioned, unpartition each disk:

   disk unpartition *disk_name*

4. Sanitize the specified disks:

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

   ⚠️ **Attention:** Do not turn off power to the node, 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.

-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:

   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. Exit the nodeshell:

   exit

8. Determine whether all of the disks were returned to spare status:

   storage aggregate show-spare-disks

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

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 *node_name*
	c. Return the disks to spare status by entering the following command for each disk:
	    storage disk unfail -disk *disk_name* -s -q
	d. Return to administrative mode:
	    set -privilege admin

Result

The specified disks are sanitized and designated as hot spares. The serial numbers of the sanitized
disks are written to /etc/log/sanitized_disks.

Sanitizing a disk in maintenance mode

Beginning in ONTAP 9.6, you can perform disk sanitization in maintenance mode. This allows you to remove data from a disk or a set of disks on decommissioned or inoperable systems so that the data can never be recovered.

Before you begin

- The disk cannot be a self-encrypting disk (SED).

  You must use the `storage encryption disk sanitize` command to sanitize an SED.

  Encryption of data at rest

Steps

1. Boot into maintenance mode.

2. If the disks you want to sanitize are partitioned, unpartition each disk:

   `disk unpartition disk_name`

3. Sanitize the specified disks:

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

   > ⚠️ **Attention:** Do not turn off power to the node, 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.

   `-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.

4. If desired, check the status of the disk sanitization process:

   `disk sanitize status [disk_list]`

5. 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`

6. Exit maintenance mode.

Setting up an active-passive configuration on nodes using root-data partitioning

When an HA pair is configured to use root-data partitioning by the factory, ownership of the data partitions is split between both nodes in the pair, for use in an active-active configuration. If you want to use the HA pair in an active-passive configuration, you must update partition ownership before creating your data aggregate.

Before you begin

- You should have decided which node will be the active node and which node will be the passive node.
- Storage failover must be configured on the HA pair.

About this task

This task is performed on two nodes: Node A and Node B.

All commands are input at the clustershell.

This procedure is designed for nodes for which no data aggregate has been created from the partitioned disks.

Steps

1. View the current ownership of the data partitions:

   storage aggregate show-spare-disks

   You can see that half of the data partitions are owned by one node and half are owned by the other node. All of the data partitions should be spare.

```
cluster1::> storage aggregate show-spare-disks

Original Owner: cluster1-01
 Pool0
  Partitioned Spares
                                                   Local    Local
                                                    Data     Root Physical
Disk                      Type   RPM Checksum      Usable   Usable     Size
------------------------- ----- ------ -------------- -------- -------- --------
1.0.0                     BSAS   7200 block         753.8GB       0B  828.0GB
1.0.1                     BSAS   7200 block         753.8GB  73.89GB  828.0GB
1.0.5                     BSAS   7200 block         753.8GB       0B  828.0GB
1.0.6                     BSAS   7200 block         753.8GB       0B  828.0GB
1.0.10                    BSAS   7200 block         753.8GB       0B  828.0GB
1.0.11                    BSAS   7200 block         753.8GB       0B  828.0GB

Original Owner: cluster1-02
 Pool0
  Partitioned Spares
                                                   Local    Local
                                                    Data     Root Physical
Disk                      Type   RPM Checksum      Usable   Usable     Size
------------------------- ----- ------ -------------- -------- -------- --------
1.0.2                     BSAS   7200 block         753.8GB       0B  828.0GB
1.0.3                     BSAS   7200 block         753.8GB       0B  828.0GB
1.0.4                     BSAS   7200 block         753.8GB       0B  828.0GB
1.0.7                     BSAS   7200 block         753.8GB       0B  828.0GB
1.0.8                     BSAS   7200 block         753.8GB  73.89GB  828.0GB
1.0.9                     BSAS   7200 block         753.8GB       0B  828.0GB
12 entries were displayed.
```

2. Enter the advanced privilege level:

   set advanced

3. For each data partition owned by the node that will be the passive node, assign it to the active node:

   storage disk assign -force -data true -owner *active_node_name* -disk *disk_name*

   You do not need to include the partition as part of the disk name.

