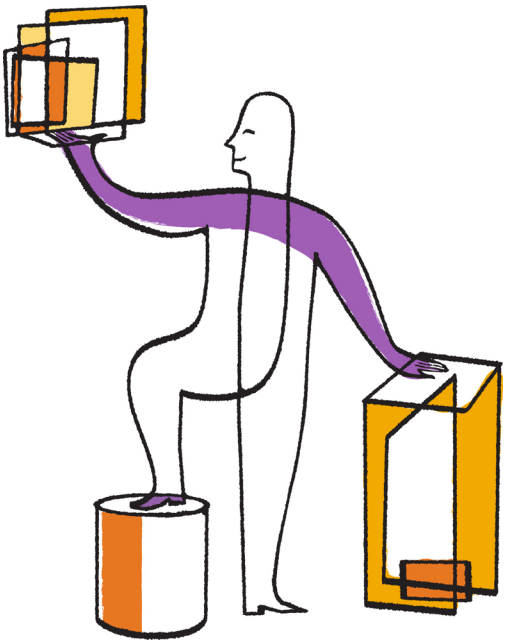
Updated for 8.2.1

Clustered Data ONTAP® 8.2

Logical Storage Management Guide



NetApp, Inc.
495 East Java Drive
Sunnyvale, CA 94089
U.S.

Telephone: +1 (408) 822-6000
Fax: +1 (408) 822-4501
Support telephone: +1 (888) 463-8277
Web: www.netapp.com
Feedback: doccomments@netapp.com

Part number: 215-08509_B0
February 2014

Contents

What logical storage is

Logical storage refers to the storage resources provided by Data ONTAP that are not tied to a physical resource.

Logical storage resources are associated with a Storage Virtual Machine (SVM, formerly known as Vserver), and they exist independently of any specific physical storage resource such as a disk, array LUN, or aggregate. Logical storage resources include volumes of all types and qtrees, as well as the capabilities and configurations you can use with these resources, such as Snapshot copies, deduplication, compression, and quotas.

For more information about SVMs, see the *Clustered Data ONTAP System Administration Guide for Cluster Administrators* and the *Clustered Data ONTAP System Administration Guide for SVM Administrators*.

Related concepts

Using FlexVol volumes on page 19
Using qtrees to partition your FlexVol volumes on page 113
Using deduplication and data compression to increase storage efficiency on page 159
Using quotas to restrict or track resource usage on page 117

How volumes work

Volumes are data containers that enable you to partition and manage your data. Understanding the types of volumes and their associated capabilities enables you to design your storage architecture for maximum storage efficiency and ease of administration.

Volumes are the highest-level logical storage object. Unlike aggregates, which are composed of physical storage resources, volumes are completely logical objects.

Data ONTAP provides two types of volumes: FlexVol volumes and Infinite Volumes. There are also volume variations, such as FlexClone volumes, FlexCache volumes, data protection mirrors, and load-sharing mirrors. Not all volume variations are supported for both types of volumes. Data ONTAP efficiency capabilities, compression and deduplication, are supported for both types of volumes.

Volumes contain file systems in a NAS environment, and LUNs in a SAN environment.

Volumes are associated with one Storage Virtual Machine (SVM). The SVM is a virtual management entity, or server, that consolidates various cluster resources into a single manageable unit. When you create a volume, you specify the SVM it is associated with. The type of the volume (FlexVol volume or Infinite Volume) is determined by an immutable SVM attribute.

Volumes have a language. The language of the volume determines the character set Data ONTAP uses to display file names and data for that volume. The default value for the language of the volume is the language of the SVM.

Volumes depend on their associated aggregates for their physical storage; they are not directly associated with any concrete storage objects, such as disks or RAID groups. If the cluster administrator has assigned specific aggregates to an SVM, then only those aggregates can be used to provide storage to the volumes associated with that SVM. This impacts volume creation, and also copying and moving FlexVol volumes between aggregates.

For more information about Infinite Volumes, see the *Clustered Data ONTAP Infinite Volumes Management Guide*.

For more information about SVMs, see the *Clustered Data ONTAP System Administration Guide for Cluster Administrators*.

For more information about data protection mirrors, see the *Clustered Data ONTAP Data Protection Guide*.

For more information about physical storage resources such as aggregates, disks, and RAID groups, see the *Clustered Data ONTAP Physical Storage Management Guide*.

What a FlexVol volume is

A FlexVol volume is a data container associated with a Storage Virtual Machine (SVM) with FlexVol volumes. It gets its storage from a single associated aggregate, which it might share with other FlexVol volumes or Infinite Volumes. It can be used to contain files in a NAS environment, or LUNs in a SAN environment.

Capabilities that FlexVol volumes provide

FlexVol volumes enable you to partition your data into individual manageable objects that can be configured to suit the needs of the users of that data.

A FlexVol volume enables you to take the following actions:

- Create a clone of the volume quickly and without having to duplicate the entire volume by using FlexClone technology.
- Reduce the space requirements of the volume by using deduplication and compression technologies.
- Create a sparse copy of the volume to balance loads or reduce network latency by using FlexCache technology.
- Create a Snapshot copy of the volume for data protection purposes.
- Limit the amount of space a user, group, or qtree can use in the volume by using quotas.
- Partition the volume by using qtrees.
- Create load-sharing mirrors to balance loads between nodes.
- Move the volume between aggregates and between storage systems.
- Make the volume available to client access using any file access protocol supported by Data ONTAP.
- Set up a volume to make more storage available when it becomes full.
- Create a volume that is bigger than the physical storage currently available to it by using thin provisioning.

Related concepts

Using FlexCache volumes to accelerate data access on page 64
Using FlexClone volumes to create efficient copies of your FlexVol volumes on page 80
Using FlexClone files and FlexClone LUNs to create efficient copies of files and LUNs on page 87
Configuring deduplication on page 159
Configuring data compression on page 163
Using a load-sharing mirror to balance loads on page 99
Moving and copying volumes (cluster administrators only) on page 59
How Data ONTAP can automatically provide more space for full FlexVol volumes on page 33

Differences among FlexVol volume features

Understanding the differences among different FlexVol volume features helps you choose the best feature to suit your requirements.

The following table summarizes those differences:

Feature	Access type (read-write or read-only)	Automatically mounted?	Full copy or shared blocks?	Location	Instantaneous or longer-running operation?
FlexCache	Same as the origin volume	Yes	Full (sparse) copy	Different aggregate; same or different node	Instantaneous (cache is populated over time)
FlexClone	Same as the parent volume	Yes	Shared blocks	Same aggregate and node	Instantaneous
Snapshot copy	Read-only	Yes	Shared blocks	Same aggregate and node	Instantaneous
Copy	Same as the original volume	No	Full copy	Same or different aggregate; same or different node	Longer-running operation
Data protection mirror	Read-only	No	Full copy	Same or different aggregate; same or different node; same or different cluster	Longer-running operation
Load-sharing mirror	Read-only	No	Full copy	Same or different aggregate; same or different node	Longer-running operation

Feature	Access type (read-write or read-only)	Automatically mounted?	Full copy or shared blocks?	Location	Instantaneous or longer-running operation?
Move (DataMotion for Volumes)	Same as the original volume	Yes	Full copy, then deletion of original	Different aggregate; same or different node	Longer-running operation

All of these volume features occur within the same Storage Virtual Machine (SVM), except for data protection mirrors, which can cross clusters and SVMs.

The time that it takes to do a longer-running operation depends on the size of the volume. For example, moving a 1-TB volume might take several hours.

Related concepts

Using FlexCache volumes to accelerate data access on page 64
Using FlexClone volumes to create efficient copies of your FlexVol volumes on page 80
Moving and copying volumes (cluster administrators only) on page 59
Using a load-sharing mirror to balance loads on page 99

What an Infinite Volume is

An Infinite Volume is a single, scalable volume that can store up to 2 billion files and tens of petabytes of data.

With an Infinite Volume, you can manage multiple petabytes of data in one large logical entity and clients can retrieve multiple petabytes of data from a single junction path for the entire volume.

An Infinite Volume uses storage from multiple aggregates on multiple nodes. You can start with a small Infinite Volume and expand it nondisruptively by adding more disks to its aggregates or by providing it with more aggregates to use.

Capabilities that Infinite Volumes provide

Infinite Volumes enable you to store multiple petabytes of data in a single volume that supports multiprotocol access, storage efficiency technologies, and data protection capabilities.

With Infinite Volumes, you can perform the following tasks:

- Manage multiple petabytes of data in a single logical entity with a single junction path and a single namespace.
- Provide multiprotocol access to that data using NFSv3, NFSv4.1, pNFS, and CIFS (SMB 1.0).
- Offer secure multi-tenancy by creating multiple Storage Virtual Machines (SVMs) with FlexVol volumes and multiple Storage Virtual Machines (SVMs) with Infinite Volume in a single cluster.

- Create an Infinite Volume that is larger than the available physical storage by using thin provisioning.
- Maximize storage efficiency by using deduplication and compression technologies.
- Optimize storage by grouping it into storage classes that correspond to specific goals
- Automatically place incoming files into the appropriate storage class according to rules based on file name, file path, or file owner.
- Protect data by creating Snapshot copies of the volume.
- Create a data protection mirror relationship between two Infinite Volumes on different clusters, and restore data when necessary.
- Back up data with CIFS or NFS over a mounted volume to tape, and restore data when necessary.
- Expand the Infinite Volume by adding more disks to the aggregates used by the Infinite Volume or by assigning more aggregates to the SVM containing the Infinite Volume and then resizing the Infinite Volume.

Comparison of FlexVol volumes and Infinite Volumes

Both FlexVol volumes and Infinite Volumes are data containers. However, they have significant differences that you should consider before deciding which type of volume to include in your storage architecture.

The following table summarizes the differences and similarities between FlexVol volumes and Infinite Volumes:

Volume capability or feature	**FlexVol volumes**	**Infinite Volumes**	**Notes**
Containing entity	SVM; single node	SVM; can span nodes	
Number of associated aggregates	One	Multiple	
Maximum size	32-bit volumes: 16 TB 64-bit volumes: model-dependent	Up to 20 PB	For information about the maximum size of 64-bit volumes, see the *Hardware Universe*.
Minimum size	20 MB	Approximately 1.33 TB for each node used	
Type of Storage Virtual Machine (SVM)	SVM with FlexVol volumes	SVM with Infinite Volume	
Maximum number per SVM	Model- and protocol-dependent	One	For more information, see the *Hardware Universe*.

Volume capability or feature	**FlexVol volumes**	**Infinite Volumes**	**Notes**
Maximum number per node	Model-dependent	Model-dependent	For more information, see the *Hardware Universe*.
Types of aggregates supported	64-bit or 32-bit	64-bit	
SAN protocols supported	Yes	No	
File access protocols supported	NFS, CIFS	NFS, CIFS	
Deduplication	Yes	Yes	
Compression	Yes	Yes	
FlexClone volumes	Yes	No	
FlexCache volumes	Yes	No	
Quotas	Yes	No	
Qtrees	Yes	No	
Thin provisioning	Yes	Yes	
Snapshot copies	Yes	Yes	
Data protection mirrors	Yes	Yes	For Infinite Volumes, only mirrors between clusters are supported.
Load-sharing mirrors	Yes	No	
Antivirus	Yes	No	
Tape backup	Yes	Yes	For Infinite Volumes, you must use NFS or CIFS rather than NDMP.
Volume security styles	UNIX, NTFS, mixed	Unified	For more information, see the *Clustered Data ONTAP File Access Management Guide for CIFS* or the *Clustered Data ONTAP File Access Management Guide for NFS*.

For more information about Infinite Volumes, see the *Clustered Data ONTAP Infinite Volumes Management Guide*.

Related references

Storage limits on page 189

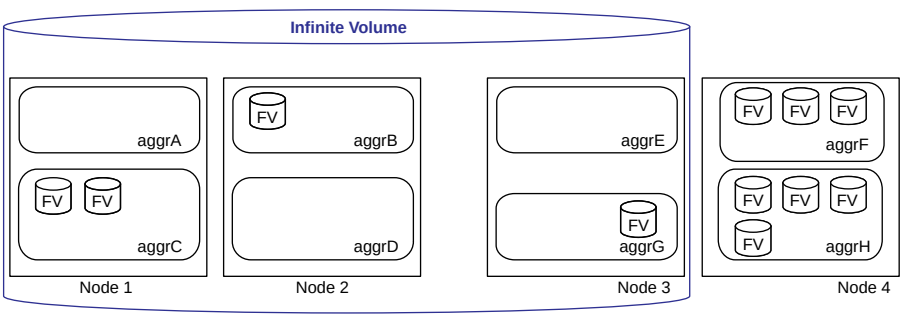
How FlexVol volumes and Infinite Volumes share aggregates

Aggregates can be shared among the volumes in a cluster. Each aggregate can contain multiple FlexVol volumes alongside multiple constituents of Infinite Volumes.

When you create an Infinite Volume, constituents of the Infinite Volume are placed on aggregates that are assigned to its containing Storage Virtual Machine (SVM). If the SVM with Infinite Volume includes aggregates that contain FlexVol volumes, one or more of the Infinite Volume's constituents might be placed on aggregates that already include FlexVol volumes, if those aggregates meet the requirements for hosting Infinite Volumes.

Similarly, when you create a FlexVol volume, you can associate that FlexVol volume with an aggregate that is already being used by an Infinite Volume.

The following diagram illustrates aggregate sharing in a four-node cluster that includes both FlexVol volumes and an Infinite Volume. The Infinite Volume uses the aggregates aggrA, aggrB, aggrC, aggrD, aggrE, and aggrG even though the aggregates aggrB, aggrC, and aggrG already provide storage to FlexVol volumes. (For clarity, the individual constituents that make up the Infinite Volume are not shown.)



How security styles affect data access

Each volume and qtree on the storage system has a security style. The security style determines what type of permissions are used for data on volumes when authorizing users. You must understand what

the different security styles are, when and where they are set, how they impact permissions, how they differ between volume types, and more.

For more information about security styles, see the *Clustered Data ONTAP File Access Management Guide for CIFS* or *Clustered Data ONTAP File Access Management Guide for NFS*.

Improving client performance with traditional and lease oplocks

Traditional oplocks (opportunistic locks) and lease oplocks enable an SMB client in certain file-sharing scenarios to perform client-side caching of read-ahead, write-behind, and lock information. A client can then read from or write to a file without regularly reminding the server that it needs access to the file in question. This improves performance by reducing network traffic.

Lease oplocks are an enhanced form of oplocks available with the SMB 2.1 protocol and later. Lease oplocks allow a client to obtain and preserve client caching state across multiple SMB opens originating from itself.

Lease oplocks are not supported on Storage Virtual Machines (SVMs) with Infinite Volumes.

For more information, see the *Clustered Data ONTAP File Access Management Guide for CIFS*.

What system volumes are

System volumes are FlexVol volumes that contain special metadata, such as metadata for file services audit logs. These volumes are visible in the cluster so that you can fully account for the storage use in your cluster.

System volumes are owned by the cluster management server (also called the admin SVM), and they are created automatically when file services auditing is enabled.

You can view system volumes by using the `volume show` command, but most other volume operations are not permitted. For example, you cannot modify a system volume by using the `volume modify` command.

This example shows four system volumes on the admin SVM, which were automatically created when file services auditing was enabled for a data SVM in the cluster:

```
cluster1::> volume show -vserver cluster1
Vserver   Volume       Aggregate    State      Type       Size  Available Used%
--------- ------------ ------------ ---------- ---- ---------- ---------- -----
cluster1  MDV_aud_1d0131843d4811e296fc123478563412
                       aggr0        online     RW          2GB     1.90GB    5%
cluster1  MDV_aud_8be27f813d7311e296fc123478563412
                       root_vs0     online     RW          2GB     1.90GB    5%
cluster1  MDV_aud_9dc4ad503d7311e296fc123478563412
                       aggr1        online     RW          2GB     1.90GB    5%
cluster1  MDV_aud_a4b887ac3d7311e296fc123478563412
```

```
                        aggr2        online     RW          2GB     1.90GB    5%
4 entries were displayed.
```

For more information about how file services auditing uses system volumes, see the *Clustered Data ONTAP File Access Management Guide for CIFS*.

Using FlexVol volumes

Most management tasks for FlexVol volumes are available to the SVM administrator. A few, such as promoting a volume to be the root volume of a Storage Virtual Machine (SVM) and moving or copying volumes, are available only to cluster administrators.

Differences between 64-bit and 32-bit FlexVol volumes

FlexVol volumes are one of two formats: 64-bit or 32-bit. A 64-bit volume has a larger maximum size than a 32-bit volume.

A newly created FlexVol volume is the same format as its associated aggregate. However, a volume can be a different format from its associated aggregate in certain cases.

The maximum size of a 64-bit volume is determined by the size of its associated aggregate, which depends on the storage system model.

A 32-bit volume has a maximum size of 16 TB.

Note: For both volume formats, the maximum size for each LUN or file is 16 TB.

Related references

Storage limits on page 189

Interoperability between 64-bit and 32-bit FlexVol volumes

Some Data ONTAP features use two FlexVol volumes; those volumes can be different formats. These features interoperate between the two volume formats.

Data ONTAP feature	Interoperates between 64-bit and 32-bit format?
FlexCache	Y
`ndmpcopy`	Y
`volume copy`	Y
Volume SnapMirror	Y
`volume move` (DataMotion for Volumes)	Y

How FlexVol volumes work with SVMs

Understanding how FlexVol volumes work with Storage Virtual Machines (SVMs) enables you to plan your storage architecture.

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

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

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

If the SVM for that volume has aggregates assigned to it, then you can use only one of those assigned aggregates to provide storage to volumes on that SVM. This can help you ensure that your SVMs are not sharing physical storage resources inappropriately. This segregation can be important in a multi-tenancy environment, because for some space management configurations, volumes that share the same aggregate can affect each other's access to free space when space is constrained for the aggregate. Aggregate assignment requirements apply to both cluster administrators and SVM administrators.

Volume move and volume copy operations are not constrained by the SVM aggregate assignments, so if you are trying to keep your SVMs on separate aggregates, you must ensure that you do not violate your SVM aggregate assignments when you perform those operations.

If the SVM for that volume has no aggregates assigned to it, then the cluster administrator can use any aggregate in the cluster to provide storage to the new volume. However, the SVM administrator cannot create volumes for SVMs with no assigned aggregates. For this reason, if you want your SVM administrator to be able to create volumes for a specific SVM, then you must assign aggregates to that SVM (`vserver modify -aggr-list`).

Changing the aggregates assigned to an SVM does not affect any existing volumes. For this reason, the list of aggregates assigned to an SVM cannot be used to determine the aggregates associated with volumes for that SVM.

For more information about configuring and managing SVMs, see the *Clustered Data ONTAP System Administration Guide for Cluster Administrators*.

How the SVM can limit how many FlexVol volumes it can have

You can limit the volumes for a Storage Virtual Machine (SVM) with FlexVol volumes to control resource usage or ensure that configuration-specific limits on the number of volumes per SVM are not exceeded.

The maximum volume limit per SVM is controlled with the `-max-volumes` parameter for the SVM. By default, there is no limit imposed on the number of volumes the SVM can have.

The SVM maximum volume limit is applied only if the SVM also has an aggregate list. It is applied for both SVM administrators and cluster administrators.

How the SVM affects the language of the FlexVol volume

The language of the Storage Virtual Machine (SVM) determines the default value for the language of a FlexVol volume, although you can override that value at volume creation time. If you change the language of the SVM, it does not affect its existing FlexVol volumes. You cannot change the language of a FlexVol volume.

For FlexCache volumes and FlexClone volumes, the default language is the language of the parent volume.

Restoring the root volume of an SVM

If the root volume of a Storage Virtual Machine (SVM) becomes unavailable, clients cannot mount the root of the namespace. In such cases, you must restore the root volume by promoting another volume to facilitate data access to the clients.

About this task

When the SVM root volume becomes unavailable, you can restore the root volume by promoting another volume, which does not have other volumes junctioned to it.

For SVMs with FlexVol volumes, you can promote one of the following volumes as the root volume:

- Load-sharing mirror copy
- Data-protection mirror copy
- A new FlexVol volume

Note: If you want to restore the root volume of an SVM with Infinite Volume, you must contact technical support.

Starting from clustered Data ONTAP 8.2, SVM root volume is created with 1 GB size to prevent any failures when mounting any volume in the SVM root volume due to lack of space or inodes. Therefore, if you are promoting a new FlexVol volume, it should be at least 1 GB in size.

Steps

1. Depending on the type of volume you select for promoting a root volume, perform the appropriate action:

If you want to promote...	Perform the following tasks...
A load-sharing mirror as the root volume of an SVM	*Promoting load-sharing mirror copy* on page 22
A data-protection mirror as the root volume of an SVM	*Promoting data-protection mirror copy* on page 23
A new FlexVol volume	*Promoting new FlexVol volume* on page 24

2. Use the `volume mount` command to remount the new root volume.

   For more information about these commands, see the man pages.

Result

When the new volume is promoted as the SVM root volume, the other data volumes get associated with the new SVM root volume.

Promoting a load-sharing mirror copy

You can promote a load-sharing mirror copy to restore the root volume of a Storage Virtual Machine (SVM).

Steps

1. Use the `set -privilege advanced` command to set the privilege level to advanced.
2. Use the `snapmirror promote` command to promote the load-sharing mirror copy as the root volume.
3. Use the `vol show` command to verify the new root volume of the SVM.
4. Use the `vol rename` command to rename the volume that was promoted as the root volume.

   For more information about these commands, see the man pages.

The following example shows how to promote a load-sharing mirror copy vol_dstls as the root volume of the SVM vs1.example.com:

```
cluster1::> set -privilege advanced

Warning: These advanced commands are potentially dangerous; use them only
when directed to do so by technical support.
Do you want to continue? {y|n}: y

cluster1::*> snapmirror promote -destination-path vs1.example.com:vol_dstls

Warning: Promote will delete the read-write volume cluster1://
```

```
vs1.example.com/vol1 and replace it with cluster1://vs1.example.com/
vol_dstls.
Do you want to continue? {y|n}: y
[Job 489] Job succeeded: SnapMirror: done

cluster1::*> volume show -volume vol_dstls -instance

              Vserver Name: vs1.example.com
              Volume Name: vol_dstls
              .
              .
              Junction Path: /
              .
              Vserver Root Volume: true
              .
              .
```

Promoting a data-protection mirror copy

You can use a data-protection mirror copy to restore the root volume of a Storage Virtual Machine (SVM).

Steps

1. Use the `snapmirror break` command to break the SnapMirror relationship.
2. Use the `set -privilege advanced` command to set the privilege level to advanced.
3. Use the `volume make-vsroot` command to promote the data-protection mirror copy as the root volume.
4. Use the `volume show` command to verify the new root volume of the SVM.
5. Use the `volume rename` command to rename the volume that was promoted as the root volume.

   For more information about these commands, see the man pages.

The following example shows how to promote a data-protection mirror copy vol_dstdp as the root volume of the SVM vs1.example.com:

```
cluster1::
> snapmirror break -destination-path vs1.example.com:vol_dstdp
[Job 521] Job succeeded: SnapMirror Break Succeeded

cluster1::> set -privilege advanced

Warning: These advanced commands are potentially dangerous; use them only
when directed to do so by technical support.
Do you want to continue? {y|n}: y

cluster1::*> volume make-vsroot -volume vol_dstdp -vserver vs1.example.com
[Job 522] Job succeeded: DONE

cluster1::*> volume show -volume vol_dstdp -instance
```

```
              Vserver Name: vs1.example.com
              Volume Name: vol_dstdp
              .
              .
              Junction Path: /
              .
              Vserver Root Volume: true
              .
              .
```

Promoting a new FlexVol volume

You can create and use a new FlexVol volume to restore the root volume of a Storage Virtual Machine (SVM).

Steps

1. Use the `set -privilege advanced` command to set the privilege level to advanced.
2. Use the `volume create` command to create a new FlexVol volume of 1 GB size.
3. Use the `volume make-vsroot` command to promote the FlexVol volume as the root volume.
4. Use the `volume show` command to verify the new root volume of the SVM.
5. Use the `volume rename` command to rename the volume that was promoted as the root volume.

   For more information about these commands, see the man pages.

The following example shows how to promote a FlexVol volume new_rootvol as the root volume of the SVM vs1.example.com:

```
cluster1::> set -privilege advanced

Warning: These advanced commands are potentially dangerous; use them only
when directed to do so by technical support.
Do you want to continue? {y|n}:

cluster1::*> vol create -vserver vs3 -volume new_rootvol -aggregate aggr0 -
size 1GB
  (volume create)

cluster1::*> volume make-vsroot -vserver vs1.example.com -volume new_rootvol

cluster1::*> volume show -volume new_rootvol -instance

              Vserver Name: vs1.example.com
              Volume Name: new_rootvol
              .
              .
              Junction Path: /
              .
              Vserver Root Volume: true
              .
              .
```

Volume junction usage rules

Volume junctions are a way to join individual volumes together into a single, logical namespace to enable data access to NAS clients. Understanding how volume junctions are formed helps you to interpret and apply the usage rules.

When NAS clients access data by traversing a junction, the junction appears to be an ordinary directory. A junction is formed when a volume is mounted to a mount point below the root and is used to create a file-system tree. The top of a file-system tree is always the root volume, which is represented by a slash (`/`). A junction leads from a directory in one volume to the root directory of another volume.

- Although specifying a junction point is optional when a volume is created, data in the volume cannot be exported (NFS) and a share cannot be created (CIFS) until the volume is mounted to a junction point in the namespace.
- A volume that was not mounted during volume creation can be mounted post-creation.
- New volumes can be added to the namespace at any time by mounting them to a junction point.
- Mounted volumes can be unmounted; however, unmounting a volume disrupts NAS client access to all data in the volume and to all volumes mounted at child junction points beneath the unmounted volume.
- Junction points can be created directly below a parent volume junction, or they can be created on a directory within a volume.

  For example, a path to a volume junction for a volume named "vol3" might be `/vol1/vol2/vol3`, or it might be `/vol1/dir2/vol3`, or even `/dir1/dir2/vol3`.

For more information, see the *Clustered Data ONTAP File Access Management Guide for CIFS* or the *Clustered Data ONTAP File Access Management Guide for NFS*.

How you use space management capabilities

To use the storage provided by FlexVol volumes as effectively as possible, you need to understand the space management capabilities that help you balance overall available storage against required user and application storage needs.

Data ONTAP enables space management using the following capabilities:

- Volume (space) guarantee

  The *volume guarantee*, also called *space guarantee* or just *guarantee*, determines how much space for the volume is preallocated from the volume's associated aggregate when the volume is created.
- Reservations

  Reservations, also called *space reservations*, *file reservations*, or *LUN reservations*, determine whether space for a particular file or LUN is preallocated from the volume.
- Fractional reserve

Fractional reserve, also called *fractional overwrite reserve* or *LUN overwrite reserve*, enables you to control the size of the overwrite reserve for a FlexVol volume.

- Automatic free space preservation

  Automatic free space preservation can either increase the size of a volume or delete Snapshot copies to prevent a volume from running out of space—all without operator intervention.

These capabilities are used together to enable you to determine, on a volume-by-volume basis, whether to emphasize storage utilization, ease of management, or something in between.

Related concepts

How volume guarantees work with FlexVol volumes on page 26
How file and LUN reservations work for FlexVol volumes on page 31
Considerations for setting fractional reserve for FlexVol volumes on page 32
How Data ONTAP can automatically provide more space for full FlexVol volumes on page 33

How volume guarantees work with FlexVol volumes

Volume guarantees (sometimes called *space guarantees*) determine how space for a volume is allocated from its containing aggregate—whether the space is preallocated for the entire volume or for only the reserved files or LUNs in the volume, or whether space for user data is not preallocated.

The guarantee is an attribute of the volume.

You set the guarantee when you create a new volume; you can also change the guarantee for an existing volume by using the `volume modify` command with the `-space-guarantee` option. You can view the guarantee type and status by using the `volume show` command.

Volume guarantee types can be `volume` (the default type), `file`, or `none`.

- A guarantee type of `volume` allocates space in the aggregate for the volume when you create the volume, regardless of whether that space is used for data yet.

  This approach to space management is called *thick provisioning*. The allocated space cannot be provided to or allocated for any other volume in that aggregate.

  When you use thick provisioning, all of the space specified for the volume is allocated from the aggregate at volume creation time. The volume cannot run out of space before the amount of data it contains (including Snapshot copies) reaches the size of the volume. However, if your volumes are not very full, this comes at the cost of reduced storage utilization.
- A guarantee type of `file` allocates space for the volume in its containing aggregate so that any reserved LUN or file in the volume can be completely rewritten, even if its blocks are being retained on disk by a Snapshot copy.

  However, writes to any file in the volume that is not reserved could run out of space.

  Before configuring your volumes with a guarantee of `file`, you should refer to *Technical Report 3965: Thin Provisioning Deployment and Implementation Guide*. You should also be aware that volume guarantees of type `file` will not be supported in a future release of Data ONTAP.
- A guarantee of `none` allocates space from the aggregate only as it is needed by the volume.

This approach to space management is called *thin provisioning*. The amount of space consumed by volumes with this guarantee type grows as data is added instead of being determined by the initial volume size, which might leave space unused if the volume data does not grow to that size. The maximum size of a volume with a guarantee of `none` is not limited by the amount of free space in its aggregate. It is possible for the total size of all volumes associated with an aggregate to exceed the amount of free space for the aggregate.

Writes to LUNs or files (including space-reserved LUNs and files) contained by that volume could fail if the containing aggregate does not have enough available space to accommodate the write. If you configure your volumes with a volume guarantee of `none`, you should refer to *Technical Report 3965: Thin Provisioning Deployment and Implementation Guide* for information about how doing so can affect storage availability.

When space in the aggregate is allocated for a `volume` or `file` guarantee for an existing volume, that space is no longer considered free in the aggregate, even if the volume is not yet using the space. Operations that consume free space in the aggregate, such as creation of aggregate Snapshot copies or creation of new volumes in the containing aggregate, can occur only if there is enough available free space in that aggregate; these operations are prevented from using space already allocated to another volume.

When the free space in an aggregate is exhausted, only writes to volumes or files in that aggregate with preallocated space are guaranteed to succeed.

Guarantees are honored only for online volumes. If you take a volume offline, any allocated but unused space for that volume becomes available for other volumes in that aggregate. When you try to bring that volume back online, if there is insufficient available space in the aggregate to fulfill its guarantee, it will remain offline. You must force the volume online, at which point the volume's guarantee will be disabled.

Related information

Technical Report: Thin Provisioning Deployment and Implementation Guide: media.netapp.com/documents/tr-3965.pdf

Enabling guarantees for FlexVol volumes

When a volume’s guarantee is disabled, the volume functions as though it has a guarantee of `none`. If you have volumes with disabled guarantees, you should address the situation by making more space available to those volumes as soon as possible.

Before you begin

The FlexVol volume must be online.

About this task

Enabled guarantees preallocate space in the aggregate. In volumes with disabled guarantees, operations that require space, such as writes and even deletions, might be disallowed. If a volume's guarantee becomes disabled, you should reenable the guarantee to be able to manually increase the

volume size. Volumes with a disabled guarantee and the autogrow feature enabled can still automatically increase in size.

You can view the status of the volume's guarantee first or try to enable the guarantee. If enabling the guarantee fails, Data ONTAP provides the reason (typically insufficient space) and specifies the amount of free space needed in the aggregate. A guarantee type of `none` is never disabled, because there is no space allocated for this guarantee type.

You can set a volume's guarantee to `none`. In this case, there is no concept of enabled or disabled, because the guarantee requires no space.

Steps

1. Optional: View the status of the volume's guarantee and the guarantee type by using the `volume show` command with the `–fields`, `-space-guarantee`, and `-space-guarantee-enabled` parameters.

   #### Example

   The command in this example displays the status of the guarantee for a volume called vol2 on the Storage Virtual Machine (SVM) named vs0. The guarantee is disabled (false).

   ```
   cluster1::> volume show -vserver vs0 -volume vol2 -fields space-guarantee,
   space-guarantee-enabled

   vserver volume space-guarantee space-guarantee-enabled
   ------- ------ --------------- ----------------------
   vs0     vol2   volume          false
   ```

   The output displays the guarantee type and whether the guarantee is enabled or disabled for the specified volumes. If the value in the `space-guarantee-enabled` column is `true`, the guarantee is enabled; if the value is `false`, the guarantee is disabled.

2. Enable or reenable the guarantee.

If you want to enable a guarantee for...	Then use this command...
A single volume	`volume modify` *`vol_name`* `-space-guarantee` *`guarantee_type`* This command enables the guarantee of a single volume with the specified guarantee type (`volume` or `file`), if there is space available to do so. If you specify a different guarantee than the one currently configured for this volume, Data ONTAP changes the guarantee to the one you specify and enables that guarantee.

If you want to enable a guarantee for...	Then use this command...

All volumes with the same guarantee type	`volume modify` { `-space-guarantee` *`guarantee_type`* `-space-guarantee-enabled` *`false`* } `-space-guarantee` *`guarantee_type`*
	This command enables the guarantees of all volumes with the specified guarantee type (`volume` or `file`).
	Make sure that the guarantee type specified in the query string (between the curly brackets) is the same as the guarantee type specified as the target guarantee type. Otherwise, this command changes the guarantee type of the volumes.
	In the following example, the command reenabled the guarantees for volumes named v1 and v3, which both have a guarantee type of `volume`:
	`cluster1::> volume modify { -space-guarantee volume -space-guarantee-enabled false } -space-guarantee volume`
	`Volume modify successful on volume: v1`
	`Volume modify successful on volume: v3`
	`2 entries were modified.`

The guarantee is enabled, or you receive an error message that tells you how much space you need to create in the aggregate before the guarantee can be enabled.

If you used the command to reenable multiple guarantees of the same type, the guarantees of all volumes with the specified guarantee type are enabled if there is enough free space to accommodate their guarantees.

3. If there is not enough space in the aggregate to enable the guarantee, you must create more space.

Example

The following is an example of an error message displayed when trying to enable a guarantee for a volume named testvol:

```
cluster1::> volume modify testvol -s volume
Error: command failed: Unable to set volume attribute "space-guarantee" for volume
"testvol"
on Vserver "vs1".
Reason: Request to enable guarantee for this volume failed because there is not enough
space
in the aggregate. Create 4.81MB of free space in the aggregate.
```

4. Try to enable the guarantee again, and view the command output to see whether the guarantee is now enabled.

   If the guarantee is still not enabled, you must try another method of creating more space.

5. Optional: If you used one of the commands to reenable multiple guarantees of the same type, verify that all of the guarantees were enabled by using the `volume show` command with the `-fields space-guarantee,space-guarantee-enabled` parameters.

   Example

```
cluster1::> volume show -aggregate testaggr -fields space-guarantee,space-
guarantee-enabled
  (volume show)
vserver volume space-guarantee space-guarantee-enabled
------- ------ -------------- ----------------------
thevs   v1     volume         true
thevs   v2     volume         true
thevs   v3     volume         true
thevs   v4     file           true
thevs   v5     none           true
5 entries were displayed.
```

   The guarantees that were enabled are displayed in the `space-guarantee-enabled` column with a value of `true`. Any guarantees that are not enabled are displayed with a value of `false`.

Related concepts

How volume guarantees work with FlexVol volumes on page 26
Methods to create space in a FlexVol volume on page 49
Methods to create space in an aggregate on page 51

How the guarantee affects FlexVol volume space requirements

The amount of space that a FlexVol volume requires from its aggregate varies depending on the volume's guarantee type. Understanding a volume's space requirement helps you predict how much space becomes available or is required when you change its guarantee or delete the volume.

A volume with a guarantee type of `none` requires space in the aggregate only for data that is already written to it.

A volume with a guarantee type of `volume` requires an amount of space in the aggregate equivalent to the volume's size, regardless of how much data (if any) is actually in the volume.

A volume with a guarantee type of `file` requires enough space in the aggregate to enable writes and overwrites to reserved files or LUNs, even if a block being overwritten is locked by a Snapshot copy or other block-sharing technology.

Related concepts

How volume guarantees work with FlexVol volumes on page 26
What the volume footprint is on page 43

Considerations for using thin provisioning with FlexVol volumes

Using thin provisioning, you can configure your volumes so that they appear to provide more storage than they have available, provided that the storage that is actually being used does not exceed the available storage.

To use thin provisioning with FlexVol volumes, you create the volume with a guarantee of `none`. With a guarantee of `none`, the volume size is not limited by the aggregate size. In fact, each volume could, if required, be larger than the containing aggregate. The storage provided by the aggregate is used up only as data is written to the LUN or file.

If the volumes associated with an aggregate show more storage as available than the physical resources available to that aggregate, the aggregate is *overcommitted*. When an aggregate is overcommitted, it is possible for writes to LUNs or files in volumes contained by that aggregate to fail if there is not sufficient free space available to accommodate the write.

If you have overcommitted your aggregate, you must monitor your available space and add storage to the aggregate as needed to avoid write errors due to insufficient space.

Aggregates can provide storage to FlexVol volumes associated with more than one Storage Virtual Machine (SVM). When sharing aggregates for thin-provisioned volumes in a multi-tenancy environment, be aware that one tenant's aggregate space availability can be adversely affected by the growth of another tenant's volumes.

For more information about thin provisioning, see the following technical reports:

- *TR 3965: NetApp Thin Provisioning Deployment and Implementation Guide*
- *TR 3483: Thin Provisioning in a NetApp SAN or IP SAN Enterprise Environment*

How file and LUN reservations work for FlexVol volumes

When reservations are enabled for one or more files or LUNs, Data ONTAP reserves enough space in the volume so that writes to those files or LUNs do not fail because of a lack of disk space.

Reservations are an attribute of the file or LUN; they are persistent across storage system reboots, takeovers, and givebacks. Reservations are enabled for new LUNs by default, but you can create a file or LUN with reservations disabled or enabled. After you create a LUN, you change the reservation attribute by using the `lun modify` command. You change the reservation attribute for files by using the `file reservation` command.

When a volume contains one or more files or LUNs with reservations enabled, operations that require free space, such as the creation of Snapshot copies, are prevented from using the reserved space. If these operations do not have sufficient unreserved free space, they fail. However, writes to the files or LUNs with reservations enabled continue to succeed.

