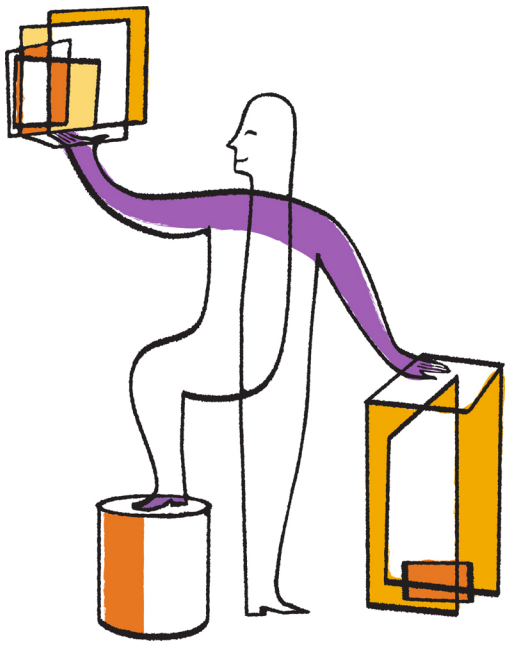
Updated for 8.3.1

Clustered Data ONTAP® 8.3

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: 215-10106_A0
June 2015

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 16
Using qtrees to partition your FlexVol volumes on page 81
Using deduplication and data compression to increase storage efficiency on page 119
Using quotas to restrict or track resource usage on page 85

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, 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:

- Using FlexClone technology, create a clone of the volume quickly and without having to duplicate the entire volume.

- Using deduplication and compression technologies, reduce the space requirements of the volume.
- Create a Snapshot copy of the volume for data protection purposes.
- Using quotas, limit the amount of space a user, group, or qtree can use in the volume.
- 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 thinly provisioned volume that can be bigger than the physical storage currently available to it.

Related concepts

Using FlexClone volumes to create efficient copies of your FlexVol volumes on page 60
Using FlexClone files and FlexClone LUNs to create efficient copies of files and LUNs on page 66
Configuring deduplication on page 119
Configuring data compression on page 123
Moving and copying volumes (cluster administrators only) on page 56

Related tasks

Configuring volumes to automatically provide more space when they are full on page 30
Configuring volumes to automatically grow and shrink their size on page 32

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?
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

Feature	Access type (read-write or read-only)	Automatically mounted?	Full copy or shared blocks?	Location	Instantaneous or 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
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 FlexClone volumes to create efficient copies of your FlexVol volumes on page 60
Moving and copying volumes (cluster administrators only) on page 56

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	Model-dependent	Up to 20 PB	For information about the maximum size of FlexVol 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*.
Maximum number per node	Model-dependent	Model-dependent	For more information, see the *Hardware Universe*.
SAN protocols supported	Yes	No	

Volume capability or feature	FlexVol volumes	Infinite Volumes	Notes
File access protocols supported	NFS, CIFS	NFS, CIFS	
Deduplication	Yes	Yes	
Compression	Yes	Yes	
FlexClone 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 149

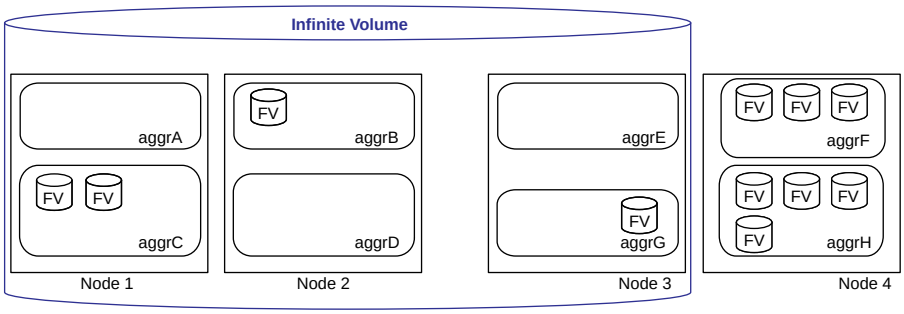
How FlexVol volumes and Infinite Volumes share aggregates

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.

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.

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.

Related information

Clustered Data ONTAP 8.3 Physical Storage Management Guide
Clustered Data ONTAP 8.3 System Administration Guide

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 FlexClone volumes, the default language is the language of the parent volume.

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`.

Related information

Clustered Data ONTAP 8.3 File Access Management Guide for NFS
Clustered Data ONTAP 8.3 File Access Management Guide for CIFS

How you use space management capabilities

You use Data ONTAP space management capabilities to maximize your data availability without overspending for storage. You accomplish this through file, LUN, and volume configuration options, and alerts that tell you when you need to take action to prevent your free space from being exhausted.

Because of powerful Data ONTAP block-sharing technologies such as Snapshot copies, you need to provide more free space than the size of the user data that has been written to a volume or LUN. The amount of free space that you need to provide depends on your application environment and your storage management practices.

You can use the following configuration options and capabilities to achieve the correct balance between data availability and storage utilization for your environment:

- Thin provisioning
  Data ONTAP provides different ways to thinly provision your storage objects.
- Volume provisioning options
  You can choose the volume provisioning option that provides the right balance of space utilization and write guarantees.
- File and LUN space reservation
  Attributes of the file or LUN that enable you to reserve free space for that specific object.
- Volume guarantees and fractional reserve
  Volume attributes that enable you to configure the way Data ONTAP reserves space for the volume.
- Automatically providing free space for full volumes
  Volume attributes that enable Data ONTAP to automatically provide more space to a volume that is getting full.
- Volume autosize
  If free space in the aggregate is more important than free space in a particular volume associated with that aggregate, you can configure the volume to give space back to the aggregate when it does not need it.
- Fullness and overallocation alerts
  Alerts that tell you when a volume or aggregate is getting full so that you can take action to prevent free space from being exhausted.
- Space usage management
  Ways to determine and affect how the used space in your volumes and aggregates is being used.

Related concepts

How volume guarantees work for FlexVol volumes on page 26
How file and LUN space reservation works on page 25
Considerations for setting fractional reserve on page 29

Related tasks

Configuring volumes to automatically provide more space when they are full on page 30

How Data ONTAP provides the ability to use thin provisioning

How you configure a storage object to use thin provisioning and how that object behaves as a result depends on the type of storage object. You can create thinly provisioned volumes, files, and LUNs.

What a thin-provisioned volume is

A thin-provisioned volume is a volume for which storage is not set aside up-front. Instead, the storage for the volume is allocated as it is needed.

You create a thin-provisioned FlexVol volume by setting its guarantee to **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 only as data is written to the volume.

What a thin-provisioned LUN is

LUNs are storage objects you use to provide access to storage using SAN protocols. The definition of a "thin-provisioned LUN" changes depending on the context. The T10 SCSI (SAN) standard uses one definition, and historically, NetApp has used a different definition.

SCSI thin-provisioned LUNs

The T10 SCSI standard defines two types of LUNs: thin provisioned and fully provisioned. Data ONTAP supports both types of T10 standard LUNs.

SCSI thin provisioning (sometimes called *T10 thin provisioning*) is a set of SCSI features and capabilities enabled by Data ONTAP. These SCSI features must be supported by your SCSI host software. SCSI thin provisioning enables host applications to support SCSI features, including LUN space reclamation and LUN space monitoring capabilities for blocks environments.

If your host software supports SCSI thin provisioning, you can use it with space-reserved and non-space-reserved LUNs, and with any volume provisioning type. See the documentation for your host software for more information about the SCSI thin provisioning capabilities it provides.

You use the Data ONTAP **`space-allocation`** setting to enable SCSI thin provisioning on a LUN.

NetApp thin-provisioned (non-space-reserved) LUNs

Historically, NetApp has used the term "thin-provisioned LUN" to mean a LUN for which space reservation is disabled (a non-space-reserved LUN). A non-space-reserved LUN shares some important characteristics with a thinly provisioned volume: its storage is allocated as it is used rather than at creation time, and its containing storage object can be overcommitted. The term "non-space-reserved LUN" is used for this configuration.

You use the Data ONTAP **`space-reserve`** setting to configure space reservation on a LUN.

Files can also be space-reserved or non-space reserved. However, non-space-reserved files are not generally referred to as thin-provisioned files.

Related information

Clustered Data ONTAP 8.3 SAN Administration Guide

What it means to overcommit a storage object

When you do not allocate the storage for a FlexVol volume or LUN up front, it enables you to overcommit the storage object that supplies its storage. Overcommitting a storage object can increase your storage efficiency but it also requires that you take an active role in monitoring your free space to prevent writes from failing due to lack of space.

A storage object is said to be overcommitted if the objects it supplies storage to are collectively larger than the amount of physical storage it can currently supply. For example, suppose you have three FlexVol volumes with volume guarantees of **none** associated with a 100 TB aggregate. If each of the FlexVol volumes has a nominal size of 40 TB, the aggregate is overcommitted. Each of the volumes could continue to receive data, providing that collectively, their physical storage requirement is not

more than 100 TB. This is only possible if the volumes have a volume guarantee of **none** (they are thinly provisioned). If all of the volumes had a volume guarantee of **volume**, you could not create the third volume.

Similarly, you can overcommit a volume that contains more than one LUN, as long as the LUNs are not space-reserved.

The ability to overcommit the supplying storage object is the promise offered by not preallocating storage for an object up front, but overcommitment requires you to manage the supply of physical storage resources carefully to avoid running out of free space. This is true in any configuration, but if your supplying storage object is overcommitted, you can run out of free space by issuing a write to space that appears to have been allocated already.

Considerations for using thin-provisioned FlexVol volumes

You can configure your thin-provisioned 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. However, you should understand how thin-provisioned volumes can act differently from fully provisioned volumes.

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 FlexVol 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.

Related information

NetApp Technical Report 3965: NetApp Thin Provisioning Deployment and Implementation Guide Data ONTAP 8.1 (7-Mode)

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

Volume provisioning options

Data ONTAP provides three basic volume provisioning options: thick provisioning, thin provisioning, and semi-thick provisioning. Each option uses different ways to manage the volume space and the space requirements for Data ONTAP block sharing technologies. Understanding how the options work enables you to choose the best option for your environment.

Thick provisioning for volumes

When a thick-provisioned volume is created, Data ONTAP sets aside enough storage from the aggregate to ensure that any block in the volume can be written to at any time. When you configure a volume to use thick provisioning, you can employ any of the Data ONTAP storage efficiency capabilities, such as compression and deduplication, to offset the larger upfront storage requirements.

Thin provisioning for volumes

When a thinly provisioned volume is created, Data ONTAP does not reserve any extra space when the volume is created. As data is written to the volume, the volume requests the storage it needs from the aggregate to accommodate the write operation. Using thin-provisioned volumes enables you to

overcommit your aggregate, which introduces the possibility of the volume not being able to secure the space it needs when the aggregate runs out of free space.

Semi-thick provisioning for volumes

When a volume using semi-thick provisioning is created, Data ONTAP sets aside storage space from the aggregate to account for the volume size. If the volume is running out of free space because blocks are in use by block-sharing technologies, Data ONTAP makes an effort to delete protection data objects (Snapshot copies and FlexClone files and LUNs) to free up the space they are holding, As long as Data ONTAP can delete the protection data objects fast enough to keep pace with the space required for overwrites, the write operations continue to succeed. This is called a “best effort” write guarantee.

You cannot employ storage efficiency technologies such as deduplication and compression on a volume that is using semi-thick provisioning.