   You would enter a command similar to the following example for each data partition you need to reassign:

   storage disk assign -force -data true -owner cluster1-01 -disk 1.0.3

4. Confirm that all of the partitions are assigned to the active node.

```
cluster1::*> storage aggregate show-spare-disks

Original Owner: cluster1-01
 Pool0
  Partitioned Spares
                                                  Local    Local
                                                   Data     Root Physical
Disk                       Type   RPM Checksum    Usable   Usable     Size
-------------------------- ----- ------ -------------- -------- -------- --------
1.0.0                      BSAS   7200 block       753.8GB       0B  828.0GB
1.0.1                      BSAS   7200 block       753.8GB 73.89GB  828.0GB
1.0.2                      BSAS   7200 block       753.8GB       0B  828.0GB
1.0.3                      BSAS   7200 block       753.8GB       0B  828.0GB
1.0.4                      BSAS   7200 block       753.8GB       0B  828.0GB
1.0.5                      BSAS   7200 block       753.8GB       0B  828.0GB
1.0.6                      BSAS   7200 block       753.8GB       0B  828.0GB
1.0.7                      BSAS   7200 block       753.8GB       0B  828.0GB
1.0.8                      BSAS   7200 block       753.8GB       0B  828.0GB
1.0.9                      BSAS   7200 block       753.8GB       0B  828.0GB
1.0.10                     BSAS   7200 block       753.8GB       0B  828.0GB
1.0.11                     BSAS   7200 block       753.8GB       0B  828.0GB

Original Owner: cluster1-02
 Pool0
  Partitioned Spares
                                                  Local    Local
                                                   Data     Root Physical
Disk                       Type   RPM Checksum    Usable   Usable     Size
-------------------------- ----- ------ -------------- -------- -------- --------
1.0.8                      BSAS   7200 block            0B 73.89GB  828.0GB
13 entries were displayed.
```

Note that cluster1-02 still owns a spare root partition.

5. Return to administrative privilege:

set admin

6. Create your data aggregate, leaving at least one data partition as spare:

storage aggregate create *new_aggr_name* -diskcount *number_of_partitions* -node *active_node_name*

The data aggregate is created and is owned by the active node.

Setting up an active-passive configuration on nodes using root-data-data partitioning

When an HA pair is configured to use root-data-data partitioning by the factory, ownership of the data partitions is split between both nodes in the pair, for use in an active-active configuration. If you want to use the HA pair in an active-passive configuration, you must update partition ownership before creating your data aggregate.

Before you begin

- You should have decided which node will be the active node and which node will be the passive node.
- Storage failover must be configured on the HA pair.

About this task

This task is performed on two nodes: Node A and Node B.

All commands are input at the clustershell.

This procedure is designed for nodes for which no data aggregate has been created from the partitioned disks.

Steps

1. View the current ownership of the data partitions:

storage aggregate show-spare-disks -original-owner *passive_node_name* -fields local-usable-data1-size, local-usable-data2-size

You should see that half of the data partitions are owned by one node and half are owned by the other node. All of the data partitions should be spare.

2. Enter the advanced privilege level:

set advanced

3. For each data1 partition owned by the node that will be the passive node, assign it to the active node:

storage disk assign -force -data1 -owner active_node_name -disk disk_name

You do not need to include the partition as part of the disk name

4. For each data2 partition owned by the node that will be the passive node, assign it to the active node:

storage disk assign -force -data2 -owner active_node_name -disk disk_name

You do not need to include the partition as part of the disk name

5. Confirm that all of the partitions are assigned to the active node:

storage aggregate show-spare-disks

```
cluster1::*> storage aggregate show-spare-disks

Original Owner: cluster1-01
 Pool0
  Partitioned Spares
                                                Local    Local
                                                 Data     Root Physical
 Disk                       Type    RPM Checksum Usable   Usable     Size
 -------------------------- ----- ------ -------------- -------- -------- --------
 1.0.0                      BSAS   7200 block     753.8GB      0B  828.0GB
 1.0.1                      BSAS   7200 block     753.8GB  73.89GB  828.0GB
 1.0.2                      BSAS   7200 block     753.8GB      0B  828.0GB
 1.0.3                      BSAS   7200 block     753.8GB      0B  828.0GB
 1.0.4                      BSAS   7200 block     753.8GB      0B  828.0GB
 1.0.5                      BSAS   7200 block     753.8GB      0B  828.0GB
 1.0.6                      BSAS   7200 block     753.8GB      0B  828.0GB
 1.0.7                      BSAS   7200 block     753.8GB      0B  828.0GB
 1.0.8                      BSAS   7200 block     753.8GB      0B  828.0GB
 1.0.9                      BSAS   7200 block     753.8GB      0B  828.0GB
 1.0.10                     BSAS   7200 block     753.8GB      0B  828.0GB
 1.0.11                     BSAS   7200 block     753.8GB      0B  828.0GB

Original Owner: cluster1-02
 Pool0
  Partitioned Spares
                                                Local    Local
                                                 Data     Root Physical
 Disk                       Type    RPM Checksum Usable   Usable     Size
 -------------------------- ----- ------ -------------- -------- -------- --------
 1.0.8                      BSAS   7200 block          0B  73.89GB  828.0GB
13 entries were displayed.
```

Note that cluster1-02 still owns a spare root partition.

