



## 集群模式 Data ONTAP® 8.3

### 逻辑存储管理指南



NetApp 北京  
北京市朝阳区东大桥路 9 号侨福  
芳草地 C 座 6 层 606 室  
邮编: 100020

电话: 86-10-59293000  
传真: 86-10-59293099  
支持电话: 86-10-59293008  
网址: <http://www.netapp.com/cn/>  
意见反馈: [ng-gpso-china-documents@netapp.com](mailto:ng-gpso-china-documents@netapp.com)

部件号: 215-09603\_A0  
2015 年 1 月



# 目录

什么是逻辑存储 .....	8
卷的工作原理 .....	9
什么是 FlexVol 卷 .....	9
FlexVol 卷提供的功能 .....	9
各项 FlexVol 卷功能之间的区别 .....	10
什么是无限卷 .....	11
无限卷提供的功能 .....	11
FlexVol 卷与无限卷的对比 .....	12
FlexVol 卷和无限卷如何共享聚合 .....	14
安全模式对数据访问具有哪些影响 .....	14
使用传统机会锁和租用机会锁提高客户端性能 .....	15
什么是系统卷 .....	15
<b>使用 FlexVol 卷 .....</b>	<b>16</b>
FlexVol 卷如何与 SVM 配合使用 .....	16
SVM 如何影响哪些聚合可与 FlexVol 卷进行关联 .....	16
SVM 如何限制自身可包含的 FlexVol 卷数 .....	16
SVM 对 FlexVol 卷的语言有哪些影响 .....	17
卷联合使用规则 .....	17
如何使用空间管理功能 .....	18
文件和 LUN 预留的工作原理 .....	19
FlexVol 卷的卷保证工作原理 .....	19
设置预留百分比的注意事项 .....	23
将卷配置为在已满时自动提供更多空间 .....	24
将卷配置为自动增长和缩减其大小 .....	27
如何解决 FlexVol 卷填满度警报和分配过度警报 .....	28
如何解决聚合填满度警报和分配过度警报 .....	30
如何确定卷或聚合中的空间使用量 .....	32
在 FlexVol 卷中创建空间的方法 .....	42
在聚合中创建空间的方法 .....	44
更改文件容量或目录容量时的警告和注意事项 .....	45
更改 FlexVol 卷上允许的最大文件数的注意事项 .....	45
为 FlexVol 卷增加最大目录大小时的注意事项 .....	45

对 Flash Pool 聚合中的卷使用缓存策略 .....	45
节点根卷和根聚合需要遵循的规则 .....	47
FlexVol 卷基本管理 .....	47
创建 FlexVol 卷 .....	47
删除 FlexVol 卷 .....	48
使用存储服务质量 (QoS) 控制和监控 FlexVol 卷的 I/O 性能 .....	48
显示文件或索引节点使用量 .....	49
用于管理 FlexVol 卷的命令 .....	50
用于显示空间使用量信息的命令 .....	50
移动和复制卷（仅限集群管理员） .....	51
FlexVol 卷移动的工作原理 .....	51
用于移动卷的命令 .....	52
移动卷时的注意事项和建议 .....	52
在 SAN 中移动卷的要求 .....	54
移动卷 .....	54
复制卷的方法 .....	55
使用 FlexClone 卷创建 FlexVol 卷的高效副本 .....	56
了解 FlexClone 卷 .....	56
FlexClone 卷和共享 Snapshot 副本 .....	57
如何将卷 SnapMirror 复制与 FlexClone 卷配合使用 .....	58
从 SnapMirror 源卷或目标卷创建 FlexClone 卷的注意事项 .....	58
如何将 FlexClone 卷从其父卷拆分 .....	58
FlexClone 卷和 LUN .....	59
了解数据保护 FlexClone 卷 .....	59
创建 FlexClone 卷 .....	59
从父卷中拆分 FlexClone 卷 .....	60
确定 FlexClone 卷所使用的空间 .....	62
使用 FlexClone 文件和 FlexClone LUN 创建文件和 LUN 的高效副本 .....	63
FlexClone 文件和 FlexClone LUN 的优势 .....	63
FlexClone 文件和 FlexClone LUN 的工作原理 .....	63
处理 FlexClone 文件和 FlexClone LUN 的注意事项 .....	65
创建经过空间优化的 FlexClone 文件和 FlexClone LUN 的示例 .....	66
分摊负载如何确定用于创建和删除 FlexClone 文件和 FlexClone LUN 的节点容量 .....	67
创建 FlexClone 文件或 FlexClone LUN .....	68
查看用于创建和删除 FlexClone 文件和 FlexClone LUN 的节点容量 ....	69

查看通过 FlexClone 文件和 FlexClone LUN 节省的空间 .....	70
用于删除 FlexClone 文件和 FlexClone LUN 的方法 .....	70
FlexVol 卷如何从 FlexClone 文件和 FlexClone LUN 回收可用空间 .....	71
用于配置 FlexClone 文件删除的命令 .....	75
支持与 FlexClone 文件和 FlexClone LUN 配合使用的功能 .....	75
使用 qtree 对 FlexVol 卷进行分区 .....	78
何时使用 qtree .....	79
qtree 与 FlexVol 卷相比有何不同之处 .....	79
获取 qtree 接合路径 .....	80
qtree 名称限制 .....	80
使用 qtree 可以对镜像执行哪些操作 .....	80
将目录转换为 qtree .....	81
用于管理 qtree 的命令 .....	83
使用配额来限制或跟踪资源使用量 .....	83
为什么使用配额 .....	83
配额流程概述 .....	84
什么是配额规则、配额策略和配额 .....	86
配额目标和类型 .....	86
特殊的配额类型 .....	87
如何应用配额 .....	92
分配配额策略的注意事项 .....	92
如何对用户和组使用配额 .....	93
如何对 qtree 使用配额 .....	97
qtree 更改如何影响配额 .....	100
如何激活配额 .....	101
如何查看配额信息 .....	104
配额报告所显示的空间使用量与 UNIX 客户端所显示的空间使用量 之间的差别 .....	108
配额配置示例 .....	111
在包含 FlexVol 卷的 SVM 上设置配额 .....	117
修改配额限制（或调整大小） .....	118
执行大量更改后重新初始化配额 .....	119
验证配额升级的状态 .....	120
用于管理配额规则和配额策略的命令 .....	121
用于激活和修改配额的命令 .....	122
使用重复数据删除和数据压缩提高存储效率 .....	<b>123</b>