The following table summarizes the major differences in how the three volume provisioning options can be used:

Volume provisioning	LUN/file space reservation	Overwrites	Protection data[2]	Storage efficiency[3]
Thick	Supported	Guaranteed[1]	Guaranteed	Supported
Thin	No effect	None	Guaranteed	Supported
Semi-thick	Supported	Best effort[1]	Best effort	Not supported

Notes

1. The ability to guarantee overwrites or provide a best-effort overwrite assurance requires that space reservation is enabled on the LUN or file.
2. Protection data includes Snapshot copies, and FlexClone files and LUNs marked for automatic deletion (backup clones).
3. Storage efficiency includes deduplication, compression, any FlexClone files and LUNs not marked for automatic deletion (active clones), and FlexClone subfiles (used for Copy Offload).

Recommended volume and file or LUN configuration combinations

There are specific combinations of FlexVol volume and file or LUN configurations you can use, depending on your application and administration requirements. Understanding the benefits and costs of these combinations can help you determine the right volume and LUN configuration combination for your environment.

The following volume and LUN configuration combinations are recommended:

- Space-reserved files or LUNs with thick volume provisioning
- Non-space-reserved files or LUNs with thin volume provisioning
- Space-reserved files or LUNs with semi-thick volume provisioning

You can use SCSI thin provisioning on your LUNs in conjunction with any of these configuration combinations.

Space-reserved files or LUNs with thick volume provisioning

Benefits:

- All write operations within space-reserved files are guaranteed; they will not fail due to insufficient space.

- There are no restrictions on storage efficiency and data protection technologies on the volume.

Costs and limitations:

- Enough space must be set aside from the aggregate up front to support the thickly provisioned volume.
- Space equal to twice the size of the LUN is allocated from the volume at LUN creation time.

Non-space-reserved files or LUNs with thin volume provisioning

Benefits:

- There are no restrictions on storage efficiency and data protection technologies on the volume.
- Space is allocated only as it is used.

Costs and restrictions:

- Write operations are not guaranteed; they can fail if the volume runs out of free space.
- You must manage the free space in the aggregate effectively to prevent the aggregate from running out of free space.

Space-reserved files or LUNs with semi-thick volume provisioning

Benefits:

Less space is reserved up front than for thick volume provisioning, and a best-effort write guarantee is still provided.

Costs and restrictions:

- Write operations can fail with this option.
  You can mitigate this risk by properly balancing free space in the volume against data volatility.
- You cannot rely on retention of data protection objects such as Snapshot copies and FlexClone files and LUNs.
- You cannot use Data ONTAP block-sharing storage efficiency capabilities that cannot be automatically deleted, including deduplication, compression, and ODX/Copy Offload.

Configuration settings for space-reserved files or LUNs with thick-provisioned volumes

This FlexVol volume and file or LUN configuration combination provides the ability to use storage efficiency technologies and does not require you to actively monitor your free space, because sufficient space is allocated up front.

The following settings are required to configure a space-reserved file or LUN in a volume using thick provisioning:

Volume setting	Value
Guarantee	Volume
Fractional reserve	100
Snapshot reserve	Any
Snapshot autodelete	Optional
Autogrow	Optional; if enabled, aggregate free space must be actively monitored.

File or LUN setting	Value
Space reservation	Enabled

Technology restrictions

None

Additional considerations

None

Configuration settings for non-space-reserved files or LUNs with thin-provisioned volumes

This FlexVol volume and file or LUN configuration combination requires the smallest amount of storage to be allocated up front, but requires active free space management to prevent errors due to lack of space.

The following settings are required to configure a non-space-reserved files or LUN in a thin-provisioned volume:

Volume setting	Value
Guarantee	None
Fractional reserve	0
Snapshot reserve	Any
Snapshot autodelete	Optional
Autogrow	Optional

File or LUN setting	Value
Space reservation	Disabled

Technology restrictions

None

Additional considerations

When the volume or aggregate runs out of space, write operations to the file or LUN can fail.

If you do not want to actively monitor free space for both the volume and the aggregate, you should enable Autogrow for the volume and set the maximum size for the volume to the size of the aggregate. In this configuration, you must monitor aggregate free space actively, but you do not need to monitor the free space in the volume.

Configuration settings for space-reserved files or LUNs with semi-thick volume provisioning

This FlexVol volume and file or LUN configuration combination requires less storage to be allocated up front than the fully provisioned combination, but places restrictions on the efficiency technologies you can use for the volume. Overwrites are fulfilled on a best-effort basis for this configuration combination.

The following settings are required to configure a space-reserved LUN in a volume using semi-thick provisioning:

Volume setting	Value
Guarantee	Volume
Fractional reserve	0
Snapshot reserve	0
Snapshot autodelete	On, with a commitment level of destroy, a destroy list that includes all objects, the trigger set to volume, and all FlexClone LUNs and FlexClone files enabled for automatic deletion.
Autogrow	Optional; if enabled, aggregate free space must be actively monitored.

File or LUN setting	Value
Space reservation	Enabled

Technology restrictions

You cannot use the following volume storage efficiency technologies for this configuration combination:

- Compression
- Deduplication
- ODX and FlexClone Copy Offload
- FlexClone LUNs and FlexClone files not marked for automatic deletion (active clones)
- FlexClone subfiles
- ODX/Copy Offload

Additional considerations

The following facts must be considered when employing this configuration combination:

- When the volume that supports that LUN runs low on space, protection data (FlexClone LUNs and files, Snapshot copies) is destroyed.
- Write operations can time out and fail when the volume runs out of free space.

Compression is enabled by default for All Flash FAS platforms. You must explicitly disable compression for any volume for which you want to use semi-thick provisioning on an All Flash FAS platform.

Determining the correct volume and LUN configuration combination for your environment

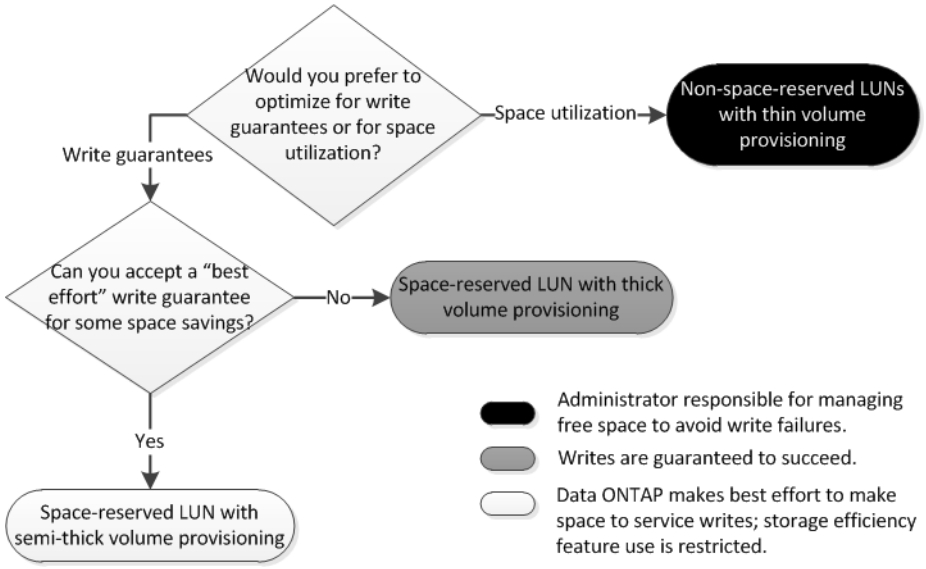
Answering a few basic questions about your environment can help you determine the best FlexVol volume and LUN configuration for your environment.

About this task

You can optimize your LUN and volume configurations for maximum storage utilization or for the security of write guarantees. You must choose which choice makes the most sense for your installation, based on your requirements for storage utilization and your ability to monitor and replenish free space quickly.

Step

1. Use the following decision tree to determine the best volume and LUN configuration combination for your environment:



How file and LUN space reservation works

Enabling space reservation for a file or LUN tells Data ONTAP to reserve the space required for that object at creation time rather than as the space is required for writes. Disabling space reservation enables you to overcommit the containing volume by creating LUNs that appear to provide more space than the volume can currently provide.

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

When a volume contains one or more files or LUNs with space reservation 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 space reservation enabled continue to succeed.

You can enable space reservation for files and LUNs contained by volumes with volume guarantees of any value. However, if the volume has a guarantee of **none**, space reservation for any files or LUNs contained by that volume has no effect.

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.

How volume guarantees work for FlexVol volumes

Volume guarantees (sometimes called *space guarantees*) determine how space for a volume is allocated from its containing aggregate—whether or not the space is preallocated for the volume.

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, provided that sufficient free space exists to honor the new guarantee.

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

- A guarantee type of **`volume`** allocates space in the aggregate for the entire volume when you create the volume, regardless of whether that space is used for data yet.
  The allocated space cannot be provided to or allocated for any other volume in that aggregate.
- A guarantee of **`none`** allocates space from the aggregate only as it is needed by the volume.
  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, although the amount of space that can actually be used is limited by the size of 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.

When space in the aggregate is allocated for a **`volume`** 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 concepts

What the volume footprint is on page 45

Related information

NetApp Technical Report 3965: NetApp Thin Provisioning Deployment and Implementation Guide Data ONTAP 8.1 (7-Mode)

Enabling volume guarantees

When a FlexVol 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.

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, if there is space available to do so. If you specify a different guarantee from 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.
	
	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     none            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 for FlexVol volumes on page 26
Methods to create space in a FlexVol volume on page 47
Methods to create space in an aggregate on page 48

Considerations for setting fractional reserve

Fractional reserve, also called *LUN overwrite reserve*, enables you to turn off overwrite reserve for space-reserved LUNs and files in a FlexVol volume. This can help you maximize your storage utilization, but if your environment is negatively affected by write operations failing due to lack of space, you must understand the requirements that this configuration imposes.

The fractional reserve setting is expressed as a percentage; the only valid values are **`0`** and **`100`** percent. The fractional reserve setting is an attribute of the volume.

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`**. With proper volume configuration and use, however, you can minimize the chance of writes failing. Data ONTAP provides a “best effort” write guarantee for volumes with fractional reserve set to **`0`** when *all* of the following requirements are met:

- Deduplication is not in use
- Compression is not in use
- FlexClone sub-files are not in use
- All FlexClone files and FlexClone LUNs are enabled for automatic deletion
  This is not the default setting. You must explicitly enable automatic deletion, either at creation time or by modifying the FlexClone file or FlexClone LUN after it is created.
- ODX and FlexClone copy offload are not in use
- Volume guarantee is set to **`volume`**
- File or LUN space reservation is **`enabled`**
- Volume Snapshot reserve is set to **`0`**
- Volume Snapshot copy automatic deletion is **`enabled`** with a commitment level of **`destroy`**, a destroy list of **`lun_clone,vol_clone,cifs_share,file_clone,sfsr`**, and a trigger of **`volume`**
  This setting also ensures that FlexClone files and FlexClone LUNs are deleted when necessary.

Note that 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 above 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 cannot meet all of the above configuration requirements and you need to ensure that the volume does not run out of space, you must 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

Related concepts

How volume guarantees work for FlexVol volumes on page 26
How file and LUN space reservation works on page 25
How to address aggregate fullness and overallocation alerts on page 35
How to address FlexVol volume fullness and overallocation alerts on page 33

Related tasks

Deleting Snapshot copies automatically on page 31
Configuring volumes to automatically grow and shrink their size on page 32

Related information

NetApp Technical Report 3965: NetApp Thin Provisioning Deployment and Implementation Guide Data ONTAP 8.1 (7-Mode)
NetApp Technical Report 3483: Thin Provisioning in a NetApp SAN or IP SAN Enterprise Environment

Configuring volumes to automatically provide more space when they are full

When FlexVol volumes get full, Data ONTAP can use various methods to attempt to automatically provide more free space for the volume. You choose which methods Data ONTAP can use, and in which order, depending on the requirements imposed by your application and storage architecture.