You can enable reservations for files and LUNs contained by volumes with volume guarantees of any value. However, if the volume has a guarantee of `none`, reservations do not provide protection against out-of-space errors.

Example

If you create a 100-GB space-reserved LUN in a 500-GB volume, that 100 GB of space is immediately allocated, leaving 400 GB remaining in the volume. In contrast, if space reservation is disabled on the LUN, all 500 GB in the volume remain available until writes are made to the LUN.

Considerations for setting fractional reserve for FlexVol volumes

Fractional reserve, also called *LUN overwrite reserve*, enables you to control the size of the overwrite reserve for reserved LUNs and files in a FlexVol volume. By using this volume attribute correctly you can maximize your storage utilization, but you should understand how it interacts with other technologies.

The fractional reserve setting is expressed as a percentage; the only valid values are 0 and 100 percent. You use the `volume modify` command to set fractional reserve.

Setting fractional reserve to 0 increases your storage utilization. However, an application accessing data residing in the volume could experience a data outage if the volume is out of free space, even with the volume guarantee set to `volume`, when any of the following technologies and Data ONTAP features are in use:

- Deduplication
- Compression
- FlexClone files
- FlexClone LUNs
- Virtual environments

If you are using one or more of these technologies with no fractional reserve and you need to prevent errors due to running out of space, you must use all of the following configuration settings for the volume:

- Volume guarantee of `volume`
- File or LUN reservations `enabled`
- Volume Snapshot copy automatic deletion enabled with a commitment level of `destroy` and a destroy list of `lun_clone,vol_clone,cifs_share,file_clone,sfsr`

  Note: If your rate of change is high, in rare cases the Snapshot copy automatic deletion could fall behind, resulting in the volume running out of space, even with all of the required configuration settings in use.

In addition, you can optionally use the volume autogrow capability to decrease the likelihood of volume Snapshot copies needing to be deleted automatically. If you enable the autogrow capability, you must monitor the free space in the associated aggregate. If the aggregate becomes full enough that the volume is prevented from growing, more Snapshot copies will probably be deleted as the free space in the volume is depleted.

If you do not want to monitor aggregate free space or have volume Snapshot copies automatically deleted, you can set the volume's fractional reserve setting to 100. This requires more free space up front, but guarantees that data modification operations will succeed even when the technologies listed above are in use.

The default value and allowed values for the fractional reserve setting depend on the guarantee of the volume:

Volume guarantee	Default fractional reserve	Allowed values
Volume	100	0, 100
None	0	0, 100
File	100	100

For more information about using fractional reserve, see the following Technical Reports:

- *TR-3965: Thin Provisioning Deployment and Implementation Guide*
- *TR-3483: Thin Provisioning in a NetApp SAN or IP SA Enterprise Environment*

Related concepts

How volume guarantees work with FlexVol volumes on page 26
How file and LUN reservations work for FlexVol volumes on page 31
How a FlexVol volume can automatically change its size on page 35
How the FlexVol volume and aggregate fullness alerts work on page 52

Related tasks

Deleting Snapshot copies automatically on page 38

Related information

Technical Report: Thin Provisioning Deployment and Implementation Guide: media.netapp.com/documents/tr-3965.pdf
Technical Report: Thin Provisioning in a NetApp SAN or IP SAN Enterprise Environment: media.netapp.com/documents/tr3483.pdf

How Data ONTAP can automatically provide more space for full FlexVol volumes

Data ONTAP uses two methods for automatically providing more space for a FlexVol volume when that volume is nearly full: allowing the volume size to increase, and deleting Snapshot copies (with

any associated storage objects). If you enable both of these methods, you can specify which method Data ONTAP should try first.

Data ONTAP can automatically provide more free space for the volume by using one of the following methods:

- Increase the size of the volume when it is nearly full (known as the *autogrow* feature).
  This method is useful if the volume's containing aggregate has enough space to support a larger volume. You can configure Data ONTAP to increase the size in increments and set a maximum size for the volume. The increase is automatically triggered based on the amount of data being written to the volume in relation to the current amount of used space and any thresholds set.
- Delete Snapshot copies when the volume is nearly full.
  For example, you can configure Data ONTAP to automatically delete Snapshot copies that are not linked to Snapshot copies in cloned volumes or LUNs, or you can define which Snapshot copies you want Data ONTAP to delete first—your oldest or newest Snapshot copies. You can also determine when Data ONTAP should begin deleting Snapshot copies—for example, when the volume is nearly full or when the volume's Snapshot reserve is nearly full.

If you enable both of these methods, you can specify which method Data ONTAP tries first when a volume is nearly full. If the first method does not provide sufficient additional space to the volume, Data ONTAP tries the other method next. By default, Data ONTAP tries to increase the size of the volume first.

Related concepts

How volume fullness alerts work with the volume autogrow capability on page 53
Methods to create space in a FlexVol volume on page 49

Related tasks

Configuring a FlexVol volume to automatically change its size on page 36
Deleting Snapshot copies automatically on page 38

Selecting the first method to increase space for full FlexVol volumes

If you enable both the volume autosize capability and automatic Snapshot deletion for the same volume, you can also specify which method Data ONTAP tries first when that volume needs additional free space. How you configure the volume depends on whether you would prefer that the volume continue to grow or that Snapshot copies are deleted.

About this task

If the first method that Data ONTAP tries to use for providing additional free space does not result in addressing the free space needs of the volume, then Data ONTAP tries the other method.

In most cases, the default configuration (growing the volume first) is preferable, because when a Snapshot copy is deleted, it cannot be restored. However, in certain situations, you might want to

avoid growing the size of a volume when possible. In this case, you can configure Data ONTAP to delete Snapshot copies before increasing the size of the volume.

Step

1. Select the first method that Data ONTAP should use to provide free space to a volume by using the `volume modify` command with the `-space-mgmt-try-first` option.

   To specify increasing the size of the volume first (the default), use `volume_grow`. To specify deleting Snapshot copies first, use `snap_delete`.

Example

The following command configures Data ONTAP to delete Snapshot copies before increasing the volume size for the volume vol0001:

```
volume modify -vserver vs2 -volume vol0001 -space-mgmt-try-first
snap_delete
```

How a FlexVol volume can automatically change its size

A volume can be configured to grow and shrink automatically in response to space usage requirements. Automatic growing occurs when used space exceeds an autogrow threshold. Automatic shrinking occurs when used space drops below an autoshrink threshold.

The autosizing feature consists of two possible functionalities:

- The autogrow functionality grows a volume's size automatically (`grow` option).

  Automatic growth can provide additional space to a volume when it is about to run out of space, as long as there is space available in the associated aggregate. When the volume's free space percentage is below the specified threshold, it continues to grow by the specified increment until either the free space percentage arrives at the threshold or the associated aggregate runs out of space.

  If a volume does not have enough space for a Snapshot copy to be created, an attempt to create a Snapshot copy does not cause the volume to automatically grow. In this case, the Snapshot copy creation fails.
- The autoshrink functionality shrinks a volume's size automatically (`grow_shrink` option).

  The autoshrink functionality is only used in combination with autogrow to meet changing space demands and is not available alone. Automatic shrinking helps to more accurately size a volume and prevents a volume from being larger than it needs to be at any given point. The volume shrinks and returns space to the aggregate if the guarantee type is `volume`.

Because the size of the Snapshot reserve is a percentage of the size of the volume, Snapshot spill can start to occur or increase as a result of a volume shrinking.

A node root volume does not support the `grow_shrink` autosize mode, but you can configure a Storage Virtual Machine (SVM) root volume for automatic sizing.

Related concepts

Methods to create space in an aggregate on page 51

Configuring a FlexVol volume to automatically change its size

You can configure a volume to grow automatically or grow and shrink automatically (known as *autosizing*) in response to space usage requirements. Automatic growing helps prevent a volume from running out of space or forcing you to delete files manually. Automatic shrinking prevents a volume from being larger than needed.

Before you begin

The FlexVol volume must be online.

About this task

The autosize capability is off by default, except for data protection mirrors, which have the `grow_shrink` option enabled by default.

Step

1. Use the applicable command to grow the volume size automatically or to grow and shrink the volume size automatically:
   - `volume autosize -vserver <vserver_name> <vol_name> -mode grow`
   - `volume autosize -vserver <vserver_name> <vol_name> -mode grow_shrink`

   You can specify the following parameters related to growing the volume size automatically:
   - `-maximum-size` sets the maximum size to which a volume can grow.

     The default is 120 percent of the volume size. If you resize the volume manually, this value is reset to 120 percent of the current volume size. A volume does not grow automatically if its current size is greater than or equal to the value of this parameter.

     If you attempt to set this parameter greater than the platform-dependent maximum volume size, it is silently reset to the maximum volume size.
   - `-increment-size` provides a target amount by which the volume size increases each time the volume grows automatically.

     The default is the lesser value of either 1 GB or 5 percent of the volume size at the time the volume was created. When increasing the size of a volume, Data ONTAP uses the specified increment as a guide; the actual size increase can be larger or smaller.
   - `-increment-percent` specifies the percent by which the volume size increases each time the volume grows automatically.

     The default depends on the size of the volume. The default is the lesser value of either 1 GB or 5 percent of the volume size at the time the volume was created. The specified percentage is converted to a fixed increment size in bytes based on the volume size when the command is issued.

- `-grow-threshold-percent` specifies the used space threshold above which growing should start.

  When the volume's used space exceeds this threshold, the volume grows automatically unless it has reached the maximum size specified for automatic growth. The default depends on the size of the volume.

You can specify the following parameters related to shrinking the volume size automatically (in addition to the parameters related to growing):

- `-minimum-size` specifies the smallest size to which the volume can shrink, enabling you to maintain a percentage of free space.

  The default minimum size is the initial volume size. Manually resizing the volume or using an invalid minimum size value when you enable the autosizing feature resets this value to the current volume size.
- `-shrink threshold percent` specifies the volume's used space percent threshold below which shrinking should start.

  When the amount of used space drops below this threshold, the volume shrinks automatically unless it has reached the specified minimum size. For example, if used space is 50 percent and the threshold is 51 percent, automatic shrinking begins. The default is 50 percent.

Example

The command in this example enables the autosizing feature on a volume called test2 with an increment size of 10 MB. The defaults are used for growing and shrinking.

```
cluster1::> volume autosize -vserver vs2 test2 -increment-size
10MB
vol autosize: Flexible volume "vs2:test2" autosize settings UPDATED.

Volume modify successful on volume: test2
```

Requirements for enabling both autoshrink and automatic Snapshot copy deletion

The autoshrink functionality can be used with automatic Snapshot copy deletion if certain configuration requirements are met.

If you want to enable both the autoshrink functionality and automatic Snapshot copy deletion, your configuration must meet the following requirements:

- Data ONTAP must be configured to attempt to increase volume size before trying to delete Snapshot copies (the `-space-mgmt-try-first` option must be set to `volume_grow`).
- The trigger for automatic Snapshot copy deletion must be volume fullness (the `trigger` parameter must be set to `volume`).

How the autoshrink functionality interacts with Snapshot copy deletion

Because the autoshrink functionality shrinks the size of a FlexVol volume, it can also affect when volume Snapshot copies are automatically deleted.

The autoshrink functionality interacts with automatic volume Snapshot copy deletion in the following ways:

- If both the `grow_shrink` autosize mode and automatic Snapshot copy deletion are enabled, when a volume size shrinks it can trigger an automatic Snapshot copy deletion.
  This is because the Snapshot reserve is based on a percentage of the volume size (5 percent by default), and that percentage is now based on a smaller volume size. This can cause Snapshot copies to spill out of the reserve and be deleted automatically.
- If the `grow_shrink` autosize mode is enabled and you manually delete a Snapshot copy, it might trigger an automatic volume shrinkage.

How a FlexVol volume can reclaim free space from FlexClone LUNs

Starting with Data ONTAP 8.2, you can configure the autodelete settings of a FlexVol volume to automatically delete FlexClone LUNs when the free space in a volume decreases below a particular threshold value.

A FlexVol volume that has the autodelete capability enabled resorts to automatic deletion of FlexClone LUNs only in the following situations:

- If the volume does not have Snapshot copies for automatic deletion.
- If the volume has Snapshot copies but the automatic deletion of Snapshot copies does not create sufficient free space in the volume.

Because the autodelete settings enable a FlexVol volume to directly delete FlexClone LUNs when the volume requires free space, you can preserve certain FlexClone LUNs by preventing them from getting automatically deleted.

Note: If the FlexVol volume contains FlexClone LUNs created using Data ONTAP versions earlier than 8.2 and if you want to delete them to increase the amount of free space in the volume, you can specify those FlexClone LUNs for automatic deletion.

Deleting Snapshot copies automatically

You can define and enable a policy for automatically deleting Snapshot copies and FlexClone LUNs. Automatically deleting Snapshot copies and FlexClone LUNs can help you manage space utilization.

About this task

You can automatically delete Snapshot copies from read-write volumes and FlexClone LUNs from read-write parent volumes. You cannot set up automatic deletion of Snapshot copies from Infinite Volumes or from read-only volumes, for example, SnapMirror destination volumes.

Step

1. You define and enable a policy for automatically deleting Snapshot copies by using the `volume snapshot autodelete modify` command.

   See the `volume snapshot autodelete modify` man page for information about the parameters that you can use with this command to define a policy that meets your needs.

   Example

   The following command enables the automatic deletion of Snapshot copies and sets the trigger to `snap_reserve` for the vol3 volume, which is part of the vs0.example.com Storage Virtual Machine (SVM):

   ```
   cluster1::> volume snapshot autodelete modify -vserver
   vs0.example.com
   -volume vol3 -enabled true -trigger snap_reserve
   ```

   Example

   The following command enables the automatic deletion of Snapshot copies and of FlexClone LUNs for the vol3 volume, which is part of the vs0.example.com Storage Virtual Machine (SVM):

   ```
   cluster1::> volume snapshot autodelete modify -vserver
   vs0.example.com
   -volume vol3 -enabled true -trigger volume -commitment try -delete-
   order
   oldest_first -destroy-list lun_clone,file_clone
   ```

Related concepts

Methods to create space in an aggregate on page 51

How to determine space usage in a volume or aggregate

Enabling a feature in Data ONTAP might consume space that you are not aware of or more space than you expected. Data ONTAP helps you determine how space is being consumed by providing three perspectives from which to view space: the volume, a volume's footprint within the aggregate, and the aggregate.

A volume can run out of space due to space consumption or insufficient space within the volume, aggregate, or a combination of both. By seeing a feature-oriented breakdown of space usage from different perspectives, you can assess which features you might want to adjust or turn off, or take other action (such as increase the size of the aggregate or volume).

You can view space usage details from any of these perspectives:

- The volume's space usage

This perspective provides details about space usage within the volume, including usage by Snapshot copies. The volume's active file system consists of user data, file system metadata, and inodes. Data ONTAP features that you enable might increase the amount of metadata, and in the case of Snapshot copies, can sometimes spill into the user data portion of the active file system.

You see a volume's space usage by using the `volume show-space` command.

- The volume's footprint within the aggregate

  This perspective provides details about the amount of space each volume is using in the containing aggregate, including the volume's metadata.

  You see a volume's footprint with the aggregate by using the `volume show-footprint` command.
- The aggregate's space usage

  This perspective includes totals of the volume footprints of all of the volumes contained in the aggregate, space reserved for aggregate Snapshot copies, and other aggregate metadata.

  You can see the aggregate's space usage by using the `storage aggregate show-space` command.

Certain features, such as tape backup and deduplication, use space for metadata both from the volume and directly from the aggregate. These features show different space usage between the volume and volume footprint perspectives.

Related concepts

Cautions and considerations for changing file or directory capacity on page 53

Considerations for changing the maximum number of files allowed on a FlexVol volume on page 53

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 and Infinite Volume constituents it contains. Measures that you take to increase space in a volume also affect space in the aggregate.

When the aggregate is offline, no values are displayed. Only non-zero values are displayed in the command output. However, you can use the `-instance` parameter to display all possible feature rows regardless of whether they are enabled and using any space. A value of `-` indicates that there is no data available to display.

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. It is also the amount of space that is freed if all volumes in the containing aggregate are destroyed. Infinite Volume

constituents appear in the output of space usage commands as if the constituents were FlexVol volumes.

- 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. The aggregate's Snapshot reserve is set to 0 percent by default.
- Total Used
  The sum of all space used or reserved in the aggregate by volumes, metadata, or Snapshot copies.

There is never a row for Snapshot spill.

The following example shows the `aggregate show-space` command output for an aggregate whose Snapshot reserve was increased to 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%
```

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

You can determine which FlexVol volumes and Infinite Volume constituents 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 does not directly correspond to the output of the `df` command, but instead bridges the gap between the output of `volume show-space` and `aggregate show-space` commands. All percentages are calculated as a percent of aggregate size.

Only non-zero values are displayed in the command output. However, you can use the `-instance` parameter to display all possible feature rows regardless of whether they are enabled and using any space. A value of `-` indicates that there is no data available to display.

Infinite Volume constituents appear in the output of space usage commands as if the constituents were FlexVol volumes.

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, so if volumes have reserved files, the volume's total used space in the `volume show-space` command output can exceed the value in this row.	• 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`. This row will go to `0`. If you configure your volumes with a volume guarantee of `none`, you should refer to Technical Report 3965 or 3483 for information about how a volume guarantee of `none` can affect storage availability.
Flexible Volume Metadata	The total amount of space used in the aggregate by the volume's metadata files.	No direct method to control.

Row/feature name	Description/contents of row	Some ways to decrease
Delayed Frees	Blocks that Data ONTAP used for performance and cannot be immediately freed. When Data ONTAP frees blocks in a FlexVol volume, this space is not always immediately shown as free in the aggregate because operations to free the space in the aggregate are batched for increased performance. Blocks that are declared free in the FlexVol volume but that are not yet free in the aggregate are called “delayed free blocks” until the associated delayed free blocks are processed. For SnapMirror destinations, this row has a value of `0` and is not displayed.	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 concepts

Methods to create space in a FlexVol volume on page 49
Methods to create space in an aggregate on page 51
What the volume footprint is on page 43
Methods to create space in an aggregate on page 51

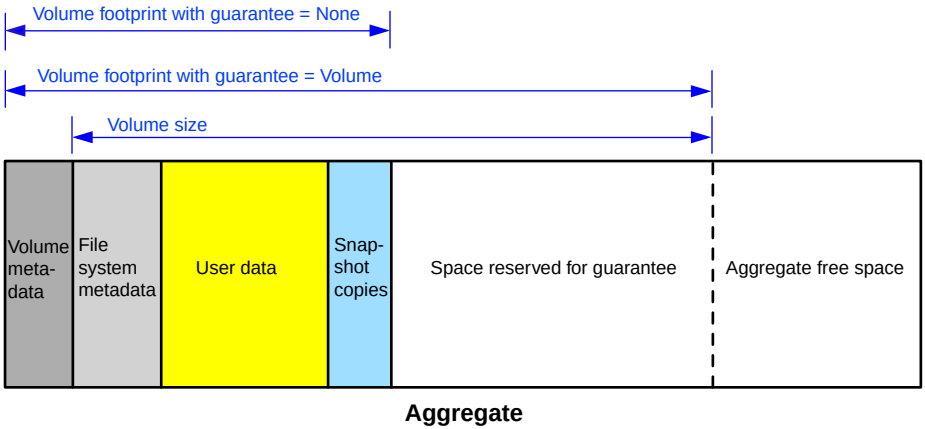
Related information

Technical Report: Thin Provisioning Deployment and Implementation Guide: media.netapp.com/documents/tr-3965.pdf
Technical Report: Thin Provisioning in a NetApp SAN or IP SAN Enterprise Environment: media.netapp.com/documents/tr3483.pdf

What the volume footprint is

A volume footprint is the amount of space a volume is using within the aggregate. Understanding what is included in the volume footprint helps you understand the space requirements for the volume.

The volume footprint consists of the space used by user data and metadata, including metadata that resides in the aggregate rather than within the volume itself. For this reason it can be larger than the volume size, as shown in the following diagram:



How you can determine and control space usage in a volume

You can view details about space usage in a volume to understand which Data ONTAP features are consuming space and what you can do to decrease that used space.

The `volume show-space` command displays the space used by each of the file system components as well as other features. Infinite Volume constituents appear in the output of space usage commands as if the constituents were FlexVol volumes. For example, you might want to understand why the `df` command output shows that a large amount of space is still used even though you deleted all of your data in a volume. In this case, output for the volume space usage command might show that it is due to Snapshot copies, inodes, or other metadata that does not shrink.

Only non-zero values are displayed in the command output. However, you can use the `-instance` parameter to display all possible feature rows regardless of whether they are enabled and using any space. A value of `-` indicates that there is no data available to display.

The following tables explain some of the common rows in the `volume show-space` command output and what you can do to try to decrease space usage by that feature:

The output for this command consists of the following main categories:

- User data
- Volume metadata
- Snapshot copy information
- Total used space

For information about how to reduce space consumed by other features (such as deduplication), see the respective Data ONTAP guides.

The available space in a volume with a guarantee type of `None` is limited by the available space in the aggregate. Checking the available space in the aggregate might show you that the aggregate is nearly full from aggregate Snapshot copies.

User data

The following output row relates to user data:

Row/feature name	Description	Some ways to decrease space usage
User Data	Everything related to user data, including the data written to the volume, including indirect blocks and directory blocks associated with user inodes, and the space reserved in the volume.	• Deleting user data • Turning off file or LUN reservations Note that turning off file or LUN reservations disables the Data ONTAP ability to guarantee writes to those files or LUNs. This can result in out of space errors being returned. Turning off reservations should be a temporary measure, and reservations should be reenabled as soon as you have provided more free space to the volume.

Volume metadata

The following output rows relate to volume metadata:

Row/feature name	Description	Some ways to decrease space usage
Deduplication / Deduplication Percent	The amount of space used by deduplication metadata files.	Comparison of the space savings you are getting from deduplication with the size of the metadata required. If the metadata requirement is larger than the savings, you can disable deduplication on the volume.
Temporary Deduplication / Temporary Deduplication Percent	The amount of space used by temporary deduplication metadata files.	No direct method to control. The temporary metadata usage decreases after deduplication scanners finish running.
Filesystem Metadata / Filesystem Metadata Percent	Internal tracking for the file system required by Data ONTAP.	No direct method to control.

Row/feature name	Description	Some ways to decrease space usage
SnapMirror Metadata / SnapMirror Metadata Percent	The amount of space in use by SnapMirror metadata files. This row relates only to logical replication. During transfers, some additional space is used temporarily.	No direct method to control. You can allow the transfer to finish so the additional space used temporarily is freed.
Tape Backup Metadata / Tape Backup Metadata Percent	The amount of space in use by tape backup metadata files in the volume.	The amount of space consumed by tape backup metadata is cleared when the next baseline (Level 0) backup is successfully run. You can initiate a baseline backup or let it run at the next scheduled time.
Quota Metadata / Quota Metadata Percent	The amount of space used by quota metadata files.	Turning off quotas.
Inodes / Inodes Percent	This row is proportional to the maximum number of files ever created in the volume.	No direct method to control current usage. You can reduce the maximum amount of space that can be used for inode allocations by lowering the maximum public inode setting (maxfiles). However, space that has already been allocated for inodes is never returned to the volume, so if those inodes have already been used, this action has no effect.

Snapshot copy information

The following output rows relate to Snapshot copies:

Row/feature name	Description	Some ways to decrease space usage
Snapshot Reserve	Based on the current volume size. The Snapshot reserve is not available to the active file system and is counted as used space, even if there are no Snapshot copies in the reserve. This row is the same as the total space used for the `.snapshot` row in the `df` command.	You can use the `volume modify` command with the `-percent-snapshot-space` parameter to lower the space allowed for Snapshot copies in the volume.

Row/feature name	Description	Some ways to decrease space usage
Snapshot Spill	The amount of space used by Snapshot copies that exceeds the Snapshot reserve size, and spills over into the active file system. This space is not available for writes to the active file system until Snapshot copies are deleted. A non-zero value in this row indicates that your Snapshot reserve has not been sized correctly for your current configuration. Volume clones, SnapMirror, and regularly scheduled Snapshot copies can cause Snapshot copy spill.	• Increasing the size of the Snapshot reserve. • Deleting volume Snapshot copies, either manually or by enabling the Snapshot autodelete capability. • Changing the SnapMirror schedule.

Total used space

The following row relates to total used space in the volume:

Row/ feature name	Description	Some ways to decrease space usage
Total Used	The total amount of used space in the volume, including the amount of space allotted for the entire Snapshot reserve and space for the active file system. This row is equivalent to the `used` field in the output of the `volume show` command. Snapshot space is treated as used space, so this row can be higher than the `df` command's output. In the `df` command, this row is equivalent to the volume's used space in the `used` column plus the Snapshot total (in the `total` column) for the Snapshot used space (`.snapshot`) row. When Snapshot spill exists, the `volume show-space` command only accounts for the used space once. However, the `df` command shows that space as used for both the active file system and for the `.snapshot` row.	Any of the methods for individual output rows.

Example output

The following example displays output for a FlexVol volume called testvol:

```
cluster1::> volume show-space testvol
  (volume show-space)

      Vserver : thevs
      Volume  : testvol

      Feature                                  Used      Used%
      -------------------------------     ----------     ------
      User Data                             853.4MB        42%
      Filesystem Metadata                     468KB         0%
      Inodes                                   16KB         0%
      Snapshot Reserve                      102.4MB         5%
      Snapshot Spill                        429.9MB        21%

      Total Used                             1.35GB        68%
```

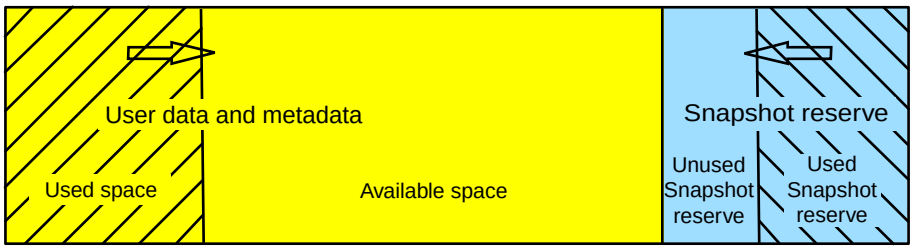
How Snapshot copies and Snapshot reserve use space in a volume

Understanding the Snapshot reserve area of a FlexVol volume or Infinite Volume and what Snapshot spill is can help you correctly size the Snapshot reserve. For FlexVol volumes, it can help you decide whether to enable the Snapshot autodelete capability.

When Snapshot copies use more space than the Snapshot reserve, they spill over and use space in the active file system. The *Snapshot reserve* area of a volume is the space reserved exclusively for Snapshot copies. It is not available to the user data or metadata area of the volume. The size of the Snapshot reserve is a specified percentage of the current volume size, and does not depend on the number of Snapshot copies or how much space they consume.

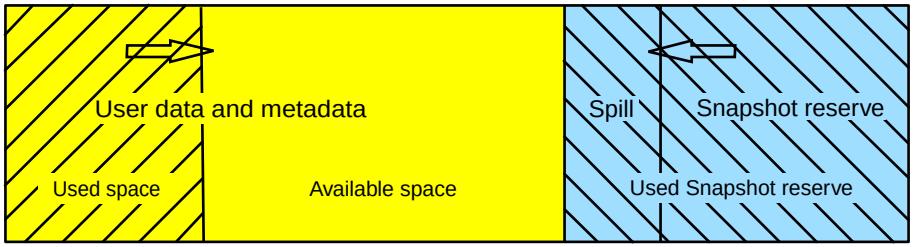
If all of the space allotted for the Snapshot reserve is used but the active file system (user data and metadata) is not full, Snapshot copies can use more space than the Snapshot reserve and spill into the active file system. This extra space is called *Snapshot spill*.

The following illustration shows a FlexVol volume with no Snapshot spill occurring. The two blocks on the left show the volume's used and available space for user data and metadata. The two blocks on the right show the used and unused portions of the Snapshot reserve. When you modify the size of the Snapshot reserve, it is the blocks on the right that change.



Volume

The following illustration shows a FlexVol volume with Snapshot spill occurring. The Snapshot reserve area is full and Snapshot copies spilling over into a Spill area that is part of the user data and metadata area's available space. The size of the Snapshot reserve remains the same.



Volume

For more information about Snapshot copies, see the *Clustered Data ONTAP Data Protection Guide*

When to use the df command and the space usage commands

You use the `df` command when you want concise information about used and available space in volumes or aggregates. If you want a detailed breakdown of space usage by feature in a volume, aggregate, or a volume's footprint within an aggregate, you use the space usage commands.

The `df` command is useful if you want a quick view of how much space each volume has available or used.

You use the `df` command (or `volume show` and `aggregate show` commands) to see total space, available space, and used space. If you need more information about how or why space is being used in your volume or aggregate, you use the `show-space` and `show-footprint` commands (the space usage commands) for the volume or aggregate.

The space usage commands, on the other hand, provide much more detail about used space and what Data ONTAP capability is causing the space to be used. For example, they could be used to help you understand why the `df` command output shows used space even though there is no data in a volume.

Used space is dynamic, even for a system that is not being accessed by clients. For this reason, you should not try to compare the output of two different space commands, or even the same command invoked twice, too closely.

Related references

Commands for displaying space information on page 58

Methods to create space in a FlexVol volume

There are multiple ways to create space in a FlexVol volume. Understanding what these methods are and their respective benefits and drawbacks helps you decide which method is best for your requirements.

Some common ways to create space in a volume are as follows:

- Increase the size of the volume.
  You can do this manually, or automatically by enabling the autogrow functionality.
- Reduce the size of the Snapshot reserve if the `df` command shows that the Snapshot reserve is not 100 percent full.
  This makes space available to the active file system.
- Make more space in the aggregate.
  This results directly or indirectly in more space being made for the volume. For example, more space in the aggregate can allow a volume to increase in size automatically with the autogrow capability.
- Enable storage efficiency technologies, such as deduplication and compression.
- Delete volume Snapshot copies if the Snapshot reserve is 100 percent full and Snapshot copies are spilling into the active file system.
  You can delete Snapshot copies manually, or automatically by enabling the Snapshot autodelete capability for the volume.
- Delete FlexClone LUNs manually or enable automatic deletion of FlexClone LUNs.
- (Temporarily) change the fractional reserve to 0 percent if your volume contains reserved files or LUNs and the fractional reserve is 100 percent.
  You should only use this as a temporary measure to create space. When the fractional reserve is set to 0 percent, overwrites might fail, and in certain deployments write failures might not be acceptable.
- Delete files.
  If the volume is 100 percent full, it might not be possible to delete a file if it participates in any block sharing, such as volume Snapshot copies or deduplication, and you cannot recover the space. In addition, modifying a directory to delete a file might require additional space, so deleting the file can actually consume space. Under these conditions, you can do one of the following:
  - You can use the `rm` command, available at the advanced privilege level, to delete files even in full volumes with Snapshot copies.
  - You can use any of the other methods listed to create more space in the volume and aggregate so that there is enough space available for file deletions.

Related concepts

Methods to create space in an aggregate on page 51
How you can determine and control space usage in a volume on page 44
How you can determine and control a volume's space usage in the aggregate on page 41
How a FlexVol volume can automatically change its size on page 35
Using deduplication and data compression to increase storage efficiency on page 159
How a FlexVol volume can reclaim free space from FlexClone LUNs on page 91
Considerations for setting fractional reserve for FlexVol volumes on page 32
Methods to create space in an aggregate on page 51

Related tasks

Deleting Snapshot copies automatically on page 38

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 volumes whose guarantee type is `volume` 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 or file-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 volumes with other guarantee types. The `Volume Guarantee` row of the `volume show-footprint` command output shows whether a volume is reserving a large amount of space in the aggregate due to its guarantee.

  If you configure your volumes with a volume guarantee of `none`, you should refer to Technical Report 3965 for information about how doing so can affect storage availability.
- 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 (visible with the `volume show-footprint` command).

Related concepts

Methods to create space in a FlexVol volume on page 49
Moving and copying volumes (cluster administrators only) on page 59
How a FlexVol volume can automatically change its size on page 35
How you can determine and control a volume's space usage in the aggregate on page 41

Related tasks

Deleting Snapshot copies automatically on page 38

Related information

Technical Report: Thin Provisioning Deployment and Implementation Guide: media.netapp.com/documents/tr-3965.pdf

Technical Report: Thin Provisioning in a NetApp SAN or IP SAN Enterprise Environment: media.netapp.com/documents/tr3483.pdf

How the FlexVol volume and aggregate fullness alerts work

Data ONTAP provides alerts when FlexVol volumes and aggregates are running out of space so that you can take corrective action by providing more space for the full volume or aggregate.

The alerts are triggered at two thresholds: nearly full and full. The thresholds are based on percentage levels of used space based on the current volume size.

The volume fullness thresholds are configurable. You use the `space-nearly-full-threshold-percent` and `space-full-threshold-percent` parameters to view and modify the volume fullness thresholds. A value of `0` disables the alert.

You cannot configure or disable the aggregate fullness thresholds; they are set at 95 and 98 percent for full and nearly full, respectively.

The following table shows valid values, EMS message type, and command parameter for the volume fullness thresholds:

Fullness threshold	Valid values	EMS when exceeded
Nearly full (volume)	0 to 99% 95% is default	monitor.volume.nearlyFull (debug level)
Full (volume)	0 to 100% 98% is default	monitor.volume.full (debug level)
Nearly full (aggregate)	95	monitor.volume.nearlyFull (debug level)
Full (aggregate)	98	monitor.volume.full (debug level)

Every time a threshold is crossed for a volume or aggregate, whether the fullness percentage is rising or falling, an EMS message is generated. When the fullness level of the volume or aggregate falls below a threshold, a `volume ok` or `aggregate ok` EMS message is generated. You can view the EMS messages by using the `ems event status` command or the `ems event log` command. In addition to EMS messages, you can configure your system to send SNMP traps.

For more information about viewing and managing events (EMS messages) and SNMP traps, see the *Clustered Data ONTAP System Administration Guide for Cluster Administrators*.

How volume fullness alerts work with the volume autogrow capability

The FlexVol volume fullness alerts can be used with the FlexVol volume autogrow capability. The two capabilities are configured independently, but if you are using both capabilities, you should consider their configuration values together to avoid unnecessary alerts.

You should make sure that your full and nearly full thresholds are set higher than the threshold at which the FlexVol volume starts to grow. Otherwise, you receive fullness alerts when the volume might still have room to grow, which automatically addresses the space constraint.

Related concepts

How Data ONTAP can automatically provide more space for full FlexVol volumes on page 33

Cautions and considerations for changing file or directory capacity

If your data requires a large number of files or very large directories, you can expand Data ONTAP file or directory capacity. However, you should understand the limitations and caveats for doing so before proceeding.

Considerations for changing the maximum number of files allowed on a FlexVol volume

FlexVol volumes have a maximum number of files that they can contain. You can change the maximum number of files for a volume, but before doing so you should understand how this change affects the volume.

The number of files a volume can contain is determined by how many inodes it has. An *inode* is a data structure that contains information about files. Volumes have both private and public inodes. Public inodes are used for files that are visible to the user; private inodes are used for files that are used internally by Data ONTAP. You can change only the maximum number of public inodes for a volume. You cannot affect the number of private inodes.

Data ONTAP automatically sets the maximum number of public inodes for a newly created volume based on the size of the volume: 1 inode per 32 KB of volume size. When the size of a volume is increased, either directly by an administrator or automatically by Data ONTAP through the autosize feature, Data ONTAP also increases (if necessary) the maximum number of public inodes so there is at least 1 inode per 32 KB of volume size, until the volume reaches approximately 1 TB in size. Growing the volume greater than 1 TB in size does not automatically result in more inodes, because Data ONTAP does not automatically create more than 33,554,409 inodes. If you need more files than the default number for any size volume, you can use the `volume modify` command to increase the maximum number of inodes for the volume.

You can also decrease the maximum number of public inodes. This does not change the amount of space currently allocated to inodes, but it does lower the maximum amount of space the public inode file can consume. However, after space has been allocated for inodes, it is never returned to the volume. Therefore, lowering the maximum number of inodes below the number of inodes currently allocated does not return the space used by the allocated but unused inodes to the volume.

Cautions for increasing the maximum directory size for FlexVol volumes

The default maximum directory size for FlexVol volumes is model-dependent, and optimized for the size of system memory. Before increasing the maximum directory size, involve technical support.

You can increase the default maximum directory size for a specific FlexVol volume by using the `-maxdir-size` option of the `volume modify` command, but doing so could impact system performance. This command has no effect for Infinite Volumes.

Rules governing node root volumes and root aggregates

A node's root volume contains special directories and configuration files for that node. The root aggregate contains the root volume. A few rules govern a node's root volume and root aggregate.

A node's root volume is a FlexVol volume that is installed at the factory and reserved for system files, log files, and core files. The directory name is `/mroot`, which is accessible only through the systemshell and with guidance from technical support.

The following rules govern the node's root volume:

- Do not change the preconfigured size for the root volume or modify the content of the root directory, unless technical support instructs you to do so.

  The minimum size for a node's root volume depends on the platform model. For information about the minimum size for the root FlexVol volume, see the *Hardware Universe* at *hwu.netapp.com*.

  Editing configuration files directly in the root directory might result in an adverse impact on the health of the node and possibly the cluster. If you need to modify system configurations, you use Data ONTAP commands to do so.
- Do not store user data in the root volume.

  Storing user data in the root volume increases the storage giveback time between nodes in an HA pair.
- Do not set the root volume's fractional reserve to any value other than 100%.
- Contact technical support if you need to designate a different volume to be the new root volume or move the root volume to another aggregate.

The node's root aggregate contains the node's root volume. Starting with Data ONTAP 8.1, new systems are shipped with the root volume in a dedicated, 64-bit root aggregate that contains three disks. By default, a node is set up to use a hard disk drive (HDD) aggregate for the root aggregate. When no HDDs are available, the node is set up to use a solid-state drive (SSD) aggregate for the root aggregate.

The root aggregate must be dedicated to the root volume only. You must not include or create data volumes in the root aggregate.

Basic FlexVol volume management

You can create and delete FlexVol volumes, change their basic attributes, and display information about how their space is being used.

Creating a FlexVol volume

You can create a FlexVol volume and specify its properties by using the `volume create` command.

Before you begin

The Storage Virtual Machine (SVM) for the new volume and the aggregate that will supply the storage to the volume must already exist. If the SVM has a list of associated aggregates, the aggregate must be included in the list.

Step

1. Use the `volume create` command to create a volume.

Example

The following command creates a new volume named dept_eng on the SVM vs1 and the aggregate aggr2. The volume is made available at `/dept/eng` in the namespace for the vs1 SVM. The volume is 750 GB in size, and its volume guarantee is of type volume (by default).