如何设置效率操作 .....	123
配置重复数据删除 .....	123
重复数据删除工作原理 .....	123
什么是重复数据删除元数据 .....	124
使用重复数据删除的准则 .....	125
在卷上启用重复数据删除 .....	126
在卷上禁用重复数据删除 .....	127
配置数据压缩 .....	127
数据压缩工作原理 .....	127
数据压缩如何检测不可压缩数据并节省系统资源 .....	128
在卷上启用数据压缩 .....	128
在卷上禁用数据压缩 .....	129
使用策略管理卷效率操作 .....	130
使用卷效率优先级确定效率操作的优先顺序 .....	130
了解预定义的效率策略 .....	130
创建卷效率策略以运行效率操作 .....	131
将卷效率策略分配给卷 .....	131
修改卷效率策略 .....	132
查看卷效率策略 .....	132
解除卷效率策略与卷的关联 .....	133
删除卷效率策略 .....	133
手动管理卷效率操作 .....	134
手动运行效率操作 .....	134
使用检查点来恢复效率操作 .....	135
对现有数据手动运行效率操作 .....	136
使用计划管理卷效率操作 .....	137
根据写入的新数据量运行效率操作 .....	137
使用计划来运行效率操作 .....	137
监控卷效率操作 .....	138
查看效率操作的状态 .....	138
查看效率空间节省 .....	138
查看 FlexVol 卷的效率统计信息 .....	139
停止卷效率操作 .....	140
有关从卷中移除空间节省的信息 .....	140
重复数据删除与 Data ONTAP 功能的互操作性 .....	141
预留百分比如何与重复数据删除配合使用 .....	142

Snapshot 副本如何与重复数据删除配合使用 .....	142
卷 SnapMirror 如何与重复数据删除配合使用 .....	142
SnapRestore 如何与重复数据删除配合使用 .....	143
OnCommand Unified Manager 服务器如何与重复数据删除配合使用 ..	143
重复数据删除如何与数据压缩配合使用 .....	144
FlexClone 卷如何与重复数据删除配合使用 .....	144
HA 对如何与重复数据删除配合使用 .....	144
DataMotion for Volumes 如何与重复数据删除配合使用 .....	144
SnapVault 备份如何与重复数据删除配合使用 .....	145
虚拟机对齐如何与重复数据删除配合使用 .....	145
MetroCluster 配置如何与重复数据删除配合使用 .....	146
数据压缩与 Data ONTAP 功能的互操作性 .....	146
预留百分比如何与数据压缩配合使用 .....	147
Snapshot 副本如何与数据压缩配合使用 .....	147
卷 SnapMirror 如何与数据压缩配合使用 .....	147
磁带备份如何与数据压缩配合使用 .....	148
基于卷的 SnapRestore 如何与数据压缩配合使用 .....	148
单文件 SnapRestore 如何与数据压缩配合使用 .....	148
重复数据删除如何与数据压缩配合使用 .....	149
FlexClone 卷如何与数据压缩配合使用 .....	149
FlexClone 文件如何与数据压缩配合使用 .....	149
HA 对如何与数据压缩配合使用 .....	149
闪存卡如何与数据压缩配合使用 .....	149
DataMotion for Volumes 如何与数据压缩配合使用 .....	150
Flash Pool 聚合如何与数据压缩配合使用 .....	150
SnapVault 备份如何与数据压缩配合使用 .....	150
存储限制 .....	<b>152</b>
版权信息 .....	<b>154</b>
商标信息 .....	<b>155</b>
意见反馈 .....	<b>156</b>
索引 .....	<b>157</b>

## 什么是逻辑存储

---

逻辑存储指的是由 Data ONTAP 提供的存储资源，它们与物理资源不相关。

逻辑存储资源与 Storage Virtual Machine (SVM，以前称为 Vserver) 关联，它们独立于任何特定的物理存储资源（例如磁盘、阵列 LUN 或聚合）而存在。逻辑存储资源包括各种类型的卷和 qtree 以及可与这些资源结合使用的功能和配置（例如 Snapshot 副本、重复数据删除、压缩和配额）。

有关 SVM 的详细信息，请参见《适用于集群管理员的集群模式 *Data ONTAP* 系统管理指南》和《适用于 *SVM* 管理员的集群模式 *Data ONTAP* 系统管理指南》。

### 相关概念

[使用 \*FlexVol\* 卷](#)（第 16 页）

[使用 \*qtree\* 对 \*FlexVol\* 卷进行分区](#)（第 78 页）

[使用重复数据删除和数据压缩提高存储效率](#)（第 123 页）

[使用配额来限制或跟踪资源使用量](#)（第 83 页）

## 卷的工作原理

---

卷是一种数据容器，可用于对数据进行分区和管理。了解卷的类型及其相关功能有助于您设计存储架构，以便实现最高的存储效率并简化管理。

卷是最高级别的逻辑存储对象。与由物理存储资源组成的聚合不同，卷完全是逻辑对象。

Data ONTAP 提供两种类型的卷：FlexVol 卷和无限卷。此外，还有多个卷变体，例如 FlexClone 卷、数据保护镜像和负载共享镜像。这两种类型的卷仅支持部分卷变体，这两种类型的卷均可支持 Data ONTAP 效率功能、数据压缩和重复数据删除。

在 NAS 环境中，卷包含文件系统；在 SAN 环境中，卷包含 LUN。

卷与一个 Storage Virtual Machine (SVM) 相关联。SVM 是一个虚拟管理实体或服务器，负责将各种集群资源整合成易于管理的单一单元。创建卷时，要指定与其关联的 SVM。卷的类型（FlexVol 卷或无限卷）由一个不可变的 SVM 属性决定。

卷具有语言属性。卷的语言决定了 Data ONTAP 显示该卷的文件名和数据时所使用的字符集。卷的语言的默认值为 SVM 的语言。

卷依靠与其关联的聚合获得物理存储；它们不直接与任何具体的存储对象（如磁盘或 RAID 组）相关联。如果集群管理员已为某个 SVM 分配了特定的聚合，则只能使用这些聚合为与该 SVM 相关联的卷提供存储。这会影响到卷的创建，以及 FlexVol 卷在聚合间的复制与移动。