About this task

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

- Increase the size of the volume (known as *autogrow*).
  This method is useful if the volume's containing aggregate has enough space to support a larger volume. You can configure Data ONTAP to 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.
  Autogrow is not triggered to support Snapshot copy creation. If you attempt to create a Snapshot copy and there is insufficient space, the Snapshot copy creation fails, even with autogrow enabled.
- Delete Snapshot copies, FlexClone files, or FlexClone LUNs.
  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. In most cases, the default configuration is preferable, because when a Snapshot copy is deleted, it cannot be restored. However, if you need to avoid growing the size of a volume whenever possible, you can configure Data ONTAP to delete Snapshot copies before increasing the size of the volume.

Steps

1. If you want Data ONTAP to attempt to increase the size of the volume when it gets full, enable the autogrow capability for the volume by using the `volume autosize` command with **`grow`** mode.

   Remember that when the volume grows, it consumes more free space from its associated aggregate. If you are depending on the volume's ability to grow whenever it needs to, you must monitor the free space in the associated aggregate and add more when needed.

2. If you want Data ONTAP to delete Snapshot copies, FlexClone files, or FlexClone LUNs when the volume gets full, enable autodelete for those object types.

3. If you enabled both the volume autogrow capability and one or more autodelete capabilities, 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`**.

Related concepts

How a FlexVol volume can reclaim free space from FlexClone files and FlexClone LUNs on page 74

Methods to create space in a FlexVol volume on page 47

Related tasks

Deleting Snapshot copies automatically on page 31

Configuring a FlexVol volume to automatically delete FlexClone files and FlexClone LUNs on page 75

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. 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 marked for autodeletion 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 tasks

Configuring a FlexVol volume to automatically delete FlexClone files and FlexClone LUNs on page 75

Configuring volumes to automatically grow and shrink their size

You can configure FlexVol volumes to automatically grow and shrink according to how much space they currently require. Automatic growing helps prevent a volume from running out of space, if the aggregate can supply more space. Automatic shrinking prevents a volume from being larger than needed, freeing space in the aggregate for use by other volumes.

Before you begin

The FlexVol volume must be online.

About this task

Autoshrink can only be used in combination with autogrow to meet changing space demands and is not available alone. When autoshrink is enabled, Data ONTAP automatically manages the shrinking behavior of a volume to prevent an endless loop of autogrow and autoshrink actions.

As a volume grows, the maximum number of files it can contain might be automatically increased. When a volume is shrunk, the maximum number of files it can contain is left unchanged, and a volume cannot be automatically shrunk below the size that corresponds to its current maximum number of files. For this reason, it might not be possible to automatically shrink a volume all the way to its original size.

By default, the maximum size a volume can grow to is 120% of the size at which autogrow is enabled. If you need to ensure that the volume can grow to be larger than that, you must set the maximum size for the volume accordingly.

Step

1. Configure the volume to grow and shrink its size automatically:

   volume autosize -vserver *vserver_name* *vol_name* -mode grow_shrink

Example

The following command enables automatic size changes for a volume called test2. The volume is configured to begin shrinking when it is 60% full. The default values are used for when it will begin to grow and its maximum size.

```
cluster1::> volume autosize -vserver vs2 test2 -shrink-threshold-
percent 60
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 to address FlexVol volume fullness and overallocation alerts

Data ONTAP issues EMS messages when FlexVol volumes are running out of space so that you can take corrective action by providing more space for the full volume. Knowing the types of alerts and how to address them helps you ensure your data availability.

When a volume is described as *full*, it means that the percentage of the space in the volume available for use by the active file system (user data) has fallen below a (configurable) threshold. When a volume becomes *overallocated*, the space used by Data ONTAP for metadata and to support basic data access has been exhausted. Sometimes space normally reserved for other purposes can be used to keep the volume functioning, but space reservation or data availability can be at risk.

Overallocation can be either logical or physical. *Logical overallocation* means that space reserved to honor future space commitments, such as space reservation, has been used for another purpose. *Physical overallocation* means that the volume is running out of physical blocks to use. Volumes in this state are at risk for refusing writes, going offline, or potentially causing a controller disruption.

A volume can be more than 100% full due to space used or reserved by metadata. However, a volume that is more than 100% full might or might not be overallocated.

The following table describes the volume fullness and overallocation alerts, the actions you can take to address the issue, and the risks of not taking action.

Alert type	EMS level	Configurable?	Definition	Ways to address	Risk if no action taken
Nearly full	Debug	Y	The file system has exceeded the threshold set for this alert (the default is 95%). The percentage is the `Used` total minus the size of the Snapshot reserve.	• Increasing volume size • Reducing user data	No risk to write operations or data availability yet.
Full	Debug	Y	The file system has exceeded the threshold set for this alert (the default is 98%). The percentage is the `Used` total minus the size of the Snapshot reserve.	• Increasing volume size • Reducing user data	No risk to write operations or data availability yet, but the volume is approaching the stage where write operations could be at risk.
Logically overallocated	SVC Error	N	In addition to the file system being full, the space in the volume used for metadata has been exhausted.	• Increasing volume size • Deleting Snapshot copies • Reducing user data • Disabling space reservation for files or LUNs	Write operations to unreserved files could fail.
Physically overallocated	Node Error	N	The volume is running out of physical blocks it can write to.	• Increasing volume size • Deleting Snapshot copies • Reducing user data	Write operations are at risk, as well as data availability; the volume could go offline.

Every time a threshold is crossed for a volume, whether the fullness percentage is rising or falling, an EMS message is generated. When the fullness level of the volume falls below a threshold, a `volume ok` EMS message is generated.

Related concepts

How to address aggregate fullness and overallocation alerts on page 35

Related tasks

Configuring volumes to automatically provide more space when they are full on page 30

Related information

Clustered Data ONTAP 8.3 System Administration Guide

How to address aggregate fullness and overallocation alerts

Data ONTAP issues EMS messages when aggregates are running out of space so that you can take corrective action by providing more space for the full aggregate. Knowing the types of alerts and how you can address them helps you ensure your data availability.

When an aggregate is described as *full*, it means that the percentage of the space in the aggregate available for use by volumes has fallen below a predefined threshold. When an aggregate becomes *overallocated*, the space used by Data ONTAP for metadata and to support basic data access has been exhausted. Sometimes space normally reserved for other purposes can be used to keep the aggregate functioning, but volume guarantees for volumes associated with the aggregate or data availability can be at risk.

Overallocation can be either logical or physical. *Logical overallocation* means that space reserved to honor future space commitments, such as volume guarantees, has been used for another purpose. *Physical overallocation* means that the aggregate is running out of physical blocks to use. Aggregates in this state are at risk for refusing writes, going offline, or potentially causing a controller disruption.

The following table describes the aggregate fullness and overallocation alerts, the actions you can take to address the issue, and the risks of not taking action.

Alert type	EMS Level	Configurable?	Definition	Ways to address	Risk if no action taken
Nearly full	Debug	N	The amount of space allocated for volumes, including their guarantees, has exceeded the threshold set for this alert (95%). The percentage is the `Used` total minus the size of the Snapshot reserve.	• Adding storage to the aggregate • Shrinking or deleting volumes • Moving volumes to another aggregate with more space • Removing volume guarantees (setting them to **none**)	No risk to write operations or data availability yet.
Full	Debug	N	The file system has exceeded the threshold set for this alert (98%). The percentage is the `Used` total minus the size of the Snapshot reserve.	• Adding storage to the aggregate • Shrinking or deleting volumes • Moving volumes to another aggregate with more space • Removing volume guarantees (setting them to **none**)	Volume guarantees for volumes in the aggregate might be at risk, as well as write operations to those volumes.

Alert type	EMS Level	Configurable?	Definition	Ways to address	Risk if no action taken
Logically overallocated	SVC Error	N	In addition to the space reserved for volumes being full, the space in the aggregate used for metadata has been exhausted.	• Adding storage to the aggregate • Shrinking or deleting volumes • Moving volumes to another aggregate with more space • Removing volume guarantees (setting them to **none**)	Volume guarantees for volumes in the aggregate are at risk, as well as write operations to those volumes.
Physically overallocated	Node Error	N	The aggregate is running out of physical blocks it can write to.	• Adding storage to the aggregate • Shrinking or deleting volumes • Moving volumes to another aggregate with more space	Write operations to volumes in the aggregate are at risk, as well as data availability; the aggregate could go offline. In extreme cases, the node could experience a disruption.

Every time a threshold is crossed for an aggregate, whether the fullness percentage is rising or falling, an EMS message is generated. When the fullness level of the aggregate falls below a threshold, an `aggregate ok` EMS message is generated.

Related concepts

How to address FlexVol volume fullness and overallocation alerts on page 33

Related information

Clustered Data ONTAP 8.3 System Administration Guide

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.
  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 49

Considerations for changing the maximum number of files allowed on a FlexVol volume on page 49

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's active file system (volume data not captured in a Snapshot copy) 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.

The `volume show-space` command displays how the used space in a volume is being used. 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 show-space` 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
- 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.

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 • Disabling file or LUN space reservation Note that turning off space reservation disables Data ONTAP's ability to guarantee writes to those files or LUNs. This can result in write operations failing due to lack of space. Turning off space reservation should be a temporary measure, and space reservation 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.

Row/feature name	Description	Some ways to decrease space usage
Filesystem Metadata / Filesystem Metadata Percent	Internal tracking for the file system required by Data ONTAP.	No direct method to control.
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.
Performance Metadata / Performance Metadata Percent	The amount of space used by performance optimization activities.	No direct method to control.
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	A percentage of the current volume size. The Snapshot reserve is counted as used space, even if there are no Snapshot copies in the reserve. When the volume is not full, the Snapshot reserve is not available to the active file system. This row is the same as the total space used for the `.snapshot` row in the `df` command.	Using 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 Reserve Unusable	If the active file system is using more space than the volume has allocated for it, it can start to use space previously allocated for Snapshot reserve. This line shows the amount of space originally allocated for Snapshot reserve that is unavailable for Snapshot copies because it is being used by the active file system. This value is displayed as a negative number.	Decreasing the size of the active file system by deleting user data or decreasing volume metadata.
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.

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.
Total Physical Used	The total amount of space that is being used right now (rather than being reserved for future use), including space used by Snapshot copies.	Any of the methods for individual output rows.

Example output with Snapshot spill and deduplication

The following example displays output for a FlexVol volume with deduplication enabled and more Snapshot copies than fit into Snapshot reserve:

```
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%
      Deduplication                           215KB         0%

      Total Used                             1.35GB        68%
```

Example output with Snapshot reserve unusable

The following example displays output for a FlexVol volume that is consuming some Snapshot reserve for the active file system because the active file system is so full:

```
cluster1::> volume show-space testvol2
     Vserver : thevs
     Volume  : testvol2

     Feature                                  Used      Used%
     -------------------------------     ----------     ------
     User Data                             19.57MB        98%
     Filesystem Metadata                     100KB         0%
     Inodes                                  108KB         1%
     Snapshot Reserve                          1MB         5%
```