```
cluster1::> volume create -vserver vs1 -volume dept_eng
-aggregate aggr2 -junction-path /dept/eng -size 750GB
```

Deleting a FlexVol volume

To delete a FlexVol volume, you might need to prepare it for deletion first.

Before you begin

No applications should be accessing the data in the volume you want to delete.

Steps

1. If the volume has been mounted, unmount it by entering the following command:
   `volume unmount -vserver` ***`vserver_name`*** **`-volume`** ***`volume_name`***
2. If the volume is part of a SnapMirror relationship, delete the relationship by using the `snapmirror delete` command.

3. If the volume is online, take the volume offline by entering the following command:

   `volume offline -vserver vserver_name volume_name`

4. Delete the volume by entering the following command:

   `volume delete -vserver vserver_name volume_name`

Result

The volume is deleted, along with any associated quota policies and qtrees.

Controlling and monitoring I/O performance to FlexVol volumes by using Storage QoS

You can control input/output (I/O) performance to FlexVol volumes by assigning volumes to Storage QoS policy groups. You might control I/O performance to ensure that workloads achieve specific performance objectives or to throttle a workload that negatively impacts other workloads.

About this task

Policy groups enforce a maximum throughput limit (for example, 100 MB/s). You can create a policy group without specifying a maximum throughput, which enables you to monitor performance before you control the workload.

You can also assign Storage Virtual Machines (SVMs) with FlexVol volumes, LUNs, and files to policy groups.

Note the following requirements about assigning a volume to a policy group:

- The volume must be contained by the SVM to which the policy group belongs.
  You specify the SVM when you create the policy group.
- If you assign a volume to a policy group, then you cannot assign the volume's containing SVM or any child LUNs or files to a policy group.

Note: Storage QoS is supported on clusters that have up to eight nodes.

For more information about how to use Storage QoS, see the *Clustered Data ONTAP System Administration Guide for Cluster Administrators*.

Steps

1. Use the `qos policy-group create` command to create a policy group.
2. Use the `volume create` command or the `volume modify` command with the `-qos-policy-group` parameter to assign a volume to a policy group.
3. Use the `qos statistics` commands to view performance data.
4. If necessary, use the `qos policy-group modify` command to adjust the policy group's maximum throughput limit.

Displaying file or inode usage

FlexVol volumes have a maximum number of files that they can contain. Knowing how many files are contained by your volumes helps you determine whether you need to increase the number of (public) inodes for your volumes to prevent them from hitting their maximum file limit.

About this task

Public inodes can be either free (they are not associated with a file) or used (they point to a file). The number of free inodes for a volume is the total number of inodes for the volume minus the number of used inodes (the number of files).

Step

1. To display inode usage for a volume, enter the following command:

   `df -i` *`volume_name`*

   You can omit the volume name; in this case, Data ONTAP displays the inode usage for all volumes in the cluster. You can also specify a Storage Virtual Machine (SVM) to see only volumes on that SVM.

Example

```
cm320c-rst::> df -i -vserver vs1
Filesystem               iused      ifree  %iused  Mounted on
/vol/cifs_test/            105       2928      3%  /home
/vol/root/                  98        468     17%  ---
/vol/vola/                 103      12047      0%  /nfsv4
3 entries were displayed.
```

Commands for managing FlexVol volumes

There are specific commands for managing FlexVol volumes using the Data ONTAP CLI.

If you want to...	Use this command...
Bring a volume online	`volume online`
Change the size of a volume	`volume size`
Determine the associated aggregate of a volume	`volume show`
Determine the associated aggregate for all volumes on a Storage Virtual Machine (SVM)	`volume show -vserver <vserver_name> -fields aggregate`

If you want to...	Use this command...
Determine the format of a volume	`volume show -fields block-type`
Mount a volume onto another volume using a junction	`volume mount`
Put a volume into the restricted state	`volume restrict`
Rename a volume	`volume rename`
Take a volume offline	`volume offline`

See the man page for each command for more information.

Commands for displaying space information

You can 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 -aggregate` `storage aggregate show-space -snap-size-total,-used-including-snapshot-reserve`
How disks and RAID groups are used in an aggregate and RAID status	`system node run -node <node_name> aggr status -r`
The amount of disk space that would be reclaimed if you deleted a specific Snapshot copy	`volume snapshot compute-reclaimable` (advanced)
The amount of space used by a volume	`volume show -fields size,used,available,percent-used` `volume show-space -vserver <vserver_name>, -volume <volume_name>`
The amount of space used by a volume in the containing aggregate	`volume show-footprint -vserver <vserver_name>, -volume <volume_name>`

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

Moving and copying volumes (cluster administrators only)

You can move or copy volumes for capacity utilization, improved performance, and to satisfy service-level agreements.

Related concepts

Methods to create space in an aggregate on page 51

How moving a FlexVol volume works

FlexVol volumes are moved from one aggregate or node to another within the same Storage Virtual Machine (SVM) for capacity utilization and improved performance, and to satisfy service-level agreements.

A volume move does not disrupt client access during the move.

Note: You cannot move 64-bit volumes to 32-bit aggregates.

Moving a volume occurs in multiple phases:

- A new volume is made on the destination aggregate.
- The data from the original volume is copied to the new volume.
  During this time, the original volume is intact and available for clients to access.
- At the end of the move process, client access is temporarily blocked.
  During this time the system performs a final replication from the source volume to the destination volume, swaps the identities of the source and destination volumes, and changes the destination volume to the source volume.
- After completing the move, the system routes client traffic to the new source volume and resumes client access.

The move is not disruptive to client access because the time in which client access is blocked ends before clients notice a disruption and time out. Client access is blocked for 45 seconds by default. If the volume move operation cannot finish in the time that access is denied, the system aborts this final phase of the volume move operation and allows client access. The system runs the final phase of the volume move operation until the volume move is complete or until the default maximum of three attempts is reached. If the volume move operation fails after the third attempt, the process goes into a cutover deferred state and waits for you to initiate the final phase.

You can change the amount of time client access is blocked or the number of times (*cutover attempts*) the final phase of the volume move operation is run if the defaults are not adequate. You can also determine what the system does if the volume move operation cannot be completed during the time client access is blocked. The `volume move start` man page contains details about moving a volume without disrupting client access.

Commands for moving volumes

There are specific Data ONTAP commands for managing volume movement.

If you want to...	Use this command...
Abort an active volume move operation.	`volume move abort`
Show status of a volume moving from one aggregate to another aggregate.	`volume move show`
Start moving a volume from one aggregate to another aggregate.	`volume move start`
Manage target aggregates for volume move.	`volume move target-aggr`
Trigger cutover of a move job.	`volume move trigger-cutover`

See the man page for each command for more information.

Considerations and recommendations when moving volumes

Moving a volume has many considerations and recommendations that are influenced by the volume you are moving. You should understand the considerations and recommendations associated with moving volumes.

Moving a volume has the following general considerations and recommendations:

- You cannot move an Infinite Volume.
- If you are upgrading the release family for a cluster, do not move a volume until after you upgrade all of the nodes in the cluster.
  This recommendation prevents you from inadvertently attempting to move a volume from a newer release family to an older release family.
- The source volume must be consistent.
- If you assigned one or more aggregates to the associated Storage Virtual Machine (SVM), the destination aggregate must be one of the assigned aggregates.
- You cannot move a volume to or from a taken over CFO aggregate.

Moving a FlexClone volume has the following considerations and recommendations:

- FlexClone volumes cannot be offline when they are being moved.
- You can move FlexClone volumes from one aggregate to another aggregate on the same node or another node in the same SVM without splitting.
- FlexClone volume Snapshot copies are not lost after moving a clone.
- You can move FlexClone parent volumes from one aggregate to another aggregate.
  When you move a FlexClone parent volume, a temporary volume is left behind that acts as a parent volume for all FlexClone volumes. No operations are allowed on the temporary volume

except to take it offline or to delete it. After all FlexClone volumes are either split or destroyed, the temporary volume is cleaned up automatically.

- After you move a FlexClone child volume, the volume is no longer a FlexClone volume.
- FlexClone move operations are mutually exclusive from FlexClone copy or split operations.
- If a clone-splitting operation is in progress, moving a volume might fail.
  In this case, you should wait and retry moving the volume after the clone-splitting operation finishes.
- You cannot move a FlexCache volume.

Moving a volume

You can move a FlexVol volume to a different aggregate, node, or both within the same Storage Virtual Machine (SVM).

Steps

1. If you are moving a data protection mirror and you have not initialized the mirror relationship, initialize the mirror relationship by using the `snapmirror initialize` command.

   Data protection mirror relationships must be initialized before you can move one of the volumes.

2. Use the `volume move target-aggr show` command to determine an aggregate to which you can move the volume.

   This step ensures that you select an aggregate that has enough space for the volume, that is, the available size is bigger than the volume that you are moving.

   #### Example

   ```
   cluster1::> volume move target-aggr show -vserver vs2 -volume user_max
   Aggregate Name   Available Size   Storage Type
   --------------   --------------   ------------
   aggr2            467.9GB          FCAL
   node12a_aggr3    10.34GB          FCAL
   node12a_aggr2    10.36GB          FCAL
   node12a_aggr1    10.36GB          FCAL
   node12a_aggr4    10.36GB          FCAL
   5 entries were displayed.
   ```

   The resulting list shows that the vs2 volume can be moved to any of the listed aggregates.

3. Run a validation check on the volume by using the `volume move start -perform-validation-only` command to ensure that the volume can be moved to the intended aggregate.

4. Use the `volume move start` command to move a volume.

   #### Example

   The following command moves a volume named user_max on an SVM named vs2 to an aggregate named node12a_aggr3. The administrator previously determined that clients accessing

the data in the volume can tolerate an I/O timeout of, at most, 120 seconds. The move runs as a background process.

```
cluster1::> volume move start -vserver vs2 -volume user_max
-destination-aggregate node12a_aggr3 -cutover-window 120
```

5. Use the `volume move show` command to determine the command's status.

Example

The following example shows the state of a volume move that completed the replication phase and is in the cutover phase.

```
cluster1::> volume move
show
Vserver   Volume     State    Move Phase  Percent-Complete Time-To-Complete
--------- ---------- -------- ----------  ---------------- ----------------
vs2       user_max   healthy  cutover     -                -
```

6. If the volume move operation does not complete the final phase after three attempts and goes into a cutover deferred state, use the `volume move trigger-cutover` command to attempt to complete the move.

Example

The following command forces the volume move operation to finish.

```
cluster1::> volume move trigger-cutover -vserver vs2
-volume user_max -force true
```

Note: Forcing the volume move operation to finish can disrupt client access to the volume you are moving.

SnapMirror transfers can affect volume move operations

If SnapMirror transfers are running at the same time as volume move operations, the volume move operations are prevented from entering the cutover phase.

The cutover phase cannot occur for the following cases:

- If there are checkpoints on the volume.
  A checkpoint can exist for one of the following reasons:
  - A relationship exists and there is no active transfer but there is a checkpoint.
  - A relationship has been deleted, but a checkpoint left.
- If the volume is the destination of an active SnapMirror transfer.

Characteristics of how Data ONTAP copies FlexVol volumes

A copy of a FlexVol volume is a full copy of the original FlexVol volume with the same access (read-only or read-write) as the original volume. Knowing the characteristics of the volume copy can help set your expectations about the volume copy result.

A volume copy has the following characteristics:

- A copy of a volume does not share blocks with the original volume.
  A copy of a 2-GB volume uses 2 GB of disk space.
- After the copy is made, no operations made on the copy or on the original affect the other.
  For instance, if you write data to the original volume, the data is not written to the copy.
- A volume copy is not automatically mounted when it is created.
- A volume copy must occur within the context of the same Storage Virtual Machine (SVM).
- A volume copy does not copy a volume's SnapMirror labels.
- A 64-bit volume can only be copied to 64-bit aggregates.
  It cannot be copied to a 32-bit aggregate.
- An offline volume cannot be copied.

Copying a FlexVol volume

Copying a volume creates a stand-alone copy of a volume that you can use for testing and other purposes.

About this task

Copying a volume has the following limitations:

- You can copy a volume within a Storage Virtual Machine (SVM) only.
- You can copy a FlexVol volume to a FlexVol volume only.
- If you assigned one or more aggregates to the associated SVM, the destination aggregate must be one of the assigned aggregates.

Steps

1. To create a stand-alone copy of a volume, use the `volume copy start` command.

   You can make the copy on the same aggregate as the original or on a different aggregate. When the copy is complete, it has no relation to its source volume; changes made to one volume are not propagated to the other.

Example

The following example creates a copy of a volume named src_builds on an SVM named vs0. The copy is named builds and is located on an aggregate named aggr4. The copy operation runs as a background process.

```
cluster1::> volume copy start -vserver vs0 -volume src_builds
-destination-volume builds -destination-aggregate aggr4 -foreground
false
```

2. Use the `job show` command to determine if the volume copy operation is complete.
3. The copy is not automatically mounted; mount it using the `volume mount` command.

Using FlexCache volumes to accelerate data access

A FlexCache volume is a sparsely-populated volume on a cluster node, that is backed by a volume, usually present on a different node within that cluster. A sparsely-populated volume or a sparse volume provides access to data in the backing volume (also called the origin volume) without requiring that all the data be in the sparse volume.

You can use only FlexVol volumes to create FlexCache volumes. However, many of the regular FlexVol volumes features are not supported on FlexCache volumes, such as Snapshot copy creation, deduplication, compression, FlexClone volume creation, volume move, and volume copy.

You can use FlexCache volumes to speed up access to data, or to offload traffic from heavily accessed volumes. FlexCache volumes help improve performance, especially when clients need to access the same data repeatedly, because the data can be served directly without having to access the source. Therefore, you can use FlexCache volumes to handle system workloads that are read-intensive.

Cache consistency techniques help in ensuring that the data served by the FlexCache volumes remains consistent with the data in the origin volumes.

What a cached file contains

When the client requests a data block of a specific file from a FlexCache volume, then the attributes of that file and the requested data block are cached. The file is then considered to be cached, even if all its data blocks are not present in the FlexCache volume. If the requested data is cached and valid, a read request for that data is fulfilled without access to the origin volume.

How FlexCache volumes serve read requests

A FlexCache volume directly serves read requests if it contains the data requested by the client. Otherwise, the FlexCache volume requests the data from the origin volume and stores the data before serving the client request. Subsequent read requests for the data are then served directly from the FlexCache volume.

FlexCache volumes serve client read requests as follows:

1. A cluster node, which corresponds to the LIF on which the client sends its read request, accepts the request.
2. The node responds to the read request based on the types of volumes it contains.

If the node contains...	Then...
A FlexCache volume that contains the requested data and the origin volume	The data is served from the origin volume. **Note:** If a cache policy is specifically configured to enable the FlexCache volume serve the read request, then the origin is bypassed and the cache volume directly serves the data.
A FlexCache volume that contains the requested data but not the origin volume	The data is served from the FlexCache volume.
A FlexCache volume that does not contain the requested data	The FlexCache volume retrieves the requested data from a volume that contains the data, stores the data, and serves the client request.
A volume that is the primary source of the requested data but does not contain a FlexCache volume	The data is served directly from the volume containing the requested data.

Note: If the node does not contain either the primary source of the data or a FlexCache volume, the client request is directly passed to a node that contains the primary source of the data.

Related concepts

How data changes affect FlexCache volumes on page 70

Why you use FlexCache volumes

FlexCache volumes are used to improve performance and balance resources during data read operations.

- Performance scaling

  A data volume when created is stored on a specific node of the cluster. That volume can move within the cluster, but at any point in time, only one node contains the source data. If there is intensive access to the data on that volume, then that node in the cluster can get overloaded, and develop a performance bottleneck.

  FlexCache volumes scale performance by enabling multiple nodes of a cluster to respond to read requests efficiently without having to overload the node containing the source data and without having to send data over the cluster interconnect (for cache hits).
- Resource balancing

  Certain nodes of a cluster can encounter spikes of high performance during certain tasks or activities to a specific data set. By caching copies of data throughout the cluster, FlexCache volumes efficiently enable each node in the cluster to handle the workload. This approach spreads the workload across the cluster, smoothing out the performance created by heavy read or metadata access.

Considerations for working with FlexCache volumes

You should take into account a list of considerations when creating and working with FlexCache volumes.

- You do not need to install any license for creating FlexCache volumes.
- Using clustered Data ONTAP, you can cache a FlexVol volume within the Storage Virtual Machine (SVM) that contains the origin volume.

  You must use a caching system running Data ONTAP operating in 7-Mode if you want to cache a FlexVol volume outside the cluster.
- You cannot use Infinite Volumes as the caching or origin volume.

  You can use only FlexVol volumes to cache data in other FlexVol volumes.
- To cache a FlexVol volume within a cluster, you must ensure that the FlexCache volumes and the origin volumes are created on storage systems supported by clustered Data ONTAP 8.2 or later.

  Note: For information about the requirements that the storage system running Data ONTAP operating in 7-Mode must meet for caching a clustered Data ONTAP FlexVol volume, see the *Data ONTAP Storage Management Guide for 7-Mode*.
- You can create FlexCache volumes on a specific cluster node or on all the cluster nodes spanned by the SVM that contains the origin volume.
- FlexCache volumes are created with a space guarantee type of partial.

  The partial guarantee type is a special guarantee type that cannot be changed. You cannot view the space guarantee type for a FlexCache volume from the command line interface. When you use commands such as `volume show` to view the volume's space guarantee type, the value for the particular field shows a dash.
- There is no specific limit on the size of the origin volume that a FlexCache volume can cache.
- A FlexCache volume is created with the same language setting as its corresponding origin volume.
- Flash Cache is supported on nodes with FlexCache volumes, and optimizes performance and efficiency accordingly for all volumes on the node, including FlexCache volumes.
- Storage Accelerator (SA) systems do not support clustered Data ONTAP. SA systems support only Data ONTAP operating in 7G or 7-Mode.
- FlexCache volumes support client access using the following protocols: NFSv3, NFSv4.0, and CIFS (SMB 1.0, 2.x, and 3.0).

  In addition, FlexCache volumes can retrieve from the origin volumes the Access Control Lists (ACLs) and the stream information for the cached data, depending on the protocol.
- For better performance of FlexCache volumes in a cluster, you must ensure that data LIFs are properly configured on the cluster nodes that contain the FlexCache volumes.

Related concepts

Limitations of FlexCache volumes on page 67

How autosizing of FlexCache volumes improves caching performance on page 75

Limitations of FlexCache volumes

You can have a maximum of 100 FlexCache volumes on a cluster node. In addition, certain features of Data ONTAP are not available on FlexCache volumes, and other features are not available on origin volumes that are backing the FlexCache volumes.

You cannot use the following Data ONTAP capabilities on FlexCache volumes (these limitations do not apply to origin volumes):

- Compression
  Compressed origin volumes are supported.
- Snapshot copy creation
- SA systems
- SnapManager
- SnapRestore
- SnapMirror
- FlexClone volume creation
- The `ndmp` command
- Quotas
- Volume move
- Volume copy
- Cache load balancing
- Qtree creation on cache
  Qtree management must be done on the origin.
- Deduplication
- Mounting the FlexCache volume as a read-only volume
- I2P
  The cache volume will not synchronize I2P information from the origin; any requests for this information are always forwarded to the origin.
- Storage QoS policy group
  An origin volume can be assigned to a policy group, which controls the origin volume and its corresponding FlexCache volumes.

Note: The following FAS systems do not support FlexCache volumes in a clustered Data ONTAP environment:3140, 3160, 3210, and 3240.

The following limitations apply to origin volumes:

- You must map FlexCache volumes to an origin volume that is inside the same Storage Virtual Machine (SVM).
  FlexCache volumes in a clustered Data ONTAP environment cannot point to an origin volume that is present outside the SVM, such as a 7-Mode origin volume.
- You cannot use a FlexCache volume to cache data from Infinite Volumes.
  The origin volume must be a FlexVol volume.

- A load-sharing mirror volume or a volume that has load-sharing mirrors attached to it cannot serve as an origin volume.
- Any volume in a SnapVault relationship cannot serve as an origin volume.
- You cannot use an origin volume as the destination of a SnapMirror migrate command.
- A FlexCache volume cannot be used as an origin volume.

Related concepts

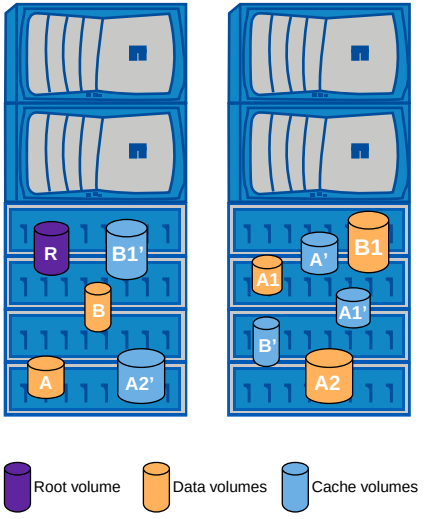
Considerations for working with FlexCache volumes on page 66

Example of FlexCache deployment in clustered Data ONTAP environment

FlexCache volumes can serve read operations from cached copies of file data from any node in the cluster, regardless of the node on which the original data resides. This enables more efficient use of resources by spreading the workload of shared data sets across more than one node.

The following illustration represents a 4-node cluster in two HA pairs. Each pair connects to four shelves of disks. This cluster contains:

- One Storage Virtual Machine (SVM), which contains one root volume (denoted as R)
- Data volumes (denoted as A, A1, A2, B, and B1)
- Multiple cache volumes (denoted as A', A1', A2', B', and B1')



In this example, volumes A, A2, B, and B1 have cache volumes on the alternate node. For effective use of FlexCache volumes, you can create cached copies for every data volume that is likely to be accessed by many clients simultaneously for read operations. However, to simplify this example, not all the volumes have cached copies.

An incoming file operation might be served from the cache volume or from the origin volume on a node, depending on the LIF used for the operation. If a cache volume exists on the node containing

the LIF that gets the incoming request, the operation might be served from the cache volume on that node.

The following illustration shows the same cluster but adds a visual representation for the following:

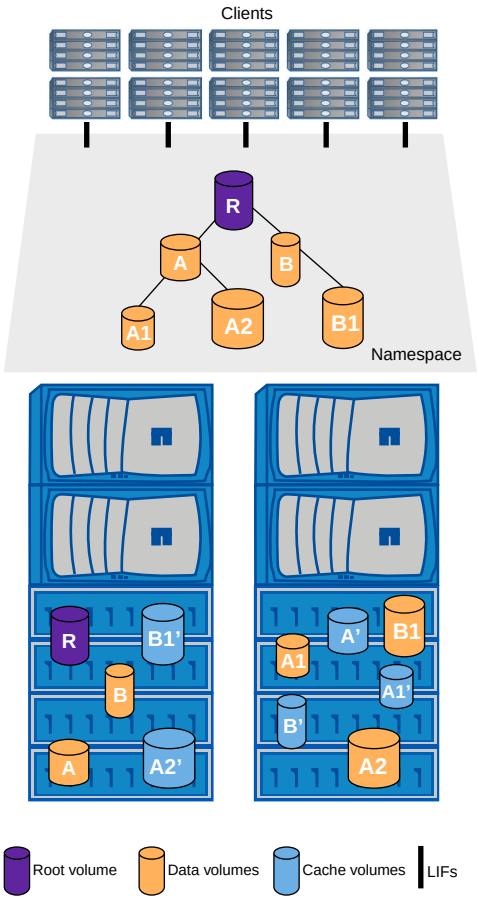
- Hosts

  Clients or applications accessing data stored on the cluster
- Namespace

  The root volume R and several data volumes (A, A1, A2, B, B1, C, C1) connected through junction points
- LIFs

  The vertical lines represent the virtual network connections into the namespace and connect the hosts to the SVM
- SVM

  A virtual container, the gray rectangle, that presents the namespace and LIFs to the hosts



What cache policies are

A cache policy is a set of parameters that help you define different properties of FlexCache volumes, such as the extent of staleness of data in FlexCache volumes, the time after which an unused delegation is returned to the origin, and the parameter that enables the FlexCache volume to serve read requests from a node that also has the origin volume.

Cache policies are defined for the Storage Virtual Machine (SVM) that contains the volumes. You can use the default cache policy or configure your own cache policies and apply them to FlexCache volumes in an SVM.

Every SVM has a default cache policy. The default cache policy is a special cache policy that is created and deleted along with the SVM. FlexCache volumes use the default cache policy when no other cache policies are present. The default cache policy can be modified, but not deleted (unless the SVM is deleted).

The different parameters in a cache policy help define the following properties of FlexCache volumes:

- Staleness of data: You can define parameters to specify the staleness of data in FlexCache volumes as a TTL (time to live) value. You can configure the length of the TTL window for objects such as files, directories, and symbolic links in a FlexCache volume depending on the amount of staleness that can be tolerated.
- Timeout value for unused cache delegation: Any delegation that remains unused beyond this value is returned by the FlexCache volume to the origin volume.
- Enabling the FlexCache volume to serve read requests: For a cluster node that contains both the origin volume and the FlexCache volume, you can enable the FlexCache volume to serve read requests instead of the origin volume by configuring the -prefer-local-cache parameter in the cache policy.

Related concepts

Cache-read delegations on page 72
TTL values in cache policies on page 73

Related references

Commands for managing FlexCache volumes on page 78

How data changes affect FlexCache volumes

How data changes affect FlexCache volumes depends on where the change is made: on the FlexCache volume, the origin volume, or another FlexCache volume. If a file is directly updated on

the origin volume, the cached copy of the file is invalidated. If the write request is relayed to the origin volume, only the changed blocks are invalidated in the FlexCache volume.

Writes to a file on the origin volume

When a change is made to a file on the origin volume, Data ONTAP revokes the delegation for that file and invalidates the entire file for all FlexCache volumes backed by that origin volume.

Note: The FlexCache copy of the file is not invalidated until an access to that file is made on the FlexCache volume.

The cache is not affected when only the access time of a file is updated.

Writes to a file on the FlexCache volume

When a write is made to a file on the FlexCache volume, the write request is relayed to the origin volume. When the origin volume acknowledges the request, the blocks that were changed are invalidated on the FlexCache volume, but the rest of the file remains valid.

Related concepts

Write operation proxy on page 74

How cache consistency is achieved

Cache consistency for FlexCache volumes is achieved by using three primary techniques: *cache-read delegations*, *TTL values in cache policies*, and *write operation proxy*.

Delegations ensure that the FlexCache volumes can directly serve client read requests without having to access the origin volume. As long as the FlexCache volume continues to have a delegation for the data on a file, the origin volume does not modify the contents of the file. The origin volume must revoke all the delegations to a file from the FlexCache volumes before modifying the contents of the file.

If a FlexCache volume does not have delegations for certain data objects (such as files, directories, and symbolic links) cached in the volume, the data is considered valid for a duration determined by the TTL value for the particular data object. You can set different TTL values for the objects while configuring cache policies for FlexCache volumes.

Any FlexCache volume that receives a write request from the client proxies the request to the origin volume. If the FlexCache volume contains a cached copy of the data that changes in the origin, the FlexCache volume removes the changed data blocks and stores the updated data during a subsequent read operation.

Cache-read delegations

A cache-read delegation is a token that the origin system grants the FlexCache volume to ensure that the FlexCache volume can serve read requests without the need for validating the data with the origin volume. Cache-read delegations are used only for files and are not mandatory.

When the FlexCache volume retrieves data from a file on the origin volume, the origin volume provides a delegation for that file to the FlexCache volume. The FlexCache volume retains the delegation until the origin volume revokes the delegation before the particular file gets modified. After the data is modified, the FlexCache volume must fetch the data before serving it to clients.

The origin volume can refuse to provide any delegation to the FlexCache volume for a file. In such a situation, the TTL values for the file decide the duration for which the data in the FlexCache volume can be considered valid.

If multiple FlexCache volumes within the cluster have delegations for a particular file, then all the delegations are revoked before the file is updated.

Delegations can cause a performance decrease for writes to the origin volume, depending on the number of FlexCache volumes that are holding delegations for the file being modified.

Note: Delegations that remain unused beyond a particular timeout value are marked unused by the FlexCache volume and returned to the origin volume. You can configure the value of the `-deleg-lru-timeout` parameter to define this timeout value while setting the cache policy for the FlexCache volumes.

The following list outlines situations when delegations cannot be used to guarantee that an object has not changed:

- Objects other than regular files: Directories, symbolic links, and other objects that are not regular files have no delegations.
- When Data ONTAP features such as SnapMirror, Single file SnapRestore, Snapshot copy restore, volume move, or aggregate relocation impact the origin volume: All these features cause the volume to be unmounted, and therefore, the delegations are invalidated.
- When the origin volume is taken offline: All the delegations owned by the volume are invalidated.
- When the FlexCache volume is taken offline: All the delegations given to the volume are destroyed.
- When connectivity is lost: If the FlexCache volume is unresponsive for a short duration of time or if the file is modified at the origin when the FlexCache volume is not connected, then delegations are considered to be revoked.
- When the maximum number of delegations is reached: If the origin volume cannot store all of its delegations, it might revoke an existing delegation to accommodate a new one.
- When the origin volume has a resource constraint: The origin volume reclaims some delegations from all its FlexCache volumes.

Related concepts

TTL values in cache policies on page 73

TTL values in cache policies

When data is retrieved from the origin volume, the file that contains that data is considered valid in the FlexCache volume as long as a delegation exists for that file. If no delegation exists, the file is considered valid for a certain length of time, specified by the TTL (time-to-live) values set for the FlexCache volume that contains the file.

The TTL values are specified as parameters in a cache policy. You can define TTL values for different parameters corresponding to the following data constituents in a FlexCache volume:

- Files (-reg-ttl)
- Directories (-dir-ttl)
- Internal ONTAP metafiles (-meta-ttl)
- Symbolic links (-sym-ttl)
- Objects that are not regular files, directories, metafiles, or symbolic links (-other-ttl)

TTL values help in ensuring consistency of data between FlexCache volumes and origin volumes. The FlexCache volume ensures this by comparing certain attributes of the cached file with those of the file in the origin volume to verify if both the sets of attributes are the same. The FlexCache volume maintains a timestamp of the latest verify operation. The value of this timestamp in conjunction with the TTL value help in deciding if the requested data can be served directly from the FlexCache volume.

For example, if the TTL value of a file from which data is requested is 15 seconds and the last verify was performed less than 15 seconds ago, the FlexCache volume allows access to the file. Otherwise, it performs a verify operation, and then serves data to the client if the attributes are the same. If the attributes are not the same, the requested data blocks are read from the origin volume, updated in the FlexCache volume, and served to the client.

If you set non-zero TTL values for FlexCache volumes, stale data might be served to clients. In addition, if the TTL values for FlexCache volumes in different nodes of a cluster are not the same, then each FlexCache volume in the cluster might contain a different version of the cached file. Therefore, to ensure that stale data is not served to clients, and to maintain consistency of data in the cached file across all the FlexCache volumes of an origin volume, you must set all the TTL values except the -meta-ttl to 0. The -meta-ttl parameter has a default value of 15 while the other parameters have a default value of 0.

Note: Setting the TTL values to 0 can negatively impact the caching performance because every data request to a volume without delegations causes an access to the origin volume.

Related concepts

Cache-read delegations on page 72

Write operation proxy

When a client writes to a cached file, the FlexCache volume proxies the write request to the origin volume, which makes the requested changes. The FlexCache volume is then updated to reflect the changes.

When writes are relayed through the FlexCache volume, if that volume contains the portion of the file that is changing, the affected data blocks are invalidated from the particular FlexCache volume. In addition, the origin volume revokes any delegation to the file. A subsequent read of those data blocks is required to store the changed data back into the cache.

As a result of the write, the changed file has updated attributes. These changed attributes impact all those FlexCache volumes in the cluster that contain cached copies of the file. After the TTL values expire on each volume containing a cached copy of the file, the FlexCache volumes compare the file attributes with those on the origin volume. This comparison forces an invalidation of the entire file from the cached copy in its FlexCache volume. The invalidation happens only on FlexCache volumes that did not proxy the write operation. In addition, the origin volume revokes delegations to all the cached copies of the file. The cache needs to be rewarmed for the FlexCache volumes to store the changed data.

Related concepts

How data changes affect FlexCache volumes on page 70

What cache hits and misses are

Cache hits and misses indicate if the data requested by the client is served directly from the FlexCache volume or from the origin volume. The occurrence of a cache hit or miss depends on factors such as availability of the requested data in the cache, the TTL (time-to-live) values, and the difference between attributes of a file in the cache and the origin.

Cache hits

When a client sends a read request, if the relevant block is cached in the FlexCache volume, the data is read directly from the FlexCache volume. This is called a *cache hit*. Cache hits are the result of a previous request.

A cache hit can be one of the following types:

- Hit: The requested data is already cached and the FlexCache volume serves the read request without accessing the origin volume.
- Hit-Verify: The requested data is already cached but the FlexCache volume needs to verify the attributes of the cached file with those on the origin. The FlexCache does not request any data from the origin volume while performing this verification process.

Cache misses

If the requested data is not on the FlexCache volume, or if the data has changed since it was cached, the caching system loads the data from the origin volume, and then returns it to the requesting client. This is called a *cache miss*.

A cache miss can be one of the following types:

- Miss: The requested data is not cached and is read from the origin volume.
- Miss-Verify: The requested data is cached but the data has changed since it was last cached. Therefore, the requested data is read from the origin. The impacted data blocks are invalidated from the cache and replaced with the changed data.

Related concepts

How FlexCache volumes share space with other volumes on page 76

How autosizing of FlexCache volumes improves caching performance

Although you can specify the size of a FlexCache volume, you should allow Data ONTAP to control the size of your FlexCache volumes automatically for better caching performance.

When no size is specified, Data ONTAP sizes FlexCache volumes based on algorithms that examine the number of files that the origin volume can contain. The resulting FlexCache volume is created with a nominal size that is proportional to, but usually smaller than, the origin volume. The FlexCache volume automatically grows in size as it caches more data.

Note: While Data ONTAP can create large FlexCache volumes to cache correspondingly large origin volumes, the space is not necessarily guaranteed. If enough space is not available in the aggregate, Data ONTAP ejects data from the FlexCache volume to make room on the aggregate for un-cached data.

If you choose to specify a size for your FlexCache volumes, making the volume very small can negatively impact your caching performance. When the FlexCache volume begins to fill up, it flushes randomly chosen, previously cached files to make room for newly requested data. When data from the flushed files is requested again, it must be retrieved again from the origin volume.

Therefore, it is best to allow Data ONTAP to increase the size of your FlexCache volumes automatically as the size of the cache working set increases. This method has the following advantages:

- If the size of the FlexCache volume's working set increases, as long as there is space in the containing aggregate, the FlexCache volume automatically increases its size rather than ejecting data from the cache, which could affect data access performance.
- These size increases happen without operator intervention.
- If you have several FlexCache volumes sharing the same aggregate, the volumes that are getting the most data accesses will also receive the most space.
- If you increase the size of an aggregate, the FlexCache volumes contained by that aggregate will automatically take advantage of the extra space if needed.

How FlexCache volumes share space with other volumes

You can have multiple FlexCache volumes in the same aggregate. You can also have regular FlexVol volumes in the same aggregate as the FlexCache volumes. If you want to set up your cluster efficiently, you must understand the way these volumes share space.

When you include multiple FlexCache volumes in the same aggregate, each FlexCache volume reserves only a small amount of space. The rest of the space is allocated as required. This means that a "hot" FlexCache volume (one that is being accessed heavily) is permitted to take up more space, while a FlexCache volume that is not being accessed as often will gradually be reduced in size.

Note: When an aggregate containing FlexCache volumes runs out of free space, Data ONTAP randomly selects a FlexCache volume in that aggregate to be truncated. Truncation means that files are removed from the FlexCache volume until the size of the volume is decreased to a predetermined percentage of its former size.

If you have regular FlexVol volumes in the same aggregate as your FlexCache volumes, and the aggregate starts filling up, the FlexCache volumes can lose some of their unreserved space (if it is not being used). In this case, when the FlexCache volume needs to fetch a new data block and it does not have enough free space to accommodate the data block, an existing data block is removed from one of the FlexCache volumes to accommodate the new data block.

If the ejected data is causing many cache misses, you can add more space to the aggregate or move some of the data to another aggregate.

How FlexCache volumes are impacted during loss of connectivity

A loss of connectivity between a FlexCache volume and its origin volume can interfere with client requests for data.

All FlexCache volumes require a mapping to an origin volume, and that origin volume must exist when the FlexCache volume is created. When requests are sent to a node containing a FlexCache volume, the FlexCache volume uses its cache policies and cache coherency mechanisms to determine how to respond.

There are situations where the FlexCache volume must communicate with the origin volume. In these situations, if the connection with the node containing the origin volume is not available (or if the origin volume is offline), the FlexCache volume cannot serve client requests. Therefore, the client needs to resend its request. The client request is served only if the connectivity is restored and the volumes responsible for serving the request are online.

Related concepts

How FlexCache volumes work during a failover on page 77

How NVRAM failures on the origin volume impact FlexCache volumes

When the origin volume experiences a failure of the Non-Volatile RAM (NVRAM) and has the `nvfail` option turned on, the corresponding FlexCache volumes inherit the state of the `nvfail` option from the origin volume. You cannot directly configure `nvfail` option on a FlexCache volume.

If you change the value of the `nvfail` option in the origin volume, the FlexCache volume also inherits the changed value.

When the `nvfail` option is `on`, the FlexCache volumes corresponding to the origin volume do not relay any requests that they receive to modify the data on the origin volume until the `-in-nvfailed` option is cleared at the origin volume.

For more information about the `nvfail` option, see the *Clustered Data ONTAP Data Protection Guide* and the *Clustered Data ONTAP Commands: Manual Page Reference*.

How FlexCache volumes work during a failover

The impact on clients accessing FlexCache volumes that move from one node to another in case of a failover is minimal.

When a failure occurs on a node that contains a FlexCache volume, it functions as follows:

- The FlexCache volume fails over without the clients becoming aware of the node failure.
- If the high availability feature is configured, the aggregate containing the FlexCache volume moves to the other node of the HA pair.

  Cached data that remains on the FlexCache volume is revalidated with the origin before usage. All the cached files go through an attribute check with the source data before responding to a client request, regardless of the TTL value.
- Failover of the data LIF corresponding to the node containing the FlexCache volume is determined by the policy configured for that LIF.

  The LIF might move to a node that does not contain a FlexCache volume. If the LIF moves to a node that contains a FlexCache volume, that FlexCache volume might not have the same cached data as the FlexCache volume that belonged to the failed node. Therefore, the FlexCache volume on the node to which the LIF moves might require additional cache warming to perform the same way as the other FlexCache volume did before the failover.

Comparison of FlexCache volumes and load-sharing mirrors

Both FlexCache volumes and load-sharing mirror volumes can serve hosts from a local node in the cluster, instead of using the cluster interconnect to access the node storing the primary source of data. However, you need to understand the essential differences between them and use them in your storage system.

The following table explains the differences between load-sharing mirror volumes and FlexCache volumes:

Load-sharing mirror volumes	FlexCache volumes
The data that load-sharing mirror volumes use to serve client requests is a complete copy of the source data.	The data that FlexCache volumes use to serve client requests is cached copy of the source data, containing only data blocks that are accessed by clients.
Can be used as a disaster-recovery solution by promoting a load-sharing mirror to a source volume.	Cannot be used for disaster recovery. A FlexCache volume does not contain a complete copy of the source data.
Are read-only volumes, with the exception of admin privileges for write access or bypass of the load-sharing mirror.	Are read and write-through cache volumes.
A user creates one load-sharing mirror volume at a time.	A user can create one FlexCache volume at a time, or can simultaneously create FlexCache volumes on all the nodes spanned by the Storage Virtual Machine (SVM) that contains the origin volume.

Commands for managing FlexCache volumes

You use the `volume flexcache` commands to create, delete, and display information about FlexCache volumes on all nodes in the cluster, or to create, modify, and delete cache policies. You can use the `volume` family of commands to perform many of the same operations on individual volumes.

If you want to...	Use this command...
Create a FlexCache volume on all the nodes spanned by a Storage Virtual Machine (SVM, formerly known as Vserver) in a cluster	`volume flexcache create`
Create a FlexCache volume on a single node	`volume create`
Display information about all FlexCache volumes in the cluster	`volume flexcache show`
Delete FlexCache volumes from all nodes in the cluster	`volume flexcache delete` **Note:** You do not need to take the FlexCache volumes offline before using the `volume flexcache delete` command.

If you want to...	Use this command...

Take a FlexCache volume offline so that you can delete the volume	`volume modify`
	or
	`volume offline`

Delete a FlexCache volume from a single node	`volume delete`
View the delegations granted by a FlexVol volume to FlexCache volumes	`vserver locks show` **Note:** The command displays the delegation information depending on the value of the protocol parameter. For example, if you want to view the delegations granted to a 7-Mode FlexCache volume, you can set the value to `fcache`. If you want to view the delegations granted to a FlexCache volume within the same cluster as the origin volume, you can set the value to `flexcache_deleg`.
Note: The following commands that provide information about creating, viewing, deleting, and modifying cache policies, require the advanced privilege level.	
Create a cache policy	`volume flexcache cache-policy create`
Display the cache policies for all SVMs	`volume flexcache cache-policy show`
Display the cache policies for a single SVM	`volume flexcache cache-policy show`
Apply a cache policy to a single volume	`volume modify`
Display the cache policy that is applied to a FlexCache volume	`volume show`
Change the TTL values for a cache policy	`volume flexcache cache-policy modify`
Delete a cache policy	`volume flexcache cache-policy delete`

For more information, see the man pages.

Using FlexClone volumes to create efficient copies of your FlexVol volumes

FlexClone volumes are writable, point-in-time copies of a parent FlexVol volume. FlexClone volumes are space-efficient because they share the same data blocks with their parent FlexVol volumes for common data. The Snapshot copy used to create a FlexClone volume is also shared with the parent volume.

You can clone an existing FlexClone volume to create another FlexClone volume. You can also create a clone of a FlexVol volume containing LUNs and LUN clones.

Starting with Data ONTAP 8.2, you can create two types of FlexClone volumes: read-write FlexClone volumes and data protection FlexClone volumes. While you can create a read-write FlexClone volume of a regular FlexVol volume, you must use only a SnapVault secondary volume to create a data protection FlexClone volume.

Understanding FlexClone volumes

FlexClone volumes can be managed similarly to regular FlexVol volumes, with a few important differences. For instance, the changes made to the parent FlexVol volume after the FlexClone volume is created are not reflected in the FlexClone volume.

The following list outlines important facts about FlexClone volumes:

Note: The following statements are applicable to both read-write and data protection FlexClone volumes unless specified otherwise.

- A FlexClone volume is a point-in-time, writable copy of the parent FlexVol volume.
- A FlexClone volume is a fully functional FlexVol volume similar to its parent.
- A FlexClone volume is always created in the same aggregate as its parent.
- A FlexClone volume is always created in the same Storage Virtual Machine (SVM) as its parent.
- An Infinite Volume cannot be used as the parent of a FlexClone volume.
- Because a FlexClone volume and its parent share the same disk space for common data, creating a FlexClone volume is instantaneous and requires no additional disk space (until changes are made to the FlexClone volume or its parent).
- A FlexClone volume is created with the same volume guarantee as its parent.
  The volume guarantee setting is enforced for the new FlexClone volume only if there is enough space in the containing aggregate.
- A FlexClone volume is created with the same space reservation and fractional reserve settings as its parent.
- A FlexClone volume is created with the same Snapshot schedule as its parent.
- A FlexClone volume is created with the same language setting as its parent.
- The common Snapshot copy shared between a FlexClone volume and its parent volume cannot be deleted while the FlexClone volume exists.

- While a FlexClone volume exists, some operations on its parent are not allowed, such as deleting the parent volume.
- You cannot create clones of volumes in a storage system that is in a partial giveback state.
- You can sever the connection between the parent volume and a read-write FlexClone volume. This is called *splitting* the FlexClone volume. Splitting removes all restrictions on the parent volume and causes the FlexClone volume to use its own additional disk space rather than sharing space with its parent.

  Note: You cannot split a data protection FlexClone volume from its parent volume.

  Attention: Splitting a FlexClone volume from its parent volume deletes all existing Snapshot copies of the FlexClone volume, and disables the creation of new Snapshot copies while the splitting operation is in progress.

  If you want to retain the Snapshot copies of the FlexClone volume, you can move the FlexClone volume to a different aggregate by using the volume move command. During the volume move operation, you can also create new Snapshot copies, if required.

- Quotas applied to the parent volume are *not* automatically applied to the FlexClone volume.
- When a FlexClone volume is created, any LUNs present in the parent volume are present in the FlexClone volume but are unmapped and offline.

Related concepts

How moving a FlexVol volume works on page 59

Related tasks

Moving a volume on page 61

FlexClone volumes and shared Snapshot copies

When volume guarantees are in effect, a new FlexClone volume uses the Snapshot copy it shares with its parent to minimize its space requirements. If you delete the shared Snapshot copy, you might increase the space requirements of the FlexClone volume.

For example, suppose that you have a 100-MB FlexVol volume that has a volume guarantee of `volume`, with 70 MB used and 30 MB free, and you use that FlexVol volume as a parent volume for a new FlexClone volume. The new FlexClone volume has an initial volume guarantee of `volume`, but it does not require a full 100 MB of space from the aggregate, as it would if you had copied the volume. Instead, the aggregate needs to allocate only 30 MB (100 MB – 70 MB) of free space to the clone.

Now, suppose that you delete the shared Snapshot copy from the FlexClone volume. The FlexClone volume can no longer optimize its space requirements, and the full 100 MB is required from the containing aggregate.

Note: If you are prevented from deleting a Snapshot copy from a FlexClone volume due to "insufficient space in the aggregate" it is because deleting that Snapshot copy requires the

allocation of more space than the aggregate currently has available. You can either increase the size of the aggregate, or change the volume guarantee of the FlexClone volume.

How to identify shared Snapshot copies in FlexClone volumes

You can identify a shared Snapshot copy by using the `volume snapshot show` command with the -instance parameter to list the Snapshot copies *in the parent volume*. Any Snapshot copy that is marked as busy in the parent volume and is also present in the FlexClone volume is a shared Snapshot copy.

How you use volume SnapMirror replication with FlexClone volumes

Because both volume SnapMirror replication and FlexClone volumes rely on Snapshot copies, there are some restrictions on how the two features can be used together. For instance, you can create a volume SnapMirror relationship using a FlexClone volume or its parent as the source volume. However, you cannot create a new volume SnapMirror relationship using either a FlexClone volume or its parent as the destination volume.

Considerations for creating a FlexClone volume from a SnapMirror source or destination volume

You can create a FlexClone volume from the source or destination volume in an existing volume SnapMirror relationship. However, doing so could prevent future SnapMirror replication operations from completing successfully.

Replication might not work because when you create the FlexClone volume, you might lock a Snapshot copy that is used by SnapMirror. If this happens, SnapMirror stops replicating to the destination volume until the FlexClone volume is destroyed or is split from its parent. You have two options for addressing this issue:

- If you require the FlexClone volume on a temporary basis and can accommodate a temporary stoppage of the SnapMirror replication, you can create the FlexClone volume and either delete it or split it from its parent when possible.

  The SnapMirror replication continues normally when the FlexClone volume is deleted or is split from its parent.
- If a temporary stoppage of the SnapMirror replication is not acceptable, you can create a Snapshot copy in the SnapMirror source volume, and then use that Snapshot copy to create the FlexClone volume. (If you are creating the FlexClone volume from the destination volume, you must wait until that Snapshot copy replicates to the SnapMirror destination volume.)

  This method of creating a Snapshot copy in the SnapMirror source volume allows you to create the clone without locking a Snapshot copy that is in use by SnapMirror.

How splitting a FlexClone volume from its parent works

Splitting a read-write FlexClone volume from its parent removes any space optimizations that are currently used by the FlexClone volume. After the split, both the FlexClone volume and the parent

volume require the full space allocation determined by their volume guarantees. The FlexClone volume becomes a normal FlexVol volume.

You must be aware of the following considerations related to clone-splitting operations:

- You can split only read-write FlexClone volumes. Data protection FlexClone volumes cannot be split from their parent volumes.
- When you split a FlexClone volume from its parent, all existing Snapshot copies of the FlexClone volume are deleted. If you want to retain the Snapshot copies of the FlexClone volume, you can move the FlexClone volume to a different aggregate by using the `volume move` command. During the volume move operation, you can also create new Snapshot copies, if required.
- No new Snapshot copies can be created of the FlexClone volume during the split operation.
- Because the clone-splitting operation is a copy operation that might take considerable time to complete, Data ONTAP provides the `volume clone split stop` and `volume clone split status` commands to stop or check the status of a clone-splitting operation.
- The clone-splitting operation proceeds in the background and does not interfere with data access to either the parent or the clone volume.
- The FlexClone volume must be online when you start the split operation.
- The parent volume must be online for the split operation to succeed.
- If the FlexClone volume has a data protection or load sharing mirror, it cannot be split from its parent volume.
- If you split a FlexClone volume from a FlexVol volume that has deduplication and compression enabled, the split volume *does not* have deduplication and compression enabled.
- After a FlexClone volume and its parent volume have been split, they cannot be rejoined.

FlexClone volumes and LUNs

You can clone FlexVol volumes that contain LUNs and FlexClone LUNs.

Note: LUNs in this context refer to the LUNs that Data ONTAP serves to clients, not to the array LUNs used for storage on a storage array.

When you create a FlexClone volume, LUNs in the parent volume are present in the FlexClone volume but they are not mapped and they are offline. To bring the LUNs in the FlexClone volume online, you need to map them to initiator groups.

If the parent volume contains FlexClone LUNs, the FlexClone volume also contains FlexClone LUNs, which share storage with the FlexClone LUNs in the parent volume.

Understanding data protection FlexClone volumes

You can use the FlexClone technology to create a space-efficient copy of a data protection volume that is used as a SnapVault secondary volume. The Snapshot copy that establishes the SnapVault

relationship between the primary and secondary volumes is the backing Snapshot copy for creating the data protection FlexClone volume.

Data protection FlexClone volumes are similar to read-write FlexClone volumes because they share common blocks with their parent FlexVol volumes. However, you can create a data protection FlexClone volume only from a parent FlexVol volume that is also a secondary SnapVault volume. In addition, you cannot split a data protection FlexClone volume from its parent volume.

For more information about volumes in a SnapVault relationship, see the *Clustered Data ONTAP Data Protection Guide*.

Creating a FlexClone volume

If you need an instantaneous copy of your data without using a lot of disk space, you can create a FlexClone volume from the parent FlexVol volume that contains the data. Depending on the type of the parent volume, you can create a read-write FlexClone or a data protection FlexClone volume.

Before you begin

- The FlexClone license must be installed on the cluster.
- The volume you want to clone must be online.

About this task

- You can create a data protection FlexClone volume only from a parent FlexVol volume that is a SnapVault secondary volume.
- After you create a FlexClone volume, you cannot delete the parent volume while the FlexClone volume exists.

Step

1. Use the `volume clone create` command to create a FlexClone volume.

   Note: While creating a read-write FlexClone volume, you do not need to specify the base Snapshot copy. Data ONTAP creates a Snapshot copy if you do not name any specific Snapshot copy to be used as the base Snapshot copy for the clone. However, you must specify the base Snapshot copy for creating a data protection FlexClone volume.

Example

- The following command creates a read-write FlexClone volume vol1_clone of the parent volume vol1:

  ```
  volume clone create -vserver vs0 -flexclone vol1_clone -type RW -
  parent-volume vol1
  ```

- The following command creates a data Protection FlexClone volume vol_dp_clone of the parent volume dp_vol by using the base Snapshot copy snap1:

```
volume clone create -vserver vs1 -flexclone vol_dp_clone -type DP -
parent-volume dp_vol -parent-snapshot snap1
```

Splitting a FlexClone volume from its parent

If you want a read-write FlexClone volume to have its own disk space rather than using that of its parent, you can split it from its parent. Because this operation creates a copy of data that is currently shared between the parent and the FlexClone, it can take some time to complete.

Before you begin

You must ensure that the FlexClone volume to be split from its parent is a read-write FlexClone volume. A data protection FlexClone volume cannot be split from its parent volume.

About this task

Splitting a FlexClone volume from its parent volume consumes free space from the containing aggregate. If you do not have enough privileges to view the space available in your aggregate, you need to contact your storage administrator to ensure that the split operation can complete.

Steps

1. Use the `volume clone show` command with the **`estimate`** parameter to determine the amount of free space needed to complete the split operation.

   #### Example

   The following example provides information about the free space required to split a FlexClone volume clone1 from its parent volume vol1:

   ```
   cluster1::> volume clone show -estimate -vserver vs1 -flexclone
   clone1 -parent-volume volume1
                                 Split
   Vserver   FlexClone       Estimate
   --------- ------------- ----------
   vs1       clone1           40.73MB
   ```

2. Determine the amount of free space in the aggregate that contains the FlexClone volume and its parent by using the `storage aggregate show` command.
3. If the containing aggregate does not have enough free space available, add storage to the aggregate by using the `storage aggregate add-disks` command.
4. Start the split operation by using the `volume clone split start` command.

   #### Example

   The following example shows how you can initiate the process to split the FlexClone volume clone1 from its parent volume vol1:

```
cluster1::> volume clone split start -vserver vs1 -flexclone clone1

Warning: Are you sure you want to split clone volume clone1 in
Vserver vs1 ?
{y|n}: y
[Job 1617] Job is queued: Split clone1.
```

5. You can monitor the progress of the splitting job by using the `job show` command.
6. You can confirm if the split volume is no longer a FlexClone volume by using the `volume show` command with the **`fields`** parameter set to **`clone-volume`**.

   The value of the **`clone-volume`** option is `false` for a volume that is not a FlexClone volume.

   #### Example

   The following example shows how you can verify if the volume clone1 that is split from its parent is not a FlexClone volume.

```
cluster1::> volume show clone1 -fields clone-volume
vserver volume clone-volume
------- ------ ------------
vs1     clone1 false
```

Determining the space used by a FlexClone volume

You can determine the space used by a FlexClone volume based on its nominal size and the amount of space it shares with the parent FlexVol volume.

About this task

When a FlexClone volume is created, it shares all of its data with its parent volume. Therefore, although the nominal size of the FlexVol volume is the same as its parent's size, it uses very little free space from the aggregate. The free space used by a newly-created FlexClone volume is approximately 0.5% of its nominal size. This space is used to store the FlexClone volume's metadata.

New data written to either the parent or the FlexClone volume is not shared between the volumes. The increase in the amount of new data that gets written to the FlexClone volume leads to an increase in the space the FlexClone volume requires from its containing aggregate.

Steps

1. Determine the nominal size of the FlexClone volume by using the `volume size` command.

   #### Example

   The following example shows the nominal size of a FlexClone volume clone1:

```
cluster1::> volume size -volume clone1

vol size: Volume "vs1:clone1" has size 200m.
```

2. Determine the amount of space that is shared between the parent and FlexClone volume by using the `volume clone split estimate` command.

   Example

   The following example shows the amount of space shared between the FlexClone volume clone1 and its parent volume vol1:

```
cluster1::>  volume clone split estimate -vserver vs1 -flexclone
clone1
                              Split
Vserver   FlexClone       Estimate
--------- ------------- ----------
vs1       clone1            2.34MB
```

3. Subtract the size of the shared space from the nominal size of the FlexClone volume to determine the amount of free space being used by the FlexClone volume.

Using FlexClone files and FlexClone LUNs to create efficient copies of files and LUNs

FlexClone files and FlexClone LUNs are writable, space-efficient clones of parent files and parent LUNs, and help in efficient utilization of the physical aggregate space. FlexClone files and FlexClone LUNs are supported only for FlexVol volumes, not for Infinite Volumes.

FlexClone files and FlexClone LUNs utilize 0.4 percent of their size to store the metadata. Clones share the data blocks of their parent files and parent LUNs and occupy negligible storage space until clients write new data either to the parent file or LUN, or to the clone.

Clients can perform all file and LUN operations on both the parent and the clone entities.

Benefits of FlexClone files and FlexClone LUNs

The process of creating FlexClone files or FlexClone LUNs is highly space-efficient and time-efficient because the cloning operation does not involve physically copying any data.

You can create space-efficient copies of your data by using FlexClone files and FlexClone LUNs in situations such as the following:

- When you need to deploy, upgrade, or redeploy thousands of standardized virtual desktops or servers.
- When you need a copy of a database for application development purposes.
- When you need to boot servers in a server farm.

You can create FlexClone LUNs of the parent boot LUN, then use the FlexClone LUN to boot a server in a server farm.

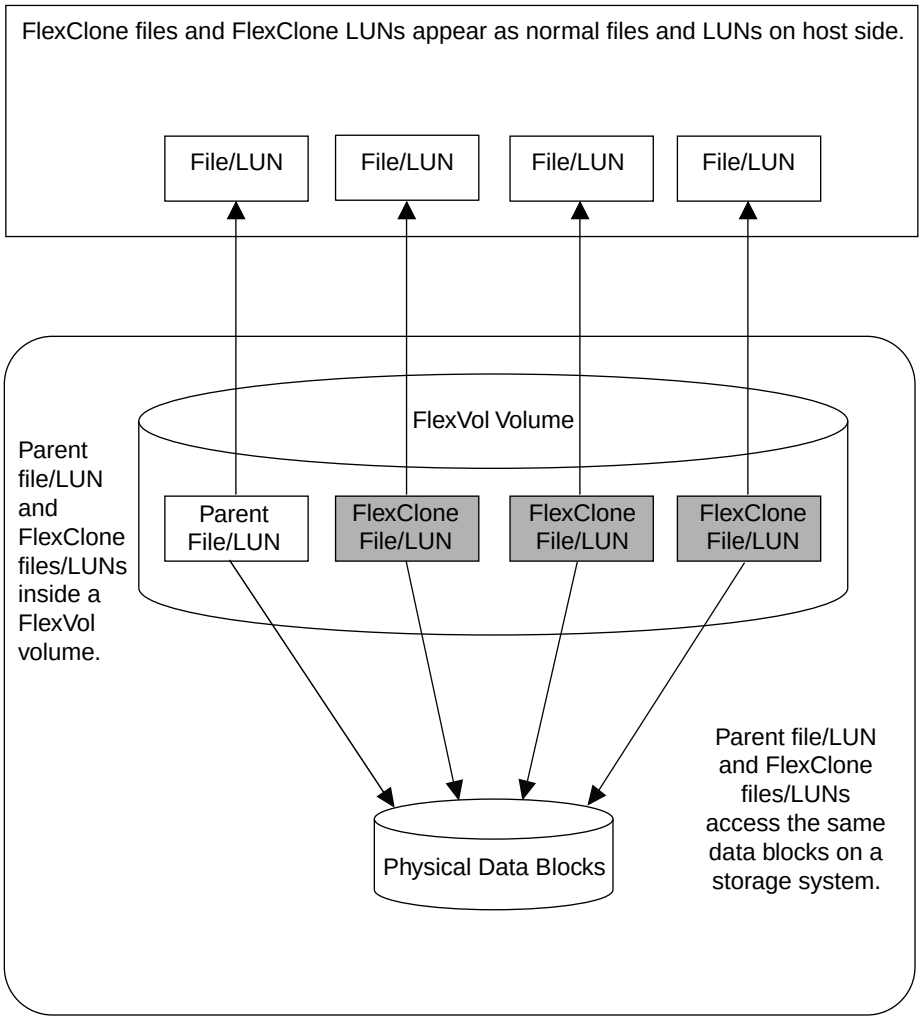
How FlexClone files and FlexClone LUNs work

FlexClone files and FlexClone LUNs share the same physical data blocks with their parent files and LUNs present in FlexVol or FlexClone volumes, and occupy negligible space in the form of metadata.

You can create a clone of a file that is present in a FlexVol volume in a NAS environment, and you can also clone a LUN in a SAN environment.

The cloned copies are highly space-efficient and time-efficient because the cloning operation does not copy physical blocks of data. When you write new data to a parent or clone, then the entity on which new data is written starts occupying extra storage space.

The following illustration shows the parent files or LUNs and FlexClone files or LUNs accessing the same data blocks on the storage system. On the host side, the parent files or LUNs and FlexClone files or LUNs appear as normal files and LUNs:



The cloning operation is instantaneous and has no impact on client access to the parent file or LUN. Clients that are accessing the parent file or LUN do not experience any disruption or outage. Clients can perform all operations on FlexClone files and FlexClone LUNs as they can on standard files and LUNs.

You can create a maximum of 32,767 FlexClone files or FlexClone LUNs from a parent file or LUN without creating a physical copy of the parent entity. If you try to create more than 32,767 clones, Data ONTAP automatically creates a new physical copy of the parent file or LUN.

Considerations for working with FlexClone files and FlexClone LUNs

You should keep several considerations in mind when working with FlexClone files and FlexClone LUNs.

- You can create FlexClone files and LUNs only in the same FlexVol volume containing the parent files and LUNs.
- You can clone a complete file, sub-file, LUN, or sub-LUN.
  To clone a sub-file or sub-LUN, you should know the block range of the parent entity and clone entity.
- The `sis` attribute is added to a FlexVol volume when a FlexClone file or FlexClone LUN is created for the first time.
- When clients write new data either to a FlexClone file or FlexClone LUN, or the parent file or parent LUN, then the new data occupies additional storage space.
- If you delete the FlexClone files or LUNs, the parent files or LUNs are not affected.
  Deleting a parent file or LUN has no impact on the FlexClone files or FlexClone LUNs.
- If you create FlexClone files or LUNs from a Snapshot copy, you cannot create new Snapshot copies until the cloning process is complete.
- If a FlexVol volume containing FlexClone files and LUNs has the fractional reserve set to zero, you must follow extra configuration and management requirements to ensure that the applications accessing the FlexVol volume do not receive errors due to lack of space (ENOSPC).

Related concepts

Considerations for setting fractional reserve for FlexVol volumes on page 32

Creating a FlexClone file or FlexClone LUN

You can create space-efficient and time-efficient clones of files and LUNs present in FlexVol volumes or FlexClone volumes by using the `volume file clone create` command.

Before you begin

- FlexClone license must be installed on the cluster.
- If multiple block ranges are used for sub-LUN cloning or sub-file cloning, the block numbers must not overlap.

About this task

Depending on the privileges assigned by the cluster administrator, an SVM administrator can create FlexClone files and FlexClone LUNs. For more information about the SVM administrator capabilities, see the *Clustered Data ONTAP System Administration Guide for SVM Administrators*.

Step

1. To create FlexClone file or FlexClone LUN, use the `volume file clone create` command.

   Example

   The following example shows how you can create a FlexClone file file1_clone of the parent file file1_source in the volume vol1:

   ```
   cluster1::> volume file clone create -vserver vs0 -volume vol1 -
   source-path /file1_source -destination-path /file1_clone
   ```

   For more information about using this command, see the man pages.

Viewing the space savings due to FlexClone files and FlexClone LUNs

You can view the percentage of disk space saved by block sharing within a volume containing FlexClone files and LUNs.

Step

1. To view the space saving achieved due to FlexClone files and FlexClone LUNs, enter the following command:

   `df -s` ***`volname`***

   `volname` is the name of the FlexVol volume.

   Note: If you run the `df -s` command on a deduplication-enabled FlexVol volume, you can view the space saved by both deduplication and FlexClone files and LUNs.

Example

The following example shows the space saving on a FlexClone volume test1:

```
systemA> df -s test1

Filesystem         used   saved   %saved Vserver
/vol/test1/        4828    5744     54%  vs1
```

How a FlexVol volume can reclaim free space from FlexClone LUNs

Starting with Data ONTAP 8.2, you can configure the autodelete settings of a FlexVol volume to automatically delete FlexClone LUNs when the free space in a volume decreases below a particular threshold value.

A FlexVol volume that has the autodelete capability enabled resorts to automatic deletion of FlexClone LUNs only in the following situations:

- If the volume does not have Snapshot copies for automatic deletion.
- If the volume has Snapshot copies but the automatic deletion of Snapshot copies does not create sufficient free space in the volume.

Because the autodelete settings enable a FlexVol volume to directly delete FlexClone LUNs when the volume requires free space, you can preserve certain FlexClone LUNs by preventing them from getting automatically deleted.

Note: If the FlexVol volume contains FlexClone LUNs created using Data ONTAP versions earlier than 8.2 and if you want to delete them to increase the amount of free space in the volume, you can specify those FlexClone LUNs for automatic deletion.

Configuring a FlexVol volume to automatically delete FlexClone LUNs

Starting with Data ONTAP 8.2, you can enable a FlexVol volume to automatically delete FlexClone LUNs when the free space in the volume decreases below a particular threshold. You can also enable automatic deletion for FlexClone LUNs created using Data ONTAP versions earlier than 8.2.

Before you begin

- The FlexVol volume must contain FlexClone LUNs and be online.
- The FlexVol volume must not be a read-only volume.

Steps

1. Enable automatic deletion of FlexClone LUNs in the FlexVol volume by using the `volume snapshot autodelete modify` command.

   #### Example

   The following example shows how you can enable volume vol1 to trigger the automatic deletion of FlexClone LUNs for space reclamation until 25% of the volume consists of free space:

   ```
   cluster1::> volume snapshot autodelete modify -vserver vs1 -volume
   vol1 -enabled true -commitment disrupt -trigger volume -target-free-
   space 25 -destroy-list lun_clone,sfsr

   Volume modify successful on volume:vol1
   ```

   The FlexClone LUNs created using Data ONTAP 8.2 or later in the FlexVol volume are automatically enabled for deletion when you enable FlexClone LUN autodelete in the volume.

   Note: While enabling FlexClone LUNs for automatic deletion, if you set the value of the commitment parameter to `destroy`, all the FlexClone LUNs created using Data ONTAP 8.2 or later, including those marked for preservation can get deleted when the free space in the volume decreases below the specified threshold value.

2. Verify if the automatic deletion of FlexClone LUNs is enabled in the FlexVol volume by using the `volume snapshot autodelete show` command.

Example

The following example shows how you can determine if volume vol1 is enabled for automatic deletion of FlexClone LUNs:

```
cluster1::> volume snapshot autodelete show -vserver vs1 -volume vol1
         Vserver Name: vs1
          Volume Name: vol1
              Enabled: true
           Commitment: disrupt
         Defer Delete: user_created
         Delete Order: oldest_first
  Defer Delete Prefix: (not specified)
    Target Free Space: 25%
              Trigger: volume
         Destroy List: lun_clone,sfsr
Is Constituent Volume: false
```

3. If the FlexVol volume has FlexClone LUNs created using Data ONTAP versions earlier than 8.2, and if you want to enable these FlexClone LUNs for automatic deletion, then perform the following steps:

a) Use the `volume file clone autodelete` command to enable automatic deletion of a particular FlexClone LUN.

Example

The following example shows how you can enable the automatic deletion of a FlexClone LUN lun1_clone contained in volume vol1:

```
cluster1::> volume file clone autodelete -vserver vs1 -clone-
path /vol/vol1/lun1_clone -enabled true
```

b) Use the `lun show` command to verify if the FlexClone LUN is enabled for automatic deletion.

Example

The following example shows how you can verify if the FlexClone LUN lun1_clone is enabled for automatic deletion:

```
cluster1::> lun show -instance -lun lun1_clone

Vserver Name: vs1
                    LUN Path: /vol/vol1/lun1_clone
                Volume Name: vol1
                 Qtree Name: ""
                    LUN Name: lun1
                    LUN Size: 50.01GB
                     OS Type: windows
          Space Reservation: disabled
              Serial Number: BZHV0]C9Pj3U
```

```
                    Comment:
 Space Reservations Honored: true
           Space Allocation: disabled
                      State: online
                   LUN UUID: b02b01da-438d-11e2-821a-123478563412
                     Mapped: unmapped
                 Block Size: 512
           Device Legacy ID: -
           Device Binary ID: -
             Device Text ID: -
                  Read Only: false
Inaccessible Due to Restore: false
                  Used Size: 0
        Maximum Resize Size: 502.0GB
              Creation Time: 12/11/2012 12:24:19
                      Class: -
                      Clone: -
   Clone Autodelete Enabled: true
           QoS Policy Group: -
```

For more information about using the commands, see the respective man pages.

Preventing a specific FlexClone LUN from being autodeleted

If you configure a FlexVol volume to automatically delete FlexClone LUNs, any FlexClone LUN that fits the criteria you configure might be deleted. If you have specific FlexClone LUNs that you want to preserve, you can exclude them from the automatic FlexClone deletion process.

Before you begin

A FlexClone license must be installed.

About this task

Starting with Data ONTAP 8.2, when you create a FlexClone LUN, by default the FlexClone LUN can be deleted automatically to reclaim space on the volume if you have configured the volume for FlexClone autodelete. If you have not configured the volume for FlexClone autodelete, FlexClone LUNs are not deleted automatically.

Note: When you create a FlexClone LUN from a Snapshot copy, a background split between the FlexClone LUN and the Snapshot copy is automatically triggered. If this background split has not completed and this Snapshot copy is autodeleted, that FlexClone LUN will be deleted even if you have disabled the FlexClone autodelete feature for that FlexClone LUN. After the background split is complete and if that Snapshot copy is deleted, the FlexClone LUN will not be deleted.

Attention: If you set the commitment level on the volume to `try` or `disrupt`, then you can individually preserve specific FlexClone LUNs by disabling FlexClone LUN autodelete for those LUNs. However, if you set the commitment level on the volume to `destroy` and the destroy lists include file_clone and lun_clone, then the volume setting overrides the LUN setting and all FlexClone LUNs can be deleted.

Steps

1. Use the `volume file clone autodelete` command to prevent a specific FlexClone LUN from being automatically deleted.

   Example

   `volume file clone autodelete -vserver vs1 -volume vol1 -clone-path lun1_clone -enable false`

   The FlexClone LUN lun1_clone cannot be deleted automatically to reclaim space on the volume.

2. To verify that the FlexClone LUN cannot be deleted automatically, use the `lun show` command.

   Example

   `lun show -instance -lun lun1_clone`