有关无限卷的详细信息，请参见《集群模式 *Data ONTAP* 无限卷管理指南》。

有关 SVM 的详细信息，请参见《适用于集群管理员的集群模式 *Data ONTAP* 系统管理指南》。

有关数据保护镜像的详细信息，请参见《集群模式 *Data ONTAP* 数据保护指南》。

有关聚合、磁盘和 RAID 组等物理存储资源的详细信息，请参见《集群模式 *Data ONTAP* 物理存储管理指南》。

## 什么是 FlexVol 卷

FlexVol 卷是一个与包含 FlexVol 卷的 Storage Virtual Machine (SVM) 关联的数据容器。该卷从单个关联聚合获取存储，可能会与其他 FlexVol 卷或无限卷共享该聚合。该卷可用于在 NAS 环境中保存文件或者在 SAN 环境中保存 LUN。

## FlexVol 卷提供的功能

FlexVol 卷可用于将数据分为可根据数据用户需求进行配置且易于管理的单个对象。

可以使用 FlexVol 卷执行以下操作：

- 使用 FlexClone 技术快速创建卷的克隆，而无需复制整个卷。

## 10 | 逻辑存储管理指南

- 使用重复数据删除和压缩技术来降低卷的空间要求。
- 创建卷的 Snapshot 副本，以便保护数据。
- 通过使用配额来限制卷中可供用户、组、或 qtree 使用的空间量。
- 使用 qtree 对卷进行分区。
- 创建负载共享镜像，从而平衡节点间的负载。
- 在聚合之间以及存储系统之间移动卷。
- 允许客户端使用 Data ONTAP 支持的任何文件访问协议对卷进行访问。
- 在卷变满时，设置一个卷以提供更多存储空间。
- 使用精简配置来创建大于当前的可用物理存储的卷。

### 相关概念

[使用 FlexClone 卷创建 FlexVol 卷的高效副本](#)（第 56 页）

[使用 FlexClone 文件和 FlexClone LUN 创建文件和 LUN 的高效副本](#)（第 63 页）

[配置重复数据删除](#)（第 123 页）

[配置数据压缩](#)（第 127 页）

[移动和复制卷（仅限集群管理员）](#)（第 51 页）

### 相关任务

[将卷配置为在已满时自动提供更多空间](#)（第 24 页）

[将卷配置为自动增长和缩减其大小](#)（第 27 页）

## 各项 FlexVol 卷功能之间的区别

了解各项 FlexVol 卷功能之间的区别有助于选择最符合自身要求的功能。

下表总结了这些区别：

功能	访问类型 (读写或只读)	是否自动挂 载?	完整副本 还是共享 块?	位置	即时操作还是 持续较长时间 的操作?
FlexClone	与父卷相同	是	共享块	相同聚合以及 相同节点	即时
Snapshot 副 本	只读	是	共享块	相同聚合以及 相同节点	即时

功能	访问类型 (读写或只读)	是否自动挂载?	完整副本还是共享块?	位置	即时操作还是持续较长时间的操作?
数据保护镜像	只读	否	完整副本	相同或不同聚合; 相同或不同节点; 相同或不同集群	持续较长时间的操作
负载共享镜像	只读	否	完整副本	相同或不同聚合; 相同或不同节点	持续较长时间的操作
移动 (DataMotion for Volumes)	与原始卷相同	是	完整副本, 然后删除原始卷	不同聚合; 相同或不同节点	持续较长时间的操作

除了数据保护镜像功能可以跨多个集群和 Storage Virtual Machine (SVM) 运行之外, 以上所有卷功能均在同一个 SVM 中运行。

执行持续较长时间的操作所需的时间取决于卷大小。例如, 移动一个大小为 1 TB 的卷可能需要几小时。

#### 相关概念

[使用 FlexClone 卷创建 FlexVol 卷的高效副本](#) (第 56 页)

[移动和复制卷 \(仅限集群管理员\)](#) (第 51 页)

## 什么是无限卷

无限卷是一个可扩展卷, 可存储最多 20 亿个文件和数十 PB 的数据。

利用无限卷, 您可以在一个庞大逻辑实体中管理多个 PB 的数据, 客户端也可以通过整个卷的单个接合路径检索多个 PB 的数据。

无限卷使用多个节点上的多个聚合所提供的存储。您可以先从小型无限卷开始, 然后通过向该无限卷的聚合添加更多磁盘或为其提供更多聚合, 无中断地对其进行扩展。

### 无限卷提供的功能

通过使用无限卷, 您可以将多个 PB 的数据存储在支持多协议访问、存储效率技术和数据保护功能的单个卷中。

通过使用无限卷, 您可以执行以下任务:

- 在具有单个接合路径和单个命名空间的单个逻辑实体中管理多个 PB 的数据。

- 使用 NFSv3、NFSv4.1、pNFS 和 CIFS (SMB 1.0) 为该数据提供多协议访问途径。
- 通过在单个集群中创建多个包含 FlexVol 卷的 Storage Virtual Machine (SVM) 和多个包含无限卷的 Storage Virtual Machine (SVM) 来提供安全多租户。
- 通过使用精简配置创建大于可用物理存储的无限卷。
- 通过使用重复数据删除和压缩技术最大限度提高存储效率。
- 通过将存储分组到与特定目标对应的存储类来进行优化。
- 自动根据规则（基于文件名、文件路径或文件所有者）将传入文件放入到适当的存储类。
- 通过创建卷的 Snapshot 副本来保护数据。
- 在位于不同集群的两个无限卷之间创建数据保护镜像关系，并在必要时恢复数据。
- 使用 CIFS 或 NFS 将已挂载卷上的数据备份到磁带，并在必要时恢复数据。
- 通过以下方法扩展无限卷：向无限卷所使用的聚合添加更多磁盘；或者将更多聚合分配给包含无限卷的 SVM，然后调整无限卷大小。

## FlexVol 卷与无限卷的对比

FlexVol 卷和无限卷都是数据容器。但是，决定将哪种卷类型包含到存储架构之前，应考虑两者存在的巨大差别。

下表总结了 FlexVol 卷与无限卷之间的差别和相似之处：

卷容量或功能	FlexVol 卷	无限卷	说明
包含实体	SVM；单节点	SVM；可以跨越节点	
关联聚合的数量	一个	多个	
大小上限	因型号而异	最多 20 PB	有关 FlexVol 卷大小上限的详细信息，请参见 <i>Hardware Universe</i> 。
大小上限	20 MB	大约 1.33 TB（对于使用的每个节点）	
Storage Virtual Machine (SVM) 的类型	带有 FlexVol 卷的 SVM	带有无限卷的 SVM	
每个 SVM 的最大卷数	取决于型号和协议	一个	有关详细信息，请参见 <i>Hardware Universe</i> 。
每个节点的最大卷数	因型号而异	因型号而异	有关详细信息，请参见 <i>Hardware Universe</i> 。