```
      Snapshot Reserve Unusable                -396KB         2%

      Total Used                              20.39MB       102%
```

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 the `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.

Row/feature name	Description/contents of row	Some ways to decrease
`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.
`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.
`File Operation Metadata`	The total amount of space reserved for file operation metadata. After space is used for file operation metadata, it is not returned as free space to the aggregate, but it is reused by subsequent file operations.	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 47
Methods to create space in an aggregate on page 48
What the volume footprint is on page 45

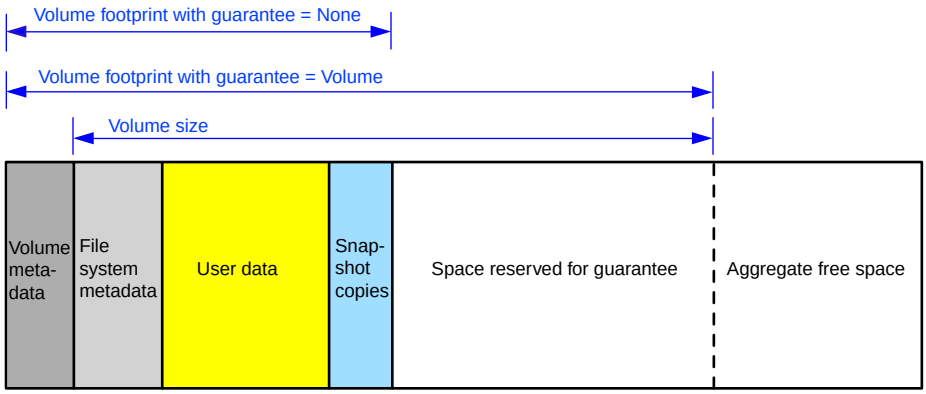
Related information

NetApp Technical Report 3965: NetApp Thin Provisioning Deployment and Implementation Guide Data ONTAP 8.1 (7-Mode)
NetApp Technical Report 3483: Thin Provisioning in a NetApp SAN or IP SAN Enterprise Environment

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 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.

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

- `Total Used`
  The sum of all space used or reserved in the aggregate by volumes, metadata, or Snapshot copies.
- `Total Physical Used`
  The amount of space being used for data now (rather than being reserved for future use). Includes space used by aggregate Snapshot copies.

There is never a row for Snapshot spill.

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

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

                      Aggregate : wqa_gx106_aggr1

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

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

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.

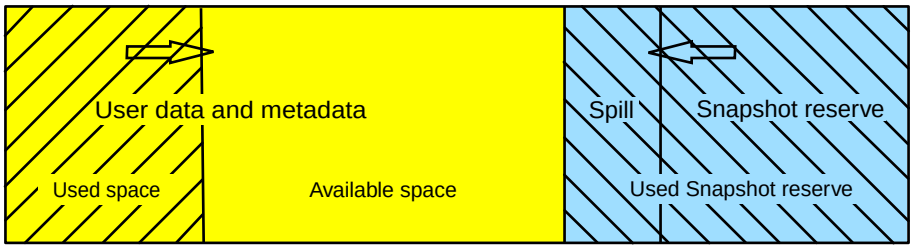
When Snapshot copies use more space than the Snapshot reserve, they spill over and use space in the active file system. The *Snapshot reserve* area of a volume is the space reserved exclusively for Snapshot copies. It is not available to the user data or metadata area of the volume. The size of the Snapshot 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 usage information on page 55

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, m
  - More space in the aggregate can allow a volume to increase in size automatically with the autogrow capability.

    - For volumes with a none guarantee, limited free space in the aggregate manifests as limited free space in the volume.

- 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 files and FlexClone LUNs manually or enable automatic deletion.
- (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 48
How you can determine and control a volume's space usage in the aggregate on page 43
Using deduplication and data compression to increase storage efficiency on page 119
How a FlexVol volume can reclaim free space from FlexClone files and FlexClone LUNs on page 74
Considerations for setting fractional reserve on page 29

Related tasks

Configuring volumes to automatically grow and shrink their size on page 32
Deleting Snapshot copies automatically on page 31

Methods to create space in an aggregate

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

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

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

- Add disks to the aggregate.
- Move some volumes to another aggregate with available space.
- Shrink the size of volume-guaranteed volumes in the aggregate.

You can do this manually or with the `autoshrink` option of the autosize capability.

- Change volume guarantee types to **none** on volumes that are using large amounts of space (large volume-guaranteed volumes with large reserved files) so that the volumes take up less space in the aggregate.
  A volume with a guarantee type of **none** has a smaller footprint in the aggregate than a volume with a guarantee type of **`volume`**. 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.
- 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 47
Moving and copying volumes (cluster administrators only) on page 56
How you can determine and control a volume's space usage in the aggregate on page 43

Related tasks

Configuring volumes to automatically grow and shrink their size on page 32
Deleting Snapshot copies automatically on page 31

Related information

NetApp Technical Report 3965: NetApp Thin Provisioning Deployment and Implementation Guide Data ONTAP 8.1 (7-Mode)
NetApp Technical Report 3483: Thin Provisioning in a NetApp SAN or IP SAN Enterprise Environment

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.

How Flash Pool aggregate caching policies work

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

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

Caching policies combine a read caching policy and a write caching policy. The policy name concatenates the names of the read caching policy and the write caching policy, separated by a hyphen. Underscores are used within a read or write caching policy name. For example, the all_read_random_write-random_write policy combines the “all_read_random_write” read caching policy and the “random_write” write caching policy. If there is no hyphen in the policy name, the write caching policy is “none”, except for the **`auto`** policy.

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

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

The following table provides a rough approximation of what types of data are inserted into cache for specific read and write caching policies.

Policy name	Insertions using read caching policy				Insertions using write caching policy	Privilege level
	Random reads	Sequential reads	Random writes	Sequential writes	Random overwrites	
auto	Yes	No	No	No	Yes	admin
none	No	No	No	No	No	admin
random_read	Yes	No	Yes	No	No	advanced
noread-random_write	No	No	No	No	Yes	advanced
meta	Metadata only	No	No	No	No	advanced
meta-random_write	Metadata only	No	No	No	Yes	advanced
random_read_write	Yes	No	Yes	No	No	advanced
random_read_write-random_write	Yes	No	Yes	No	Yes	advanced
all_read	Yes	Yes	No	No	No	advanced
all_read-random_write	Yes	Yes	No	No	Yes	advanced
all_read_random_write	Yes	Yes	Yes	No	No	advanced
all_read_random_write-random_write	Yes	Yes	Yes	No	Yes	advanced
all	Yes	Yes	Yes	Yes	No	advanced
all-random_write	Yes	Yes	Yes	Yes	Yes	advanced

Metadata is cached for all policies except **none**.

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

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

Related information

Clustered Data ONTAP 8.3 Physical Storage Management Guide
NetApp Technical Report 4070: Flash Pool Design and Implementation Guide

Rules governing node root volumes and root aggregates

A node's root volume contains special directories and 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 or by setup software. It is reserved for system files, log files, and core files. The directory name is `/mroot`, which is accessible only through the systemshell by technical support. The minimum size for a node's root volume depends on the platform model.

- The following rules govern the node's root volume:
  - Unless technical support instructs you to do so, do not modify the configuration or content of the root volume.
  - 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.
  - 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 root aggregate is dedicated to the node's root volume only.
  Data ONTAP prevents you from creating other volumes in the root aggregate.

Related information

NetApp Hardware Universe

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.

About this task

If you create a volume in an All Flash FAS platform, and if all of the nodes in the cluster are running Data ONTAP 8.3.1, then by default, adaptive compression is enabled and *inline-only* efficiency policy is assigned to the volume.

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

You can delete a FexVol volume that is no longer required or if it contains corrupted data.

Before you begin

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

Note: If you accidentally delete a volume, contact technical support for assistance.

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`

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

You use the `storage aggregate` and `volume` commands to see how space is being used in your aggregates and volumes and their Snapshot copies.

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

Related concepts

How you can determine and control space usage in a volume on page 38
How you can determine and control a volume's space usage in the aggregate on page 43
How to determine space usage in an aggregate on page 45
How Snapshot copies and Snapshot reserve use space in a volume on page 46

Related information

Clustered Data ONTAP 8.3 Commands: Manual Page Reference

Moving and copying volumes (cluster administrators only)

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

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.

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 attempts the final phase three times by default. After the third attempt, the system waits an hour before attempting the final phase sequence again. The system runs the final phase of the volume move operation until the volume move is complete.

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`*

If you want to...	Use this command...
Change the amount of time client access is blocked if the default is not adequate.	*volume move start* or *volume move modify* with the `-cutover-window` parameter. The `volume move modify` command is an advanced command and the `-cutover-window` is an advanced parameter.
Determine what the system does if the volume move operation cannot be completed during the time client access is blocked.	*volume move start* or *volume move modify* with the `-cutover-action` parameter. The `volume move modify` command is an advanced command and the `-cutover-action` is an advanced parameter.

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 or by the system configuration, such as a MetroCluster configuration. You should understand the considerations and recommendations associated with moving volumes.

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.
- If a volume that contains LUNs is not NVFAIL enabled before you move it, the volume will be NVFAIL enabled after you move it.
- You can move a volume from a Flash Pool aggregate to another Flash Pool aggregate.
  - The caching policies of that volume are also moved.
  - The move might affect volume performance.
- You can move volumes between a Flash Pool aggregate and a non-Flash Pool aggregate.
  - If you move a volume from a Flash Pool aggregate to a non-Flash Pool aggregate, Data ONTAP displays a message warning you that the move might affect volume performance and asks whether you want to continue.
  - If you move a volume from a non-Flash Pool aggregate to a Flash Pool aggregate, Data ONTAP assigns the **`auto`** caching policy.

FlexClone volume 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.
  You should not move a volume until after clone-splitting operations are finished.

MetroCluster configuration considerations

- If a MetroCluster switchover occurs before the cutover, the destination volume has a record and is a temporary volume (a volume of type TMP).
  This temporary volume is not deleted by Data ONTAP; it must be deleted manually.
  When moving a volume in a MetroCluster configuration, a record of the temporary volume corresponding to the volume in the mirrored but unassimilated aggregate is created when the temporary volume is created on the destination aggregate on the source cluster.
- If a Metrocluster switchover occurs after the cutover phase has started, but before the move job has completed, the volume move operation on the destination aggregate runs to completion, but Data ONTAP will not delete the volume on the source.
  You must delete this volume manually.
- Neither forced nor unforced MetroCluster switchbacks are allowed if there are any volume move operations in progress.
- Unforced Metrocluster switchovers are blocked, but forced Metrocluster switchovers are not if there are any volume move operations in progress.

Requirement for moving volumes in SAN environments

Before you move a volume that contains one or more LUNs, you should have a minimum of two paths per LUN (LIFs) connecting to each node in the cluster. This eliminates single points of failure and enables the system to survive component failures.

Moving a volume

You can move a FlexVol volume to a different aggregate, node, or both within the same Storage Virtual Machine (SVM) to balance storage capacity after determining that there is a storage capacity imbalance.

About this task

The administrator previously determined that clients accessing the data in the volume can tolerate an I/O timeout of, at most, 120 seconds.

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. Determine an aggregate to which you can move the volume by using the `volume move target-aggr show` command.

   The aggregate that you select must have enough space for the volume; that is, the available size is bigger than the volume that you are moving.

   Example

   The following example shows that the vs2 volume can be moved to any of the listed aggregates:

   ```
   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.
   ```

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

4. Move the volume by using the `volume move start` command.

   Example

   The following command moves the user_max volume on the vs2 SVM to the node12a_aggr3 aggregate. The move runs as a background process.

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

5. Determine the status of the volume move operation by using the `volume move show` command.

   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     -                -
   ```

Methods for copying a volume

Copying a volume creates a stand-alone copy of a volume that you can use for testing and other purposes. The method you use to copy a volume depends on the use case.

The method you use for copying a volume depends on whether you are copying it to the same aggregate or a different aggregate, and whether you want to retain Snapshot copies from the original volume. The following table lists characteristics of the copy and the methods used to create that copy.

If you want to copy a volume...	Then the method you use is...
Within the same aggregate and you do not want to copy Snapshot copies from the original volume.	Creating a FlexClone volume of the original volume.
To another aggregate and you do not want to copy Snapshot copies from the original volume.	Creating a FlexClone volume of the original volume, and then moving the volume to another aggregate by using the `volume move` command.
To another aggregate and preserve all of the Snapshot copies from the original volume.	Replicating the original volume using SnapMirror, and then breaking the SnapMirror relationship to make a read-write volume copy.