6. Return to administrative privilege:

set admin

7. Create your data aggregate, leaving at least one data partition as spare:

storage aggregate create new_aggr_name -diskcount number_of_partitions -node active_node_name

The data aggregate is created and is owned by the active node.

8. Alternatively, you can use ONTAP's recommend aggregate layout which includes best practices for RAID group layout and spare counts:

storage aggregate auto-provision

Commands for managing disks

You can use the `storage disk` and `storage aggregate` commands to manage your disks.

If you want to...	Use this command...
Display a list of spare disks, including partitioned disks, by owner	`storage aggregate show-spare-disks`
Display the disk RAID type, current usage, and RAID group by aggregate	`storage aggregate show-status`
Display the RAID type, current usage, aggregate, and RAID group, including spares, for physical disks	`storage disk show -raid`
Display a list of failed disks	`nstorage disk show -broken`
Display the pre-cluster (nodescope) drive name for a disk	`storage disk show -primary-paths` (advanced)
Illuminate the LED for a particular disk or shelf	`storage disk set-led`
Display the checksum type for a specific disk	`storage disk show -fields checksum-compatibility`
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`
Display the pre-cluster disk names for specific disks	`storage disk show -disk -fields diskpathnames`
Display the list of disks in the maintenance center	`storage disk show -maintenance`
Display SSD wear life	`storage disk show -ssd-wear`
Unpartition a disk	`system node run -node local -command disk unpartition`
Unpartition a shared disk	`storage disk unpartition` (available at diagnostic level)
Zero all non-zeroed disks	`storage disk zerospares`
Stop an ongoing sanitization process on one or more specified disks	`disk sanitize abort disk_list`
Display storage encryption disk information	`storage encryption disk show`
Retrieve authentication keys from all linked key management servers	`security key-manager restore`

Related information

[ONTAP 9 commands](#)

Commands for displaying space usage information

You use the `storage aggregate` and `volume` commands to see how 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`
	
	`storage aggregate show-space -fields snap-size-total,used-including-snapshot-reserve`

How disks and RAID groups are used in an aggregate, and RAID status	`storage aggregate show-status`
The amount of disk space that would be reclaimed if you deleted a specific Snapshot copy	`volume snapshot compute-reclaimable`(advanced)

The amount of space used by a volume	`volume show -fields size,used,available,percent-used`
	
	`volume show-space`

The amount of space used by a volume in the containing aggregate	`volume show-footprint`

Related information

[ONTAP 9 commands](#)

Commands for displaying information about storage shelves

You use the `storage shelf show` command to display configuration and error information for your disk shelves.

If you want to display...	Use this command...
General information about shelf configuration and hardware status	`storage shelf show`
Detailed information for a specific shelf, including stack ID	`storage shelf show -shelf`
Unresolved, customer actionable, errors by shelf	`storage shelf show -errors`
Bay information	`storage shelf show -bay`
Connectivity information	`storage shelf show -connectivity`
Cooling information, including temperature sensors and cooling fans	`storage shelf show -cooling`
Information about I/O modules	`storage shelf show -module`
Port information	`storage shelf show -port`
Power information, including PSUs (power supply units), current sensors, and voltage sensors	`storage shelf show -power`

Related information

[ONTAP 9 commands](#)

Managing RAID groups

You can convert from RAID-DP to RAID-TEC for triple-parity protection, or from RAID-TEC to RAID-DP when a smaller aggregate no longer needs triple-parity protection. You can also customize the size of your RAID groups.

Converting from RAID-DP to RAID-TEC

If you want the added protection of triple-parity, you can convert from RAID-DP to RAID-TEC. RAID-TEC is recommended if the size of the disks used in your aggregate is greater than 4 TiB.

Before you begin

The aggregate that is to be converted must have a minimum of six disks.

About this task

Hard disk drive (HDD) aggregates can be converted from RAID-DP to RAID-TEC. This includes HDD tiers in Flash Pool aggregates.