卷容量或功能	FlexVol 卷	无限卷	说明
支持的 SAN 协议	是	否	
支持的文件访问协议	NFS、CIFS	NFS、CIFS	
重复数据删除	是	是	
数据压缩	是	是	
FlexClone 卷	是	否	
配额	是	否	
qtree	是	否	
精简配置	是	是	
Snapshot 副本	是	是	
数据保护镜像	是	是	对于无限卷，仅支持集群之间的镜像。
负载共享镜像	是	否	
防病毒	是	否	
磁带备份	是	是	对于无限卷，必须使用 NFS 或 CIFS，而不能使用 NDMP。
卷安全模式	UNIX、NTFS、混合	统一	有关详细信息，请参见《适用于 CIFS 的集群模式 Data ONTAP 文件访问管理指南》或《适用于 NFS 的集群模式 Data ONTAP 文件访问管理指南》。

有关无限卷的详细信息，请参见《集群模式 Data ONTAP 无限卷管理指南》。

#### 相关参考

[存储限制](#)（第 152 页）

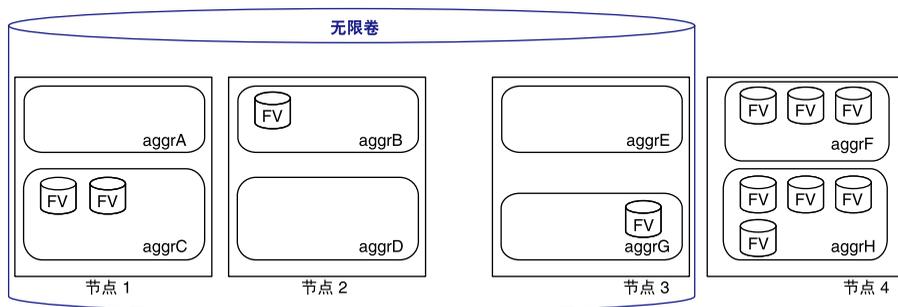
## FlexVol 卷和无限卷如何共享聚合

可以在集群的各个卷之间共享聚合。每个聚合可以包含多个 FlexVol 卷以及无限卷的多个成分卷。

创建无限卷时，系统会将该无限卷的成分卷放到分配给无限卷所在 Storage Virtual Machine (SVM) 的聚合上。如果带有无限卷的 SVM 包含带有 FlexVol 卷的聚合，无限卷的一个或多个成分卷可能会放到已含有 FlexVol 卷的聚合上，前提是这些聚合满足无限卷的托管要求。

同样，创建 FlexVol 卷时，您可以将该 FlexVol 卷与当前已由无限卷使用的某个聚合相关联。

下图说明了在一个包含 FlexVol 卷和无限卷的四节点集群中共享的聚合。无限卷使用聚合 aggrA、aggrB、aggrC、aggrD、aggrE 和 aggrG，但聚合 aggrB、aggrC 和 aggrG 已将存储提供给 FlexVol 卷。（为清晰起见，组成无限卷的各个成分卷未显示。）



## 安全模式对数据访问具有哪些影响

存储系统上的每个卷和 qtree 都具有安全模式。安全模式决定了向用户授权时用于卷上的数据的权限。必须了解各种安全模式的定义、何时应设置安全模式及其设置方法、安全模式对权限具有的影响以及不同卷类型的安全模式之间的差别等。

有关安全模式的详细信息，请参见《适用于 CIFS 的集群模式 Data ONTAP 文件访问管理指南》或《适用于 NFS 的集群模式 Data ONTAP 文件访问管理指南》。

## 使用传统机会锁和租用机会锁提高客户端性能

在某些文件共享场景中，SMB 客户端可以利用传统机会锁和租用机会锁对预读、后写及锁定信息执行客户端缓存。随后，客户端可对文件进行读取或写入操作，期间无需定期提醒服务器它需要获取对该文件的访问权限。这可以减少网络流量，提高性能。

租用机会锁是 SMB 2.1 协议和更高版本中提供的机会锁的增强型。租用机会锁允许客户端获取和保留多个 SMB 打开文件（源自客户端本身）的缓存状态。

包含无限卷的 Storage Virtual Machine (SVM) 不支持租用机会锁。

有关详细信息，请参见《适用于 CIFS 的集群模式 Data ONTAP 文件访问管理指南》。

## 什么是系统卷

系统卷是包含特殊元数据（例如用于文件服务审核日志的元数据）的 FlexVol 卷。这些卷在集群中可见，以便您能完整计算集群中的存储使用量。

系统卷由集群管理服务器（也称为管理 SVM）拥有，它们是在启用文件服务审核时自动创建的。

可以通过使用 `volume show` 命令来查看系统卷，但不允许执行大多数其他的卷操作。例如，无法通过使用 `volume modify` 命令来修改系统卷。

以下示例显示了管理 SVM 上的四个系统卷，这些系统卷是在为集群中的数据 SVM 启用文件服务审核时自动创建的：

```
cluster1::> volume show -vserver cluster1
Vserver   Volume                               Aggregate      State          Type          Size   Available Used%
-----
cluster1  MDV_aud_id0131843d4811e296fc123478563412
          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.
```

有关文件服务审核如何使用系统卷的详细信息，请参见《适用于 CIFS 的集群模式 Data ONTAP 文件访问管理指南》。

## 使用 FlexVol 卷

---

FlexVol 卷的大多数管理任务可由 SVM 管理员执行。少数任务（例如，将卷升级为 Storage Virtual Machine (SVM) 的根卷以及移动或复制卷）只能由集群管理员执行。

### FlexVol 卷如何与 SVM 配合使用

了解 FlexVol 卷处理 Storage Virtual Machine (SVM) 的方式有助于规划存储架构。

#### SVM 如何影响哪些聚合可与 FlexVol 卷进行关联

FlexVol 卷始终与一个 Storage Virtual Machine (SVM) 以及一个为该卷提供存储的聚合相关联。SVM 可以限制将哪些聚合与 FlexVol 卷进行关联，具体取决于 SVM 的配置。