Related tasks

Creating a FlexClone volume on page 64

Related information

Clustered Data ONTAP 8.3 Data Protection Guide

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 56

Related tasks

Moving a volume on page 58

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 from a SnapMirror destination volume or 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.

You can use multiple methods to delete FlexClone files and FlexClone LUNs.

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.

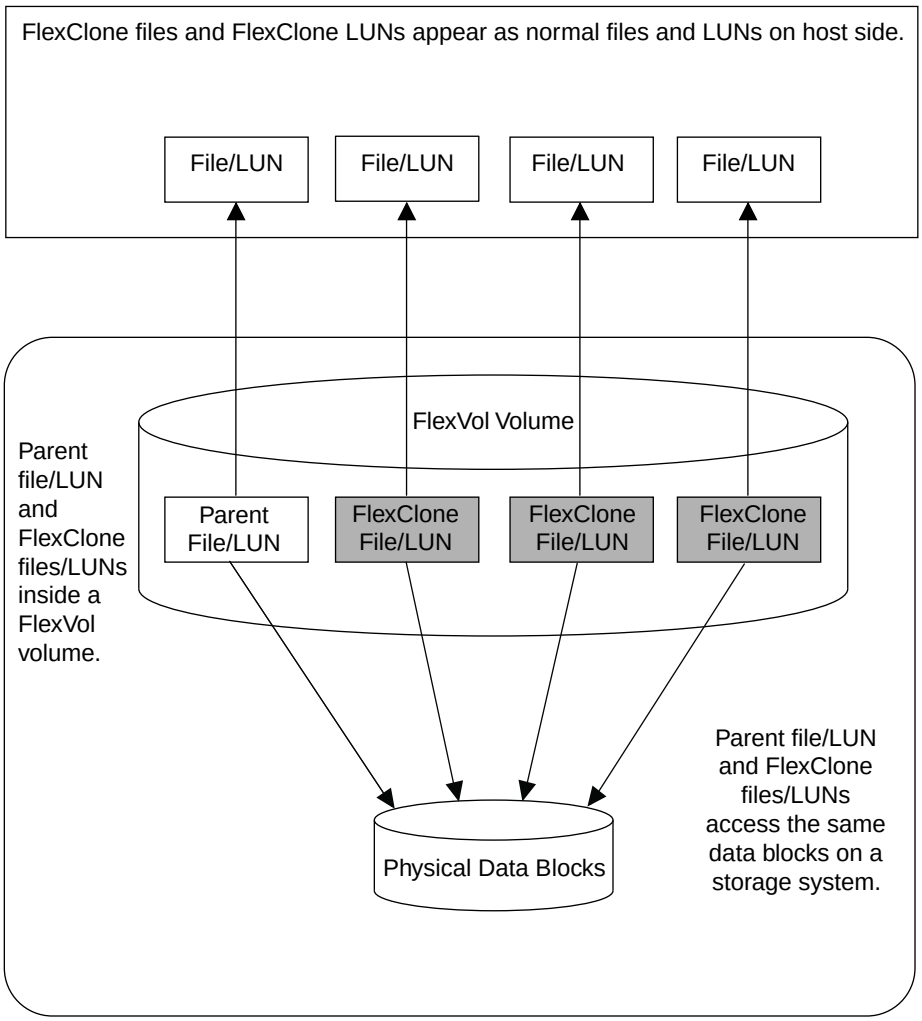
How FlexClone files and FlexClone LUNs work

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. Only when you write new data to a parent or clone does the entity on which new data is written start 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:



Each node accepts requests to create FlexClone files and FlexClone LUNs until the node reaches its maximum split load. When the node reaches its maximum split load, the node temporarily stops accepting requests to create clones and issues an EBUSY error message. When the split load for the node falls below the maximum, the node accepts requests to create clones again.

The cloning operation 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.

Related concepts

How the split load identifies node capacity to create and delete FlexClone files and FlexClone LUNs on page 71

Considerations for working with FlexClone files and FlexClone LUNs

You should keep certain considerations in mind when creating, managing, and deleting FlexClone files and FlexClone LUNs.

Considerations for creating 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.
  You should know the block range of the parent entity and clone entity to clone a sub-file or sub-LUN.
- The `sis` attribute is added to a FlexVol volume when a FlexClone file or FlexClone LUN is created for the first time.
- If you want to clone a file or LUN between qtrees of a FlexVol volume, the destination qtree must not be a secondary of a SnapVault destination.

Considerations for managing FlexClone files and FlexClone LUNs

- When clients write new data either to a FlexClone file or FlexClone LUN, or the parent file or parent LUN, the new data occupies additional storage space.
- 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 fractional reserve set to zero, and the applications accessing the volume encounter errors due to lack of space (ENOSPC), observe the configuration restrictions.
  You should also enable the autodelete settings of a FlexVol volume to automatically delete FlexClone files and LUNs to help the applications accessing the FlexVol volume avoid encountering errors due to lack of space.

Considerations for deleting FlexClone files and FlexClone LUNs

- 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 delete FlexClone files without using the NetApp Manageability SDK, you can configure the system for faster deletion by using the `volume file clone deletion` commands.
  If you use the NetApp Manageability SDK to delete FlexClone files and FlexClone LUNs, configuration of the system is not required because the faster deletion method is always used.

Related concepts

Considerations for setting fractional reserve on page 29

Related tasks

Configuring a FlexVol volume to automatically delete FlexClone files and FlexClone LUNs on page 75

Related references

Commands for configuring deletion of FlexClone files on page 77

Examples of creating space-optimized FlexClone files and FlexClone LUNs

The examples illustrate how Data ONTAP 8.3 and later requires no extra space to clone files and LUNs with space reservation enabled in a FlexVol volume with fractional reserve set to zero. Scenarios that do not support space-optimized clones are also described.

Example of creating FlexClone files with space reservation enabled

Consider that you want to clone a 100 GB file without holes in a FlexVol volume with fractional reserve set to zero. Space reservation is enabled for the file and will be enabled for the FlexClone file.

The following example shows how to clone file1_source to create a FlexClone file named file1_clone. No space reservation is specified. As a result, the FlexClone file inherits the same space reservation setting as the source file, which has space reservation enabled:

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

In Data ONTAP 8.3 and later, you do not need temporary additional space to create the FlexClone file. In previous releases, you need at least 200 GB of available space to create a FlexClone file with space reservation enabled from a 100 GB file with space reservation enabled in a FlexVol volume with fractional reserve set to zero. Extra space is no longer needed because the FlexClone files and FlexClone LUNs are space-optimized.

Example of creating FlexClone files with space reservation disabled

Consider that you want to clone a 100 GB file without holes in a FlexVol volume with fractional reserve set to zero. Space reservation is enabled for the file, but will be disabled for the FlexClone file.

The following example shows how to clone file1_source to create a FlexClone file with space reservation disabled named file1_clone:

```
cluster1::> volume file clone create -vserver vs0 -volume vol1 -
source-path /file1_source -destination-path /file1_clone -no-reserve
```

In Data ONTAP 8.3 and later, you do not need temporary additional space to create the FlexClone file. In previous releases, you need at least 100 GB of available space to create a FlexClone file with space reservation disabled from a 100 GB file with space reservation enabled in a FlexVol volume with fractional reserve set to zero. Extra space is no longer needed.

Scenarios where space-optimized clones are not supported

The following scenarios do not support space-optimized clones:

- You cannot create space-optimized clones of files, sub-files, LUNs, and sub-LUNs in a Snapshot copy.
  Extra space is required in the volume to create the clones.
- When existing FlexClone files and FlexClone LUNs are not space-optimized, you cannot create space-optimized clones of the clones.
  Extra space is required in the volume to create clones of non-space-optimized clones.
- In some situations space-optimized clones are converted to clones without space optimization and require extra space in the volume.

For example, when you clone a sub-file or sub-LUN in a Snapshot copy to overwrite a space-optimized clone in the active file system of a volume, the space-optimized clone is converted to a clone without space optimization, and extra space is required for the clone.

Related concepts

Considerations for setting fractional reserve on page 29

How the split load identifies node capacity to create and delete FlexClone files and FlexClone LUNs

You can view the maximum, current, token-reserved, and allowable *split load* for each node to determine when the node has capacity to create and delete FlexClone files and FlexClone LUNs. The allowable split load identifies when a node has capacity to create and delete clones.

The following table lists the split load information that is available for each node:

Split load type	Description
Max Split Load	The maximum amount of capacity on the node that can be used to create and delete FlexClone files and FlexClone LUNs When the maximum split load is reached, an `EBUSY` error message is issued, and new requests are not accepted until the load falls below the maximum.
Current Split Load	The amount of capacity on the node that is being used to create and delete FlexClone files and FlexClone LUNs
Token-Reserved Load	The amount of capacity on the node that is reserved by clients with tokens to create and delete FlexClone files and FlexClone LUNs You can configure clients to use the NetApp Manageability SDK to retrieve a token that reserves space on a node to create FlexClone files and FlexClone LUNs.
Allowable Split Load	The remaining amount of capacity on the node that is available for new requests to create and delete FlexClone files and FlexClone LUNs. (The Max Split Load minus the Current Split Load minus the Token-Reserved Load equals the Allowable Split Load.)

Related tasks

Viewing node capacity for creating and deleting FlexClone files and FlexClone LUNs on page 73

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

- The 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.
- If you are creating a sub-LUN or sub-file on volumes with adaptive compression enabled, the block ranges must not be misaligned.

This means that the source start block number and destination start block number must either be even aligned or odd aligned.

About this task

Depending on the privileges assigned by the cluster administrator, an SVM administrator can create FlexClone files and FlexClone LUNs.

You can specify the autodelete setting for FlexClone files and FlexClone LUNs when you create and modify clones. By default, the autodelete setting is disabled.

You can overwrite an existing FlexClone file or FlexClone LUN when you create a clone by using the `volume file clone create` command with the `-overwrite-destination` parameter.

When the node reaches its maximum split load, the node temporarily stops accepting requests to create FlexClone files and FlexClone LUNs and issues an `EBUSY` error message. When the split load for the node falls below the maximum, the node accepts requests to create FlexClone files and FlexClone LUNs again. You should wait until the node has capacity to create the clones before trying the create request again.

Step

1. Create a FlexClone file or FlexClone LUN by using 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.

Related concepts

How the split load identifies node capacity to create and delete FlexClone files and FlexClone LUNs on page 71

Related tasks

Viewing node capacity for creating and deleting FlexClone files and FlexClone LUNs on page 73

Related information

Clustered Data ONTAP 8.3 Commands: Manual Page Reference

Viewing node capacity for creating and deleting FlexClone files and FlexClone LUNs

You can view whether a node has capacity to receive new requests to create and delete FlexClone files and FlexClone LUNs by viewing the split load for the node. If the maximum split load is reached, no new requests are accepted until the split load falls below the maximum.

About this task

When the node reaches its maximum split load, an `EBUSY` error message is issued in response to create and delete requests. When the split load for the node falls below the maximum, the node accepts requests to create and delete FlexClone files and FlexClone LUNs again.

A node can accept new requests when the Allowable Split Load field displays capacity, and the create request fits in the available capacity.

Step

1. View how much capacity a node has to create and delete FlexClone files and FlexClone LUNs by using the `volume file clone split load show` command.