Steps

1. Verify that the aggregate is online and has a minimum of six disks:

   storage aggregate show-status -aggregate *aggregate_name*

2. Convert the aggregate from RAID-DP to RAID-TEC:

   storage aggregate modify -aggregate *aggregate_name* **-raidtype raid_tec**

3. Verify that the aggregate RAID policy is RAID-TEC:

   storage aggregate show *aggregate_name*

Converting RAID-TEC to RAID-DP

If you reduce the size of your aggregate and no longer need triple parity, you can convert your RAID policy from RAID-TEC to RAID-DP and reduce the number of disks you need for RAID parity.

About this task

The maximum RAID group size for RAID-TEC is larger than the maximum RAID group size for RAID-DP. If the largest RAID-TEC group size is not within the RAID-DP limits, you cannot convert to RAID-DP.

Steps

1. Verify that the aggregate is online and has a minimum of six disks:

   storage aggregate show-status -aggregate *aggregate_name*

2. Convert the aggregate from RAID-TEC to RAID-DP:

   storage aggregate modify -aggregate *aggregate_name* **-raidtype raid_dp**

3. Verify that the aggregate RAID policy is RAID-DP:

   storage aggregate show *aggregate_name*

Considerations for sizing RAID groups

Configuring an optimum RAID group size requires a trade-off of factors. You must decide which factors—speed of RAID rebuild, assurance against risk of data loss due to drive failure, optimizing

I/O performance, and maximizing data storage space—are most important for the aggregate that you are configuring.

When you create larger RAID groups, you maximize the space available for data storage for the same amount of storage used for parity (also known as the "parity tax"). On the other hand, when a disk fails in a larger RAID group, reconstruction time is increased, impacting performance for a longer period of time. In addition, having more disks in a RAID group increases the probability of a multiple disk failure within the same RAID group.

HDD or array LUN RAID groups

You should follow these guidelines when sizing your RAID groups composed of HDDs or array LUNs:

- All RAID groups in an aggregate should have the same number of disks.
  While you can have up to 50% less or more than the number of disks in different raid groups on one aggregate, this might lead to performance bottlenecks in some cases, so is best avoided.
- The recommended range of RAID group disk numbers is between 12 and 20.
  The reliability of performance 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 disk numbers, you should choose the larger number of disks.

SSD RAID groups in Flash Pool aggregates

The SSD RAID group size can be different from the RAID group size for the HDD RAID groups in a Flash Pool aggregate. Usually, you should ensure that you have only one SSD RAID group for a Flash Pool aggregate, to minimize the number of SSDs required for parity.

SSD RAID groups in SSD aggregates

You should follow these guidelines when sizing your RAID groups composed of SSDs:

- All RAID groups in an aggregate should have a similar number of drives.
  The RAID groups do not have to be exactly the same size, but you should avoid having any RAID group that is less than one half the size of other RAID groups in the same aggregate when possible.
- For RAID-DP, the recommended range of RAID group size is between 20 and 28.

Customizing the size of your RAID groups

You can customize the size of your RAID groups to ensure that your RAID group sizes are appropriate for the amount of storage you plan to include for an aggregate.

About this task

For standard aggregates, you change the size of RAID groups on a per-aggregate basis. For Flash Pool aggregates, you can change the RAID group size for the SSD RAID groups and the HDD RAID groups independently.

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

- By default, if the number of disks or array LUNs in the most recently created RAID group is less than the new RAID group size, 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 can never cause a RAID group to become larger than the current maximum RAID group size for the aggregate.
- You cannot decrease the size of already created RAID groups.

- The new size applies to all RAID groups in that aggregate (or, in the case of a Flash Pool aggregate, all RAID groups for the affected RAID group type—SSD or HDD).

Step

Use the applicable command:

If you want to...	Enter the following command...
Change the maximum RAID group size for the SSD RAID groups of a Flash Pool aggregate	`storage aggregate modify -aggregate aggr_name -cache-raid-group-size size`
Change the maximum size of any other RAID groups	`storage aggregate modify -aggregate aggr_name -maxraidsize size`

Examples

The following command changes the maximum RAID group size of the aggregate n1_a4 to 20 disks or array LUNs:

```
storage aggregate modify -aggregate n1_a4 -maxraidsize 20
```

The following command changes the maximum RAID group size of the SSD cache RAID groups of the Flash Pool aggregate n1_cache_a2 to 24:

```
storage aggregate modify -aggregate n1_cache_a2 -cache-raid-group-size 24
```

Related concepts

Considerations for sizing RAID groups on page 48
Configuring an optimum RAID group size requires a trade-off of factors. You must decide which factors—speed of RAID rebuild, assurance against risk of data loss due to drive failure, optimizing I/O performance, and maximizing data storage space—are most important for the aggregate that you are configuring.