创建 FlexVol 卷时，您需要指定将在哪个 SVM 上创建该卷，以及该卷将从哪个聚合获取其存储。新创建的 FlexVol 卷的所有存储都来自关联的聚合。

如果该卷的 SVM 分配有聚合，则只能使用其中一个已分配的聚合为该 SVM 上的卷提供存储。此做法有助于确保您的 SVM 以适当的方式共享物理存储资源。在多租户环境中，此隔离非常重要，因为对于某些空间管理配置而言，当某个聚合的空间有限时，共享该聚合的卷会影响彼此对可用空间的访问。聚合分配要求适用于集群管理员和 SVM 管理员。

卷移动操作和卷复制操作不受 SVM 聚合分配的影响，因此，当您尝试将 SVM 保存在单独的聚合上时，必须确保在执行这些操作时不会违反 SVM 聚合分配。

如果该卷的 SVM 未分配有聚合，则集群管理员可以使用集群中的任何聚合为新卷提供存储。但是，SVM 管理员不能为未分配聚合的 SVM 创建卷。因此，如果您希望 SVM 管理员能够为特定 SVM 创建卷，您必须为该 SVM 分配聚合。

更改分配给 SVM 的聚合不会影响任何现有卷。因此，不能使用分配给 SVM 的聚合列表来确定与该 SVM 的卷相关联的聚合。

相关信息

[《集群模式 Data ONTAP 8.3 物理存储管理指南》](#)

[《适用于集群管理员的集群模式 Data ONTAP 8.3 系统管理指南》](#)

#### SVM 如何限制自身可包含的 FlexVol 卷数

可以限制 Storage Virtual Machine (SVM) 可包含的 FlexVol 卷数，以控制资源使用量或确保不超过对每 SVM 卷数实施的特定于配置的限制。

每 SVM 最大卷数限制由 SVM 的 `-max-volumes` 参数控制。默认情况下，不对 SVM 可包含的卷数实施任何限制。

只有在 SVM 同时具有聚合列表时，才会应用 SVM 最大卷数限制。该限制同样适用于 SVM 管理员和集群管理员。

## SVM 对 FlexVol 卷的语言有哪些影响

虽然 Storage Virtual Machine (SVM) 的语言决定了 FlexVol 卷的默认语言值，但可以在创建卷时覆盖该值。更改 SVM 的语言时，不会影响其现有 FlexVol 卷。不能更改 FlexVol 卷的语言。

FlexClone 卷使用其父卷的语言作为默认语言。

## 卷联合使用规则

卷联合是将各个卷联合在一起形成一个逻辑命名空间以允许 NAS 客户端访问数据的方法。了解卷联合的形成原理有助于理解和应用这些使用规则。

当 NAS 客户端通过遍历联合访问数据时，联合看起来是普通目录。当卷挂载到根目录下的某个挂载点并用于创建文件系统树时，则会形成联合。文件系统树的顶部始终是根卷，用斜杠 (/) 表示。联合从一个卷的目录指向另一个卷的根目录。

- 尽管在创建卷时指定联合点是可选的，但是在卷挂载到命名空间中的联合点之前，卷中的数据无法导出 (NFS)，并且也无法创建共享 (CIFS)。
- 在卷创建期间未挂载的卷可以在创建之后挂载。
- 新卷可以通过挂载到联合点来随时添加到命名空间。
- 挂载的卷可以取消挂载；然而，取消挂载某个卷会中断 NAS 客户端对于该卷中所有数据以及取消挂载卷下子联合点挂载的所有卷的访问。
- 联合点可以在父卷联合下直接创建，也可以在卷中的目录内创建。  
例如，名为“vol3”的卷的卷联合路径可以为 /vol1/vol2/vol3、/vol1/dir2/vol3，甚至 /dir1/dir2/vol3。

### 相关信息

《适用于 NFS 的集群模式 Data ONTAP 8.3 文件访问管理指南》

《适用于 CIFS 的集群模式 Data ONTAP 8.3 文件访问管理指南》

## 如何使用空间管理功能

可以使用 Data ONTAP 的空间管理功能将数据可用性最大化，以避免过度使用存储。可以通过文件、LUN 和卷的配置选项来实现该目的，而且警报会告诉您何时需要采取措施以防止用尽可用空间。

由于 Data ONTAP 使用功能强大的块共享技术（例如 Snapshot 副本），因此您需要提供比已写入到卷或 LUN 中的用户数据大小更多的可用空间。所需提供的可用空间量取决于应用程序环境和存储管理实践。

可以使用以下配置选项和功能实现环境在数据可用性与存储利用率之间的合理平衡：

- 文件预留和 LUN 预留  
通过使用文件或 LUN 的这些属性，可为该特定对象预留可用空间。
- 卷保证和预留百分比  
通过使用这些卷属性，可以配置 Data ONTAP 为卷预留空间的方式。
- 自动为已满的卷提供可用空间  
通过使用此卷属性，可使 Data ONTAP 自动为将近已满的卷提供更多空间。
- 卷自动调整  
如果聚合中的可用空间比特定卷（与该聚合关联）中的可用空间更重要，则可以将该卷配置为不需要时将空间归还给该聚合。
- 填满度警报和分配过度警报  
这些警报可告诉您卷或聚合将在何时变满，以便您能采取措施以防止用尽可用空间。
- 空间使用量管理  
这些方法可用于确定和影响卷和聚合中的已用空间的使用方式。

### 相关概念

[FlexVol 卷的卷保证工作原理](#)（第 19 页）

[文件和 LUN 预留的工作原理](#)（第 19 页）

[设置预留百分比的注意事项](#)（第 23 页）

### 相关任务

[将卷配置为在已满时自动提供更多空间](#)（第 24 页）

## 文件和 LUN 预留的工作原理

启用文件或 LUN 预留在创建时即告诉 Data ONTAP 预留该文件或 LUN 所需的空間，而非等到需要空间写入时才告诉。禁用预留允许您通过创建空间似乎比可用所属卷空间多的 LUN 来过量使用该卷。

预留是文件或 LUN 的属性；它们在存储系统重新启动、接管和归还的过程中永久生效。默认情况下为新的 LUN 启用预留，但是您可以创建禁用或启用了预留的文件或 LUN。当您创建了 LUN 之后，您可以使用 `lun modify` 命令更改预留属性。您可以使用 `file reservation` 命令更改文件的预留属性。