Example

In the following example, the split load is displayed for all of the nodes in cluster1. All nodes in the cluster have capacity to create and delete FlexClone files and FlexClone LUNs as indicated by the Allowable Split Load field:

```
cluster1::> volume file clone split load show
Node       Max        Current    Token         Allowable
           Split Load Split Load Reserved Load Split Load
---------- ---------- ---------- ------------- ----------
node1         15.97TB         0B         100MB    15.97TB
node2         15.97TB         0B         100MB    15.97TB
2 entries were displayed.
```

Related concepts

How the split load identifies node capacity to create and delete FlexClone files and FlexClone LUNs on page 71

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
```

Methods to delete FlexClone files and FlexClone LUNs

You can use multiple methods to delete FlexClone files and FlexClone LUNs. Understanding what methods are available helps you plan how to manage clones.

You can use the following methods to delete FlexClone files and FlexClone LUNs:

- You can configure a FlexVol volume to automatically delete clones with autodelete enabled when the free space in a FlexVol volume decreases below a particular threshold.
- You can configure clients to delete clones by using the NetApp Manageability SDK.
- You can use clients to delete clones by using the NAS and SAN protocols.
  The slower deletion method is enabled by default because this method does not use the NetApp Manageability SDK. However, you can configure the system to use the faster deletion method when you delete FlexClone files by using the `volume file clone deletion` commands.

Related concepts

How a FlexVol volume can reclaim free space from FlexClone files and FlexClone LUNs on page 74

Related references

Commands for configuring deletion of FlexClone files on page 77

How a FlexVol volume can reclaim free space from FlexClone files and FlexClone LUNs

You can configure the autodelete settings of a FlexVol volume to automatically delete FlexClone files and FlexClone LUNs that have autodelete enabled when a volume is nearly full to reclaim a target amount of free space in the volume.

You can configure a volume to automatically start deleting FlexClone files and FlexClone LUNs when the free space in the volume decreases below a particular threshold value and automatically stop deleting clones when a target amount of free space in the volume is reclaimed. Although you cannot specify the threshold value that starts the automatic deletion of clones, you can specify whether a clone is eligible for deletion, and you can specify the target amount of free space for a volume to reclaim by deleting clones.

A volume automatically deletes FlexClone files and FlexClone LUNs when the free space in the volume decreases below a particular threshold and when *both* the following requirements are met:

- The autodelete capability is enabled for the volume that contains the FlexClone files and FlexClone LUNs.
  You can enable the autodelete capability for a FlexVol volume by using the `volume snapshot autodelete modify` command. You must set the `-trigger` parameter to **`volume`** or **`snap_reserve`** for a volume to automatically delete FlexClone files and FlexClone LUNs.
- The autodelete capability is enabled for the FlexClone files and FlexClone LUNs.
  You can enable autodelete for a FlexClone file or FlexClone LUN by using the `file clone create` command with the `-autodelete` parameter. As a result, you can preserve certain FlexClone files and FlexClone LUNs by disabling autodelete for the clones and ensuring that other volume settings do not override the clone setting.

You can specify the autodelete setting for FlexClone LUNs created using Data ONTAP 8.2 and later, and you can specify the autodelete setting for FlexClone files created using Data ONTAP 8.3 and later. After upgrading to Data ONTAP 8.3, FlexClone files created using Data ONTAP versions earlier than 8.3 have autodelete disabled. You can enable autodelete for the FlexClone files by using the `volume file clone autodelete` command.

Related tasks

Configuring a FlexVol volume to automatically delete FlexClone files and FlexClone LUNs on page 75

Configuring a FlexVol volume to automatically delete FlexClone files and FlexClone LUNs

You can enable a FlexVol volume to automatically delete FlexClone files and FlexClone LUNs with autodelete enabled when the free space in the volume decreases below a particular threshold.

Before you begin

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

Steps

1. Enable automatic deletion of FlexClone files and FlexClone LUNs in the FlexVol volume by using the `volume snapshot autodelete modify` command.
   - For the `-trigger` parameter, you can specify **`volume`** or **`snap_reserve`**.
   - For the `-destroy-list` parameter, you must always specify **`lun_clone,file_clone`** regardless of whether you want to delete only one type of clone.

   #### Example

   The following example shows how you can enable volume vol1 to trigger the automatic deletion of FlexClone files and 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,file_clone

   Volume modify successful on volume:vol1
   ```

   Note: While enabling FlexVol volumes for automatic deletion, if you set the value of the `-commitment` parameter to **`destroy`**, all the FlexClone files and FlexClone LUNs with the `-autodelete` parameter set to **`true`** might be deleted when the free space in the volume decreases below the specified threshold value. However, FlexClone files and FlexClone LUNs with the `-autodelete` parameter set to **`false`** will not be deleted.

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

   #### Example

   The following example shows that volume vol1 is enabled for automatic deletion of FlexClone files and 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,file_clone
Is Constituent Volume: false
```

3. Ensure that autodelete is enabled for the FlexClone files and FlexClone LUNs in the volume that you want to delete by performing the following steps:

   a. Enable automatic deletion of a particular FlexClone file or FlexClone LUN by using the `volume file clone autodelete` command.

      You can force a specific FlexClone file or FlexClone LUN to be automatically deleted by using the `volume file clone autodelete` command with the `-force` parameter.

      Example

      The following example shows that automatic deletion of the FlexClone LUN lun1_clone contained in volume vol1 is enabled:

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

      You can enable autodelete when you create FlexClone files and FlexClone LUNs.

   b. Verify that the FlexClone file or FlexClone LUN is enabled for automatic deletion by using the `volume file clone show-autodelete` command.

      Example

      The following example shows that the FlexClone LUN lun1_clone is enabled for automatic deletion:

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

      Vserver Name: vs1
                                                                  Clone
      Path: vol/vol1/lun1_clone

      Autodelete Enabled: true
      ```

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

Related tasks

Deleting Snapshot copies automatically on page 31

Preventing a specific FlexClone file or FlexClone LUN from being automatically deleted

If you configure a FlexVol volume to automatically delete FlexClone files and FlexClone LUNs, any clone that fits the criteria you specify might be deleted. If you have specific FlexClone files or

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 8.3, when you create a FlexClone file or FlexClone LUN, by default the autodelete setting for the clone is disabled. FlexClone files and FlexClone LUNs with autodelete disabled are preserved when you configure a FlexVol volume to automatically delete clones to reclaim space on the volume.

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

Steps

1. Prevent a specific FlexClone file or FlexClone LUN from being automatically deleted by using the `volume file clone autodelete` command.

   Example

   The following example shows how you can disable autodelete for FlexClone LUN lun1_clone contained in vol1:

   ```
   cluster1::> volume file clone autodelete -vserver vs1 -volume vol1 -
   clone-path lun1_clone -enable false
   ```

   A FlexClone file or FlexClone LUN with autodelete disabled cannot be deleted automatically to reclaim space on the volume.

2. Verify that autodelete is disabled for the FlexClone file or FlexClone LUN by using the `volume file clone show-autodelete` command.

   Example

   The following example shows that autodelete is false for the FlexClone LUN lun1_clone:

   ```
   cluster1::> volume file clone show-autodelete -vserver vs1 -clone-
   path vol/vol1/lun1_clone
                                                                Vserver
   Name: vs1
                                                                Clone
   Path: vol/vol1/lun1_clone
   Autodelete Enabled: false
   ```

Commands for configuring deletion of FlexClone files

When clients delete FlexClone files without using the NetApp Manageability SDK, you can use the `volume file clone deletion` commands to enable faster deletion of FlexClone files from a

FlexVol volume. Extensions for and minimum size of FlexClone files are used to enable faster deletion.

You can use the `volume file clone deletion` commands to specify a list of supported extensions and a minimum size requirement for FlexClone files in a volume. The faster deletion method is used only for FlexClone files that meet the requirements. For FlexClone files that do not meet the requirements, the slower deletion method is used.

When clients delete FlexClone files and FlexClone LUNs from a volume by using the NetApp Manageability SDK, the extension and size requirements do not apply because the faster deletion method is always used.

To...	Use this command...
Add an extension to the supported list of extensions for the volume	`volume file clone deletion add-extension`
Change the minimum size of FlexClone files that can be deleted from the volume by using the faster deletion method	`volume file clone deletion modify`
Remove an extension from the supported list of extensions for the volume	`volume file clone deletion remove-extension`
View the supported list of extensions and the minimum size of FlexClone files that clients can delete from the volume by using the faster deletion method	`volume file clone deletion show`

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

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
- 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.

However, 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 occurs in the background. Therefore, when you try to delete 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 is deleted after the block sharing is complete.

Related information

Clustered Data ONTAP 8.3 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 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 by default. However, you can create FlexClone files and FlexClone LUNs with space reservation disabled from a parent file and parent LUN with space reservation enabled if the FlexVol volume lacks space.

If the FlexVol volume does not contain enough space to create a FlexClone file or FlexClone LUN with the same space reservation as that of the parent, then the cloning operation fails.

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 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

Functionality	Qtree	FlexVol volume
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

Obtaining a qtree junction path

You can mount an individual qtree by obtaining the junction path or namespace path of the qtree. The qtree path displayed by the CLI command `qtree show -instance` is of the format `/vol/<volume-name>/<qtree-name>`. However, this path does not refer to the junction path or namespace path of the qtree.

About this task

You need to know the junction path of the volume to obtain the junction path or namespace path of the qtree.

Step

1. Use the `vserver volume junction-path` command to obtain the junction path of a volume.

Example

The following example displays the junction path of the volume named vol1 located on the Storage Virtual Machine (SVM) named vs0:

```
cluster1::> volume show -volume vol1 -vserver vs0 -fields junction-
path

------- ------ -------------

vs0 vol1 /vol1
```

From the above output, the volume's junction path is `/vol1`. Since qtrees are always rooted at the volume, the junction path or namespace path of the qtree will be `/vol1/qtree1`.

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`

If you want to...	Use this command...
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 109

Related tasks

Setting up quotas on an SVM with FlexVol volumes on page 113

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 92

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)

When this occurs...	This event is sent to the EMS...
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 93
How quotas work with qtrees on page 97

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 90

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 88
How default user and group quotas create derived quotas on page 94
How default tree quotas on a FlexVol volume create derived tree quotas on page 97
How default user quotas on a FlexVol volume affect quotas for the qtrees in that volume on page 98

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 97

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 90
How default user quotas on a FlexVol volume affect quotas for the qtrees in that volume on page 98

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 96

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 96

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 93

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 90

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 94

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 100
When a full quota reinitialization is required on page 101

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 100

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 100

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 104

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
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 105

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 105

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 107
How the df command accounts for file size on page 107
How the du command accounts for space usage on page 108

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 106
How the df command accounts for file size on page 107
How the du command accounts for space usage on page 108

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 106
How the ls command accounts for space usage on page 107
How the du command accounts for space usage on page 108

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 106
How the ls command accounts for space usage on page 107
How the df command accounts for file size on page 107

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 113

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 87
Quota targets and types on page 87
Special kinds of quotas on page 88
How quotas are activated on page 100
How you can view quota information on page 102

Modifying (or Resizing) 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 reinitializing 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 the 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 requirements.

Related tasks

Reinitializing quotas after making extensive changes on page 115

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 101
How you can view quota information on page 102

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`
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 87
Considerations for assigning quota policies on page 92
How you can view quota information on page 102

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 100
Why enforced quotas differ from configured quotas on page 104

Related tasks

Modifying (or Resizing) quota limits on page 114
Reinitializing quotas after making extensive changes on page 115

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

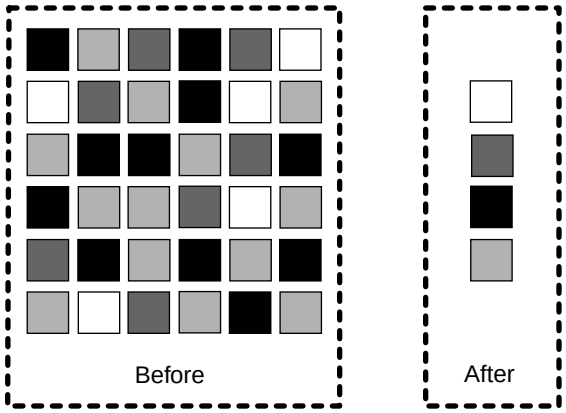
Enabling deduplication on a volume on page 122

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 128
Managing volume efficiency operations using schedules on page 135

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 must understand how data compression works, the types of data compression, and when you must run data compression to achieve maximum saving and performance from your storage system. You must also understand about the incompressible data detection feature to quickly identify the files and compression groups that cannot be compressed.

You must be aware of how to enable data compression on the volume, assign a compression type to a volume, and move between different types of compression.

You can use data compression on primary, secondary, and tertiary storage tiers.

Related tasks

Enabling data compression on a volume on page 126

How data compression works

Data compression stores more data in less space and reduce the time and bandwidth required to replicate data during volume SnapMirror transfers. You need to choose the type of compression based on your requirement and the configurations of your storage system. You also need to understand and choose when you want to run data compression on your volumes.

You can use data compression on regular files and LUNs, but not on storage system internal files, Windows NT streams, and volume metadata.

Data compression works by combining a set of data blocks in a *compression group*. This compression group is compressed and stored as fewer blocks, thereby saving space in the storage system.

The following are the types of data compression:

- Secondary compression:
  Secondary compression divides the file to be compressed into compression groups of 32k blocks. This compression group is then compressed and stored as fewer blocks, which reduces the size of the data to a large extent, thereby increasing the space in the storage system.
  You must be aware of the following considerations related to secondary compression:
  - Secondary compression is supported on all releases of Data ONTAP.
  - If you enable data compression on HDD aggregates, then by default, secondary compression is enabled. You can enable secondary compression on All Flash FAS platforms and Flash Pool aggregates based on the configurations of your storage system.
  - Inline compression and postprocess compression can perform secondary compression on HDD aggregates and Flash Pool aggregates and only inline compression can perform secondary compression on All Flash FAS platforms.
- Adaptive compression:
  Adaptive compression divides the file to be compressed into compression groups of 8k blocks. This compression group is then compressed and stored as a single block. When a user requests data from this compression group, it takes much less time to decompress and provide that data to the user, thereby improving the read performance. Adaptive compression provides relatively less savings as compared to secondary compression, but with better performance.
  You must be aware of the following considerations related to adaptive compression:

- Adaptive compression is supported only in Data ONTAP 8.3.1 and later releases.
- If you enable data compression on All Flash FAS platforms and Flash Pool aggregates, then by default, adaptive compression is enabled. You can enable adaptive compression on HDD aggregates based on the configurations of your storage system.
- If you create a volume in an All Flash FAS platform, then by default, adaptive compression is enabled.
- Inline compression and postprocess compression can perform adaptive compression on HDD aggregates and Flash Pool aggregates and only inline compression can perform adaptive compression on All Flash FAS platforms.

Adaptive compression is more suited for All Flash FAS platforms and Flash Pool aggregates and provides better performance, specifically for databases and other workloads that have 8K input/output (I/O) sizes.

You also need to understand when you want to run data compression on your volumes as this helps in space savings and reducing cost of the data stored. You can run data compression while data is being written to the volume (inline compression), after the complete data is written to the volume (postprocess compression), or both.

Note:

- In All Flash FAS platforms, only inline compression is supported.
- In HDD aggregates and Flash Pool aggregates, both, inline and postprocess compression are supported.

- Inline compression:
  Inline compression checks if data can be compressed, compresses data, and then writes data to the volume. This provides space savings for more data to be stored. However, data that cannot be compressed or data bypassed by inline compression is written in the uncompressed format to the volume.
  If you create a volume in an All Flash FAS platform, and if all of the nodes in the cluster are running Data ONTAP 8.3.1, then by default, inline compression is enabled.
- Postprocess compression
  Postprocess compression runs on all files, irrespective of whether the file is compressible or not, after all the data is written to the volume. Post postprocess compression compresses the new data writes to the volume that were not compressed initially (if inline compression is enabled), and rewrites them as compressed data to the volume. The postprocess compression operation runs as a low-priority background process.
  Postprocess compression is not supported on volumes in an All Flash FAS platform. If a volume is upgraded from an earlier release to Data ONTAP 8.3.1 or is moved from another platform to an All Flash FAS platform, no postprocess compression takes place on those volumes even if postprocess compression is enabled. Each time the postprocess compression is run, an EMS message is generated informing you that postprocess compression was skipped.

If both inline and postprocess compression are enabled, then postprocess compression compresses only the blocks on which inline compression is not run. This includes blocks that were bypassed by inline compression, such as small, partial compression group overwrites.

For 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 detects whether a file is compressible and, for large files, whether a compression group within a file is compressible. This helps in saving system resources used by inline compression for compressing incompressible files or compression groups.

When you enable incompressible data detection, inline compression checks whether a compression group can be compressed (for files under 500 MB). 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 whether the incompressible data flag is set for the file. If the flag is set, then inline compression is not attempted on the file, thereby saving system resources. When postprocess compression is enabled, postprocess compression runs on all files, regardless of whether the file is compressible. If postprocess compression compresses at least one compression group in an incompressible file, then it clears the incompressible data flag for that file. During the next compression attempt, as the flag is cleared, inline compression can run on this file to achieve space savings.

Inline compression performs a quick check (for files equal to or greater than 500 MB) on the first 4-KB block of each compression group to determine whether 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.

Note: Incompressible data detection and quick check are not supported on All Flash FAS platforms.

The `volume efficiency modify` command man page contains more information about enabling or disabling incompressible data detection and modifying the minimum file size to attempt a quick check on a file.

Configurations supported in All Flash FAS platforms, HDD aggregates, and Flash Pool aggregates

You must understand the data compression features that are supported in All Flash FAS platforms, HDD aggregates, and Flash Pool aggregates to effectively save space and improve the performance of your storage system.

Data compression features	Mode of compression	Volumes in All Flash FAS platforms	Volumes in HDD and Flash Pool aggregates
Secondary compression	Inline compression	Supported	Supported
	Postprocess compression	Not Supported	Supported
Adaptive compression	Inline compression	Supported	Supported
	Postprocess compression	Not supported	Supported
Incompressible data detection		Not supported	Supported
Quick check		Not supported	Supported

Enabling data compression on a volume

You can enable data compression on a FlexVol volume or an Infinite Volume to achieve space savings by using the `volume efficiency modify` command. You can also assign a compression type to your volume, if you do not require the default compression type.

Before you begin

You must have enabled deduplication on the volume.

Note: Deduplication only needs to be enabled and does not need to be running on the volume.

Enabling deduplication on a volume on page 122

About this task

- In HDD aggregates and Flash Pool aggregates, you can enable both inline and postprocess compression or only postprocess compression on a volume.
  If you are enabling both, then you must enable postprocess compression on the volume before enabling inline compression.
- In All Flash FAS platforms, only inline compression is supported.
  Before enabling inline compression, you must enable postprocess compression on the volume. However, as postprocess compression is not supported in All Flash FAS platforms, no postprocess compression takes place on those volumes and an EMS message is generated informing you that postprocess compression was skipped.
- The compression type is automatically assigned based on the aggregates, platform, and the Data ONTAP release of your storage system:

Platform/Aggregates	Data ONTAP release	Compression Type
All Flash FAS	Data ONTAP 8.3.1	Adaptive compression
All Flash FAS	Data ONTAP 8.3.0	Secondary compression
Flash Pool aggregates	Data ONTAP 8.3.1	Adaptive compression
Flash Pool aggregates	Data ONTAP 8.3.0	Secondary compression
HDD aggregates	Data ONTAP 8.3.1	Secondary compression
HDD aggregates	Data ONTAP 8.3.0	Secondary compression

Choices

- Use the `volume efficiency modify` command to enable data compression with the default compression type.

  #### Example

  The following command enables postprocess compression on volume VolA:

  ```
  volume efficiency modify -vserver vs1 -volume VolA -compression true
  ```

  The following command enables both postprocess and inline compression on volume VolA:

  ```
  volume efficiency modify -vserver vs1 -volume VolA -compression true -
  inline-compression true
  ```

- Use the `volume efficiency modify` command, in the advanced privilege level, to enable data compression and assign a specific compression type.

1. Use the `set -privilege advanced` command to change the privilege level to advanced.
2. Use the `volume efficiency modify` command to assign a compression type to a volume.

   Example

   The following command enables postprocess compression and assigns the adaptive compression type to volume VolA:

   `volume efficiency modify -vserver vs1 -volume VolA -compression true -compression-type adaptive`

   The following command enables both postprocess and inline compression and assigns the adaptive compression type to volume VolA:

   `volume efficiency modify -vserver vs1 -volume VolA -compression true -compression-type adaptive -inline-compression true`
3. Use the `set -privilege admin` command to change the privilege level to admin.

Related concepts

Managing volume efficiency operations using policies on page 128
Managing volume efficiency operations using schedules on page 135

Moving between secondary and adaptive compression

You can move between secondary compression and adaptive compression depending on the amount of data read. Adaptive compression is typically preferred when there are lot of random reads on the system and higher performance is required. Secondary compression is preferred when data is written sequentially, and higher compression savings are required.

About this task

The default compression is selected based on your aggregates and platform.

Steps

1. Use the `volume efficiency modify` command to disable data compression on the volume.

   Example

   The following command disables data compression on the volume vol1:

   `volume efficiency modify -compression false -inline-compression false -volume vol1`
2. Use the `set -privilege advanced` command to change the privilege level to advanced.
3. Use the `volume efficiency undo` command to decompress the compressed data.

   Example

   The following command decompress the compressed data on the volume vol1:

   `volume efficiency undo -vserver vs1 -volume vol1 -compression true`
4. Use the `volume efficiency show` command to ensure that the status of the operation is idle.

   Example

   The following command displays the status of an efficiency operation on volume vol1:

   `volume efficiency show -vserver vs1 -volume vol1`

5. Use the `volume efficiency modify` command to enable data compression and set the type of compression.

   Example

   The following command enables data compression and sets the compression type as secondary compression on the volume vol1:

   `volume efficiency modify -vserver vs1 -volume vol1 -compression true -compression-type secondary`

6. Use the `set -privilege admin` command to change the privilege level to admin.
7. Optional: Use the `volume efficiency modify` command to enable inline compression.

   Example

   The following command enables inline compression on the volume vol1:

   `volume efficiency modify -vserver vs1 -volume vol1 -inline-compression true`

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 run deduplication or data compression on a FlexVol volume or Infinite Volume either by scheduling the operations to start at a specific time or by specifying a threshold percentage after which the operations are triggered.

You can schedule a deduplication or data compression operation by creating job schedules that are enclosed within the efficiency policies or you can specify a threshold percentage, which waits for the new data to exceed the specified percentage and then triggers the deduplication or data compression

operation. You can disassociate an efficiency policy to stop running any further scheduled based deduplication and data compression operations on the volume by using the `volume efficiency modify` command.

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

Understanding predefined efficiency policies

Starting with Data ONTAP 8.3, you can configure a volume with efficiency policies to achieve additional space savings. You can configure a volume to run inline compression without a scheduled or manually started background efficiency operation configured on the volume.

When you create a Storage Virtual Machine (SVM), the following efficiency policies are created automatically and cannot be deleted:

- Default
  You can configure a volume with the efficiency policy to run the scheduled deduplication operations on the volume.
- Inline-only
  You can configure a volume with the inline-only efficiency policy and enable inline compression, to run inline compression on the volume without any scheduled or manually started background efficiency operations.

For more information about the inline-only and default efficiency policies, see the man pages.

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*.

Note: You can run deduplication or data compression operations at a scheduled time, or by creating a schedule with a specific duration, or by specifying a threshold percentage, which waits for the new data to exceed the specified percentage and then triggers the deduplication or data compression operation.

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 pol1 that triggers an efficiency operation daily:

```
volume efficiency policy create -vserver vs1 -policy pol1 -schedule
daily
```

The following command creates a volume efficiency policy named pol2 that triggers an efficiency operation when the threshold percentage reaches 20%:

```
volume efficiency policy create -vserver vs1 -policy pol2 -type
threshold -start-threshold-percent 20%
```

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 and 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.

Examples

The following command modifies the volume efficiency policy named policy1 to run every hour:

```
volume efficiency policy modify -vserver vs1 -policy policy1 -schedule
hourly
```

The following command modifies a volume efficiency policy named pol2 to threshold 30%:

```
volume efficiency policy modify -vserver vs1 -policy pol1 -type
threshold -start-threshold-percent 30%
```

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
```

Disassociating a volume efficiency policy from a volume

You can disassociate a volume efficiency policy from a volume to stop running any further schedule-based deduplication and data compression operations on the volume. Once you disassociate a volume efficiency policy, you have it trigger it manually.