```
               Vserver Name: vs1
                   LUN Path: /vol/vol1/lun1_clone
                Volume Name: vol1
                 Qtree Name: ""
                   LUN Name: lun1_clone
                   LUN Size: 50.01GB
                    OS Type: windows
          Space Reservation: disabled
              Serial Number: 2FiMg]C2FFHF
                    Comment:
 Space Reservations Honored: true
           Space Allocation: disabled
                      State: online
                   LUN UUID: b02b01da-438d-11e2-821a-123478563412
                     Mapped: unmapped
                 Block Size: 512
           Device Legacy ID: -
           Device Binary ID: -
             Device Text ID: -
                  Read Only: false
Inaccessible Due to Restore: false
                  Used Size: 0
        Maximum Resize Size: 502.0GB
              Creation Time: 12/11/2013 12:24:19
                      Class: -
                      Clone: -
   Clone Autodelete Enabled: false
           QoS Policy Group: -
```

Result

FlexClone LUN lun1_clone cannot be autodeleted.

Features supported with FlexClone files and FlexClone LUNs

FlexClone files and FlexClone LUNs work with different Data ONTAP features such as deduplication, Snapshot copies, quotas, and volume SnapMirror.

The following features are supported with FlexClone files and FlexClone LUNs:

- Deduplication
- Snapshot copies
- Access control lists
- Quotas
- FlexClone volumes
- NDMP
- Volume SnapMirror
- The `volume move` command
- The `volume copy` command
- Space reservation
- HA configuration

How deduplication works with FlexClone files and FlexClone LUNs

You can efficiently use the physical storage space of the data blocks by creating a FlexClone file or FlexClone LUN of the parent file and parent LUN in a deduplication-enabled volume.

The block-sharing mechanism used by FlexClone files and LUNs is also used by deduplication. You can maximize the space savings in a FlexVol volume by enabling deduplication on the volume and then cloning the deduplication-enabled volume.

Note: While executing the `sis undo` command on a deduplication-enabled volume, you cannot create FlexClone files and FlexClone LUNs of the parent files and parent LUNs residing in that volume.

How Snapshot copies work with FlexClone files and FlexClone LUNs

You can create FlexClone files and FlexClone LUNs from an existing Snapshot copy of the parent files and parent LUNs contained in a FlexVol volume. If the base Snapshot copy of a FlexVol volume is deleted, the dependent FlexClone files are not deleted.

You must consider the following while using a Snapshot copy for creating FlexClone files and FlexClone LUNs:

- If the Snapshot copy gets automatically deleted because of using the `volume snapshot autodelete` command with the commitment parameter set to `destroy`, the dependent FlexClone files are not deleted but the dependent FlexClone LUNs are deleted. However, if the block sharing on the FlexClone files is not completed, data might be lost.

- You cannot manually delete a Snapshot copy from which FlexClone files or FlexClone LUNs are being created until the block-sharing process between the parent and clone entities is complete. The Snapshot copy remains locked until the completion of the block-sharing process, which happens in the background. Therefore, when you try deleting a locked Snapshot copy, the system displays a message asking you to retry the operation after some time. In such a situation, if you want to manually delete the particular Snapshot copy, you must keep retrying the deletion operation so that the Snapshot copy gets deleted after the block sharing is complete.

For more information about the automatic and manual deletion of Snapshot copies, see the *Clustered Data ONTAP Data Protection Guide*.

How access control lists work with FlexClone files and FlexClone LUNs

FlexClone files and FlexClone LUNs inherit the access control lists of their parent files and LUNs.

If the parent files contain Windows NT streams, the FlexClone files also inherit the stream information. However, parent files containing more than six streams cannot be cloned.

How quotas work with FlexClone files and FlexClone LUNs

Quota limits are applied on the total logical size of the FlexClone files or FlexClone LUNs. Starting from Data ONTAP 8.1, cloning operations do not fail block sharing even if it causes quotas to exceed.

When you create a FlexClone file or FlexClone LUN, quotas do not recognize any space savings. For example, if you create a FlexClone file of a parent file of 10 GB, you are only using 10 GB of physical space, but the quota utilization is recorded as 20 GB (10 GB for the parent and 10 GB for the FlexClone file).

If the creation of a FlexClone file or LUN results in the group or user quota's being exceeded, the clone operation succeeds provided the FlexVol volume has enough space to hold the metadata for the clone. However, the quota for that user or group is oversubscribed.

How FlexClone volumes work with FlexClone files and FlexClone LUNs

You can create a FlexClone volume of a FlexVol volume that has both a FlexClone file and FlexClone LUN and its parent file or LUN in it.

FlexClone files or FlexClone LUNs and their parent files or LUNs that are present in the FlexClone volume continue to share blocks the same way they do in the parent FlexVol volume. In fact, all the FlexClone entities and their parents share the same underlying physical data blocks, minimizing physical disk space usage.

If the FlexClone volume is split from its parent volume, then the FlexClone files or FlexClone LUNs and their parent files or LUNs stop sharing the blocks in the clone of the FlexClone volume. Thereafter they exist as independent files or LUNs. This means that the clone of the volume uses more space than before the splitting operation.

How NDMP works with FlexClone files and FlexClone LUNs

NDMP works at the logical level with FlexClone files and FlexClone LUNs. All FlexClone files or LUNs are backed up as separate files or LUNs.

When you use NDMP services to back up a qtree or a FlexVol volume that contains FlexClone files or FlexClone LUNs, block sharing between parent and clone entities is not preserved, and clone entities are backed up to tape as separate files or LUNs. The space saving is lost. Therefore, the tape onto which you are backing up should have sufficient space to store the expanded amount of data. When you restore, all the FlexClone files and FlexClone LUNs are restored as separate physical files and LUNs. You can enable deduplication on the volume to restore the block-sharing benefits.

Note: When FlexClone files and FlexClone LUNs are being created from an existing Snapshot copy of a FlexVol volume, you cannot back up the volume to tape until the block-sharing process, which happens in the background, is complete. If you use NDMP on the volume when the block-sharing process is in progress, the system displays a message asking you to retry the operation after some time. In such a situation, you must keep retrying the tape backup operation so that it succeeds after the block sharing is complete.

For more information about tape backup, see the *Clustered Data ONTAP Data Protection Tape Backup and Recovery Guide*

How volume SnapMirror works with FlexClone files and FlexClone LUNs

Volume SnapMirror used with FlexClone files and FlexClone LUNs helps in maintaining space savings because the cloned entities are replicated only once.

If a FlexVol volume is a volume SnapMirror source and contains FlexClone files or FlexClone LUNs, volume SnapMirror transfers only the shared physical block and a small amount of metadata to the volume SnapMirror destination. The destination stores only one copy of the physical block, and this block is shared between the parent and cloned entities. Therefore, the destination volume is an exact copy of the source volume and all the clone files or LUNs on the destination volume share the same physical block.

For more information about volume SnapMirror, see the *Clustered Data ONTAP Data Protection Guide*.

How volume move affects FlexClone files and FlexClone LUNs

During the cutover phase of a volume move operation, you cannot create FlexClone files or FlexClone LUNs of a FlexVol volume.

How volume copy works with FlexClone files and FlexClone LUNs

You can perform a volume copy operation on a FlexVol volume that has FlexClone files and FlexClone LUNs in it.

After the volume copy operation is done, the FlexClone files and FlexClone LUNs and their parents on the destination FlexVol volume share the same data blocks as they did on the source FlexVol volume.

For more information about the `vol copy` command, see the *Clustered Data ONTAP Data Protection Guide*.

How space reservation works with FlexClone files and FlexClone LUNs

FlexClone files and FlexClone LUNs inherit the space reservation attribute from the parent file and parent LUN.

FlexClone files and FlexClone LUNs inherit the space reservation settings of the parent file and the parent LUN. Therefore, if there is not enough space in the FlexVol volume to create a FlexClone LUN with the same space reservation as that of the parent, then the cloning operation fails.

Note: The space required according to space reservation attribute is separate for the parent LUN and the FlexClone LUN.

How an HA configuration works with FlexClone files and FlexClone LUNs

FlexClone file and FlexClone LUN operations are supported in an HA configuration.

In an HA pair, you cannot create FlexClone files or FlexClone LUNs on the partner while the takeover or giveback operation is in progress. All the pending block sharing operations on the partner are resumed after the takeover or giveback operation is complete.

Using a load-sharing mirror to balance loads

A load-sharing mirror reduces the network traffic to a FlexVol volume by providing additional read-only access to clients. You can create and manage load-sharing mirrors to distribute read-only traffic away from a FlexVol volume. Load-sharing mirrors do not support Infinite Volumes.

A set of load-sharing mirrors consists of a source volume that can fan out to one or more destination volumes. Each load-sharing mirror in the set must belong to the same Storage Virtual Machine (SVM) as the source volume of the set. The load-sharing mirrors should also be created on different aggregates and accessed by different nodes in the cluster to achieve proper load balancing of client requests.

Note: A source volume can have only one set of load-sharing mirrors.

A load-sharing mirror is mounted in the same area of the SVM namespace as its source volume. This is useful for frequently read, but infrequently updated data, such as shared binary files. For example,

you can set up one or more load-sharing mirrors to a source volume that is mounted at the `/bin` directory. Client requests to read the binaries on that volume are routed to the load-sharing mirrors, not to the source volume.

Considerations when working with load-sharing mirrors

You should be aware of certain considerations when working with load-sharing mirrors.

You should make the following considerations when working with load-sharing mirrors:

- Infinite Volumes are not supported with read-only load-sharing mirrors.
- When creating a load-sharing mirror, the creation can fail.
  If the `snapmirror create` command fails, wait 10 seconds and retry the command.
- NFSv4 clients are not supported with read-only load-sharing mirrors.
  Data ONTAP routes NFSv4 clients to the source of the load-sharing mirror for direct read and write access.
- If a read-write volume has no load-sharing mirrors, you can mount it normally and client requests are always directed to it.
  However, when a read-write volume has a load-sharing mirror, client requests are directed to the mirror only after the mirror has been initialized (has a Snapshot copy exported to clients for read-only access). As a result, it might take several minutes before client requests are directed to a newly created load-sharing mirror.
- When a load-sharing mirror is removed, or when the last one is removed if there are multiple load-sharing mirrors, all clients are immediately directed to the read-write volume.
  If a client requires read-write access to a read-write volume that has one or more load-sharing mirrors, you must mount it through the `/.` admin link at the root of the virtual server.
- If a Windows client performs a CIFS create operation or a UNIX client performs an NFS mount operation to a source volume with load-sharing mirrors, you must manually update the load-sharing mirrors first.
  If you do not, newly written data on the source volume will not be visible to clients unless they access the source volume through the `/.` admin link.
- You can create a fan-out of load-sharing mirrors from a common source volume, but you cannot cascade mirrors.

Mounting a FlexVol volume that has load-sharing mirrors for NFS clients

If you are mounting a volume that has one or more mirrors from an NFS client that must have access to the source read-write volume (for instance, to make changes to it), you must mount it through the `/.admin` link at the root of the Storage Virtual Machine (SVM).

Step

1. Mount the volume using the `/.admin` link.

   The `/.admin` link routes requests to the parent read-write volume regardless of whether the volume has any mirrors.

Example

For example, if you are mounting a client that requires read-write access to a mirrored volume mounted at `/dept/eng/sandbox` on an SVM named engdata, the following command can be used:

```
# mount -t nfs engdata:/.admin/dept/eng/sandbox mountpoint
```

If you are mounting a client that does not require read-write access to the same volume, the following command can be used:

```
# mount -t nfs engdata:/dept/eng/sandbox mountpoint
```

Mounting a FlexVol volume that has load-sharing mirrors for CIFS clients

If you are mounting a volume that has one or more mirrors from a CIFS client that must have access to the parent read-write volume (for instance, to make changes to it), you must create a new CIFS share.

Step

1. Use the `vserver cifs share create` command to maintain access to the parent read-write volume.

Example

For example, if you are mounting a client that requires read-write access to a mirrored volume mounted at `/dept/eng/sandbox` on an SVM named engdata, the following command can be used:

```
cluster1::> vserver cifs share create -vserver engdata -share-name
engdata -path /.admin/dept/eng/sandbox
```

To mount a client that requires only read access, you can use the command without the `/.admin` link.

Requests from this CIFS share will be redirected to a load-sharing mirror.

```
cluster1::> vserver cifs share create -vserver engdata -share-name
engdata -path /dept/eng/sandbox
```

Restriction when modifying the source volume

You must manually replicate the source volume to the load-sharing mirror or mirrors immediately after creating, mounting, deleting, or unmounting a FlexVol volume. Access issues occur if you do not.

Failure to manually replicate the source volume has the following results:

- If you create or mount a new volume on a source volume without replicating the load-sharing mirrors, the new or mounted volume is not visible unless you access it through the `/.admin` link.
- If you delete or unmount a volume on a source volume without replicating the load-sharing mirrors, a client can hang if it attempts to do a directory listing at the read-only mount point of the load-sharing mirrors.

Creating a set of load-sharing mirrors

You can create a set of one or more load-sharing mirrors to a read-write volume to reduce the amount of work that a volume would otherwise have to do.

About this task

This is useful for frequently read but infrequently updated data. An example might be shared binary files. You can set up one or more load-sharing mirrors to a volume that is mounted at the `/bin` directory. Client requests to read the binaries on that volume are routed to the load-sharing mirrors, not to the read-write volume.

Steps

1. Creating a FlexVol volume for load-sharing on page 102
2. Creating load-sharing mirror relationships on page 103
3. Creating a baseline for a set of load-sharing mirrors on page 104

Creating a FlexVol volume for load-sharing

You must create a FlexVol volume and designate it as a load-sharing mirror destination before you can create mirror relationships for load-sharing.

Steps

1. Use the `volume create` command with the `-type` parameter set to `DP` to create a destination volume for the load-sharing mirror.

   Note: The destination volume that you create must be the same size or greater than the source volume.

   See the *Data ONTAP Cluster-Mode Administration Reference* for details about the `volume create` command.

Example

The following example creates a volume named dept_eng_ls_mirror1 that will be the load-sharing mirror of a source volume that is 2 GB in size. The volume is located on the SVM named vs0, on the aggregate named aggr3.

```
cluster1::> volume create -vserver vs0 -volume dept_eng_ls_mirror1
-aggregate aggr3 -size 2GB -type DP
```

2. Repeat the previous step for each load-sharing mirror that you want.

Example

The following example creates two more destination volumes that will be used as load-sharing mirrors.

```
cluster1::> volume create -vserver vs0 -volume dept_eng_ls_mirror2
-aggregate aggr4 -size 2GB -type DP

cluster1::> volume create -vserver vs0 -volume dept_eng_dr_mirror3
-aggregate aggr5 -size 2GB -type DP
```

Creating load-sharing mirror relationships

Before you can replicate data from the source FlexVol volume to the load-sharing mirror destination volumes, you must create the mirror relationships by using the `snapmirror create` command.

Steps

1. Use the `snapmirror create` command with the `-type LS` parameter to create a load-sharing mirror relationship between the source endpoint and a destination endpoint.

Example

The following command creates a load-sharing mirror of the source volume named dept_eng. The source volume and the mirrors are located on the Storage Virtual Machine (SVM) named vs0.

```
cluster1::> snapmirror create -source-path //vs0/dept_eng
-destination-path //vs0/dept_eng_ls_mirror1 -type LS
[Job 171] Job is queued: snapmirror create the relationship with
destination //v
[Job 171] Job succeeded: SnapMirror: done
```

When you create a relationship for a load-sharing mirror, the attributes for that load-sharing mirror (throttles, update schedules, and so on) are shared by all of the load-sharing mirrors that share the same source volume.

2. Repeat Step 1 to add load-sharing mirror relationships to the source volume, up to the maximum number allowed for a load-sharing mirror fanout, which is 24 for NAS-enabled SVMs.

Example

The following command creates load-sharing mirror relationships between the source volume named dept_eng and the destination volumes dept_eng_ls_mirror2 and dept_eng_ls_mirror3.

```
cluster1::> snapmirror create -source-path //vs0/dept_eng
-destination-path //vs0/dept_eng_ls_mirror2 -type LS
[Job 172] Job is queued: snapmirror create the relationship with
destination //v
[Job 172] Job succeeded: SnapMirror: done

cluster1::> snapmirror create -source-path //vs0/dept_eng
-destination-path //vs0/dept_eng_ls_mirror3 -type LS
[Job 173] Job is queued: snapmirror create the relationship with
destination //v
[Job 173] Job succeeded: SnapMirror: done
```

Creating a baseline for a set of load-sharing mirrors

You initialize the set of load-sharing mirrors to create a baseline of the source FlexVol volume to the load-sharing mirror destination volumes.

Step

1. Use the `snapmirror initialize-ls-set` command to initialize all of the load-sharing mirrors in the set.

   Note: Do not use the `snapmirror initialize` command to initialize a set of load-sharing mirrors. The `snapmirror initialize` command is for initializing individual volumes.

Example

The following example creates a baseline copy of the source volume named dept_eng to all of the load-sharing mirrors created for the source volume. The source volume is located on the Storage Virtual Machine (SVM) named vs0.

```
cluster1::> snapmirror initialize-ls-set //vs0/dept_eng
[Job 174] Job is queued: snapmirror load-share initialize for source //vs1/
dept_
eng.
```

Adding a load-sharing mirror to a set of load-sharing mirrors

You might want to add a load-sharing mirror to a set of load-sharing mirrors if the current number of load-sharing mirrors in that set continually has a large number of requests for data.

About this task

When you add a load-sharing mirror to a load-sharing set, you create the volume that will be the load-sharing mirror, you create the SnapMirror relationship between the source volume and the new

load-sharing mirror, and then you initialize the load sharing mirror to the same Snapshot copy of data that every other load sharing mirror in the set has.

Steps

1. Initializing an individual load-sharing mirror on page 105

Initializing an individual load-sharing mirror

You add a load-sharing mirror to the load-sharing set by initializing the load-sharing mirror. Clients cannot access the added load-sharing mirror until you initialize it.

Before you begin

You must have done the following tasks before you can initialize a load-sharing mirror and add it to the set of load-sharing mirrors:

- Have a set of load-sharing mirrors to which you want to add. That is, you have already created the set of load-sharing mirrors by creating destination volumes, SnapMirror relationships, and initializing those relationships using the `snapmirror initialize-ls-set` command.
- Created the destination volume of the load-sharing mirror you want to add by specifying the DP type option of the `volume create` command.

About this task

Initializing a load-sharing mirror creates a baseline copy of the source volume's data that is equivalent to the most up-to-date copies on the other load-sharing mirrors in the set.

Step

1. Use the `snapmirror initialize` command with the -type LS option to initialize the volume that you are adding to the load-sharing set.

   Note: Do not use the `snapmirror initialize-ls-set` command. The `snapmirror initialize-ls-set` command is for initializing volumes for an entire set of load-sharing mirrors, not for initializing an individual volume.

Example

The following example adds a load-sharing mirror named dept_eng_ls_mirror4 to a set of load-sharing mirrors of a source volume named dept_eng. The source volume and load-sharing mirrors are on the Storage Virtual Machine (SVM) named vs0.

```
cluster1::> snapmirror initialize -source-path node://vs0/dept_eng
-destination-path //vs0/dept_eng_ls_mirror4 -type LS
[Job 187] Job is queued: snapmirror initialize of destination //vs0/
dept_eng_ls_
mirror4.
```

After this initial copy, Data ONTAP sees the new load-sharing mirror as part of the set of load-sharing mirrors and will update the set when a scheduled update or manual update of the set occurs.

Updating a set of load-sharing mirrors

You can update a set of load-sharing mirrors if you think an update is necessary before the next scheduled update.

Step

1. Use the `snapmirror update-ls-set` command to update all of the load-sharing mirrors in the set.

Example

The following example updates all of the load-sharing mirrors created for the source volume named dept_eng on the Storage Virtual Machine (SVM) named vs0.

```
cluster1::> snapmirror update-ls-set -source-path //vs0/dept_eng
[Job 193] Job is queued: snapmirror load-share update for source
cluster1://vs0/dept_
eng.
```

Aborting an update to a load-sharing mirror

You can end an update to a set of load-sharing mirrors if the update started, but did not finish.

About this task

Load-sharing mirrors are either up to date and serving data to clients, or they are lagging and not serving data to clients.

- If you are aborting an update to an up-to-date load-sharing mirror, transfers to associated up-to-date load-sharing mirrors in the set of load-sharing mirrors are also aborted.
- If you are aborting an update to a lagging load-sharing mirror, then only the SnapMirror transfer associated with the lagging load-sharing mirror is aborted.

Step

1. Use the `snapmirror abort` command to end an update to a particular load-sharing mirror.

   Note: You can use the `snapmirror abort` command on other load-sharing mirrors in the set if those load-sharing mirrors also prevent updates to other load-sharing mirrors in the set.

Example

The following example ends the update to the load-sharing mirror named dept_eng_ls_mirror2 on a Storage Virtual Machine (SVM) named vs0.

```
cluster1::> snapmirror abort -source-path //vs0/dept_eng
-destination-path //vs0/dept_eng_ls_mirror2
[Job 184] Job is queued: snapmirror abort for the relationship with
destination
[Job 184] Job succeeded: SnapMirror: done
```

Scheduling SnapMirror transfers

If you want scheduled SnapMirror transfers, you can add a schedule to a mirror relationship after you initially create the relationship.

About this task

Unless you create and implement a schedule for SnapMirror transfers, you are limited to manually updating destination FlexVol volumes or Infinite Volumes with mirror relationships. The following are characteristics of adding a SnapMirror transfer schedule:

- When you add a schedule for a data protection mirror copy of Infinite Volumes, do not schedule updates for less than one-hour intervals.

  If you schedule updates for less than one-hour intervals, Data ONTAP tries but cannot meet the schedule for Infinite Volumes, and the data protection mirror relationship is displayed as unhealthy.
- Scheduled SnapMirror transfers (or even manual updates) can disrupt Snapshot copy schedules when a transfer lasts longer than the full retention period of the schedule.

Steps

1. Create the schedule you want to implement by using the `job schedule cron create` command.

   Note: You cannot use the `job schedule interval create` command to schedule SnapMirror transfers.

2. Apply the schedule to the mirror relationship by using the `-schedule` option of the `snapmirror modify` command.

   See the `snapmirror modify` command man page for more information about the command.

Changing mirror relationship schedules

You can change a schedule that updates mirror relationships for FlexVol volumes and Infinite Volumes if the schedule impacts other backups or updates.

Before you begin

- You must have created the cluster and Storage Virtual Machine (SVM) peering relationship.
  To know about creating cluster and SVM peering, see the *Clustered Data ONTAP System Administration Guide for Cluster Administrators*.

About this task

Changing a schedule affects load-sharing mirror copies differently than it does for data-protection mirror copies. If you change a schedule to a load-sharing mirror relationship, Data ONTAP makes the change to the relationships of all the load-sharing mirror copies in the group. Data ONTAP determines the load-sharing mirror group by the SVM and source volume specified by the command. See the *Clustered Data ONTAP Logical Storage Management Guide* for more information about load-sharing mirror copies.

Steps

1. Create the new schedule by using the `job schedule cron create` command.

   Creating a schedule is described in the cron job creation section of the *Clustered Data ONTAP System Administration Guide for Cluster Administrators*. See the man page for details about the `job schedule cron create` command.

2. Change the schedule for a mirror relationship by using the `snapmirror modify -schedule` command.

   This command must be used from the destination SVM.

Example

The following command changes the update schedule used by a data protection mirror relationship for destination volume named “dept_eng_ls1” to a schedule named “dept_eng_mirror_sched”:

```
vs2::> snapmirror modify -source-path vs1:dept_eng
-destination-path vs2:dept_eng_ls1 -schedule dept_eng_mirror_sched
```

Listing the schedule state of a mirror relationship

You might want to see what state a scheduled transfer for a mirror relationship is in to ensure that jobs are running as they should.

About this task

The state of a scheduled job might be dormant. The dormant state means that the job is waiting for the scheduled start time to begin the transfer. There is nothing wrong with the job and you do not need to do anything.

Step

1. To see the state of scheduled jobs, use the `snapmirror show` command.

Showing the status of a particular load-sharing mirror

You can show the status of a particular load-sharing mirror if you are uncertain of its state. For example, you might want to check the status of a load-sharing mirror if you are synchronizing it with other load-sharing mirrors in a set and you want to know whether Data ONTAP is done synchronizing.

Step

1. Use the `snapmirror show` command to show the status of a particular load-sharing mirror.

Example

The following example shows the relationship for the load-sharing mirror named dept_eng_ls_mirror2.

```
cluster1::> snapmirror show -fields status -destination-volume
dept_eng_ls_mirror2

source-path          destination-path                    status
-------------------- ----------------------------------- ------
clus1://vs1/dept_eng clus1://vs1/dept_eng_ls_mirror2     Idle
```

Note: You can use one or more `snapmirror show` command options to show specific information about a load-sharing mirror relationship. See the man page for more information.

Determining whether load-sharing mirrors are up-to-date

You can determine whether load-sharing mirrors are up to date by using the `snapmirror show` command and looking at the exported-snapshot field. You might do this to ensure that all of the load-sharing mirrors in the set get updated after an update.

About this task

If a load-sharing mirror is lagging behind the most up-to-date load-sharing mirror in the set, the exported-snapshot field shows a dash (-).

Step

1. Use the `snapmirror show` command with the `-fields` option to list the status of load-sharing mirrors for a particular source volume.

Example

The following example lists the load-sharing mirrors for the source volume named dept_eng and shows the load-sharing mirrors that are lagging behind the up-to-date mirrors (identified by the dash in the exported-snapshot field).

```
cluster1::> snapmirror show -fields type,exported-snapshot
-S clus1://vs1/dept_eng
source-path           destination-path          type exported-snapshot
--------------------- ------------------------- ---- -----------------
clust1://vs1/dept_eng clust1://vs1/dept_eng_ls1 LS   -
clust1://vs1/dept_eng clust1://vs1/dept_eng_ls2 LS   -
clust1://vs1/dept_eng clust1://vs1/dept_eng_ls3 LS   -
clust1://vs1/dept_eng clust1://vs1/dept_eng_ls4 LS   snapmirror.
5_2147484688.2010-04-16_173522
clust1://vs1/dept_eng clust1://vs1/dept_eng_ls5 LS   -
clust1://vs1/dept_eng clust1://vs1/dept_eng_ls6 LS   snapmirror.
5_2147484688.2010-04-16_173522
clust1://vs1/dept_eng clust1://vs1/dept_eng_ls7 LS   -
7 entries were displayed.
```

Note: You can use one or more `snapmirror show` command options to list more specific information about a set of load-sharing mirrors. See the `snapmirror show` command in the *Clustered Data ONTAP Data Protection Guide* for more information.

Recovering a lost source FlexVol volume from a mirror

You might be able to use a load-sharing mirror to recover a source volume that is not accessible. This might be a good solution if you lose a source volume because a component failed, such as a shelf or port failure.

About this task

- This is not a procedure you use for recovering from a site disaster, such as a fire or flood, because both source volume and destination volume of the mirror are on the same cluster and same Storage Virtual Machine (SVM).
  A site disaster would affect both sides of the mirror because of this configuration restriction. For a failure the magnitude of a site failure, you should consider using a cross-cluster disaster recovery solution.
- You should not perform the following procedure if you are backing up data to tape.
  If you attempt to replace the source read-write volume with a mirrored volume using this procedure, the attempt will fail because the tape backup locks the Snapshot copy.
- If you created a junction path to a data protection destination volume and you promote the destination using the `snapmirror promote` command, the junction path that you created is deleted.

Step

1. Use the `snapmirror promote` command to make a mirror a read-write volume that replaces the source read-write volume.

   #### Example

   The following example recovers a lost source volume from the load-sharing mirror named dept_eng_ls_mirror3 on an SVM named vs0 and a cluster named cluster1:

   ```
   cluster1::> snapmirror promote -destination-path
   cluster1://vs0/dept_eng_ls_mirror3 -source-path
   cluster1://vs0/eng
   ```

   Data ONTAP makes the destination volume of the mirror a read-write volume, destroys the original read-write volume if it is accessible, and redirects mirrors and clients that accessed the original read-write volume to the new read-write volume.

   Note: The recovered source volume might not have all of the data that the original source volume had because the SnapMirror relationship for load-sharing and data protection mirrors is a scheduled, asynchronous update and the update might not have occurred recently.

Deleting a mirror copy

You can delete a mirror relationship and the destination FlexVol volume or Infinite Volume if you no longer want the mirror copy.

About this task

When you delete a mirror copy, you must delete the mirror relationship and the destination volume. Deleting the mirror relationship does not delete SnapMirror created Snapshot copies on either the source or destination volumes. Deleting the mirror relationship attempts to delete Snapshot copy owners for the SnapMirror created Snapshot copies on both source and destination volumes.

When you delete a load-sharing mirror copy from a set of load-sharing mirror copies, the destination volume of the deleted load-sharing mirror relationship cannot be used again as a destination volume of a load-sharing relationship if it contains any data or Snapshot copies.

Steps

1. Optional: On the source Storage Virtual Machine (SVM), use the `snapmirror list-destination` command to view the list of destination volumes for that source volume.

   Example

   ```
   vs1::> snapmirror list-destinations
                                                          Progress
   Source             Destination         Transfer   Last         Relationship
   Path         Type  Path         Status Progress   Updated      Id
   ----------- ----- ------------ ------- --------- ------------ ---------------
   vs1:src_ui  DP    vs2:vsrc_ui_ls_mir2
                                 Idle    -         -            3672728c-
   ad06-11e2-981e-123478563412
   ```

2. Use the `snapmirror delete` command to delete a mirror relationship.

   Example

   The following command deletes a mirror relationship between a destination volume named src_ui_ls_mir2 and a source volume named src_ui:

   ```
   vs2::> snapmirror delete -source-path vs1:src_ui
   -destination-path vs2:src_ui_ls_mir2
   ```

   The command deletes the mirror relationship, but does not delete the destination volume. In the case of a load-sharing mirror copy, the destination volume will be in the restricted state. If you want to use it as a destination volume of a data protection relationship, you must wait at least 10 minutes. This is the amount of time required to refresh internal caches and place the volume back online.

3. Use the `snapmirror release` command from the source SVM to remove the configuration information and Data ONTAP created Snapshot copies on the source volume.

   Example

   The following command removes the DP or XDP relationship from the source SVM named vs1:

   ```
   vs1::> snapmirror release -source-path vs1:src_ui -destination-path
   vs2:src_ui_ls_mir2
   ```

   This command removes the DP or XDP relationship information from the source SVM and does not delete any volumes. This command deletes the base Snapshot copies for the destination volume named src_ui_ls_mir2 from the source volume named src_ui.

4. Optional: Use the `volume delete` command to delete the destination volume.

   Delete the destination volume if you no longer need the volume.

Using qtrees to partition your FlexVol volumes

Qtrees enable you to partition your FlexVol volumes into smaller segments that you can manage individually. You can use qtrees to manage quotas, security style, and CIFS oplocks.

Data ONTAP creates a default qtree, called *qtree0*, for each volume. If you do not put data into a qtree, it resides in qtree0.

Qtree names must have no more than 64 characters.

Directories cannot be moved between qtrees. Only files can be moved between qtrees.

When to use qtrees

Qtrees enable you to partition your data without incurring the overhead associated with a FlexVol volume. You might create qtrees to organize your data, or to manage one or more of the following factors: quotas, security style, and CIFS oplocks setting.

The following list describes examples of qtree usage strategies:

- Quotas
  You can limit the size of the data used by a particular project, by placing all of that project's files into a qtree and applying a tree quota to the qtree.
- Security style
  If you have a project that needs to use NTFS-style security, because the members of the project use Windows files and applications, you can group the data for that project in a qtree and set its security style to NTFS, without requiring that other projects also use the same security style.
- CIFS oplocks settings
  If you have a project using a database that requires CIFS oplocks to be off, you can set CIFS oplocks to `off` for that project's qtree, while allowing other projects to retain CIFS oplocks.

How qtrees compare with FlexVol volumes

In general, qtrees are similar to FlexVol volumes. However, the two technologies have some key differences. Understanding these differences helps you choose between them when you design your storage architecture.

The following table compares qtrees and FlexVol volumes:

Functionality	Qtree	FlexVol volume
Enables organizing user data	Yes	Yes
Enables grouping users with similar needs	Yes	Yes
Accepts a security style	Yes	Yes
Accepts oplocks configuration	Yes	Yes
Can be resized	Yes (using quota limits)	Yes
Supports Snapshot copies	No (qtree data can be extracted from volume Snapshot copies)	Yes
Supports quotas	Yes	Yes
Can be cloned	No (except as part of a FlexVol volume)	Yes
Can serve as the root of a Storage Virtual Machine (SVM)	No	Yes
Can serve as a junction	No	Yes
Can be exported using NFS	Yes	Yes

Qtree name restrictions

Qtree names can be no more than 64 characters in length. In addition, using some special characters in qtree names, such as commas and spaces, can cause problems with other Data ONTAP capabilities, and should be avoided.

What you can do with qtrees on a mirror

You can see but not modify qtrees that exist within a mirror.

For example, you can use the `volume qtree statistics` command on the mirror. Note that information displayed about the qtrees (including name, security style, oplock mode, and other attributes) may not be synchronized between the read-write volume and the mirror, depending on the

mirror's replication schedule. But after the read-write volume is replicated to the mirror, qtree information is synchronized.

However, you cannot create, modify, or delete the qtrees on the mirror.

Converting a directory to a qtree

If you have a directory at the root of a FlexVol volume that you want to convert to a qtree, you must migrate the data contained in the directory to a new qtree with the same name, using your client application.

About this task

The steps you take to convert a directory to a qtree depend on what client you use. The following process outlines the general tasks you need to complete:

Steps

1. Rename the directory to be made into a qtree.
2. Create a new qtree with the original directory name.
3. Use the client application to move the contents of the directory into the new qtree.
4. Delete the now-empty directory.

   Note: You cannot delete a directory if it is associated with an existing CIFS share.

Converting a directory to a qtree using a Windows client

To convert a directory to a qtree using a Windows client, you rename the directory, create a qtree on the storage system, and move the directory's contents to the qtree.

About this task

You must use Windows Explorer for this procedure. You cannot use the Windows command-line interface or the DOS prompt environment.

Steps

1. Open Windows Explorer.
2. Click the folder representation of the directory you want to change.

   Note: The directory must reside at the root of its containing volume.
3. From the **File** menu, select **Rename** to give this directory a different name.
4. On the storage system, use the `volume qtree create` command to create a new qtree with the original name of the directory.
5. In Windows Explorer, open the renamed directory folder and select the files inside it.

6. Drag these files into the folder representation of the new qtree.

   Note: The more subfolders contained in the folder that you are moving, the longer the move operation takes.

7. From the **File** menu, select **Delete** to delete the renamed, now-empty directory folder.

Converting a directory to a qtree using a UNIX client

To convert a directory to a qtree in UNIX, you rename the directory, create a qtree on the storage system, and move the directory's contents to the qtree.

Steps

1. Open a UNIX client window.
2. Use the `mv` command to rename the directory.

   Example

   ```
   client: mv /n/user1/vol1/dir1 /n/user1/vol1/olddir
   ```

3. From the storage system, use the `volume qtree create` command to create a qtree with the original name.

   Example

   ```
   system1: volume qtree create /n/user1/vol1/dir1
   ```

4. From the client, use the `mv` command to move the contents of the old directory into the qtree.

   Note: The more subdirectories contained in a directory that you are moving, the longer the move operation will take.

   Example

   ```
   client: mv /n/user1/vol1/olddir/* /n/user1/vol1/dir1
   ```

5. Use the `rmdir` command to delete the old, now-empty directory.

   Example

   ```
   client: rmdir /n/user1/vol1/olddir
   ```

After you finish

Depending on how your UNIX client implements the `mv` command, file ownership and permissions might not be preserved. If this occurs, update file owners and permissions to their previous values.

Commands for managing qtrees

There are specific Data ONTAP commands for managing and configuring qtrees.

Many qtree commands cannot be performed while a volume move operation is in progress. If you are prevented from completing a qtree command for this reason, wait until the volume move is complete and then retry the command.

If you want to...	Use this command...
Create a qtree	`volume qtree create`
Display a filtered list of qtrees	`volume qtree show`
Delete a qtree	`volume qtree delete`
Modify a qtree's UNIX permissions	`volume qtree modify -unix-permissions`
Modify a qtree's CIFS oplocks setting	`volume qtree oplocks`
Modify a qtree's security setting	`volume qtree security`
Rename a qtree	`volume qtree rename`
Display a qtree's statistics	`volume qtree statistics`
Reset a qtree's statistics	`volume qtree statistics -reset`

Using quotas to restrict or track resource usage

Quotas provide a way to restrict or track the disk space and number of files used by a user, group, or qtree. Quotas are applied to a specific FlexVol volume or qtree.

Why you use quotas

You can use quotas to limit resource usage in FlexVol volumes, to provide notification when resource usage reaches specific levels, or to track resource usage.

You specify a quota for the following reasons:

- To limit the amount of disk space or the number of files that can be used by a user or group, or that can be contained by a qtree
- To track the amount of disk space or the number of files used by a user, group, or qtree, without imposing a limit

- To warn users when their disk usage or file usage is high

Related concepts

Examples of quota configuration on page 147

Related tasks

Setting up quotas on an SVM with FlexVol volumes on page 152

Overview of the quota process

Quotas can be soft or hard. Soft quotas cause Data ONTAP to send a notification when specified thresholds are exceeded, and hard quotas prevent a write operation from succeeding when specified thresholds are exceeded.

When Data ONTAP receives a request to write to a FlexVol volume, it checks to see whether quotas are activated for that volume. If so, Data ONTAP determines whether any quota for that volume (and, if the write is to a qtree, for that qtree) would be exceeded by performing the write operation. If any hard quota is exceeded, the write operation fails, and a quota notification is sent. If any soft quota is exceeded, the write operation succeeds, and a quota notification is sent.

Related concepts

How quotas are applied on page 126

Differences among hard, soft, and threshold quotas

Hard quotas prevent operations while soft quotas trigger notifications.

Hard quotas impose a hard limit on system resources; any operation that would result in exceeding the limit fails. The following settings create hard quotas:

- Disk Limit parameter
- Files Limit parameter

Soft quotas send a warning message when resource usage reaches a certain level, but do not affect data access operations, so you can take appropriate action before the quota is exceeded. The following settings create soft quotas:

- Threshold for Disk Limit parameter
- Soft Disk Limit parameter
- Soft Files Limit parameter

Threshold and Soft Disk quotas enable administrators to receive more than one notification about a quota. Typically, administrators set the Threshold for Disk Limit to a value that is only slightly smaller than the Disk Limit, so that the threshold provides a "final warning" before writes start to fail.

Understanding quota notifications

Quota notifications are messages that are sent to the event management system (EMS) and also configured as SNMP traps.

Notifications are sent in response to the following events:

- A hard quota is reached; in other words, an attempt is made to exceed it
- A soft quota is exceeded
- A soft quota is no longer exceeded

Thresholds are slightly different from other soft quotas. Thresholds trigger notifications only when they are exceeded, not when they are no longer exceeded.

Hard-quota notifications are configurable by using the `volume quota modify` command. You can turn them off completely, and you can change their frequency, for example, to prevent sending of redundant messages.

Soft-quota notifications are not configurable because they are unlikely to generate redundant messages and their sole purpose is notification.

The following table lists the events that quotas send to the EMS system:

When this occurs...	This event is sent to the EMS...
A hard limit is reached in a tree quota	`wafl.quota.qtree.exceeded`
A hard limit is reached in a user quota on the volume	`wafl.quota.user.exceeded` (for a UNIX user) `wafl.quota.user.exceeded.win` (for a Windows user)
A hard limit is reached in a user quota on a qtree	`wafl.quota.userQtree.exceeded` (for a UNIX user) `wafl.quota.userQtree.exceeded.win` (for a Windows user)
A hard limit is reached in a group quota on the volume	`wafl.quota.group.exceeded`
A hard limit is reached in a group quota on a qtree	`wafl.quota.groupQtree.exceeded`
A soft limit, including a threshold, is exceeded	`quota.softlimit.exceeded`
A soft limit is no longer exceeded	`quota.softlimit.normal`

The following table lists the SNMP traps that quotas generate:

When this occurs...	This SNMP trap is sent...
A hard limit is reached	quotaExceeded
A soft limit, including a threshold, is exceeded	quotaExceeded and softQuotaExceeded
A soft limit is no longer exceeded	quotaNormal and softQuotaNormal

For more information about viewing and managing events and SNMP traps, see the *Clustered Data ONTAP System Administration Guide for Cluster Administrators*.

Note: Notifications contain qtree ID numbers rather than qtree names. You can correlate qtree names to ID numbers by using the `volume qtree show -id` command.

What quota rules, quota policies, and quotas are

Quotas are defined in quota rules specific to FlexVol volumes. These quota rules are collected together in a quota policy of a Storage Virtual Machine (SVM), and then activated on each volume on the SVM.

A quota rule is always specific to a volume. Quota rules have no effect until quotas are activated on the volume defined in the quota rule.

A quota policy is a collection of quota rules for all the volumes of an SVM. Quota policies are not shared among SVMs. An SVM can have up to five quota policies, which enable you to have backup copies of quota policies. One quota policy is assigned to an SVM at any given time.

A quota is the actual restriction that Data ONTAP enforces or the actual tracking that Data ONTAP performs. A quota rule always results in at least one quota, and might result in many additional derived quotas. The complete list of enforced quotas is visible only in quota reports.

Activation is the process of triggering Data ONTAP to create enforced quotas from the current set of quota rules in the assigned quota policy. Activation occurs on a volume-by-volume basis. The first activation of quotas on a volume is called initialization. Subsequent activations are called either reinitialization or resizing, depending on the scope of the changes.

Note: When you initialize or resize quotas on a volume, you are activating the quota rules in the quota policy that is currently assigned to the SVM.

Quota targets and types

Quotas have a type: they can be either user, group, or tree. Quota targets specify the user, group, or qtree for which the quota limits are applied.

The following table lists the kinds of quota targets, what types of quotas each quota target is associated with, and how each quota target is represented:

Quota target	Quota type	How target is represented	Notes
user	user quota	UNIX user name UNIX UID A file or directory whose UID matches the user Windows user name in pre-Windows 2000 format Windows SID A file or directory with an ACL owned by the user's SID	User quotas can be applied for a specific volume or qtree.
group	group quota	UNIX group name UNIX GID A file or directory whose GID matches the group	Group quotas can be applied for a specific volume or qtree. **Note:** Data ONTAP does not apply group quotas based on Windows IDs.
qtree	tree quota	qtree name	Tree quotas are applied to a particular volume and do not affect qtrees in other volumes.
*	user quota group quota tree quota	The asterisk character (*)	A quota target of * denotes a *default quota*. For default quotas, the quota type is determined by the value of the type field.

Related concepts

How quotas work with users and groups on page 127
How quotas work with qtrees on page 132

Special kinds of quotas

You use default, explicit, derived and tracking quotas to manage disk usage in the most efficient manner.

How default quotas work

You can use default quotas to apply a quota to all instances of a given quota type. For example, a default user quota affects all users on the system for the specified FlexVol volume or qtree.. In addition, default quotas enable you to modify your quotas easily.

You can use default quotas to automatically apply a limit to a large set of quota targets without having to create separate quotas for each target. For example, if you want to limit most users to 10 GB of disk space, you can specify a default user quota of 10 GB of disk space instead of creating a quota for each user. If you have specific users for whom you want to apply a different limit, you can create explicit quotas for those users. (Explicit quotas—quotas with a specific target or list of targets —override default quotas.)

In addition, default quotas enable you to use resizing rather than reinitialization when you want quota changes to take effect. For example, if you add an explicit user quota to a volume that already has a default user quota, you can activate the new quota by resizing.

Default quotas can be applied to all three types of quota target (users, groups, and qtrees).

Default quotas do not necessarily have specified limits; a default quota can be a tracking quota.

A quota is indicated by a target that is either an empty string ("") or an asterisk (*), depending on the context:

- When you create a quota using the `volume quota policy rule create` command, setting the `-target` parameter to an empty string ("") creates a default quota.
- In the output of the `volume quota policy rule show` command, a default quota appears with an empty string ("") as the Target.
- In the output of the `volume quota report` command, a default quota appears with an asterisk (*) as the ID and Quota Specifier.

Default user quota example

The following command creates a default user quota to apply a 50-MB limit for each user in vol1:

```
volume quota policy rule create -vserver vs1 -policy-name
quota_policy_vs1_1 -volume vol1 -type user -target "" -disk-limit 50MB
-qtree ""
```

The `volume quota policy rule show` command displays the following output:

```
Vserver: vs1              Policy: quota_policy_vs1_1  Volume:
                                                          vol1
```

```
                                                  Soft              Soft
                            User          Disk    Disk    Files     Files
Type    Target    Qtree     Mapping      Limit   Limit   Limit     Limit  Threshold
-----   --------  -------   -------   --------  -------  ------  -------  ---------
user    ""        ""        off           50MB        -       -        -          -
```

If any user on the system performs an action that would cause that user's data to take up more than 50 MB in vol1 (for example, writing to a file from an editor), the command fails.

Related concepts

How derived quotas work on page 124

How you use explicit quotas

You can use explicit quotas to specify a quota for a specific quota target, or to override a default quota for a specific target.

An explicit quota specifies a limit for a particular user, group, or qtree. An explicit quota replaces any default quota that is in place for the same target.

When you add an explicit user quota for a user that has a derived user quota, you must use the same user mapping setting as the default user quota. Otherwise, when you resize quotas, the explicit user quota is rejected because it is considered a new quota.

Explicit quotas only affect default quotas at the same level (volume or qtree). For example, an explicit user quota for a qtree does not affect the default user quota for the volume that contains that qtree. However, the explicit user quota for the qtree overrides (replaces the limits defined by) the default user quota for that qtree.

Examples of explicit quotas

One user, chen, is allowed 80 MB of space on vol1 with the following command:

```
volume quota policy rule create -vserver vs1 -policy-name
quota_policy_vs1_1 -volume vol1 -type user -target corp\chen -disk-
limit 80MB -qtree ""
```

A group, eng1, is allowed 150 MB of disk space and an unlimited number of files in qtree proj1 on vol2 with the following command:

```
volume quota policy rule create -vserver vs1 -policy-name
quota_policy_vs1_1 -volume vol2 -type group -target eng1 -disk-limit
150MB -qtree proj1
```

The qtree, proj1, is allowed 750 MB of disk space and 76,800 files on the vol2 volume with the following command:

```
volume quota policy rule create -vserver vs1 -policy-name
quota_policy_vs1_1 -volume vol2 -type tree -target proj1 -disk-limit
750MB -file-limit 76800 -qtree ""
```

The `volume quota policy rule show` command displays the following output:

```
Vserver: vs1              Policy: quota_policy_vs1_1  Volume:
                                                            vol1

                                                 Soft            Soft
                          User         Disk      Disk   Files    Files
  Type   Target    Qtree  Mapping      Limit     Limit  Limit    Limit  Threshold
  -----  --------  ------ -------  --------  -------  ------  -------  ---------
  user   corp\chen      ""     off      80MB        -       -        -          -

                                                           vol2

                                                 Soft            Soft
                          User         Disk      Disk   Files    Files
  Type   Target    Qtree  Mapping      Limit     Limit  Limit    Limit  Threshold
  -----  --------  ------ -------  --------  -------  ------  -------  ---------
  group      eng1   proj1     off     150MB        -       -        -          -
  tree      proj1      ""     off     750MB        -   76800        -          -
```

How derived quotas work

A quota enforced as a result of a default quota, rather than an explicit quota (a quota with a specific target), is referred to as a *derived quota*.

The number and location of the derived quotas depends on the quota type:

- A default tree quota on a volume creates derived tree quotas for every qtree on the volume.
- A default user or group quota creates a derived user or group quota for every user or group that owns a file at the same level (volume or qtree).
- A default user or group quota on a volume creates a default user or group quota on every qtree that also has a tree quota.

The settings—including limits and user mapping—of derived quotas are the same as the settings of the corresponding default quotas. For example, a default tree quota with a 20-GB disk limit on a volume creates derived tree quotas with 20-GB disk limits on the qtrees in the volume. If a default quota is a tracking quota (with no limits), the derived quotas are also tracking quotas.

To see derived quotas, you can generate a quota report. In the report, a derived user or group quota is indicated by a Quota Specifier that is either blank or an asterisk (*). A derived tree quota, however, has a Quota Specifier; to identify a derived tree quota, you must look for a default tree quota on the volume with the same limits. Since they are not manually configured quota rules, derived quotas do not appear in the output of the `quota policy rule show` command.

Explicit quotas interact with derived quotas in the following ways:

- Derived quotas are not created if an explicit quota already exists for the same target.
- If a derived quota exists when you create an explicit quota for a target, you can activate the explicit quota by resizing rather than having to perform a full quota initialization.

Related concepts

How default quotas work on page 122

How default user and group quotas create derived quotas on page 128
How default tree quotas on a FlexVol volume create derived tree quotas on page 133
How default user quotas on a FlexVol volume affect quotas for the qtrees in that volume on page 134

How you use tracking quotas

Tracking quotas generate reports of disk and file usage and do not limit resource usage. When tracking quotas are used, modifying quota values is less disruptive, because you can resize quotas rather than turning them off and back on.

To create a tracking quota, you omit the Disk Limit and Files Limit parameters. This tells Data ONTAP to monitor disk and files usage for that target at that level (volume or qtree), without imposing any limits. Tracking quotas are indicated in the output of `show` commands and the quota report with a dash ("-") for all limits.

You can also specify a *default tracking quota*, which applies to all instances of the target. Default tracking quotas enable you to track usage for all instances of a quota type (for example, all qtrees or all users). In addition, they enable you use resizing rather than reinitialization when you want quota changes to take effect.

Examples of explicit tracking quotas

A user, chen, is tracked on vol1 with the following command:

```
volume quota policy rule create -vserver vs1 -policy-name
quota_policy_vs1_1 -volume vol1 -type user -target corp\chen -qtree ""
```

A group, eng1, is tracked on vol1 with the following command:

```
volume quota policy rule create -vserver vs1 -policy-name
quota_policy_vs1_1 -volume vol1 -type group -target eng1 -qtree ""
```

A qtree, proj1, is tracked on vol1 with the following command:

```
volume quota policy rule create -vserver vs1 -policy-name
quota_policy_vs1_1 -volume vol1 -type tree -target proj1 -qtree ""
```

The `volume quota policy rule show` command displays the following output:

```
Vserver: vs1              Policy: quota_policy_vs1_1  Volume:
                                                           vol1

                                          Soft             Soft
                          User         Disk     Disk   Files    Files
  Type   Target    Qtree  Mapping     Limit    Limit   Limit    Limit  Threshold
  -----  --------  ------ -------  --------  -------  ------  -------  ---------
  user  corp\chen      ""     off         -        -       -        -          -
  group      eng1      ""     off         -        -       -        -          -
  tree      proj1      ""     off         -        -       -        -          -
```

Examples of default tracking quotas

All users are tracked on vol1 with the following command:

```
volume quota policy rule create -vserver vs1 -policy-name
quota_policy_vs1_1 -volume vol1 -type user -target "" -qtree ""
```

All groups are tracked on vol1 with the following command:

```
volume quota policy rule create -vserver vs1 -policy-name
quota_policy_vs1_1 -volume vol1 -type group -target "" -qtree ""
```

All qtrees on vol1 are tracked with the following command:

```
volume quota policy rule create -vserver vs1 -policy-name
quota_policy_vs1_1 -volume vol1 -type tree -target "" -qtree ""
```

The `volume quota policy rule show` command displays the following output:

```
Vserver: vs1              Policy: quota_policy_vs1_1  Volume:
                                                          vol1

                                             Soft            Soft
                          User         Disk    Disk   Files    Files
  Type   Target    Qtree  Mapping     Limit   Limit   Limit    Limit  Threshold
  -----  --------  ------ -------  --------  ------- ------  -------  ---------
  user         ""      ""     off         -       -       -        -          -
  group        ""      ""     off         -       -       -        -          -
  tree         ""      ""     off         -       -       -        -          -
```

How quotas are applied

Understanding how quotas are applied enables you to configure quotas and set the expected limits.

Whenever an attempt is made to create a file or write data to a file in a FlexVol volume that has quotas enabled, the quota limits are checked before the operation proceeds. If the operation exceeds either the disk limit or the files limit, the operation is prevented.

Quota limits are checked in the following order:

1. The tree quota for that qtree (This check is not relevant if the file is being created or written to qtree0.)
2. The user quota for the user that owns the file on the volume
3. The group quota for the group that owns the file on the volume
4. The user quota for the user that owns the file on the qtree (This check is not relevant if the file is being created or written to qtree0.)
5. The group quota for the group that owns the file on the qtree (This check is not relevant if the file is being created or written to qtree0.)

The quota with the smallest limit might not be the one that is exceeded first. For example, if a user quota for volume vol1 is 100 GB, and the user quota for qtree q2 contained in volume vol1 is 20 GB,

the volume limit could be reached first if that user has already written more than 80 GB of data in volume vol1 (but outside of qtree q2).

Considerations for assigning quota policies

A quota policy is a grouping of the quota rules for all the FlexVol volumes of a Storage Virtual Machine (SVM). You must be aware of certain considerations when assigning the quota policies.

- An SVM has one assigned quota policy at any given time. When an SVM is created, a blank quota policy is created and assigned to the SVM. This default quota policy has the name "default" unless a different name is specified when the SVM is created.
- An SVM can have up to five quota policies. If an SVM has five quota policies, you cannot create a new quota policy for the SVM until you delete an existing quota policy.
- When you need to create a quota rule or change quota rules for a quota policy, you can choose either of the following approaches:
  - If you are working in a quota policy that is assigned to an SVM, then you need not assign the quota policy to the SVM.
  - If you are working in an unassigned quota policy and then assigning the quota policy to the SVM, then you must have a backup of the quota policy that you can revert to if required.

    For example, you can make a copy of the assigned quota policy, change the copy, assign the copy to the SVM, and rename the original quota policy.
- You can rename a quota policy even when it is assigned to the SVM.

How quotas work with users and groups

When you specify a user or group as the target of a quota, the limits imposed by that quota are applied to that user or group. However, some special groups and users are handled differently. There are different ways to specify IDs for users, depending on your environment.

Related concepts

How user and group quotas work with qtrees on page 133

How you specify UNIX users for quotas

You can specify a UNIX user for a quota using one of three formats: the user name, the UID, or a file or directory owned by the user.

To specify a UNIX user for a quota, you can use one of the following formats:

- The user name, such as jsmith.

  Note: You cannot use a UNIX user name to specify a quota if that name includes a backslash (\) or an @ sign. This is because Data ONTAP treats names containing these characters as Windows names.
- The UID, such as 20.
- The path of a file or directory owned by that user, so that the file's UID matches the user.

Note: If you specify a file or directory name, you must select a file or directory that will last as long as the user account remains on the system.

Specifying a file or directory name for the UID does not cause Data ONTAP to apply a quota to that file or directory.

How you specify Windows users for quotas

You can specify a Windows user for a quota using one of three formats: the Windows name in pre-Windows 2000 format, the SID, or a file or directory owned by the SID of the user.

To specify a Windows user for a quota, you can use one of the following formats:

- The Windows name in pre-Windows 2000 format. Include the domain in NetBIOS form, for example, corp\Bob. If the name contains a space, enclose the value of the quota target in quotes, for example "corp\John Smith".
- The security ID (SID), as displayed by Windows in text form, such as S-1-5-32-544.
- The name of a file or directory that has an ACL owned by that user's SID.

  Note: If you specify a file or directory name, you must select a file or directory that will last as long as the user account remains on the system.

  For Data ONTAP to obtain the SID from the ACL, the ACL must be valid.

  If the file or directory exists in a UNIX-style qtree, or if the storage system uses UNIX mode for user authentication, Data ONTAP applies the user quota to the user whose *UID*, not SID, matches that of the file or directory.

  Specifying a file or directory name to identify a user for a quota does not cause Data ONTAP to apply a quota to that file or directory.

How default user and group quotas create derived quotas

When you create default user or group quotas, corresponding derived user or group quotas are automatically created for every user or group that owns files at the same level.

Derived user and group quotas are created in the following ways:

- A default user quota on a FlexVol volume creates derived user quotas for every user that owns a file anywhere on the volume.
- A default user quota on a qtree creates derived user quotas for every user that owns a file in the qtree.
- A default group quota on a FlexVol volume creates derived group quotas for every group that owns a file anywhere on the volume.
- A default group quota on a qtree creates derived group quotas for every group that owns a file in the qtree.

If a user or group does not own files at the level of a default user or group quota, derived quotas are not created for the user or group. For example, if a default user quota is created for qtree proj1 and the user jsmith owns files on a different qtree, no derived user quota is created for jsmith.

The derived quotas have the same settings as the default quotas, including limits and user mapping. For example, if a default user quota has a 50-MB disk limit and has user mapping turned on, any resulting derived quotas also have a 50-MB disk limit and user mapping turned on.

However, no limits exist in derived quotas for three special users and groups. If the following users and groups own files at the level of a default user or group quota, a derived quota is created with the same user-mapping setting as the default user or group quota, but it is only a tracking quota (with no limits):

- UNIX root user (UID 0)
- UNIX root group (GID 0)
- Windows BUILTIN\Administrators group

  Since quotas for Windows groups are tracked as user quotas, a derived quota for this group is a user quota that is derived from a default user quota, not a default group quota.

Example of derived user quotas

If you have volume where three users—root, jsmith, and bob—own files, and you create a default user quota on the volume, Data ONTAP automatically creates three derived user quotas. Therefore, after you reinitialize quotas on the volume, four new quotas appear in the quota report:

```
cluster1::> volume quota report
  Vserver: vs1
                                    ----Disk----  ----Files-----   Quota
  Volume   Tree      Type    ID        Used  Limit    Used   Limit   Specifier
  -------  --------  ------  -------  -----  -----  ------  ------   ---------
  vol1               user    *          0B   50MB       0       -   *
  vol1               user    root       5B      -       1       -
  vol1               user    jsmith    30B   50MB      10       -   *
  vol1               user    bob       40B   50MB      15       -   *
  4 entries were displayed.
```

The first new line is the default user quota that you created, which is identifiable by the asterisk (*) as the ID. The other new lines are the derived user quotas. The derived quotas for jsmith and bob have the same 50-MB disk limit as the default quota. The derived quota for the root user is a tracking quota without limits.

Related concepts

How derived quotas work on page 124

How default user quotas on a FlexVol volume affect quotas for the qtrees in that volume on page 134

How quotas are applied to the root user

The root user (UID=0) on UNIX clients is subject to tree quotas, but not user quotas or group quotas. This allows the root user to take actions on behalf of other users that would otherwise be prevented by a quota.

When root carries out a file or directory ownership change or other operation (such as the UNIX `chown` command) on behalf of a user with less privileges, Data ONTAP checks the quotas based on the new owner but does not report errors or stop the operation, even if the hard quota restrictions of the new owner are exceeded. This can be useful when an administrative action, such as recovering lost data, results in temporarily exceeding quotas.

Note: After the ownership transfer is carried out, however, a client system will report a disk space error if the user attempts to allocate more disk space while the quota is still exceeded.

How quotas work with special Windows groups

Quotas are applied to the Everyone group and the BUILTIN\Administrators group differently than to other Windows groups.

The following list describes what happens if the quota target is a special Windows group ID:

- If the quota target is the Everyone group, a file whose ACL shows that the owner is Everyone is counted under the SID for Everyone.
- If the quota target is BUILTIN\Administrators, the entry is considered a user quota, for tracking only.

  You cannot impose restrictions on BUILTIN\Administrators.

  If a member of BUILTIN\Administrators creates a file, the file is owned by BUILTIN\Administrators and is counted under the SID for BUILTIN\Administrators, not the user's personal SID.

Note: Data ONTAP does not support group quotas based on Windows group IDs. If you specify a Windows group ID as the quota target, the quota is considered to be a user quota.

How quotas are applied to users with multiple IDs

A user can be represented by multiple IDs. You can set up a single user quota for such a user by specifying a list of IDs as the quota target. A file owned by any of these IDs is subject to the restriction of the user quota.

Suppose a user has the UNIX UID 20 and the Windows IDs corp\john_smith and engineering\jsmith. For this user, you can specify a quota where the quota target is a list of the UID and Windows IDs. When this user writes to the storage system, the specified quota applies, regardless of whether the write originates from UID 20, corp\john_smith, or engineering\jsmith.

Note: Separate quota rules are considered separate targets, even if the IDs belong to the same user.

For example, for the same user you can specify one quota that limits UID 20 to 1 GB of disk space and another quota that limits corp\john_smith to 2 GB of disk space, even though both IDs represent the same user. Data ONTAP applies quotas to UID 20 and corp\john_smith separately.

In this case, no limits are applied to engineering\jsmith, even though limits are applied to the other IDs used by the same user.

How Data ONTAP determines user IDs in a mixed environment

If you have users accessing your Data ONTAP storage from both Windows and UNIX clients, then both Windows and UNIX security are used to determine file ownership. Several factors determine whether Data ONTAP uses a UNIX or Windows ID when applying user quotas.

If the security style of the qtree or FlexVol volume that contains the file is only NTFS or only UNIX, then the security style determines the type of ID used when applying user quotas. For qtrees with the mixed security style, the type of ID used is determined by whether the file has an ACL.

The following table summarizes what type of ID is used:

Security Style	ACL	No ACL
UNIX	UNIX ID	UNIX ID
Mixed	Windows ID	UNIX ID
NTFS	Windows ID	Windows ID

Related concepts

How you link UNIX and Windows names for quotas on page 132

How quotas with multiple users work

When you put multiple users in the same quota target, the quota limits defined by that quota are not applied to each individual user; in this case, the quota limits are *shared* among all users listed in the quota target.

Note: If you combine separate user quotas into one multi-user quota, you can activate the change by resizing quotas. However, if you want to remove users from a quota target with multiple users, or add users to a target that already has multiple users, you must reinitialize quotas before the change takes effect.

Example of more than one user in a quota target

In the following example, there are two users listed in the quota target:

```
volume quota policy rule create -vserver vs0 -policy-name
quota_policy_0 -volume vol0 -type user -target corp\jsmith,corp\chen -
disk-limit 80MB
```

The two users can use up to 80 MB of space combined. If one uses 75 MB, then the other one can use only 5 MB.

How you link UNIX and Windows names for quotas

In a mixed environment, users can log in as either Windows users or UNIX users. You can configure quotas to recognize that a user's UNIX id and Windows ID represent the same user.

Quotas for Windows user name are mapped to a UNIX user name, or vice versa, when both of the following conditions are met:

- The `user-mapping` parameter is set to "on" in the quota rule for the user.
- The user names have been mapped with the `vserver name-mapping` commands.

When a UNIX and Windows name are mapped together, they are treated as the same person for determining quota usage.

Related concepts

How Data ONTAP determines user IDs in a mixed environment on page 131

How quotas work with qtrees

You can create quotas with a qtree as their target; these quotas are called *tree quotas*. You can also create user and group quotas for a specific qtree. In addition, quotas for a FlexVol volume are sometimes inherited by the qtrees contained by that volume.

How tree quotas work

You can create a quota with a qtree as its target to limit how large the target qtree can become. These quotas are also called *tree quotas*.

When you apply a quota to a qtree, the result is similar to a disk partition, except that you can change the qtree's maximum size at any time by changing the quota. When applying a tree quota, Data ONTAP limits the disk space and number of files in the qtree, regardless of their owners. No users, including root and members of the BUILTIN\Administrators group, can write to the qtree if the write operation causes the tree quota to be exceeded.

Note: The size of the quota does not guarantee any specific amount of available space. The size of the quota can be larger than the amount of free space available to the qtree. You can use the `volume quota report` command to determine the true amount of available space in the qtree.

How user and group quotas work with qtrees

Tree quotas limit the overall size of the qtree. To prevent individual users or groups from consuming the entire qtree, you specify a user or group quota for that qtree.

Example user quota in a qtree

Suppose you have no user quotas on vol2. It comes to your attention that a certain user, corp \kjones, is taking up too much space in a critical qtree, qt1, that resides in vol2. You can restrict this user's space in the critical qtree with the following command:

```
volume policy rule create -vserver vs0 -policy-name quota_policy_0 -
volume vol2 -type user -target corp\kjones -qtree qt1 -disk-limit 20MB
-threshold 15MB
```

Related concepts

How quotas work with users and groups on page 127

How default tree quotas on a FlexVol volume create derived tree quotas

When you create a default tree quota on a FlexVol volume, corresponding derived tree quotas are automatically created for every qtree in that volume.

These derived tree quotas have the same limits as the default tree quota. If no additional quotas exist, the limits have the following effects:

- Users can use as much space in a qtree as they are allotted for the entire volume (provided they did not exceed the limit for the volume by using space in the root or another qtree).
- Each of the qtrees can grow to consume the entire volume.

The existence of a default tree quota on a volume continues to affect all new qtrees that are added to the volume. Each time a new qtree is created, a derived tree quota is also created.

Like all derived quotas, derived tree quotas display the following behaviors:

- Are created only if the target does not already have an explicit quota.
- Appear in quota reports but do not appear when you show quota rules with the `volume quota policy rule show` command.

Example of derived tree quotas

You have a volume with three qtrees (proj1, proj2, and proj3) and the only tree quota is an explicit quota on the proj1 qtree limiting its disk size to 10 GB. If you create a default tree quota on the volume and reinitialize quotas on the volume, the quota report now contains four tree quotas:

```
                                    ----Disk----  ----Files-----   Quota
  Volume   Tree      Type    ID        Used  Limit    Used   Limit   Specifier
```

```
  -------  --------  ------  -------  -----  -----  ------  ------   ---------
  vol1     proj1     tree    1           0B   10GB       1       -   proj1
  vol1               tree    *           0B   20GB       0       -   *
  vol1     proj2     tree    2           0B   20GB       1       -   proj2
  vol1     proj3     tree    3           0B   20GB       1       -   proj3
...
```

The first line shows the original explicit quota on the proj1 qtree. This quota remains unchanged.

The second line shows the new default tree quota on the volume. The asterisk (*) Quota Specifier indicates it is a default quota. This quota is a result of the quota rule that you created.

The last two lines show new derived tree quotas for the proj2 and proj3 qtrees. Data ONTAP automatically created these quotas as a result of the default tree quota on the volume. These derived tree quotas have the same 20-GB disk limit as the default tree quota on the volume. Data ONTAP did not create a derived tree quota for the proj1 qtree because the proj1 qtree already had an explicit quota.

Related concepts

How derived quotas work on page 124

How default user quotas on a FlexVol volume affect quotas for the qtrees in that volume

If a default user quota is defined for a FlexVol volume, a default user quota is automatically created for every qtree contained by that volume for which an explicit or derived tree quota exists.

If a default user quota on the qtree already exists, it remains unaffected when the default user quota on the volume is created.

The automatically created default user quotas on the qtrees have the same limits as the default user quota you create for the volume.

An explicit user quota for a qtree overrides (replaces the limits applied by) the automatically created default user quota, the same way as it overrides a default user quota on that qtree that was created by an administrator.

Related concepts

How default user and group quotas create derived quotas on page 128

How qtree changes affect quotas

When you delete, rename, or change the security style of a qtree, the quotas applied by Data ONTAP might change, depending on the current quotas being applied.

How deleting a qtree affects tree quotas

When you delete a qtree, all quotas applicable to that qtree, whether they are explicit or derived, are no longer applied by Data ONTAP.

Whether the quota rules persist depends on where you delete the qtree:

- If you delete a qtree using Data ONTAP, the quota rules for that qtree are automatically deleted, including tree quota rules and any user and group quota rules configured for that qtree.
- If you delete a qtree using your CIFS or NFS client, you must delete any quota rules for that qtree to avoid getting errors when you reinitialize quotas. If you create a new qtree with the same name as the one you deleted, the existing quota rules are not applied to the new qtree until you reinitialize quotas.

How renaming a qtree affects quotas

When you rename a qtree using Data ONTAP, the quota rules for that qtree are automatically updated. If you rename a qtree using your CIFS or NFS client, you must update any quota rules for that qtree.

Note: If you rename a qtree using your CIFS or NFS client and do not update quota rules for that qtree with the new name before you reinitialize quotas, quotas will not be applied to the qtree and explicit quotas for the qtree—including tree quotas and user or group quotas for the qtree—might be converted into derived quotas.

How changing the security style of a qtree affects user quotas

You can apply Access Control Lists (ACLs) on qtrees by using NTFS or mixed security styles, but not by using the UNIX security style. Therefore, changing the security style of a qtree might affect how quotas are calculated. You should always reinitialize quotas after you change the security style of a qtree.

If you change the security style of a qtree from NTFS or mixed to UNIX, any ACLs on files in that qtree are ignored and the file usage is charged against the UNIX user IDs.

If you change the security style of a qtree from UNIX to either mixed or NTFS, the previously hidden ACLs become visible. In addition, any ACLs that were ignored become effective again, and the NFS user information is ignored. If no ACL existed before, the NFS information continues to be used in the quota calculation.

Note: To make sure that quota usages for both UNIX and Windows users are properly calculated after you change the security style of a qtree, you must reinitialize quotas for the volume containing that qtree.

Example

The following example shows how a change in the security style of a qtree results in a different user being charged for the usage of a file in the particular qtree.

Suppose NTFS security is in effect on qtree A, and an ACL gives Windows user corp\joe ownership of a 5 MB file. User corp\joe is charged with 5 MB of disk space usage for qtree A.

Now you change the security style of qtree A from NTFS to UNIX. After quotas are reinitialized, Windows user corp\joe is no longer charged for this file; instead, the UNIX user corresponding to the UID of the file is charged for the file. The UID could be a UNIX user mapped to corp\joe or the root user.

How quotas are activated

New quotas and changes to quotas do not take effect until they are activated. Knowing how quota activation works can help you manage your quotas less disruptively.

You can activate quotas at the volume level.

Quotas are activated either by *initializing* (turning them on) or by *resizing*. Turning off quotas and turning them on again is called reinitializing.

The length of the activation process and its impact on quota enforcement depends on the type of activation:

- The initialization process involves two parts: a `quota on` job and a quota scan of the volume's entire file system. The scan begins after the `quota on` job completes successfully. The quota scan can take some time; the more files that the volume has, the longer it takes. Until the scan is finished, quota activation is not complete and quotas are not enforced.
- The resize process involves only a `quota resize` job. Because it does not involve a quota scan, resizing takes less time than a quota initialization. During a resize process, quotas are enforced.

By default, the `quota on` and `quota resize` jobs run in the background, which permits you to use other commands at the same time.

Note: If your quota changes are made in a quota policy that is not currently assigned, you must assign the quota policy to the volume before resizing or reinitializing the quotas.

Errors and warnings from the activation process are sent to the event management system. If you use the `-foreground` parameter with the `volume quota on` or `volume quota resize` commands, the command does not return until the job is complete; this is useful if you are reinitializing from a script. To display errors and warnings later, you can use the `volume quota show` command with the `-instance` parameter.

Quota activation persists across halts and reboots. The process of quota activation does not affect the availability of the storage system data.

Related concepts

When you can use resizing on page 137
When a full quota reinitialization is required on page 138

When you can use resizing

Because quota resizing is faster than quota initialization, you should use resizing whenever possible. However, resizing only works for certain types of quota changes.

You can resize quotas when making the following types of changes to the quota rules:

- Changing an existing quota.
  For example, changing the limits of an existing quota.
- Adding a quota for a quota target for which a default quota or a default tracking quota exists.
- Deleting a quota for which a default quota or default tracking quota entry is specified.
- Combining separate user quotas into one multi-user quota.

Attention: After you have made extensive quotas changes, you should perform a full reinitialization to ensure that all of the changes take effect.

Note: If you attempt to resize and not all of your quota changes can be incorporated by using a resize operation, Data ONTAP issues a warning.

You can determine from the quota report whether your storage system is tracking disk usage for a particular user, group, or qtree. If you see a quota in the quota report, it means that the storage system is tracking the disk space and the number of files owned by the quota target.

Example quotas changes that can be made effective by resizing

Some quota rule changes can be made effective by resizing. Consider the following quotas:

```
cluster1::>volume quota policy rule show
  Vserver: vs1              Policy: quota_policy_0  Volume:
                                                                  vol2
                                                 Soft               Soft
                           User          Disk     Disk    Files     Files
  Type    Target    Qtree  Mapping      Limit    Limit    Limit     Limit    Threshold
  -----   --------  ------ -------   --------  -------  --------  -------  ---------
  user    ""        ""     -            50MB        -     15360         -          -
  group   ""        ""     -           750MB        -     87040         -          -
  tree    ""        ""     -               -        -         -         -          -
  user    corp\jdoe ""     -           100MB        -     76800         -          -
  user    corp\kbuck""     -           100MB        -     76800         -          -
```

Suppose you make the following changes:

- Increase the number of files for the default user target.
- Add a new user quota for a new user, boris, that needs more disk limit than the default user quota.
- Delete the kbuck user's explicit quota entry; the new user now needs only the default quota limits.

These changes result in the following quotas:

```
cluster1::>volume quota policy rule show
  Vserver: vs1              Policy: quota_policy_0  Volume:
                                                              vol2
                                                  Soft          Soft
                                User          Disk    Disk    Files    Files
  Type   Target    Qtree   Mapping      Limit   Limit   Limit    Limit    Threshold
  -----  --------  ------- -------  --------  -------  --------  -------  ---------
  user   ""        ""      -            50MB        -    25600        -          -
  group  ""        ""      -           750MB        -    87040        -          -
  tree   ""        ""      -               -        -        -        -          -
  user   corp\jdoe ""      -           100MB        -    76800        -          -
  user   corp\boris""      -           100MB        -    76800        -          -
```

Resizing activates all of these changes; a full quota reinitialization is not necessary.

Related concepts

How quotas are activated on page 136

When a full quota reinitialization is required

Although resizing quotas is faster, you must do a full quota reinitialization if you make certain small or extensive changes to your quotas.

A full quota reinitialization is necessary in the following circumstances:

- You create a quota for a target that has not previously had a quota.
- You change user mapping (with the `vserver name-mapping` commands) of users that are targets of quota rules where the user-mapping parameter is enabled.
- You change the security style of a qtree from UNIX to either mixed or NTFS.
- You change the security style of a qtree from mixed or NTFS to UNIX.
- You remove users from a quota target with multiple users, or add users to a target that already has multiple users.
- You make extensive changes to your quotas.

Example of quotas changes that require initialization

Suppose you have a volume that contains three qtrees and the only quotas in the volume are three tree quotas. You decide to make the following changes:

- Add a new qtree and create a new tree quota for it.
- Add a default user quota for the volume.

Both of these changes require a full quota initialization. Resizing does not make the quotas effective.

Related concepts

How quotas are activated on page 136

How you can view quota information

You can use quota reports to view details such as the configuration of quota rules and policies, enforced and configured quotas, and errors that occur during quota resizing and reinitialization.

Viewing quota information is useful in situations such as the following:

- Configuring quotas—for example, to configure quotas and verify the configurations
- Responding to notifications that disk space or file limits will soon be reached or that they have been reached
- Responding to requests for more space

How you can use the quota report to see what quotas are in effect

Because of the various ways that quotas interact, more quotas are in effect than just the ones you have explicitly created. To see what quotas are in effect, you can view the quota report.

The following examples show quota reports for different types of quotas applied on a FlexVol volume vol1, and a qtree q1 contained in that volume:

Example with no user quotas specified for the qtree

In this example, there is one qtree, q1, which is contained by the volume vol1. The administrator has created three quotas:

- A default tree quota limit on vol1 of 400 MB
- A default user quota limit on vol1 of 100 MB
- An explicit user quota limit on vol1 of 200 MB for the user jsmith

The quota report for these quotas looks similar to the following excerpt:

```
cluster1::> volume quota report
  Vserver: vs1
                                    ----Disk----  ----Files-----   Quota
  Volume   Tree      Type    ID        Used  Limit    Used   Limit   Specifier
  -------  --------  ------  -------  -----  -----  ------  ------   ---------
  vol1     -          tree   *           0B  400MB       0       -   *
  vol1     -          user   *           0B  100MB       0       -   *
  vol1     -          user   corp/jsmith
                                      150B  200MB       7       -   corp/jsmith
  vol1     q1         tree   1           0B  400MB       6       -   q1
  vol1     q1         user   *           0B  100MB       0       -
  vol1     q1         user   corp/jsmith 0B  100MB       5       -
  vol1     -          user   root        0B   0MB        1       -
  vol1     q1         user   root        0B   0MB        8       -
```

The first three lines of the quota report display the three quotas specified by the administrator. Since two of these quotas are default quotas, Data ONTAP automatically creates derived quotas.

The fourth line displays the tree quota that is derived from the default tree quota for every qtree in vol1 (in this example, only q1).

The fifth line displays the default user quota that is created for the qtree as a result of the existence of the default user quota on the volume and the qtree quota.

The sixth line displays the derived user quota that is created for jsmith on the qtree because there is a default user quota for the qtree (line 5) and the user jsmith owns files on that qtree. Note that the limit applied to the user jsmith in the qtree q1 is not determined by the explicit user quota limit (200 MB). This is because the explicit user quota limit is on the volume, so it does not affect limits for the qtree. Instead, the derived user quota limit for the qtree is determined by the default user quota for the qtree (100 MB).

The last two lines display more user quotas that are derived from the default user quotas on the volume and on the qtree. A derived user quota was created for the root user on both the volume and the qtree because the root user owned files on both the volume and the qtree. Since the root user gets special treatment in terms of quotas, its derived quotas are tracking quotas only.

Example with user quotas specified for the qtree

This example is similar to the previous one, except that the administrator has added two quotas on the qtree.

There is still one volume, vol1, and one qtree, q1. The administrator has created the following quotas:

- A default tree quota limit on vol1 of 400 MB
- A default user quota limit on vol1 of 100 MB
- An explicit user quota limit on vol1 for the user jsmith of 200 MB
- A default user quota limit on qtree q1 of 50 MB
- An explicit user quota limit on qtree q1 for the user jsmith of 75 MB

The quota report for these quotas looks like this:

```
cluster1::> volume quota report
  Vserver: vs1
                                     ----Disk----  ----Files-----   Quota
  Volume   Tree      Type    ID        Used  Limit    Used   Limit   Specifier
  -------  --------  ------  -------  -----  -----  ------  ------   ---------
  vol1     -          tree   *           0B  400MB       0       -   *
  vol1     -          user   *           0B  100MB       0       -   *
  vol1     -          user   corp/jsmith
                                    2000B  200MB       7       -   corp/jsmith
  vol1     q1         user   *           0B   50MB       0       -   *
  vol1     q1         user   corp/jsmith 0B  75MB        5       -   corp/jsmith
  vol1     q1         tree   1           0B 400MB        6       -   q1
  vol1     -          user   root        0B   0MB        2       -
  vol1     q1         user   root        0B   0MB        1       -
```

The first five lines of the quota report display the five quotas created by the administrator. Since some of these quotas are default quotas, Data ONTAP automatically creates derived quotas.

The sixth line displays the tree quota that is derived from the default tree quota for every qtree in vol1 (in this example, only q1).

The last two lines display the user quotas that are derived from the default user quotas on the volume and on the qtree. A derived user quota was created for the root user on both the volume and the qtree because the root user owned files on both the volume and the qtree. Since the root user gets special treatment in terms of quotas, its derived quotas are tracking quotas only.

No other default quotas or derived quotas were created for the following reasons:

- A derived user quota was not created for the jsmith user even though the user owns files on both the volume and the qtree because the user already has explicit quotas at both levels.
- No derived user quotas were created for other users because no other users own files on either the volume or the qtree.
- The default user quota on the volume did not create a default user quota on the qtree because the qtree already had a default user quota.

Related concepts

Why enforced quotas differ from configured quotas on page 141

Why enforced quotas differ from configured quotas

Enforced quotas differ from configured quotas because derived quotas are enforced without being configured but configured quotas are enforced only after they are successfully initialized. Understanding these differences can help you compare the enforced quotas that are shown in quota reports to the quotas that you configured.

Enforced quotas, which appear in quota reports, might differ from the configured quota rules for the following reasons:

- Derived quotas are enforced without being configured as quota rules; Data ONTAP creates derived quotas automatically in response to default quotas.
- Quotas might not have been reinitialized on a volume after quota rules were configured.
- Errors might have occurred when quotas were initialized on a volume.

Using the quota report to determine which quotas limit writes to a specific file

You can use the `volume quota report` command with a specific file path to determine which quota limits affect write operations to a file. This can help you understand which quota is preventing a write operation.

Step

1. Use the `volume quota report` command with the `-path` parameter.

Example of showing quotas affecting a specific file

The following example shows the command and output to determine what quotas are in effect for writes to the file file1, which resides in the qtree q1 in the FlexVol volume vol2:

```
cluster1:> volume quota report -vserver vs0 -volume vol2 -path /vol/
vol2/q1/file1
Virtual Server: vs0
                                     ----Disk----  ----Files-----
Quota
Volume   Tree      Type    ID        Used  Limit    Used   Limit
Specifier
-------  --------  ------  -------  -----  -----  ------  ------
---------
vol2     q1        tree    jsmith     1MB  100MB       2   10000
q1
vol2     q1        group   eng        1MB  700MB       2   70000
vol2               group   eng        1MB  700MB       6   70000   *
vol2               user    corp\jsmith
                                      1MB   50MB       1       -   *
vol2     q1        user    corp\jsmith
                                      1MB   50MB       1       -
5 entries were displayed.
```

Commands for displaying information about quotas

You can use commands to display a quota report containing enforced quotas and resource usage, display information about quota state and errors, or about quota policies and quota rules.

Note: You can run the following commands only on FlexVol volumes.

If you want to...	Use this command...
View information about enforced quotas	`volume quota report`
View resource usage (disk space and number of files) of quota targets	`volume quota report`
Determine which quota limits are affected when a write to a file is allowed	`volume quota report` with the `-path` parameter
Display the quota state, such as `on`, `off`, and `initializing`	`volume quota show`
View information about quota message logging	`volume quota show` with the `-logmsg` parameter
View errors that occur during quota initialization and resizing	`volume quota show` with the `-instance` parameter

If you want to...	Use this command...
View information about quota policies	`volume quota policy show`
View information about quota rules	`volume quota policy rule show`
View the name of the quota policy that is assigned to a Storage Virtual Machine (SVM, formerly known as Vserver)	`vserver show` with the `-instance` parameter

See the man page for each command for more information.

Related concepts

When to use the volume quota policy rule show and volume quota report commands on page 143

When to use the volume quota policy rule show and volume quota report commands

Although both commands show information about quotas, the `volume quota policy rule show` quickly displays configured quota rules while the `volume quota report` command, which consumes more time and resources, displays enforced quotas and resource usage.

The `volume quota policy rule show` command is useful for the following purposes:

- Check the configuration of quota rules before activating them
  This command displays all configured quota rules regardless of whether the quotas have been initialized or resized.
- Quickly view quota rules without affecting system resources
  Because it does not display disk and file usage, this command is not as resource intensive as a quota report.
- Display the quota rules in a quota policy that is not assigned to the Storage Virtual Machine (SVM)

The `volume quota report` command is useful for the following purposes:

- View enforced quotas, including derived quotas
- View the disk space and number of files used by every quota in effect, including targets affected by derived quotas
  (For default quotas, the usage appears as "0" because the usage is tracked against the resulting derived quota.)
- Determine which quota limits affect when a write to a file will be allowed
  Add the `-path` parameter to the `volume quota report` command.

Note: The quota report is resource-intensive operation. If you run it on many FlexVol volumes in the cluster, it might take a long time to complete. A more efficient way would be to view the quota report for a particular volume in an SVM.

Related references

Commands for displaying information about quotas on page 142

Difference in space usage displayed by a quota report and a UNIX client

The value of used disk space that is displayed in a quota report for a FlexVol volume or qtree can be different from the value displayed by a UNIX client for the same volume or qtree. The difference in usage values is because of the difference in methods followed by the quota report and the UNIX commands for calculating the data blocks in the volume or qtree.

For example, if a volume contains a file that has empty data blocks (to which data is not written), the quota report for the volume does not count the empty data blocks while reporting the space usage. However, when the volume is mounted on a UNIX client and the file is shown as the output of the `ls` command, the empty data blocks are also included in the space usage. Therefore, the `ls` command displays a higher file size when compared to the space usage displayed by the quota report.

Similarly, the space usage values shown in a quota report can also differ from the values shown as a result of UNIX commands such as `df` and `du`.

How a quota report accounts for disk space and file usage

The number of files used and the amount of disk space specified in a quota report for a FlexVol volume or a qtree depend on the count of the used data blocks corresponding to every inode in the volume or the qtree.

The block count includes both direct and indirect blocks used for regular and stream files. The blocks used for directories, Access Control Lists (ACLs), stream directories, and metafiles do not get accounted for in the quota report. In case of UNIX sparse files, empty data blocks are not included in the quota report.

Related concepts

How the ls command accounts for space usage on page 144
How the df command accounts for file size on page 145
How the du command accounts for space usage on page 146

How the ls command accounts for space usage

When you use the `ls` command to view the contents of a FlexVol volume mounted on a UNIX client, the file sizes displayed in the output could be lesser or more than the space usage displayed in the quota report for the volume depending on the type of data blocks for the file.

The output of the `ls` command displays only the size of a file and does not include indirect blocks used by the file. Any empty blocks of the file also get included in the output of the command.

Therefore, if a file does not have empty blocks, the size displayed by the `ls` command might be less than the disk usage specified by a quota report because of the inclusion of indirect blocks in the quota

report. Conversely, if the file has empty blocks, then the size displayed by the `ls` command might be more than the disk usage specified by the quota report.

The output of the `ls` command displays only the size of a file and does not include indirect blocks used by the file. Any empty blocks of the file also get included in the output of the command.

Example of the difference between space usage accounted by the ls command and a quota report

The following quota report shows a limit of 10 MB for a qtree q1:

```
                                  ----Disk----  ----Files-----   Quota
  Volume   Tree      Type    ID        Used  Limit    Used   Limit   Specifier
  -------  --------  ------  -------  -----  -----  ------  ------   ---------
  vol1     q1          tree    user1   10MB  10MB        1       -   q1

...
```

A file present in the same qtree can have a size exceeding the quota limit when viewed from a UNIX client by using the `ls` command, as shown in the following example:

```
[user1@lin-sys1 q1]$ ls -lh
-rwxr-xr-x  1 user1 nfsuser  27M Apr 09  2013 file1
```

Related concepts

How a quota report accounts for disk space and file usage on page 144
How the df command accounts for file size on page 145
How the du command accounts for space usage on page 146

How the df command accounts for file size

When you run the `df` command from the mount point of a qtree for which a quota rule is configured, the output of the command shows the same space usage as the value specified by the quota report.

If quotas are enabled for the volume that contains the qtree, the space usage reported by the `df` command excludes blocks used by directories, ACLs, stream directories, and metafiles. Therefore, the reported space usage exactly matches the value specified by the quota report.

However, if the qtree does not have a quota rule configured or if quotas are not enabled for the FlexVol volume, then the reported space usage includes the blocks consumed by directories, Access Control Lists (ACLs), stream directories and metafiles for the entire volume, including other qtrees within the volume. In such a situation, the space usage reported by the `df` command is more than the value specified by the quota report.

Example of space usage accounted by the df command and a quota report

The following quota report shows a limit of 10 MB for a qtree q1:

```
                                    ----Disk----  ----Files-----   Quota
  Volume   Tree      Type    ID       Used  Limit    Used   Limit   Specifier
  -------  --------  ------  -------  -----  -----  ------  ------   ---------
  vol1     q1          tree    user1   10MB  10MB       1       -   q1

...
```

In the following example, the space usage as the output of the `df` command shows the same limit of 10 MB (in terms of 1K blocks) because quota rules are configured for the qtree:

```
[user1@lin-sys1 q1]$ df -k
192.0.2.245:/vol/vol1/q1
                     10240  10240 0  100% /q1
```

Related concepts

How a quota report accounts for disk space and file usage on page 144
How the ls command accounts for space usage on page 144
How the du command accounts for space usage on page 146

How the du command accounts for space usage

When you run the `du` command to check the disk space usage for a qtree or FlexVol volume mounted on a UNIX client, the usage value might be higher than the value displayed by a quota report for the qtree or volume.

The output of the `du` command contains the combined space usage of all the files through the directory tree beginning at the level of the directory where the command is issued. Because the usage value displayed by the `du` command also includes the data blocks for directories, it is higher than the value displayed by a quota report.

Example of the difference between space usage accounted by the du command and a quota report

The following quota report shows a limit of 10 MB for a qtree q1:

```
                                    ----Disk----  ----Files-----   Quota
  Volume   Tree      Type    ID       Used  Limit    Used   Limit   Specifier
  -------  --------  ------  -------  -----  -----  ------  ------   ---------
  vol1     q1          tree    user1   10MB   10MB       1       -   q1

...
```

In the following example, the disk space usage as the output of the `du` command shows a higher value that exceeds the quota limit:

```
[user1@lin-sys1 q1]$ du -sh
11M     q1
```

Related concepts

How a quota report accounts for disk space and file usage on page 144
How the ls command accounts for space usage on page 144
How the df command accounts for file size on page 145

Examples of quota configuration

These examples help you understand how to configure quotas and read quota reports.

For the following examples, assume that you have a storage system that includes a Storage Virtual Machine (SVM, formerly known as Vserver), vs1, with one volume, vol1. To start setting up quotas, you create a new quota policy for the SVM with the following command:

```
cluster1::>volume quota policy create -vserver vs1 -policy-name
quota_policy_vs1_1
```

Since the quota policy is new, you assign it to the SVM by entering the following command:

```
cluster1::>vserver modify -vserver vs1 -quota-policy quota_policy_vs1_1
```

Example 1: Default user quota

You decide to impose a hard limit of 50 MB for each user in vol1 by entering the following command:

```
cluster1::>volume quota policy rule create -vserver vs1 -policy-
name quota_policy_vs1_1 -volume vol1 -type user -target "" -disk-
limit 50MB -qtree ""
```

To activate the new rule, you initialize quotas on the volume by entering the following command:

```
cluster1::>volume quota on -vserver vs1 -volume vol1 -foreground
```

To view the quota report, you enter the following command:

```
cluster1::>volume quota report
```

The resulting quota report is similar to the following report:

```
Vserver: vs1
                                     ----Disk----  ----Files-----   Quota
  Volume   Tree      Type    ID        Used  Limit    Used   Limit   Specifier
  -------  --------  ------  -------  -----  -----  ------  ------   ---------
```

```
  vol1              user    *           0B   50MB       0       -   *
  vol1              user    jsmith     49MB   50MB      37       -   *
  vol1              user    root        0B      -       1       -
```

The first line shows the default user quota that you created, including the disk limit. Like all default quotas, this default user quota does not display information about disk or file usage. In addition to the quota that was created, two other quotas appear—one quota for each user that currently owns files on vol1. These additional quotas are user quotas that were derived automatically from the default user quota. The derived user quota for the user jsmith has the same 50-MB disk limit as the default user quota. The derived user quota for the root user is a tracking quota (without limits).

If any user on the system (other than the root user) tries to perform an action that would use more than 50 MB in vol1 (for example, writing to a file from an editor), the action fails.

Example 2: Explicit user quota overriding a default user quota

If you need to provide more space in volume vol1 to the user jsmith, then you enter the following command:

```
cluster1::>volume quota policy rule create -vserver vs1 -policy-
name quota_policy_vs1_1 -volume vol1 -type user -target jsmith -
disk-limit 80MB -qtree ""
```

This is an explicit user quota, because the user is explicitly listed as the target of the quota rule.

This is a change to an existing quota limit, because it changes the disk limit of the derived user quota for the user jsmith on the volume. Therefore, you do not need to reinitialize quotas on the volume to activate the change. You can resize quotas by entering the following command:

```
cluster1::>volume quota resize -vserver vs1 -volume vol1 -foreground
```

Quotas remain in effect while you resize, and the resizing process is short.

The resulting quota report is similar to the following report:

```
cluster1::> volume quota report
  Vserver: vs1
                                    ----Disk----  ----Files-----   Quota
  Volume   Tree      Type    ID        Used  Limit    Used   Limit   Specifier
  -------  --------  ------  -------  -----  -----  ------  ------   ---------
  vol1              user    *           0B   50MB       0       -   *
  vol1              user    jsmith     50MB   80MB      37       -   jsmith
  vol1              user    root        0B      -       1       -
  3 entries were displayed.
```

The second line now shows a Disk Limit of 80 MB and a Quota Specifier of jsmith.

Therefore, jsmith can use up to 80 MB of space on vol1, even though all other users are still limited to 50 MB.

Example 3: Thresholds

Suppose you want to receive a notification when users reach within 5 MB of their disk limits. To create a threshold of 45 MB for all users, and a threshold of 75 MB for jsmith, you change the existing quota rules by entering the following commands:

```
cluster1::>volume quota policy rule modify -vserver vs1 -policy
quota_policy_vs1_1 -volume vol1 -type user -target "" -qtree "" -
threshold 45MB
cluster1::>volume quota policy rule modify -vserver vs1 -policy
quota_policy_vs1_1 -volume vol1 -type user -target jsmith -qtree ""
-threshold 75MB
```

Since the sizes of the existing rules are changed, you resize quotas on the volume in order to activate the changes. You wait until the resize process is finished.

To see the quota report with thresholds, you add the `-thresholds` parameter to the `volume quota report` command:

```
cluster1::>volume quota report -thresholds
  Vserver: vs1
                                    ----Disk----  ----Files-----
  Volume   Tree      Type    ID        Used  Limit   Used    Limit  Quota
                                           (Thold)                 Specifier
  -------  --------  ------  --------  -----  -----  ------  ------  ---------
  vol1               user    *            0B   50MB       0       -  *
                                           (45MB)
  vol1               user    jsmith      59MB   80MB      55       -  jsmith
                                           (75MB)
  vol1               user    root         0B      -       1       -
                                           ( -)
  3 entries were displayed.
```

The thresholds appear in parentheses in the Disk Limit column.

Example 4: Quotas on qtrees

Suppose you need to partition some space for two projects. You can create two qtrees, named proj1 and proj2, to accommodate those projects within vol1.

Currently, users can use as much space in a qtree as they are allotted for the entire volume (provided they did not exceed the limit for the volume by using space in the root or another qtree). In addition, each of the qtrees can grow to consume the entire volume. If you want to ensure that neither qtree grows beyond 20 GB, you can create default tree quota on the volume by entering the following command:

```
cluster1:>>volume quota policy rule create -vserver vs1 -policy-
name quota_policy_vs1_1 -volume vol1 -type tree -target "" -disk-
limit 20GB
```

Note that the correct type is *tree*, not qtree.

Because this is a new quota, you cannot activate it by resizing. You reinitialize quotas on the volume by entering the following commands:

```
cluster1:>>volume quota off -vserver vs1 -volume vol1
cluster1:>>volume quota on -vserver vs1 -volume vol1 -foreground
```

Note: You must ensure that you wait for about five minutes before reactivating the quotas on each affected volume, as attempting to activate them almost immediately after running the `volume quota off` command might result in errors.

Alternatively, you can run the commands to re-initialize the quotas for a volume from the node that contains the particular volume.

Quotas are not enforced during the reinitialization process, which takes longer than the resizing process.

When you display a quota report, it has several new lines: some lines are for tree quotas and some lines are for derived user quotas.

The following new lines are for the tree quotas:

```
                                    ----Disk----  ----Files-----   Quota
  Volume   Tree      Type    ID        Used  Limit    Used   Limit   Specifier
  -------  --------  ------  -------  -----  -----  ------  ------   ---------
...
  vol1               tree    *           0B   20GB       0       -   *
  vol1     proj1     tree    1           0B   20GB       1       -   proj1
  vol1     proj2     tree    2           0B   20GB       1       -   proj2
...
```

The default tree quota that you created appears in the first new line, which has an asterisk (*) in the ID column. In response to the default tree quota on a volume, Data ONTAP automatically creates derived tree quotas for each qtree in the volume. These are shown in the lines where proj1 and proj2 appear in the Tree column.

The following new lines are for derived user quotas:

```
                                    ----Disk----  ----Files-----   Quota
  Volume   Tree      Type    ID        Used  Limit    Used   Limit   Specifier
  -------  --------  ------  -------  -----  -----  ------  ------   ---------
...
  vol1     proj1     user    *           0B   50MB       0       -
  vol1     proj1     user    root        0B      -       1       -
  vol1     proj2     user    *           0B   50MB       0       -
  vol1     proj2     user    root        0B      -       1       -
...
```

Default user quotas on a volume are automatically inherited for all qtrees contained by that volume, if quotas are enabled for qtrees. When you added the first qtree quota, you enabled quotas on qtrees. Therefore, derived default user quotas were created for each qtree. These are shown in the lines where ID is asterisk (*).

Because the root user is the owner of a file, when default user quotas were created for each of the qtrees, special tracking quotas were also created for the root user on each of the qtrees. These are shown in the lines where ID is root.

Example 5: User quota on a qtree

You decide to limit users to less space in the proj1 qtree than they get in the volume as a whole. You want to keep them from using any more than 10 MB in the proj1 qtree. Therefore, you create a default user quota for the qtree by entering the following command:

```
cluster1::>volume quota policy rule create -vserver vs1 -policy-
name quota_policy_vs1_1 -volume vol1 -type user -target "" -disk-
limit 10MB -qtree proj1
```

This is a change to an existing quota, because it changes the default user quota for the proj1 qtree that was derived from the default user quota on the volume. Therefore, you activate the change by resizing quotas. When the resize process is complete, you can view the quota report.

The following new line appears in the quota report showing the new explicit user quota for the qtree:

```
                                  ----Disk----  ----Files-----   Quota
Volume   Tree      Type    ID       Used  Limit    Used   Limit   Specifier
-------  --------  ------  -------  -----  -----  ------  ------   ---------
vol1     proj1     user    *          0B   10MB       0       -   *
```

However, the user jsmith is being prevented from writing more data to the proj1 qtree because the quota you created to override the default user quota (to provide more space) was on the volume. As you have added a default user quota on the proj1 qtree, that quota is being applied and limiting all the users' space in that qtree, including jsmith. To provide more space to the user jsmith, you add an explicit user quota rule for the qtree with an 80 MB disk limit to override the default user quota rule for the qtree by entering the following command:

```
cluster1::>volume quota policy rule create -vserver vs1 -policy-
name quota_policy_vs1_1 -volume vol1 -type user -target jsmith -
disk-limit 80MB -qtree proj1
```

Since this is an explicit quota for which a default quota already existed, you activate the change by resizing quotas. When the resize process is complete, you display a quota report.

The following new line appears in the quota report:

```
                                  ----Disk----  ----Files-----   Quota
Volume   Tree      Type    ID       Used  Limit    Used   Limit   Specifier
-------  --------  ------  -------  -----  -----  ------  ------   ---------
vol1     proj1     user    jsmith    61MB   80MB      57       -   jsmith
```

The final quota report is similar to the following report:

```
cluster1::>volume quota report
  Vserver: vs1
                                    ----Disk----  ----Files-----   Quota
  Volume   Tree      Type    ID        Used  Limit    Used   Limit   Specifier
  -------  --------  ------  -------  -----  -----  ------  ------   ---------
  vol1               tree    *          0B   20GB       0       -   *
  vol1               user    *          0B   50MB       0       -   *
  vol1               user    jsmith    70MB   80MB      65       -   jsmith
  vol1     proj1     tree    1          0B   20GB       1       -   proj1
  vol1     proj1     user    *          0B   10MB       0       -   *
  vol1     proj1     user    root       0B      -       1       -
  vol1     proj2     tree    2          0B   20GB       1       -   proj2
  vol1     proj2     user    *          0B   50MB       0       -
  vol1     proj2     user    root       0B      -       1       -
  vol1               user    root       0B      -       3       -
  vol1     proj1     user    jsmith    61MB   80MB      57       -   jsmith
  11 entries were displayed.
```

The user jsmith is required to meet the following quota limits to write to a file in proj1:

1. The tree quota for the proj1 qtree.
2. The user quota on the proj1 qtree.
3. The user quota on the volume.

Related tasks

Setting up quotas on an SVM with FlexVol volumes on page 152

Setting up quotas on an SVM with FlexVol volumes

To set up quotas on a new Storage Virtual Machine (SVM, formerly known as Vserver) with FlexVol volumes, you must create a quota policy, add quota policy rules to the policy, assign the policy to the SVM, and initialize quotas on each FlexVol volume on the SVM.

Steps

1. Use the `vserver show` command with the `-instance` option to display the name of the default quota policy that was automatically created when the SVM was created.

   If a name was not specified when the SVM was created, the name is "default". You can also use the `vserver quota policy rename` command to give the default policy a name.

   Note: You can also create a new policy by using the `volume quota policy create` command.

2. Use the `volume quota policy rule create` command to create *any* of the following quota rules for each volume on the SVM:

   - Default quota rules for all users
   - Explicit quota rules for specific users
   - Default quota rules for all groups
   - Explicit quota rules for specific groups
   - Default quota rules for all qtrees

- Explicit quota rules for specific qtrees

3. Use the `volume quota policy rule show` command to check that the quota rules are configured correctly.
4. If you are working on a new policy, use the `vserver modify` command to assign the new policy to the SVM.
5. Use the `volume quota on` command to initialize the quotas on each volume on the SVM.

   You can monitor the initialization process in the following ways:

   - When you use the `volume quota on` command, you can add the `-foreground` parameter to run the quota on job in the foreground. (By default, the job runs in the background.) When the job runs in the background, you can monitor its progress by using the `job show` command.
   - You can use the `volume quota show` command to monitor the status of the quota initialization.
6. Use the `volume quota show -instance` command to check for initialization errors, such as quota rules that failed to initialize.
7. Use the `volume quota report` command to display a quota report so that you can ensure the enforced quotas match your expectations.

Related concepts

What quota rules, quota policies, and quotas are on page 120
Quota targets and types on page 120
Special kinds of quotas on page 122
How quotas are activated on page 136
How you can view quota information on page 139

Making minor changes to existing quota limits

When you make changes to the size of existing quotas, you can resize the quotas on all affected volumes, which is faster than re-initializing quotas on those volumes.

About this task

You have a Storage Virtual Machine (SVM, formerly known as Vserver) with enforced quotas and you want either to change the size limits of existing quotas or to add or delete quotas for targets that already have derived quotas.

Steps

1. Use the `vserver show` command with the `-instance` parameter to determine the name of the policy that is currently assigned to the SVM.
2. Modify quota rules by performing any of the following actions:

- Use the `volume quota policy rule modify` command to modify the disk or file limits of existing quota rules.
- Use the `volume quota policy rule create` command to create explicit quota rules for targets (users, groups, or qtrees) that currently have derived quotas.
- Use the `volume quota policy rule delete` command to delete explicit quota rules for targets (users, groups, or qtrees) that also have default quotas.

3. Use the `volume quota policy rule show` command to check that the quota rules are configured correctly.
4. Use the `volume quota resize` command on each volume where you changed quotas to activate the changes on each volume.

   You can monitor the resize process in either of the following ways:

   - When you use the `volume quota resize` command, you can add the `-foreground` parameter to run the resize job in the foreground. (By default, the job runs in the background.) When the job runs in background, you can monitor its progress by using the `job show` command.
   - You can use the `volume quota show` command to monitor the resize status.
5. Use the `volume quota show -instance` command to check for resize errors, such as quota rules that failed to get resized.

   In particular, check for "new definition" errors, which occur when you resize quotas after adding an explicit quota for a target that does not already have a derived quota.
6. Use the `volume quota report` command to display a quota report so that you can ensure the enforced quotas match your expectations.

Related tasks

Reinitializing quotas after making extensive changes on page 154

Reinitializing quotas after making extensive changes

When you make extensive changes to existing quotas; for example, by adding or deleting quotas for targets that have no enforced quotas— you must make the changes and re-initialize quotas on all affected volumes.

About this task

You have a Storage Virtual Machine (SVM) with enforced quotas and you want to make changes that require a full reinitialization of quotas.

Steps

1. Use the `vserver show` command with the `-instance` parameter to determine the name of the policy that is currently assigned to the SVM.

2. Modify quota rules by performing any of the following actions:

If you want to...	Then...
Create new quota rules	Use the `volume quota policy rule create` command
Modify the settings of existing quota rules	Use the `volume quota policy rule modify` command
Delete existing quota rules	Use the `volume quota policy rule delete` command

3. Use the `volume quota policy rule show` command to check that the quota rules are configured correctly.
4. Re-initialize quotas on each volume where you changed quotas by turning quotas off and then turning quotas on for those volumes.
   a) Use the `volume quota off` command on each affected volume to deactivate quotas on that volume.
   b) Use the `volume quota on` command on each affected volume to activate quotas on that volume.

      Note: You must ensure that you wait for about five minutes before reactivating the quotas on each affected volume, as attempting to activate them almost immediately after running the `volume quota off` command might result in errors.

      Alternatively, you can run the commands to re-initialize the quotas for a volume from the node that contains the particular volume.

   You can monitor the initialization process in either of the following ways:

   - When you use the `volume quota on` command, you can add the `-foreground` parameter to run the quota on job in the foreground. (By default, the job runs in the background.)
     When the job runs in the background, you can monitor its progress by using the `job show` command.
   - You can use the `volume quota show` command to monitor the status of the quota initialization.
5. Use the `volume quota show -instance` command to check for initialization errors, such as quota rules that failed to initialize.
6. Use the `volume quota report` command to display a quota report so that you can ensure the enforced quotas match your expectations.

Related concepts

When a full quota reinitialization is required on page 138
How you can view quota information on page 139

Verifying status of quota upgrades

When you initiate the transition of a FlexVol volume containing quotas from Data ONTAP 7.3.x to a clustered Data ONTAP version such as Data ONTAP 8.2 or later, you can verify if the quotas are getting upgraded to the clustered Data ONTAP environment.

Before you begin

You must have initiated the transition of the FlexVol volume containing quotas from Data ONTAP operating in 7-Mode to clustered Data ONTAP.

Steps

1. View the quota details of a particular clustered Data ONTAP volume to which data from the 7-Mode volume is being transitioned by using the `volume quota show -instance` command.

   The following example shows the quota details of a volume vol3. The details also include the quota state, which is set to `initializing`.

   ```
   cluster1::*> volume quota show -instance -vserver vs1 -volume vol3
                Vserver Name: vs1
                 Volume Name: vol3
                 Quota State: initializing
                  Scan Status: 3%
            Logging Messages: -
            Logging Interval: -
            Sub Quota Status: upgrading
     Last Quota Error Message: -
   Collection of Quota Errors: -
          User Quota enforced: -
         Group Quota enforced: -
          Tree Quota enforced: -
   ```

   Note: The `initializing` state is also shown for volumes where quotas are getting activated. However, the sub quota status for such volumes shows `scanning`.

2. Monitor the progress of the quota upgrade process by using the `job show` command.

3. Verify the completion of the quota upgrade process for the clustered Data ONTAP volume by using the `volume quota show -instance` command.

   The following example shows quota details of the volume vol3 after the upgrade is complete. The quota state is `on`.

   ```
   cluster1::> volume quota show -instance -vserver vs1 -volume vol3
             Vserver Name: vs1
              Volume Name: vol3
              Quota State: on
   ```

```
           Scan Status: -
      Logging Messages: on
      Logging Interval: 1h
      Sub Quota Status: none
  Last Quota Error Message: -
Collection of Quota Errors: -
```

Note: You can also use the `quota.upgrade` event to track the beginning and completion of the quota upgrade process.

Commands to manage quota rules and quota policies

You can use the `volume quota policy rule` commands to configure quota rules, and use the `volume quota policy` commands and some `vserver` commands to configure quota policies.

Note: You can run the following commands only on FlexVol volumes.

Commands for managing quota rules

If you want to...	Use this command...
Create a new quota rule	`volume quota policy rule create`
Delete an existing quota rule	`volume quota policy rule delete`
Modify an existing quota rule	`volume quota policy rule modify`
Display information about configured quota rules	`volume quota policy rule show`

Commands for managing quota policies

If you want to...	Use this command...
Duplicate a quota policy and the quota rules it contains	`volume quota policy copy`
Create a new, blank quota policy	`volume quota policy create`
Delete an existing quota policy that is not currently assigned to a Storage Virtual Machine (SVM)	`volume quota policy delete`
Rename a quota policy	`volume quota policy rename`
Display information about quota policies	`volume quota policy show`
Assign a quota policy to an SVM	`vserver modify`

If you want to...	Use this command...
Display the name of the quota policy assigned to an SVM	`vserver show`

See the man page for each command for more information.

Related concepts

What quota rules, quota policies, and quotas are on page 120
Considerations for assigning quota policies on page 127
How you can view quota information on page 139

Commands to activate and modify quotas

You can use the `volume quota` commands to change the state of quotas and configure message logging of quotas.

If you want to...	Use this command...
Turn quotas on (also called *initializing* them)	`volume quota on`
Resize existing quotas	`volume quota resize`
Turn quotas off	`volume quota off`
Change the message logging of quotas, turn quotas on, turn quotas off, or resize existing quotas	`volume quota modify`

See the man page for each command for more information.

Related concepts

How quotas are activated on page 136
Why enforced quotas differ from configured quotas on page 141

Related tasks

Making minor changes to existing quota limits on page 153
Reinitializing quotas after making extensive changes on page 154

Using deduplication and data compression to increase storage efficiency

You can run deduplication and data compression together or independently on a FlexVol volume or an Infinite Volume to achieve optimal space savings. Deduplication eliminates the duplicate data blocks and data compression compresses the data blocks to reduce the amount of physical storage required.

How to set up efficiency operations

Depending on your storage environment setup, you can first estimate the space savings that can be achieved and then configure deduplication and data compression or only deduplication. You can run the efficiency operations on a volume by using schedules or policies.

You can use the space savings estimation tool to estimate the savings you can achieve in an existing environment. The space savings estimation tool can evaluate a maximum of 2 TB of data. You can download the space savings estimation tool from *communities.netapp.com/docs/DOC-18699*.

Configuring deduplication

Deduplication is a Data ONTAP feature that reduces the amount of physical storage space required by eliminating duplicate data blocks within a FlexVol volume or an Infinite Volume. You should not enable deduplication on the root volume.

You can decide to deduplicate only the new data that is written to the volume after enabling deduplication or both the new data and the data existing in the volume prior to enabling deduplication.

Related tasks

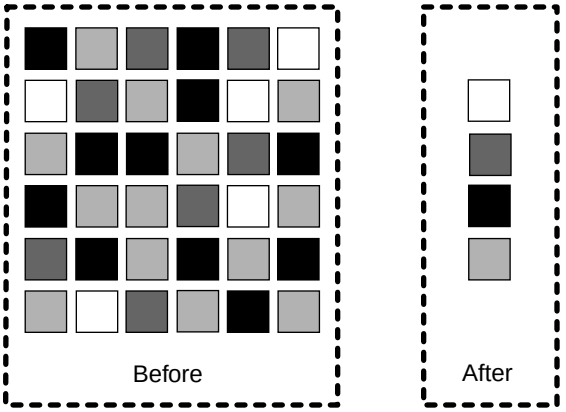
Enabling deduplication on a volume on page 162

How deduplication works

Deduplication operates at the block level within the entire FlexVol volume or an Infinite Volume, eliminating duplicate data blocks, and storing only unique data blocks.

Each block of data has a digital signature that is compared with all other signatures in a data volume. If an exact block signature match exists, a byte-by-byte comparison is done for all the bytes in the block. Only if all the bytes match, the duplicate block is discarded and its disk space is reclaimed resulting in no data loss.

Deduplication removes data redundancies, as shown in the following illustration:



Data ONTAP writes all data to a storage system in 4-KB blocks. When deduplication runs for the first time on a volume with existing data, it scans all the blocks in the volume and creates a digital fingerprint for each of the blocks. Each of the fingerprints is compared to all the other fingerprints within the volume. If two fingerprints are found to be identical, a byte-by-byte comparison is done for all data within the block. If the byte-by-byte comparison detects identical data, the pointer to the data block is updated, and the duplicate block is removed.

Note: When deduplication is run on a volume with existing data, it is best to configure deduplication to scan all the blocks in the volume for better space savings.

Deduplication runs on the active file system. Therefore, as additional data is written to the deduplicated volume, fingerprints are created for each new block and written to a change log file. For subsequent deduplication operations, the change log is sorted and merged with the fingerprint file, and the deduplication operation continues with fingerprint comparisons as previously described.

For more information about deduplication on an Infinite Volume, see the *Clustered Data ONTAP Infinite Volumes Management Guide*.

What deduplication metadata is

The deduplication metadata includes the fingerprint file and change logs. Fingerprints are the digital signatures for every 4-KB data block in a FlexVol volume or an Infinite Volume.

The deduplication metadata contains two change log files. When deduplication is running, the fingerprints of the new data blocks from one change log file are merged into the fingerprint file, and the second change log file stores the fingerprints of the new data that is written to the volume during the deduplication operation. The roles of the change log files are reversed when the next deduplication operation is run.

In Data ONTAP 8.0.1, the deduplication metadata is located within the aggregate. Starting with Data ONTAP 8.1, two copies of deduplication metadata are maintained per volume. A copy of the deduplication metadata resides in the volume and another copy is in the aggregate. The deduplication metadata in the aggregate is used as the working copy for all the deduplication operations. An additional copy of the deduplication metadata resides in the volume.

When a volume is moved, the deduplication metadata is also transferred with the volume. If the volume ownership changes, the next time deduplication is run, then the deduplication metadata which resides in the aggregate is created automatically by using the copy of deduplication metadata in the volume. This method is a faster operation than creating a new fingerprint file.

Starting with Data ONTAP 8.2, the fingerprints are stored for each physical block, this reduces the amount of space required to store the deduplication metadata.

Deduplication metadata can occupy up to 7 percent of the total physical data contained within the volume, as follows:

- In a volume, deduplication metadata can occupy up to 4 percent of the total amount of data contained within the volume.

  For an Infinite Volume, the deduplication metadata within an individual data constituent can occupy up to 4 percent of the total amount of data contained within the data constituent.
- In an aggregate, deduplication metadata can occupy up to 3 percent of the total physical data contained within the volume.

You can use the `storage aggregate show` command to check the available space in an aggregate and the `volume show` command to check the available space in a volume. For more information about these commands, see the man pages.

Example

A 2 TB aggregate has four volumes, each 400 GB in size, in the aggregate. You need three volumes to be deduplicated with varying savings percentage on each volume.

The space required in the different volumes for deduplication metadata is as follows:

- 2 GB [4% × (50% of 100 GB)] for a 100 GB of logical data with 50 percent savings
- 6 GB [4% × (75% of 200 GB)] for a 200 GB of logical data with 25 percent saving
- 3 GB [4% × (25% of 300 GB)] for a 300 GB of logical data with 75 percent savings

The aggregate needs a total of 8.25 GB [(3% × (50% of 100 GB)) + (3% × (75% of 200 GB)) + (3% × (25% of 300 GB)) = 1.5+4.5+2.25= 8.25 GB] of space available in the aggregate for deduplication metadata.

Guidelines for using deduplication

Deduplication runs as a system operation and consumes system resources when the deduplication operation is running on FlexVol volumes or Infinite Volumes.

If the data does not change often in a volume, it is best to run deduplication less frequently. If you run multiple concurrent deduplication operations on a storage system, these operations lead to a higher consumption of system resources. It is best to begin with fewer concurrent deduplication operations. Increasing the number of concurrent deduplication operations gradually enables you to better understand the impact on the system.

Note: It is best not to have multiple volumes with the volume size nearing the logical data limit of a volume with deduplication enabled.

Performance considerations for deduplication

Various factors affect the performance of deduplication. You should check the performance impact of deduplication in a test setup, including sizing considerations, before deploying deduplication in performance-sensitive or production environments.

The following factors can affect the performance of deduplication:

- The data access pattern (for example, sequential versus random access, the size, and pattern of the input and output)
- The amount of duplicate data, the amount of total data, and the average file size
- The nature of data layout in the volume
- The amount of changed data between deduplication operations
- The number of concurrent deduplication operations
- Hardware platform (system memory and CPU module)
- Load on the system
- Disk types (for example, ATA/FC and RPM of the disk)

For more information about performance aspects of deduplication, see *Technical Report 3966: Data Compression and Deduplication Deployment and Implementation Guide for Clustered Data ONTAP*.

Related information

Data Compression and Deduplication Deployment and Implementation Guide for Clustered Data ONTAP: media.netapp.com/documents/tr-3966.pdf

Enabling deduplication on a volume

You can enable deduplication on a FlexVol volume or an Infinite Volume to achieve storage efficiency by using the `volume efficiency on` command.

Before you begin

For a FlexVol volume, you must have verified that enough free space exists for deduplication metadata in the volumes and aggregates.

Step

1. Use the `volume efficiency on` command to enable deduplication.

Example

The following command enables deduplication on the volume VolA:

```
volume efficiency on -vserver vs1 -volume VolA
```

Related concepts

Managing volume efficiency operations using policies on page 166
Managing volume efficiency operations using schedules on page 170

Disabling deduplication on a volume

You can disable deduplication on a volume by using the `volume efficiency off` command.

About this task

If you have enabled data compression on the volume, running the `volume efficiency off` command disables data compression.

Steps

1. Use the `volume efficiency stop` command to stop any volume efficiency operation that is currently active on the volume.
2. Use the `volume efficiency off` command to disable the deduplication operation.

Example

The following command disables deduplication on the volume VolA:

```
volume efficiency off -vserver vs1 -volume VolA
```

Configuring data compression

Data compression is a Data ONTAP feature that enables you to reduce the physical capacity that is required to store data on storage systems by compressing the data blocks within a FlexVol volume or an Infinite Volume. You can use data compression only on volumes contained within 64-bit aggregates.

You can use data compression on primary, secondary, and tertiary storage tiers.

Related tasks

Enabling data compression on a volume on page 165

How data compression works

Data compression enables you to store more data in less space. Further, you can use data compression to reduce the time and bandwidth required to replicate data during volume SnapMirror transfers. Data compression can save space on regular files or LUNs.

However, storage system internal files, Windows NT streams, and volume metadata are not compressed.

Data compression works by compressing a small group of consecutive blocks known as a compression group. Data compression can be done in the following ways:

- Inline compression
  If inline compression is enabled on a volume, during subsequent data writes the compressible data is compressed and written to the volume. However, data which cannot be compressed or data bypassed by inline compression is written in the uncompressed format to the volume.
- Postprocess compression
  If postprocess compression is enabled on a volume, the new data writes to the volume which were not compressed initially (if inline compression is enabled), are rewritten as compressed data to the volume when postprocess compression is run. The postprocess compression operation runs as a low-priority background process.

If both inline and postprocess compression are enabled, then postprocess compression compresses only the blocks on which inline compression was not run. This includes blocks that were bypassed by inline compression such as small, partial compression group overwrites.

Note: You cannot enable data compression on the storage system root volumes or on the volumes that are contained within 32-bit aggregates.

For more information about compression on an Infinite Volume, see the *Clustered Data ONTAP Infinite Volumes Management Guide*.

How data compression detects incompressible data and saves system resources

Incompressible data detection enables you to check if a file is compressible and, for large files, if a compression group within a file is compressible. Enabling detection of incompressible data saves the system resources used by inline compression for compressing incompressible files or compression groups.

By default, for files under 500 MB (this value can be changed), inline compression checks if a compression group can be compressed. If incompressible data is detected within a compression group, then a flag is set for the file containing the compression group to indicate that the file is incompressible. During subsequent compression attempts, inline compression first checks if the incompressible data flag is set for the file. If the flag is set, then inline compression is not attempted on the file.

For files equal to or greater than 500 MB (this value can be changed), inline compression performs a quick check on the first 4-KB block of each compression group to determine if it can be compressed. If the 4-KB block cannot be compressed, the compression group is left uncompressed. However, if compression of the 4-KB block is successful, then compression is attempted on the whole compression group.

Postprocess compression runs on all files, irrespective of whether the file is compressible or not. If postprocess compression compresses at least one compression group in an incompressible file, then the incompressible data flag for that file is cleared. During the next compression attempt, inline compression can run on this file to achieve space savings.

For more information about enabling or disabling incompressible data detection and modifying the minimum file size to attempt quick check on a file, see the `volume efficiency modify` command man page.

Enabling data compression on a volume

You can enable data compression only on FlexVol volumes or Infinite Volumes that are contained within 64-bit aggregates. You can enable data compression to achieve space savings by using the `volume efficiency modify` command.

Before you begin

You must have enabled deduplication on the volume.

About this task

You can enable both inline and postprocess compression or only postprocess compression on a volume. Before enabling inline compression, you must enable postprocess compression on the volume.

Step

1. Use the `volume efficiency modify` command to enable data compression.

Examples

The following command enables postprocess compression on the volume VolA:

```
volume efficiency modify -vserver vs1 -volume VolA -compression true
```

The following command enables both postprocess and inline compression on the volume VolA:

```
volume efficiency modify -vserver vs1 -volume VolA -compression true -
inline-compression true
```

Related concepts

Managing volume efficiency operations using policies on page 166
Managing volume efficiency operations using schedules on page 170

Disabling data compression on a volume

You can disable data compression on a FlexVol volume or an Infinite Volume by using the `volume efficiency modify` command.

About this task

If you want to disable postprocess compression, you must first disable inline compression on the volume.

Steps

1. Use the `volume efficiency stop` command to stop any volume efficiency operation that is currently active on the volume.
2. Use the `volume efficiency modify` command to disable data compression.

Examples

The following command disables inline compression on the volume VolA:

```
volume efficiency modify -vserver vs1 -volume VolA -inline-compression
false
```

The following command disables both postprocess and inline compression on the volume VolA:

```
volume efficiency modify -vserver vs1 -volume VolA -compression false
-inline-compression false
```

Managing volume efficiency operations using policies

You can schedule deduplication or data compression to run on a FlexVol volume or Infinite Volume by defining volume efficiency policies. You can create job schedules that are enclosed within the efficiency policies to run deduplication or data compression for a specific duration.

A volume efficiency policy exists in the context of a Storage Virtual Machine (SVM).

The volume efficiency policies support only job schedules that are of type cron. For more information about creating job schedules of type cron, see the *Clustered Data ONTAP System Administration Guide for Cluster Administrators*.

Using volume efficiency priorities to prioritize efficiency operations

A functionality of Quality of Service (QoS) policies enable you to prioritize the volume efficiency operations that are running on a volume as a `best-effort` or `background` operation.

Scheduling the volume efficiency operations as `best-effort` or `background` allows you to maximize the utilization of system resources along with other system operations on a storage system.

You can assign a volume efficiency priority for every volume, irrespective of whether an efficiency policy is assigned to the volume.

For more information about assigning a priority for a volume efficiency operation, see the `volume efficiency policy modify` command man page and the technical report *TR-3966: Data Compression and Deduplication Deployment and Implementation Guide for Clustered Data ONTAP.*

Related information

Data Compression and Deduplication Deployment and Implementation Guide for Clustered Data ONTAP: media.netapp.com/documents/tr-3966.pdf

Creating a volume efficiency policy to run efficiency operations

You can create a volume efficiency policy to run deduplication or data compression followed by deduplication on a FlexVol volume or an Infinite Volume for a specific duration, and specify the job schedule using the `volume efficiency policy create` command.

Before you begin

You must have created a cron schedule using the `job schedule cron create` command. For more information about managing the cron schedules, see the *Clustered Data ONTAP System Administration Guide for Cluster Administrators*.

About this task

An SVM administrator with default predefined roles cannot manage the deduplication policies. However, the cluster administrator can modify the privileges assigned to an SVM administrator by using any customized roles. For more information about the SVM administrator capabilities, see the *Clustered Data ONTAP System Administration Guide for SVM Administrators*.

Step

1. Use the `volume efficiency policy create` command to create a volume efficiency policy.

Examples

The following command creates a volume efficiency policy named policy1 that runs an efficiency operation daily and for a duration of 10 hours:

```
volume efficiency policy create -vserver vs1 -policy policy1 -schedule
daily -duration 10
```

Assigning a volume efficiency policy to a volume

You can assign an efficiency policy to a volume to run deduplication or data compression operation by using the `volume efficiency modify` command.

About this task

If an efficiency policy is assigned to a SnapVault secondary volume, only the volume efficiency priority attribute is considered when running volume efficiency operations. The job schedules are ignored and the deduplication operation is run when incremental updates are made to the SnapVault secondary volume.

Step

1. Use the `volume efficiency modify` command to assign a policy to a volume.

Example

The following command assigns the volume efficiency policy named new_policy with volume VolA:

```
volume efficiency modify -vserver vs1 -volume VolA -policy new_policy
```

Modifying a volume efficiency policy

You can modify a volume efficiency policy to run deduplication or data compression for a different duration or change the job schedule using the `volume efficiency policy modify` command.

Step

1. Use the `volume efficiency policy modify` command to modify a volume efficiency policy.

Example

The following command modifies the volume efficiency policy named policy1 to run every hour:

```
volume efficiency policy modify -vserver vs1 -policy policy1 -schedule
hourly
```

Viewing a volume efficiency policy

You can view the volume efficiency policy name, schedule, duration, and description by using the `volume efficiency policy show` command.

About this task

When you run the `volume efficiency policy show` command from the cluster scope, the cluster-scoped policies are not displayed. However, you can view the cluster-scoped policies in the Storage Virtual Machine (SVM) context.

Step

1. Use the `volume efficiency policy show` command to view information about a volume efficiency policy.

   The output depends on the parameters you specify. For more information about displaying detailed view and other parameters, see the man page for this command.

Examples

The following command displays information about the policies created for the SVM vs1:

```
volume efficiency policy show -vserver vs1
```

The following command displays the policies for which the duration is set as 10 hours:

```
volume efficiency policy show -duration 10
```

Deleting a volume efficiency policy

You can delete a volume efficiency policy by using the `volume efficiency policy delete` command.

Before you begin

You must have ensured that the policy you want to delete is not associated with any FlexVol volume or an Infinite Volume.

Step

1. Use the `volume efficiency policy delete` command to delete a volume efficiency policy.

Example

The following command deletes a volume efficiency policy named policy1:

```
volume efficiency policy delete -vserver vs1 -policy policy1
```

Managing volume efficiency operations using schedules

You can manage how the efficiency operations run on a FlexVol volume or Infinite Volume by running efficiency operations manually, or using a schedule, or depending on the amount of new data written to the FlexVol volume or Infinite Volume.

You can also control how the efficiency operations run based on the following conditions:

- Use checkpoints or not
- Run efficiency operations on existing data or only new data
- Stop efficiency operations if required

You can use the `volume efficiency show` command with `schedule` as value for the `-fields` option to view the schedule assigned to the volumes.

Running efficiency operations manually

You can run efficiency operations manually on a FlexVol volume or an Infinite Volume by using the `volume efficiency start` command.

Before you begin

Depending on the efficiency operation you want to run manually, you must have enabled deduplication or both data compression and deduplication.

About this task

If only deduplication is enabled on a volume, then deduplication runs on the data. However, if deduplication and data compression are enabled on a volume, data compression is run initially followed by deduplication.

Deduplication is a background process that consumes system resources while it is running. If the data does not change often in a volume, it is best to run deduplication less frequently. Multiple concurrent deduplication operations running on a storage system lead to a higher consumption of system resources.

You can run a maximum of eight concurrent deduplication or data compression operations per node. If any more efficiency operations are scheduled, the operations are queued.

When you run deduplication or data compression on an Infinite Volume, a separate operation runs on each data constituent in the volume, with a maximum of eight concurrent operations per node.

Step

1. Use the `volume efficiency start` command to start the efficiency operation on a volume.

Examples

The following command allows you to manually start only deduplication or data compression followed by deduplication on the volume VolA:

```
volume efficiency start -vserver vs1 -volume VolA
```

The following command allows you to manually run the efficiency operations on the existing data in the volume VolA:

```
volume efficiency start -vserver vs1 -volume VolA -scan-old-data true
```

Running efficiency operations depending on the amount of new data written

You can modify the efficiency operation schedule to run deduplication or data compression when the number of new blocks written to the volume after the previous efficiency operation (performed manually or scheduled) exceeds a specified threshold percentage.

About this task

If the `schedule` option is set to `auto`, the scheduled efficiency operation runs when the amount of new data exceeds the specified percentage. The default threshold value is 20 percent. This threshold value is the percentage of the total number of blocks already processed by the efficiency operation.

Step

1. Use the `volume efficiency modify` command with the `auto@`*`num`* option to modify the threshold percentage value.

   `num` is a two-digit number to specify the percentage.

Example

The following command modifies the threshold percentage value to 30 percent for the volume VolA:

```
volume efficiency modify -vserver vs1 -volume -VolA -schedule auto@30
```

Using checkpoints to resume efficiency operation

You can use the checkpoints to periodically log the execution process of an efficiency operation. When an efficiency operation is stopped for any reason (such as system halt, system disruption, reboot, or because last efficiency operation failed or stopped) and checkpoint data exists, an efficiency operation can resume from the latest checkpoint file.

For an Infinite Volume, checkpoints are created on each individual data constituents. You cannot view the checkpoints on an Infinite Volume, but you can resume the operation.

A checkpoint is created at the end of each stage or substage of the efficiency operation. When an efficiency operation scans the existing data in a volume, a checkpoint is created approximately every hour.

Resuming an efficiency operation using the checkpoint option

You can resume an efficiency operation by using the `volume efficiency start` command with the checkpoint option.

About this task

If only deduplication is enabled on the volume, deduplication runs on the data. However, if deduplication and data compression are enabled, data compression runs on the existing data followed by deduplication.

You can view the details of the checkpoint for a volume by using the `volume efficiency show` command.

By default the efficiency operations resume from checkpoints. However, if a checkpoint corresponding to a previous efficiency operation (the phase when the `volume efficiency start -scan-old-data` command is run) is older than 24 hours, then the efficiency operation does not resume from the previous checkpoint automatically. In this case, the efficiency operation starts from the beginning. However, if you know that significant changes have not occurred in the volume since the last scan, you can force continuation from the previous checkpoint by using the `-use-checkpoint` option.

Step

1. Use the `volume efficiency start` command to resume an efficiency operation with the checkpoint option.

Example

The following command enables you to resume an efficiency operation by using the checkpoint option on the volume VolA:

```
volume efficiency start -vserver vs1 -volume VolA -scan-old-data true
-use-checkpoint true
```

Running efficiency operations manually on existing data

You can manually run the efficiency operations on the data that exists in the FlexVol volume or an Infinite Volume prior to enabling deduplication or data compression. Deduplication or data

compression followed by deduplication can be run by using the `volume efficiency start -scan-old-data` command.

About this task

If only deduplication is enabled on a volume, then deduplication runs on the data. However, If deduplication and data compression are enabled on a volume, data compression is run initially followed by deduplication.

When you run data compression on existing data, by default the data compression operation skips the data blocks that are shared by deduplication and the data blocks that are locked by Snapshot copies. If you choose to run data compression on shared blocks, then optimization is turned off, the fingerprint information is captured, and used for sharing again. You can change the default behavior of data compression when compressing existing data. For more information, see the *TR-3966: Data Compression and Deduplication Deployment and Implementation Guide for Clustered Data ONTAP*.

When you run deduplication or data compression on an Infinite Volume, a separate operation runs on each data constituent in the volume.

You can run a maximum of eight deduplication or data compression operations concurrently per node and the remaining operations are queued.

Step

1. Use the `volume efficiency start -scan-old-data` command to manually run deduplication or data compression followed by deduplication on the existing data.

Example

The following command allows you to manually run deduplication or data compression followed by deduplication on the existing data in the volume VolA:

```
volume efficiency start -vserver vs1 -volume VolA -scan-old-data true
```

Related information

Data Compression and Deduplication Deployment and Implementation Guide for Clustered Data ONTAP: media.netapp.com/documents/tr-3966.pdf

Modifying scheduling of efficiency operations

You can modify the scheduling of deduplication or data compression operation on a FlexVol volume or Infinite Volume by using the `volume efficiency modify` command. The configuration options of a schedule and volume efficiency policy are mutually exclusive.

Step

1. Use the `volume efficiency modify` command to modify the scheduling of deduplication or data compression operation on a volume.

Examples

The following command modifies the scheduling of efficiency operation for VolA to run at 11 p.m., Monday through Friday:

```
volume efficiency modify -vserver vs1 -volume VolA -schedule mon-
fri@23
```

Monitoring volume efficiency operations

You can monitor the progress of efficiency operations on a FlexVol volume or an Infinite Volume by viewing the status of the efficiency operations and the space savings achieved on the FlexVol volume or an Infinite Volume.

For more information about efficiency operations and space savings on an Infinite Volume, see the *Clustered Data ONTAP Infinite Volumes Management Guide*.

Viewing the status of efficiency operations

You can view whether deduplication or data compression is enabled on a FlexVol volume or an Infinite Volume and check the status, state, and progress of the efficiency operations on a FlexVol volume or individual data constituents in an Infinite Volume by using the `volume efficiency show` command.

Step

1. Use the `volume efficiency show` command to view the status of an efficiency operation on a volume.

Example

The following command displays the status of an efficiency operation on volume VolA:

```
volume efficiency show -vserver vs1 -volume VolA
```

If the efficiency operation is enabled on volume VolA and the operation is idle, then you can see the following in the system output:

```
cluster1::> volume efficiency show -vserver vs1 -volume VolA

                    Vserver Name: vs1
                     Volume Name: VolA
                     Volume Path: /vol/VolA
                           State: Enabled
                          Status: Idle
                        Progress: Idle for 00:03:20
                           .......
```

```
                                .......
                                .......
```

Viewing efficiency space savings

You can view the amount of space savings achieved through deduplication and data compression on a volume by using the `volume show` command.

About this task

The space savings in Snapshot copies are not included when calculating the space savings achieved on a volume. Using deduplication does not affect volume quotas. Quotas are reported at the logical level, and remain unchanged.

Step

1. Use the `volume show` command to view space savings achieved on a volume using deduplication and data compression.

Example

The following command enables you to view the space savings achieved by using deduplication and data compression on volume VolA:

`volume show -vserver vs1 -volume VolA`

```
cluster1::> volume show -vserver vs1 -volume VolA

                                Vserver Name: vs1
                                 Volume Name: VolA


..........


..........
            Space Saved by Storage Efficiency: 115812B
       Percentage Saved by Storage Efficiency: 97%
                  Space Saved by Deduplication: 13728B
             Percentage Saved by Deduplication: 81%
                 Space Shared by Deduplication: 1028B
                   Space Saved by Compression: 102084B
        Percentage Space Saved by Compression: 97%


..........


..........
```

Viewing efficiency statistics of a FlexVol volume

You can view the details of the efficiency operations run on a FlexVol volume by using the `volume efficiency stat` command.

Step

1. Use the `volume efficiency stat` command to view the statistics of efficiency operations on a FlexVol volume.

Example

The following command enables you to view the statistics of the efficiency operations on the volume VolA:

`volume efficiency stat -vserver vs1 -volume VolA`

```
cluster1::> volume efficiency stat -vserver vs1 -volume VolA

               Vserver Name: vs1
                Volume Name: VolA
                Volume Path: /vol/VolA
Inline Compression Attempts: 0
```

Stopping volume efficiency operations

You can stop a deduplication or postprocess compression operation by using the `volume efficiency stop` command. This command automatically generates a checkpoint.

Step

1. Use the `volume efficiency stop` command to stop an active deduplication or postprocess compression operation.

   If you specify the `-all` option, active and queued efficiency operations are aborted.

Examples

The following command stops the deduplication or postprocess compression operation that is currently active on volume VolA:

`volume efficiency stop -vserver vs1 -volume VolA`

The following command aborts both active and queued deduplication or postprocess compression operations on volume VolA:

`volume efficiency stop -vserver vs1 -volume VolA -all true`

Information about removing space savings from a volume

You can choose to remove the space savings achieved by running efficiency operations on a volume. You must ensure that you contact technical support before removing or undoing the space savings on a volume.

For more information about removing space savings from a volume, see the technical report *TR-3966: Data Compression and Deduplication Deployment and Implementation Guide for Clustered Data ONTAP*.

Related information

Data Compression and Deduplication Deployment and Implementation Guide for Clustered Data ONTAP: media.netapp.com/documents/tr-3966.pdf

Deduplication interoperability with Data ONTAP features

When you use deduplication, you should be aware of the features supported by deduplication and how they work with deduplication.

The following features are supported by deduplication:

- Snapshot copies
- Volume SnapMirror
- SnapRestore
- OnCommand Unified Manager server
- Volume copy
- Data compression
- FlexClone volumes
- HA pair
- DataMotion for Volumes
- SnapVault backup

You can enable extents on deduplicated volumes. You can perform read reallocation to improve the file layout and the sequential read performance of a deduplicated volume.

However, deduplicated volumes cannot be replicated using synchronous SnapMirror and semi-synchronous SnapMirror.

Related concepts

How fractional reserve works with deduplication on page 178
How Snapshot copies work with deduplication on page 178
How volume SnapMirror works with deduplication on page 178

How fractional reserve works with deduplication

If you are using deduplication for a volume with a fractional reserve setting of 0, there are additional configuration requirements if you need to ensure that your applications never receive an ENOSPC (out of space) error. For more information, see the documentation on setting fractional reserve.

Related concepts

How Snapshot copies work with deduplication

You can run deduplication only on the active file system. However, this data can get locked in Snapshot copies created before you run deduplication, resulting in reduced space savings.

To avoid conflicts between deduplication and Snapshot copies, you should follow these guidelines:

- Run deduplication before creating new Snapshot copies.
- Remove unnecessary Snapshot copies stored in deduplicated volumes.
- Reduce the retention time of Snapshot copies stored in deduplicated volumes.
- Schedule deduplication only after significant new data has been written to the volume.
- Configure appropriate reserve space for the Snapshot copies.
- If snap reserve is 0, you should turn off the schedule for automatic creation of Snapshot copies (which is the case in most LUN deployments).

How volume SnapMirror works with deduplication

You can use volume SnapMirror to replicate a deduplicated volume regardless of size of the volume and logical data in the volume.

When using volume SnapMirror with deduplication, you must consider the following information:

- You can enable deduplication on the source system, the destination system, or both systems.
- The shared blocks are transferred only once.
  Therefore, deduplication also reduces the use of network bandwidth.
- When the volume SnapMirror relationship is broken, the default deduplication schedule is applied at the destination storage system.

When configuring volume SnapMirror and deduplication, you should coordinate the deduplication schedule and the volume SnapMirror schedule. You should start volume SnapMirror transfers of a deduplicated volume after the deduplication operation is complete. This schedule prevents the sending of undeduplicated data and additional temporary metadata files over the network. If the temporary metadata files in the source volume are locked in Snapshot copies, these files consume extra space in the source and destination volumes.

How SnapRestore works with deduplication

The metadata created during a deduplication operation is located both in the FlexVol volume and in the aggregate. Therefore, when you initiate a SnapRestore operation on a volume, the metadata is restored to the volume and the restored data retains the original space savings.

After a SnapRestore operation is completed, if deduplication is enabled on the volume, any new data written to the volume continues to be deduplicated.

How OnCommand Unified Manager server works with deduplication

Deduplication is supported with the NetApp Management Console data protection capability, the NetApp Management Console provisioning capability, and Operations Manager in OnCommand Unified Manager server.

Deduplication and the NetApp Management Console data protection capability in OnCommand Unified Manager server

In releases earlier than OnCommand Unified Manager server, the NetApp Management Console data protection capability waits for an active deduplication operation to complete, before renaming the Snapshot copies. While the NetApp Management Console data protection capability waits, it does not allow clients to list the Snapshot copies or restore from them. Therefore, in releases prior to OnCommand Unified Manager server, the use of deduplication with the NetApp Management Console data protection capability is not optimal.

However, this limitation is removed in OnCommand Unified Manager server.

Deduplication and the NetApp Management Console provisioning capability in OnCommand Unified Manager server

With the NetApp Management Console provisioning capability in OnCommand Unified Manager server, you can enable the provisioning policies to support all three modes of deduplication, namely, on-demand deduplication, automated deduplication, and scheduled deduplication.

For more information about using deduplication with the NetApp Management Console provisioning capability and the NetApp Management Console data protection capability, see the *Provisioning Manager and Protection Manager Guide to Common Workflows for Administrators*.

Deduplication and Operations Manager in OnCommand Unified Manager server

You can perform deduplication operations from Operations Manager in OnCommand Unified Manager server.

You can configure deduplication on the system and generate reports or graphs summarizing space savings for file and LUN clones.

For more information about using deduplication with Operations Manager, see the *Operations Manager Administration Guide*.

Related information

Documentation on the NetApp Support Site: support.netapp.com

How volume copy works with deduplication

When deduplicated data is copied by using the `volume copy` command, the copy of the data at the destination inherits all the deduplication attributes and storage savings of the source data.

The metadata created during a deduplication operation (fingerprint files and change log files) is located inside the FlexVol volume and in the active copy within the aggregate. Therefore, when you run the volume copy operation on a FlexVol volume, the fingerprint files and change log files are copied to the destination volume. After a volume copy operation, if deduplication is enabled on the destination volume, any new data written to the volume continues to be deduplicated.

The first deduplication operation that is run on the copied FlexVol volume automatically reconstructs the deduplication metadata in the aggregate based on the copy that is available in the volume.

How deduplication works with data compression

When both data compression and deduplication are enabled on a FlexVol volume, the data is first compressed and then deduplicated. Depending on the type of data, the combined savings can yield higher savings than running deduplication alone.

How FlexClone volumes work with deduplication

Deduplication is supported on FlexClone volumes. The FlexClone volume of a deduplicated volume is a deduplicated volume. The cloned volume inherits the deduplication configuration of the parent volume (for example, deduplication schedules).

The metadata created during a deduplication operation (fingerprint files and change log files) is cloned. This metadata is located both in the FlexVol volume and in the aggregate.

If you run deduplication on a clone volume, the clone is deduplicated, but the parent volume remains non-deduplicated.

To run deduplication manually for all new data in the cloned volume, you should use the `volume efficiency start` command.

When a cloned volume is split from the parent volume, deduplication of all data in the clone that was part of the parent volume is removed after the volume-split operation. However, if deduplication is running on the clone volume, the data is deduplicated in the subsequent deduplication operation.

How HA pairs work with deduplication

Starting with Data ONTAP 8.1, deduplication operations can be run on volumes from either of the nodes during takeover in an HA pair. The maximum number of concurrent deduplication operations allowed on each node of an HA pair is eight.

If one of the nodes fails, the other node takes over the deduplication operations managed by the failed node. In takeover mode, the working node continues with the deduplication operations. The working node can start deduplication operations on volumes that belong to the failed node. When the working node is managing the deduplication operations on volumes that belong to both the nodes, the maximum number of concurrent deduplication operations is still eight.

How DataMotion for Volumes works with deduplication

Deduplication savings on a FlexVol volume are retained when the volume is moved by using the DataMotion for Volumes (volume move) operation. If deduplication operations are running when a volume move operation is active, then these operations are stopped shortly before the final cutover is complete.

After the volume move is complete, the efficiency operations cannot be resumed from the previous checkpoint and the efficiency operations start from the beginning.

If you try to nondisruptively move a FlexVol volume that has deduplication operations running, then the deduplication operation is aborted.

How SnapVault backup works with deduplication

You can logically replicate a deduplication-enabled primary volume to a SnapVault secondary volume. You can preserve the space savings achieved on the primary volume during replication or enable deduplication only on the SnapVault secondary volume.

During replication, the space savings achieved by running deduplication on the primary volume are preserved over the network and when the data is written to the SnapVault secondary volume.

You might enable deduplication on the SnapVault secondary volumes in the following scenarios:

- Deduplication cannot be configured on the primary volume, but space savings is required on the SnapVault secondary volumes where the retention time is longer for Snapshot copies.
- Deduplication could not be completed on the primary volume before the replication transfer to the SnapVault secondary volume starts.

When using SnapVault backup with deduplication, you must observe the following guidelines:

- You cannot run efficiency operations manually on the SnapVault secondary volumes.
  However, you can run the `volume efficiency start -scan-old-data` command on the secondary volume.
- If deduplication is configured to run on the SnapVault secondary volumes, then deduplication is triggered each time the replication transfer from the primary volume to the SnapVault secondary volume is completed.

- During the data transfer from the primary to the SnapVault secondary volume, you cannot modify the state of deduplication on the SnapVault secondary volume.
- When you initiate a volume copy or volume clone operation on a SnapVault secondary volume, it does not replicate any changes made to the volume since the last Snapshot copy was created.

For more information about volumes in a SnapVault relationship, see the *Clustered Data ONTAP Data Protection Guide*.

How virtual machine alignment works with deduplication

The virtual machine alignment feature can be used with Virtual Storage Console (VSC) for VMware vSphere to nondisruptively fix a misaligned Virtual Machine Disk (VMDK). For users with a large number of misaligned VMs, this feature can help improve both system performance and increased deduplication savings.

The virtual machine alignment is a volume level setting which needs to be specified when the volume is created. The virtual machine alignment attributes ensure that the virtual machine disks created within the volume, align with underlying layers of the storage system.

Note: The virtual machine alignment feature can be used only on NFS based storage systems.

For more information about installing and administering the Virtual Storage Console for VMware vSphere, see the *Virtual Storage Console for VMware vSphere Installation and Administration Guide*.

Data compression interoperability with Data ONTAP features

When you use data compression, you should be aware of the features supported by data compression and how they work with data compression.

The following features are supported by data compression:

- Snapshot copies
- Volume SnapMirror
- Tape backup
- Volume-based SnapRestore
- Single file SnapRestore
- Volume copy
- Deduplication
- FlexClone volumes
- FlexClone files
- HA pair
- Flash cache cards
- DataMotion for Volumes

- Flash Pool aggregates
- SnapVault backup

Compressed volumes cannot be replicated using synchronous SnapMirror and semi-synchronous SnapMirror. Read reallocation and extents are not supported on compressed volumes.

Related concepts

How fractional reserve works with data compression on page 183
How Snapshot copies work with data compression on page 183
How volume SnapMirror works with data compression on page 184
How tape backup works with data compression on page 184
How volume-based SnapRestore works with data compression on page 185
How single file SnapRestore works with data compression on page 185
How volume copy works with data compression on page 185
How deduplication works with data compression on page 186
How FlexClone volumes work with data compression on page 186
How FlexClone files work with data compression on page 186
How HA pairs work with data compression on page 186
How Flash cache cards work with data compression on page 186
How DataMotion for Volumes works with data compression on page 186
How Flash Pool aggregates work with data compression on page 187
How SnapVault backup works with data compression on page 187

How fractional reserve works with data compression

If you are using data compression for a volume with a fractional reserve setting of 0, there are additional configuration requirements if you need to ensure that your applications never receive an ENOSPC (out of space) error. For more information, see the documentation on setting fractional reserve.

Related concepts

Considerations for setting fractional reserve for FlexVol volumes on page 32

How Snapshot copies work with data compression

When you run data compression in the default mode after a Snapshot copy is created, the existing data that is locked by the Snapshot copy is compressed.

Snapshot copies lock blocks of data that cannot be freed until the Snapshot copy expires or is deleted. On any volume on which data compression is enabled, when a Snapshot copy of the data is created, any subsequent changes to the data temporarily requires additional disk space, until the Snapshot copy is deleted or expires.

How volume SnapMirror works with data compression

Because volume SnapMirror operates at the physical block level, when data compression is enabled on the source storage system, the data remains compressed when it is replicated to the destination storage system. This operation can significantly reduce the amount of required network bandwidth during replication.

When using volume SnapMirror with data compression, you must observe the following guidelines:

- For SnapMirror transfer to occur, the source and destination volumes must be contained in 64-bit aggregates and the destination storage system must be running the same or later version of Data ONTAP.

  If the source storage system is running on Data ONTAP 8.1, then the destination storage system must be running on Data ONTAP 8.1 or later.
- You can enable, run, and manage data compression only from the primary storage system.

  However, the FlexVol volume in the secondary storage system inherits all the data compression attributes and storage savings through the volume SnapMirror transfer.
- If you plan to compress existing data in the disk with the `-shared-blocks` or `-snapshot-blocks` options on a FlexVol volume that has data blocks locked in Snapshot copies and has existing volume SnapMirror relationships, then this operation might result in a large transfer of data blocks.

  You can specify these options only from the advanced privilege level.

  The data compression operation rewrites data as new compressed blocks and these blocks are transferred in the next incremental transfer.

For more information about volume SnapMirror, see the *Clustered Data ONTAP Data Protection Guide*.

Related information

Documentation on the NetApp Support Site: support.netapp.com

How tape backup works with data compression

When you back up compressed data using NDMP, data from the source volume is written to the tape in an uncompressed format. Therefore, if you want to regain space savings on a volume restored from tape, you must enable compression on the volume before initiating restore.

You can restore the compressed data from a tape to a destination volume and the saving will be retained. You must enable inline compression only if you want the new data that is written from the client to be compressed on the restored volume.

How volume-based SnapRestore works with data compression

When you initiate a volume-based SnapRestore operation on a FlexVol volume that contains compressed data, the compression setting is restored to that of the Snapshot copy and the restored data retains the original space savings of the Snapshot copy.

For more information about volume-based SnapRestore, see the *Clustered Data ONTAP Data Protection Guide*.

Related information

Documentation on the NetApp Support Site: support.netapp.com

How single file SnapRestore works with data compression

When you initiate a single file SnapRestore operation, the data is restored from the Snapshot copy to the active file system and the original space savings are restored.

If incompressible data detection is enabled on the volume, then the incompressible data flag is restored to the active file system from the Snapshot copy.

For more information about single file SnapRestore, see the *Clustered Data ONTAP Data Protection Guide*.

Related information

Documentation on the NetApp Support Site: support.netapp.com

How volume copy works with data compression

When a volume with compressed data is copied to the destination system by using the `vol copy` command, the copy of the data at the destination system inherits all the compression attributes and storage savings of the original data.

For more information about volume copy, see the *Clustered Data ONTAP Data Protection Guide*.

Related information

Documentation on the NetApp Support Site: support.netapp.com

How deduplication works with data compression

When both data compression and deduplication are enabled on a FlexVol volume, the data is first compressed and then deduplicated. Depending on the type of data, the combined savings can yield higher savings than running deduplication alone.

How FlexClone volumes work with data compression

If you split a FlexClone volume from the parent volume, then the new volume inherits data compression attributes from the parent volume. The attributes inherited indicate whether deduplication, postprocess compression, and inline compression are enabled. The space savings achieved in the parent volume are inherited by the new volume.

If you create a FlexClone volume when the decompression operation is active on the parent volume, then the decompression operation does not run on the cloned volume.

How FlexClone files work with data compression

You can run data compression on a FlexVol volume that contains FlexClone files. Only fully cloned files can be created on FlexVol volumes that have data compression enabled. Partially cloned files cannot be created on FlexVol volumes that have data compression enabled.

If incompressible data detection is enabled on the volume, then the incompressible flag is inherited by the cloned file.

How HA pairs work with data compression

You can enable data compression in an HA pair. If one of the nodes fails, the other node takes over the operations of the failed node. In takeover mode, the working node continues to perform the data compression operations.

For more information about HA pairs, see the *Data ONTAP High Availability and MetroCluster Configuration Guide for 7-Mode*.

Related information

Documentation on the NetApp Support Site: support.netapp.com

How Flash cache cards work with data compression

Although data compression and Flash cache cards work in the same storage system, the read performance of the compressed data remains the same with or without Flash cache cards.

How DataMotion for Volumes works with data compression

Data compression savings on a FlexVol volume are retained when the volume is moved using the DataMotion for Volumes (volume move) operation. For a volume move operation to be successful,

the destination volumes must be contained within a 64-bit aggregate and both the source and destination volumes must run the same version of Data ONTAP.

How Flash Pool aggregates work with data compression

Data compression is supported on Flash Pool aggregates, but the compressed blocks are not read or write cached in the solid-state disk (SSD) of the Flash Pool aggregates. For a volume that has data compression enabled, only the blocks which not compressed are read or write cached.

How SnapVault backup works with data compression

You can logically replicate a SnapVault primary volume to a SnapVault secondary volume. During replication, the space savings achieved through data compression on the primary volume is preserved only if data compression is not configured on the SnapVault secondary volume.

If data compression is not enabled on the SnapVault secondary volume, the space savings achieved through data compression on the primary volume is preserved over the network during transfers and when the data is written to the secondary volume.

However, if the SnapVault secondary volume is configured to run compression, then uncompressed data is transferred over the network and written to the secondary volume. Even during incremental updates, uncompressed data is transferred over the network.

When using SnapVault backup with data compression, you must observe the following guidelines:

- When the data transfer between the primary and SnapVault secondary volume is in progress, you cannot modify the state of data compression on the SnapVault secondary volume.
- Although data compression is disabled on the SnapVault secondary volume, you can enable deduplication on the SnapVault secondary volume.

  Deduplication is triggered each time the replication transfer from the primary volume to the SnapVault secondary volume is completed.
- If you require data compression savings on the SnapVault secondary volume and not on the primary volume, then you can use inline compression on the SnapVault secondary volume.

  However, if you want to run data compression on the primary and the SnapVault secondary volume, then the space savings achieved on the primary volume are not preserved during the data transfer and when the data is written to the secondary volume.
- If you need to run only inline compression without running deduplication on the SnapVault secondary volume, then you must change the schedule for the SnapVault secondary volume to `manual`.
- If you disable data compression on a SnapVault secondary volume, then before the subsequent transfer to the secondary volume is initiated, a SnapRestore operation is performed on the secondary volume.

  However, if data compression was enabled on the secondary volume when the previous Snapshot copy was created, then the SnapRestore operation is not performed. Although now data compression is disabled on the SnapVault secondary volume, uncompressed data is written to the secondary volume.

- When you initiate a volume copy or volume clone operation on a SnapVault secondary volume, it does not replicate any changes made to the volume since the last Snapshot copy was created.

You cannot run efficiency operations manually on the SnapVault secondary volumes. However, you can run the `volume efficiency start -scan-old-data` command on the secondary volume.

For more information about volumes in a SnapVault relationship, see the *Clustered Data ONTAP Data Protection Guide*.

Storage limits

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

Limits are listed in the following sections:

Volume limits

Limit	Native storage	Storage arrays	Virtual storage (Data ONTAP-v)	Notes
Array LUNs Minimum size for root volume	N/A	Model-dependent	N/A	See the *Hardware Universe*.
Files Maximum size	16 TB	16 TB	16 TB	
Files Maximum per volume	Volume size dependent, up to 2 billion	Volume size dependent, up to 2 billion	Volume size dependent, up to 2 billion	2 billion = 2 × 10 to the 9th power.
FlexCache volumes Maximum per node	100	100	N/A	
FlexClone volumes Hierarchical clone depth	499	499	499	The maximum depth of a nested hierarchy of FlexClone volumes that can be created from a single FlexVol volume.
FlexVol volumes Maximum per node	Model-dependent	Model-dependent	200	See the *Hardware Universe*.

Limit	Native storage	Storage arrays	Virtual storage (Data ONTAP-v)	Notes
FlexVol volumes Maximum per node per SVM	Model-dependent	Model-dependent	200	This limit applies only in SAN environments. See the *Clustered Data ONTAP SAN Configuration Guide*.
FlexVol volumes Minimum size	20 MB	20 MB	20 MB	
FlexVol volumes (32-bit) Maximum size	16 TB	16 TB	16 TB	
FlexVol volumes (64-bit) Maximum size	Model-dependent	Model-dependent	Model-dependent	See the *Hardware Universe*.
FlexVol root volumes Minimum size	Model-dependent	Model-dependent	Model-dependent	See the *Hardware Universe*.
LUNs Maximum per node	Model-dependent	Model-dependent	1,024	See the *Clustered Data ONTAP SAN Configuration Guide*.
LUNs Maximum per cluster	Model-dependent	Model-dependent	1,024	See the *Clustered Data ONTAP SAN Configuration Guide*.
LUNs Maximum per volume	Model-dependent	Model-dependent	512	See the *Clustered Data ONTAP SAN Configuration Guide*.
LUNs Maximum size	16 TB	16 TB	16 TB	
Qtrees Maximum per FlexVol volume	4,995	4,995	4,995	

Limit	Native storage	Storage arrays	Virtual storage (Data ONTAP-v)	Notes
Snapshot copies Maximum per FlexVol volume or Infinite Volume	255	255	255	The use of certain Data ONTAP capabilities could reduce this limit. For more information, see the *Clustered Data ONTAP Data Protection Guide*.
Volumes Maximum per cluster for NAS	12,000	12,000	200	Infinite Volumes do not count against this limit, but their constituent volumes do.
Volumes Maximum per cluster with SAN protocols configured	Model-dependent	Model-dependent	200	See the *Clustered Data ONTAP SAN Configuration Guide*. Infinite Volumes do not count against this limit, but their constituent volumes do.

FlexClone file and FlexClone LUN limits

Limit	Native storage	Storage arrays	Virtual storage (Data ONTAP-v)	Notes
Maximum per file or LUN	32,767	32,767	32,767	If you try to create more than 32,767 clones, Data ONTAP automatically creates a new physical copy of the parent file or LUN. This limit might be lower for FlexVol volumes that use deduplication.

Limit	Native storage	Storage arrays	Virtual storage (Data ONTAP-v)	Notes
Maximum total shared data per FlexVol volume	640 TB	640 TB	100 TB	

Copyright information

Trademark information

How to send your comments

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

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

You can also contact us in the following ways:

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

Index

D

E

F

N

O

P

Q

R

S

T

U

V

W