Appendix: Mirrored and unmirrored aggregates

You can use an optional feature called *SyncMirror* to synchronously mirror aggregate data in copies, or *plexes*, stored in different RAID groups. Plexes ensure against data loss if more disks fail than the RAID type protects against, or if there is a loss of connectivity to RAID group disks.

How unmirrored aggregates work

Unless you are using SyncMirror, all of your aggregates are unmirrored. Unmirrored aggregates have only one *plex* (copy of their data), which contains all of the RAID groups belonging to that aggregate.

The following diagram shows an unmirrored aggregate composed of disks, with its one plex. The aggregate has four RAID groups: rg0, rg1, rg2, and rg3. Each RAID group has 6 data disks, one parity disk, and one dparity (double parity) disk. All disks used by the aggregate come from the same pool, pool0.
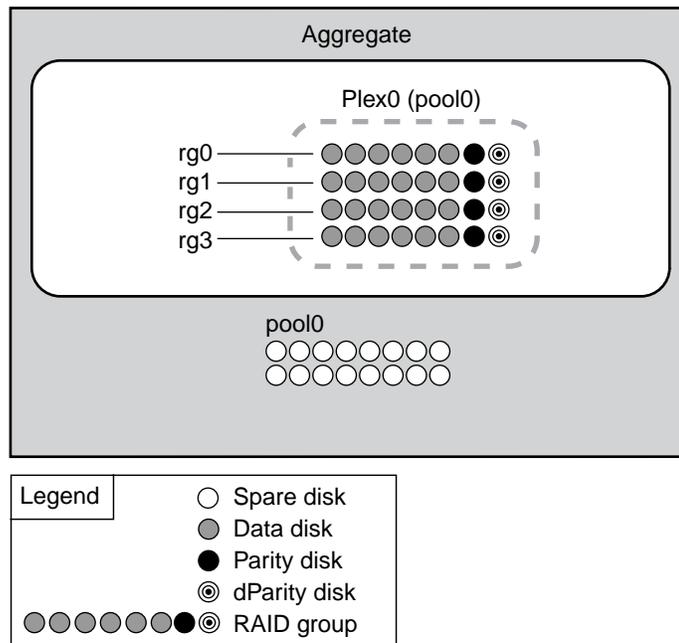
The following diagram shows an unmirrored aggregate with array LUNs, with its one plex. It has two RAID groups, rg0 and rg1. All array LUNs used by the aggregate come from the same pool, pool0.
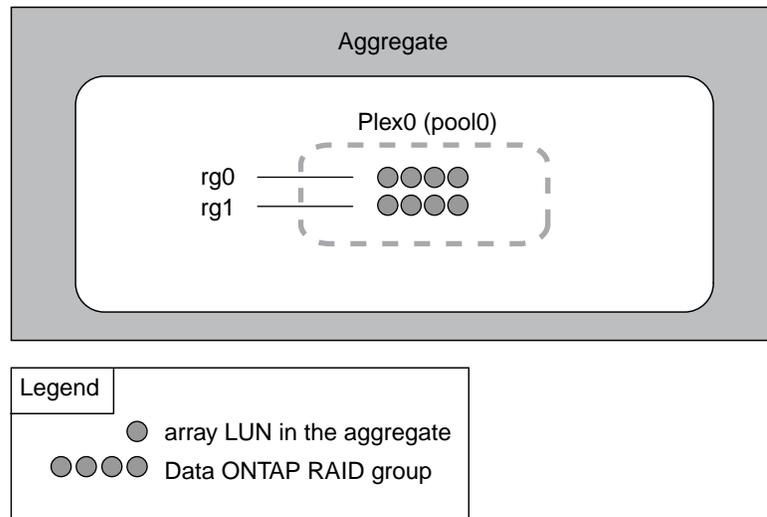
How mirrored aggregates work

Mirrored aggregates have two *plexes* (copies of their data), which use the SyncMirror functionality to duplicate the data to provide redundancy.