Step

1. Use the `volume efficiency modify` command to disassociate a volume efficiency policy from a volume.

Example

The following command disassociates the volume efficiency policy from volume VolA:

```
volume efficiency modify -vserver vs1 -volume VolA -policy -
```

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.

Note: You cannot delete the *inline-only* and the *default* predefined efficiency policy.

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 manually

You can manage how the efficiency operations run on a FlexVol volume or Infinite Volume by running efficiency operations manually.

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 on a volume.

About this task

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.

Example

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
```

Using checkpoints to resume efficiency operation

The checkpoints are used internally to 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, the 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:

- in each stage or substage of the operation
- when you run the `sis stop` command
- when the duration expires

Resuming a halted efficiency operation

If an efficiency operation is halted due to a system halt, system disruption, or reboot, you can resume the efficiency operation from the same point by using the `volume efficiency start` command

with the checkpoint option. This helps in saving time and resources by not having to restart the efficiency operation from the beginning.

About this task

If you enabled only deduplication on the volume, deduplication runs on the data. If you enabled both deduplication and data compression on a volume, then data compression runs first, 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 with the `-use-checkpoint` option to resume an efficiency operation.

Example

The following command enables you to resume an efficiency operation on new data on volume VolA:

```
volume efficiency start -vserver vs1 -volume VolA -use-checkpoint true
```

The following command enables you to resume an efficiency operation on existing data on 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 run the efficiency operations manually on the data that exists in FlexVol volumes 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 you enable only deduplication on a volume, then deduplication runs on the data. if you enable both deduplication and data compression on a volume, then data compression runs first, 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 and 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 *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; the remaining operations are queued.

Note: In All Flash FAS platforms, you cannot compress the existing data. Only new data that is written is compressed in All Flash FAS platforms. If you try to compress existing data on All Flash FAS platforms, an EMS message is generated informing the user that this operation was skipped.

Step

1. Use the `volume efficiency start -scan-old-data` command to run deduplication, or data compression followed by deduplication, manually on the existing data.

Example

The following command enables you to run deduplication, or data compression followed by deduplication, manually on the existing data in 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

Managing volume efficiency operations using schedules

You can manage how the efficiency operations run on a FlexVol volume or Infinite Volume by using a schedule or by depending on the amount of new data written to the FlexVol volume or Infinite Volume.

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
```

Running efficiency operations using scheduling

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. You can also view the status, state, type of compression, 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 that is assigned the adaptive compression type:

`volume efficiency show -instance -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
- MetroCluster configuration

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.

Related concepts

How fractional reserve works with deduplication on page 139
How Snapshot copies work with deduplication on page 140
How volume SnapMirror works with deduplication on page 140
How SnapRestore works with deduplication on page 140
How OnCommand Unified Manager server works with deduplication on page 140
How deduplication works with data compression on page 141
How FlexClone volumes work with deduplication on page 141
How HA pairs work with deduplication on page 142
How DataMotion for Volumes works with deduplication on page 142
How SnapVault backup works with deduplication on page 142
How virtual machine alignment works with deduplication on page 143
How a MetroCluster configuration works with deduplication on page 143

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

Considerations for setting fractional reserve on page 29

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 5.2R1, 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 5.2R1, 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 5.2R1.

For more information about using deduplication with the NetApp Management Console data protection capability, see the *OnCommand Unified Manager Administration Guide*.

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, see the *OnCommand Unified Manager Administration Guide*.

Deduplication and Operations Manager in OnCommand Unified Manager server

You can perform deduplication operations from Operations Manager in OnCommand Unified Manager server.

You can generate reports or graphs summarizing space savings for file and LUN clones.

For more information about using deduplication with Operations Manager, see the *OnCommand Unified Manager Administration Guide*.

Related information

Documentation on the NetApp Support Site: mysupport.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 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.

Starting from Data ONTAP 8.3, you can configure the storage system to retry the cutover for the volume move operation using the `retry-on-failure` option. When this option is configured, the volume move operation repeats the cutover phase and waits (hibernates) for an hour.

The deduplication volumes are unlocked between the waiting period of the cutover phase for the volume move operation. This can result in large amount of space consumption and loss of metadata. Therefore, you must run deduplication during the cutover phase of the volume move operation.

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*.

How a MetroCluster configuration works with deduplication

During a MetroCluster switchback process, all active deduplication operations are stopped on volumes whose containing aggregates are being switched back and a checkpoint is recorded for each of the deduplication operation that is stopped.

After the MetroCluster switchback process is complete, if a volume has a schedule or policy associated with it, then the deduplication operations restart from the recorded checkpoint. If the volume does not have any schedule or policy associated with it, you can manually restart the deduplication operations from the recorded checkpoint.

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
- MetroCluster configuration

Read reallocation and extents are not supported on compressed volumes.

Related concepts

How fractional reserve works with data compression on page 144
How Snapshot copies work with data compression on page 144
How volume SnapMirror works with data compression on page 144
How tape backup works with data compression on page 145
How volume-based SnapRestore works with data compression on page 145
How single file SnapRestore works with data compression on page 145
How deduplication works with data compression on page 146
How FlexClone volumes work with data compression on page 146
How FlexClone files work with data compression on page 146
How HA pairs work with data compression on page 146
How Flash cache cards work with data compression on page 146
How DataMotion for Volumes works with data compression on page 146
How Flash Pool aggregates work with data compression on page 147
How SnapVault backup works with data compression on page 147
How MetroCluster configurations work with data compression on page 147

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 on page 29

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 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: mysupport.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: mysupport.netapp.com

How single file SnapRestore works with data compression

Data compression is supported with single file SnapRestore operation. The single file SnapRestore operation restores data from the Snapshot copy to the active file system and retains the original space savings.

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: mysupport.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

Data compression can run on FlexVol volume that contain FlexClone files. Fully cloned files can be created on FlexVol volumes that have data compression enabled. Sub-file cloned files can be created on FlexVol volumes only if adaptive compression is enabled.

If incompressible data detection is enabled on the volume, then the incompressible flag is inherited by the cloned file.

Note: FlexClone file operation is not supported if the Snapshot copy and the active file system have different compression types.

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 *Clustered Data ONTAP High-Availability Configuration Guide*.

Related information

Documentation on the NetApp Support Site: mysupport.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, both the source and destination volumes must run the same version of Data ONTAP.

You can move volumes between other platforms and All Flash FAS platforms and the destination volume will retain the same compression type as that of the source. If the source volume has postprocess compression enabled and the destination volume is on an All Flash FAS platform, an EMS message is generated informing the user that postprocess compression was skipped.

How Flash Pool aggregates work with data compression

Data compression is supported on Flash Pool aggregates; the compressed blocks can be used for read and write caching in the solid-state disk (SSD) cache of the Flash Pool aggregates.

How SnapVault backup works with data compression

You can logically replicate a SnapVault primary volume to a SnapVault secondary volume. During replication, 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`** or set the efficiency policy to **`inline-only`**.
- 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 is enabled on the secondary volume when the previous Snapshot copy is 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*.

How MetroCluster configurations work with data compression

Data compression is supported on MetroCluster configurations. Starting with Data ONTAP 8.3.1, the compression type is automatically assigned based on the aggregates, platform, and the Data ONTAP release of your storage system. For a MetroCluster switchback process to be successful, both the

source and destination volumes must be running on Data ONTAP 8.3.1. If not, the MetroCluster switchback of the volume fails.

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*
- *FlexClone file and FlexClone LUN limits*

Storage limits for Cloud ONTAP are documented in the *Cloud ONTAP Release Notes*.

Volume limits

Storage object	Limit	Native storage	Storage arrays	Data ONTAP Edge
Array LUNs	Minimum size for root volume [1]	N/A	Model-dependent	N/A
Files	Maximum size	16 TB	16 TB	16 TB
	Maximum per volume[2]	Volume size dependent, up to 2 billion	Volume size dependent, up to 2 billion	Volume size dependent, up to 2 billion
FlexClone volumes	Hierarchical clone depth[3]	499	499	499
FlexVol volumes	Maximum per node[1]	Model-dependent	Model-dependent	200
	Maximum per node per SVM[4]	Model-dependent	Model-dependent	200
	Minimum size	20 MB	20 MB	20 MB
	Maximum size[1]	Model-dependent	Model-dependent	Model-dependent
FlexVol root volumes	Minimum size[1]	Model-dependent	Model-dependent	Model-dependent
LUNs	Maximum per node[4]	Model-dependent	Model-dependent	1,024
	Maximum per cluster[4]	Model-dependent	Model-dependent	1,024
	Maximum per volume[4]	Model-dependent	Model-dependent	512
	Maximum size	16 TB	16 TB	16 TB
Qtrees	Maximum per FlexVol volume	4,995	4,995	4,995
Snapshot copies	Maximum per FlexVol volume or Infinite Volume[5]	255	255	255

Storage object	Limit	Native storage	Storage arrays	Data ONTAP Edge
Volumes	Maximum per cluster for NAS[6]	12,000	12,000	200
	Maximum per cluster with SAN protocols configured[7]	Model-dependent	Model-dependent	200

Notes:

1. See the *Hardware Universe*.
2. 2 billion = 2 × 10 to the 9th power.
3. The maximum depth of a nested hierarchy of FlexClone volumes that can be created from a single FlexVol volume.
4. This limit applies only in SAN environments.
   See the *Clustered Data ONTAP SAN Configuration Guide*.
5. The use of certain Data ONTAP capabilities could reduce this limit.
   See the *Clustered Data ONTAP Data Protection Guide*.
6. Infinite Volumes do not count against this limit, but their constituent volumes do.
7. Infinite Volumes do not count against this limit, but their constituent volumes do.
   See the *Clustered Data ONTAP SAN Configuration Guide*.

FlexClone file and FlexClone LUN limits

Limit	Native storage	Storage arrays	Data ONTAP Edge
Maximum per file or LUN[1]	32,767	32,767	32,767
Maximum total shared data per FlexVol volume	640 TB	640 TB	100 TB

Note:

1. 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.

Related information

Find the Release Notes for your version of Cloud ONTAP

Copyright information

Trademark information

How to send comments about documentation and receive update notifications

You can help us to improve the quality of our documentation by sending us your feedback. You can receive automatic notification when production-level (GA/FCS) documentation is initially released or important changes are made to existing production-level documents.

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.

If you want to be notified automatically when production-level documentation is released or important changes are made to existing production-level documents, follow Twitter account @NetAppDoc.

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

G

H

I

J

L

M

N

O

P

Q

R

S

T

U

V