当卷包含一个或多个启用了预留的文件或 LUN 时，需要可用空间的操作（例如快照副本的创建）不得使用预留的空间。如果这些操作没有足够的未预留可用空间，则会失败。然而，写入启用了预留的文件或 LUN 仍然可以成功。

您可以为卷保证为任意值的卷中的文件和 LUN 启用预留。但是，如果卷保证为 `none`，则预留不具有任何效果。

### 示例

如果您在 500-GB 的卷中创建一个预留了 100-GB 空间的 LUN，则会立即分配该 100 GB 的空间，卷中剩余的空间为 400 GB。相反，如果对 LUN 禁用了空间预留，则卷中的全部 500 GB 都可供使用，直到对 LUN 执行写入操作。

## FlexVol 卷的卷保证工作原理

卷保证（有时称为空间保证）可确定如何从卷所属聚合分配卷空间 — 是否为卷预分配空间。

保证是卷的一个属性。

可以在创建新卷时设置保证；也可以更改现有卷的保证，前提是存在用于支持新保证的足够可用空间。

卷保证类型可以是 `volume`（默认类型）或 `none`。

- `volume` 保证类型可在创建卷时为整个卷分配聚合中的空间，而不管该空间是否已用于数据。  
这种空间管理方法称为厚配置。已分配的空间无法提供或分配给该聚合中的任何其他卷。使用厚配置时，为卷指定的所有空间都是在创建卷时从聚合中分配的。在卷中的数据量（包括 Snapshot 副本）达到卷大小之前，卷不会用尽空间。但是，如果卷并未全部占满，则存储利用率会降低。
- `none` 保证类型只有在卷需要空间时才会从聚合中分配空间。  
这种空间管理方法称为精简配置。使用此保证类型时，卷占用的空间量会随着数据的添加而增长，而不是由初始卷大小决定，如果由初始卷大小决定，则当卷数据未增长到初始卷大小时，可能会出现未使用的空间。如果使用保证类型 `none`，则卷的大小上限不受其聚合中可用空间量的限制。与聚合关联的所有卷的总大小可以超过聚合的可用空间量，即使实际可以使用的空间量受聚合大小的限制。

在向卷中的 LUN 或文件（包括空间预留 LUN 和文件）写入时，如果卷所在的聚合没有足够的可用空间来支持该写入，则该写入将失败。

在为现有卷的 `volume` 保证分配聚合中的空间之后，该空间在聚合中将不再视为可用，即使该卷尚未使用该空间。只有当聚合中有足够的可用空间时，才能执行占用聚合中可用空间的操作（如创建聚合 Snapshot 副本或在卷所在的聚合中创建新卷等）；这些操作不能使用已分配给其他卷的空间。

如果聚合中的可用空间用尽，则只有向该聚合中具有预分配空间的卷或文件执行的写入操作才能保证成功。

只有联机卷才支持保证。如果使某个卷脱机，则已为该卷分配但尚未使用的空间将可供聚合中的其他卷使用。在尝试使该卷重新联机时，如果聚合中没有足够的可用空间来实现卷保证，则该卷将仍保持脱机状态。您必须强制使该卷联机，此时，该卷的保证将被禁用。

相关概念

[什么是卷占用空间](#)（第 39 页）

相关信息

[NetApp 技术报告 3965: 《NetApp 精简配置部署和实施指南（7 模式 Data ONTAP 8.1）》](#)

### 启用卷保证

如果禁用了 FlexVol 卷保证，则该卷就会假定保证类型为 `none`。如果卷禁用了卷保证，则应尽快为这些卷创建更多可用空间以解决此类情况。

开始之前

FlexVol 卷必须处于联机状态。

关于本任务

已启用的保证会在聚合中预先分配空间。如果卷禁用了保证，则可能会禁止执行需要更多空间的操作（例如写入，甚至还包括删除）。如果卷保证被禁用，则应重新启用该保证，才能手动增加卷大小。如果卷禁用了保证而启用了自动增长功能，则该卷的大小仍然可能自动增加。

您可以先查看卷的保证状态，或者尝试启用该保证。如果启用保证失败，则 Data ONTAP 会指明原因（通常是空间不足）并指定聚合中需要具有的可用空间量。保证类型 `none` 始终不会被禁用，因为不会为此保证类型分配任何空间。

步骤

1. 可选：使用带有 `-fields`、`-space-guarantee` 和 `-space-guarantee-enabled` 参数的 `volume show` 命令查看卷的保证状态和保证类型。

## 示例

以下示例中的命令显示了名为 vs0 的 Storage Virtual Machine (SVM) 上卷 vol2 的保证状态。保证已禁用 (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
```

以上输出显示了指定卷的保证类型以及保证已启用还是已禁用。如果 `space-guarantee-enabled` 列中的值为 `true`，则表示保证已启用；如果该值为 `false`，则表示保证已禁用。

## 2. 启用或重新启用保证。

要启用保证的卷	命令
单个卷	<pre>volume modify vol_name -space-guarantee guarantee_type</pre> <p>此命令将为单个卷启用指定类型的保证，前提是有足够空间可用于执行此操作。如果指定的保证与当前为该卷配置的保证不同，则 Data ONTAP 会将保证更改为指定的保证并启用该保证。</p>
所有具有相同保证类型的卷	<pre>volume modify { -space-guarantee guarantee_type - space-guarantee-enabled false } -space-guarantee guarantee_type</pre> <p>此命令将为所有具有指定保证类型的卷启用保证。</p> <p>请确保查询字符串中用花括号指定的保证类型与指定的目标保证类型相同。否则，此命令会更改卷的保证类型。</p> <p>以下示例中的命令为名为 v1 和 v3 的卷重新启用了保证，而且这两个卷的保证类型均为 <code>volume</code>：</p> <pre>cluster1::&gt; 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.</pre>

此时将启用保证，或者您会收到一条错误消息，指出要启用保证，需要在聚合中创建多少空间。

如果已使用此命令重新启用了多个相同类型的保证，则会为所有具有指定保证类型的卷启用保证，但前提有足够的可用空间可以启用这些保证。

3. 如果聚合中没有足够空间可用于启用保证，则必须创建更多空间。

示例

以下示例显示了尝试为名为 `testvol` 的卷启用保证时显示的错误消息：