When a mirrored aggregate is created (or when a second plex is added to an existing unmirrored aggregate), ONTAP copies the data in the original plex (plex0) to the new plex (plex1). The plexes are physically separated (each plex has its own RAID groups and its own pool), and the plexes are updated simultaneously. This provides added protection against data loss if more disks fail than the RAID level of the aggregate protects against or there is a loss of connectivity, because the unaffected plex continues to serve data while you fix the cause of the failure. After the plex that had a problem is fixed, the two plexes resynchronize and reestablish the mirror relationship.

The disks and array LUNs on the system are divided into two pools: pool0 and pool1. Plex0 gets its storage from pool0 and plex1 gets its storage from pool1.

The following diagram shows an aggregate composed of disks with the SyncMirror functionality enabled and implemented. A second plex has been created for the aggregate, plex1. The data in plex1 is a copy of the data in plex0, and the RAID groups are also identical. The 32 spare disks are allocated to pool0 or pool1, 16 disks for each pool.
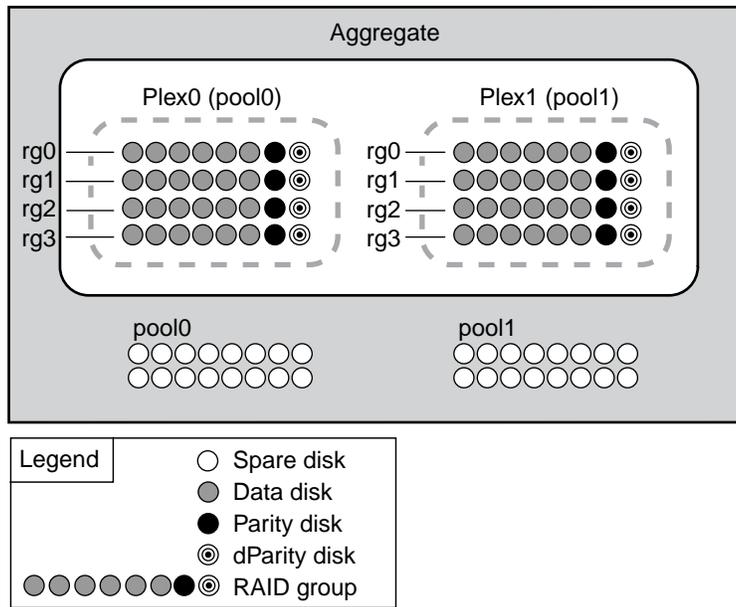
The following diagram shows an aggregate composed of array LUNs with the SyncMirror
functionality enabled and implemented. A second plex has been created for the aggregate, plex1.
Plex1 is a copy of plex0, and the RAID groups are also identical.
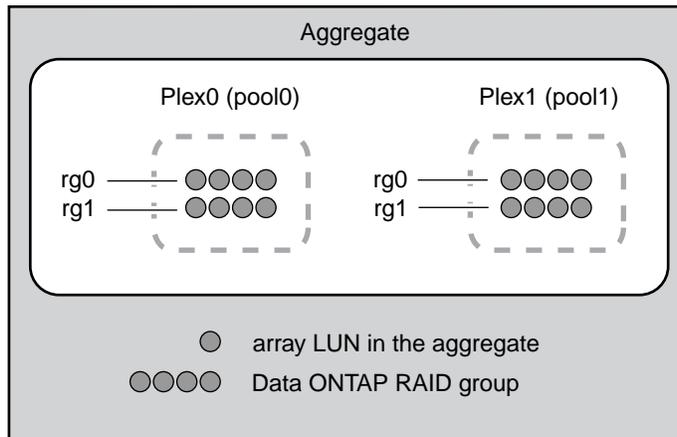
Where to find additional information

After you have setup the disks and aggregates for your storage system, you can perform additional tasks such as setting up volumes to contain your data.

- *ONTAP 9 commands*

  Describe commands for creating and managing aggregates in reference format.

- *Logical storage management*

  Describes how to manage logical storage resources in clusters, including FlexVol volumes, FlexClone volumes, FlexCache volumes, files, and LUNs.

- *SAN administration*

  Describes how to configure and manage iSCSI, FCoE, and FC protocols including configuration of LUNs, igroups and targets.

- *NetApp Technical Report 3483: Thin Provisioning in a NetApp SAN or IP SAN Enterprise Environment*

  Describes thin provisioning in a SAN or IP SAN enterprise environment.

- *NFS management*

  Describes how to manage file access to clusters using the NFS protocol, including authentication, authorization, and security

- *SMB/CIFS management*

  Describes how to configure CIFS servers and manage file access using the SMB protocol, including security, file policy management, and CIFS client- and server-based services.

Copyright and trademark

Copyright

Trademark