```
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. 尝试重新启用保证，并查看命令输出以检查是否已启用保证。  
如果仍未启用保证，则必须尝试通过其他方法创建更多空间。
5. 可选：如果已使用其中一个命令重新启用了多个相同类型的保证，请使用 `volume show` 命令与 `-fields space-guarantee,space-guarantee-enabled` 参数来验证是否已启用所有保证。

示例

```
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.
```

已启用的保证会在 `space-guarantee-enabled` 列中显示值 `true`。任何未启用的保证则在该列中显示值 `false`。

相关概念

- [FlexVol 卷的卷保证工作原理](#)（第 19 页）
- [在 FlexVol 卷中创建空间的方法](#)（第 42 页）
- [在聚合中创建空间的方法](#)（第 44 页）

将精简配置与 **FlexVol** 卷结合使用的注意事项

通过使用精简配置，您可以对卷进行配置，以便这些卷看起来像是提供了比可用存储更多的存储一样（前提是实际正在使用的存储不超过可用存储）。

要将精简配置与 FlexVol 卷结合使用，请创建保证类型为 `none` 的卷。使用 `none` 保证类型时，卷大小不受聚合大小限制。事实上，如果需要，每个卷都可以大于其所属的聚合。仅当数据写入到 LUN 或文件时，才会使用聚合提供的存储。

如果与某个聚合关联的卷所显示的可用存储多于可供该聚合使用的物理资源，则该聚合即已过量使用。过量使用某个聚合时，如果没有足够的可用空间供写入使用，则对该聚合所包含的卷中的 LUN 或文件的写入可能会失败。

如果已过量使用聚合，必须监控可用空间并根据需要为聚合添加存储，以避免由于空间不足引起写入错误。

聚合可向与多个 Storage Virtual Machine (SVM) 关联的 FlexVol 卷提供存储。请注意，在多租户环境中将共享聚合用于精简配置卷时，一个租户的聚合空间可用性可能会由于另一个租户的卷的增长而受到不利影响。

#### 相关信息

- [NetApp 技术报告 3965: 《NetApp 精简配置部署和实施指南（7-模式 Data ONTAP 8.1）》](#)
- [NetApp 技术报告 3483: 《NetApp SAN 或 IP SAN 企业环境中的精简配置》](#)

## 设置预留百分比的注意事项

预留百分比（又称 LUN 覆盖预留）允许您为 FlexVol 卷中预留了空间的 LUN 和文件关闭覆盖预留。这可以帮助您最大限度地提高存储利用率，但是如果您的环境因空间不足导致写入操作失败而受到严重影响，则您必须了解此配置的各项要求。

预留百分比设置以百分比的形式表示；有效值只有 0% 和 100%。预留百分比设置是卷的属性。

将预留百分比设置为 0 可以提高存储利用率。但是，访问卷中数据的应用程序在卷空间不足的情况下可能会遇到数据中断错误，即使在卷保证设置为 `volume` 的情况下也一样。然而，通过正确配置和使用卷，您可以将写入失败的可能性降至最低。Data ONTAP 为预留百分比设置为 0 的卷提供“尽力确保最佳结果”的写入保证，前提是满足以下所有要求：

- 未使用重复数据删除
- 未使用压缩
- 未使用 FlexClone 子文件
- 所有 FlexClone 文件和 FlexClone LUN 都已启用自动删除  
这不是默认设置。您必须在创建时或者在创建后通过修改 FlexClone 文件或 FlexClone LUN 来显式启用自动删除。
- 未使用 ODX 和 FlexClone 副本负载转移
- 卷保证设置为 `volume`
- 文件或 LUN 预留为 `enabled`
- 卷快照预留设置为 0
- 卷快照副本自动删除为 `enabled`，并且承诺等级为 `destroy`、销毁列表包括 `lun_clone,vol_clone,cifs_share,file_clone,sfsr`，触发器为 `volume`

此设置还确保根据需要删除 FlexClone 文件和 FlexClone LUN。

请注意，如果您的更改率较高，则偶尔快照副本自动删除有可能会落后，从而导致卷空间不足，即使使用了以上所有必要的配置设置也一样。

此外，您还可以选择使用卷自动增长功能降低需要自动删除卷快照副本的可能性。如果您启用自动增长功能，则必须监控关联聚合中的可用空间。如果聚合变得足够满，使得卷无法增长，则随着卷中的可用空间耗尽，可能会删除更多的快照副本。

如果您无法满足以上所有配置要求，并且需要确保卷不会出现空间不足的情况，则必须将卷的预留百分比设置为 100。这需要预先准备更多的可用空间，但是可以保证即使使用了上面列出的技术，数据修改操作也能成功。

预留百分比设置的默认值和允许值视卷保证而定：

卷保证	默认的预留百分比	允许值
卷	100	0、100
无	0	0、100

#### 相关概念

[FlexVol 卷的卷保证工作原理](#)（第 19 页）

[文件和 LUN 预留的工作原理](#)（第 19 页）

[如何解决聚合填满度警报和分配过度警报](#)（第 30 页）

[如何解决 FlexVol 卷填满度警报和分配过度警报](#)（第 28 页）

#### 相关任务

[自动删除 Snapshot 副本](#)（第 26 页）

[将卷配置为自动增长和缩减其大小](#)（第 27 页）

#### 相关信息

[NetApp 技术报告 3965: 《NetApp 精简配置部署和实施指南（7-模式 Data ONTAP 8.1）》](#)

[NetApp 技术报告 3483: 《NetApp SAN 或 IP SAN 企业环境中的精简配置》](#)

## 将卷配置为在已满时自动提供更多空间

当 FlexVol 卷已满时，Data ONTAP 可以使用多种方法来尝试自动为该卷提供更多可用空间。您可以根据应用程序和存储架构来选择 Data ONTAP 能够使用的方法以及各种方法的优先顺序。

#### 关于本任务

Data ONTAP 可使用以下一种或两种方法自动为已满的卷提供更多可用空间：

- 增加卷的大小（称为自动增长）。

如果包含该卷的聚合具有足够的空间来支持更大的卷，则此方法很有用。您可以对 Data ONTAP 进行配置，使其为卷设置一个最大大小。增加大小的操作是根据要写入到卷中的数据量与当前已用空间量及所设置的任何阈值之间的关系自动触发的。

自动增长不会出于支持 Snapshot 副本创建的目的而触发。如果尝试创建 Snapshot 副本并且没有足够空间，则即使启用了自动增长，Snapshot 副本创建也将失败。

- 删除 Snapshot 副本、FlexClone 文件或 FlexClone LUN。

例如，可以将 Data ONTAP 配置为自动删除与克隆卷或 LUN 中的 Snapshot 副本无关联的 Snapshot 副本，或者也可以定义您希望 Data ONTAP 首先删除的 Snapshot 副本 — 最旧或最新的 Snapshot 副本。此外，还可以决定 Data ONTAP 应在何时开始删除 Snapshot 副本，例如，当卷快满或卷的 Snapshot 预留快满时。

如果您同时启用了这两种方法，则可以指定 Data ONTAP 在卷快满时首先尝试哪种方法。如果第一种方法无法为卷提供足够的额外空间，则 Data ONTAP 会接着尝试另一种方法。

默认情况下，Data ONTAP 会首先尝试增加卷的大小。在大多数情况下，最好使用默认配置，因为删除 Snapshot 副本之后，该副本将无法恢复。但是，如果需要尽可能避免增加卷的大小，可将 Data ONTAP 设置为先删除 Snapshot 副本再增加卷的大小。

#### 步骤

1. 如果希望 Data ONTAP 在卷已满时尝试增加该卷的大小，请将 `volume autosize` 命令与 `grow` 模式结合使用，以便为该卷启用自动增长功能。  
请记住，当卷增长时，它会在关联的聚合中占据更多可用空间。如果依靠卷在需要时自动增长，则必须监控关联的聚合中的可用空间，并在需要时添加更多可用空间。
2. 如果希望 Data ONTAP 在卷已满时删除 Snapshot 副本、FlexClone 文件或 FlexClone LUN，请为这些对象类型启用自动删除。
3. 如果同时启用了卷自动增长功能和一项或多项自动删除功能，请将 `volume modify` 命令与 `-space-mgmt-try-first` 选项结合使用，以便选择 Data ONTAP 用于向卷提供可用空间的第一种方法。  
要指定先增加卷的大小（默认行为），请使用 `volume_grow`。要指定先删除 Snapshot 副本，请使用 `snap_delete`。

#### 相关概念

[FlexVol 卷如何从 FlexClone 文件和 FlexClone LUN 回收可用空间](#)（第 71 页）  
[在 FlexVol 卷中创建空间的方法](#)（第 42 页）

#### 相关任务

[自动删除 Snapshot 副本](#)（第 26 页）  
[将 FlexVol 卷配置为自动删除 FlexClone 文件和 FlexClone LUN](#)（第 72 页）

## 自动删除 Snapshot 副本

您可以定义并启用用来自动删除 Snapshot 副本和 FlexClone LUN 的策略。自动删除 Snapshot 副本和 FlexClone LUN 有助于管理空间利用率。

### 关于本任务

您可以从读写卷自动删除 Snapshot 副本，并从读写父卷自动删除 FlexClone LUN。您不能设置从无限卷或只读卷（例如 SnapMirror 目标卷）自动删除 Snapshot 副本。

### 步骤

1. 您可以使用 `volume snapshot autodelete modify` 命令定义并启用用于自动删除 Snapshot 副本的策略。

有关可在 `volume snapshot autodelete modify` 命令中使用哪些参数来根据需要定义策略的信息，请参见此命令的手册页。

### 示例

以下命令将为 `vs0.example.com` Storage Virtual Machine (SVM) 中的卷 `vol3` 启用 Snapshot 副本自动删除功能，并将触发条件设置为 `snap_reserve`：

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

### 示例

以下命令将为 `vs0.example.com` Storage Virtual Machine (SVM) 中的卷 `vol3` 启用自动删除功能，以便自动删除 Snapshot 副本和标记为自动删除的 FlexClone LUN：

```
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
```

### 相关任务

[将 FlexVol 卷配置为自动删除 FlexClone 文件和 FlexClone LUN](#)（第 72 页）

## 将卷配置为自动增长和缩减其大小

您可以将 FlexVol 卷配置为根据其当前所需的空间自动增长和缩减。自动增长有助于防止卷出现空间不足的情况（如果聚合可以提供更多的空间）。自动缩减可以防止卷浪费空间，它将空间释放到聚合中以供其他卷使用。

### 开始之前

FlexVol 卷必须联机。

### 关于本任务

自动缩减只能与自动增长结合使用，以满足不断变化的空间需求，不能单独使用。当启用自动缩减时，Data ONTAP 自动管理卷的缩减行为以防止自动增长和自动缩减操作出现死循环。

当卷增长时，它能够包含的最大文件数可以自动增加。当卷缩减时，它能够包含的最大文件数保持不变，并且卷无法自动缩减到小于当前最大文件数对应的大小。因此，不太可能将卷一直自动缩减到其原始大小。

默认情况下，卷可以增长至的最大大小为启用自动增长时大小的 120%。如果您需要确保卷可以增长到超过该值，则必须相应地设置卷的最大大小。

### 步骤

#### 1. 将卷配置为自动增长和缩减其大小：

```
volume autosize -vserver vs_server_name vol_name -mode grow_shrink
```

#### 示例

下面的命令为名为 test2 的卷启用自动更改大小。该卷配置为当使用空间降到 60% 时开始自动缩减。针对何时开始增长以及其最大大小使用了默认值。

```
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
```

## 同时启用自动缩减和 Snapshot 副本自动删除的要求

如果满足特定配置要求，则自动缩减功能可与 Snapshot 副本自动删除结合使用。

如果要同时启用自动缩减功能和 Snapshot 副本自动删除，必须满足以下配置要求：

- 必须将 Data ONTAP 配置为先尝试增加卷大小再尝试删除 Snapshot 副本（`-space-mgmt-try-first` 选项必须设置为 `volume_grow`）。

- Snapshot 副本自动删除的触发条件必须是卷容量填满度（`trigger` 参数必须设置为 `volume`）。

### 自动缩减功能如何与 Snapshot 副本删除功能交互

由于自动缩减功能会缩减 FlexVol 卷的大小，因此，它可能还会影响自动删除卷 Snapshot 副本的时间。

自动缩减功能会通过以下方式与卷 Snapshot 副本自动删除功能进行交互：

- 如果既启用了 `grow_shrink` 自动调整大小模式，又启用了 Snapshot 副本自动删除，则在卷大小缩减时，可能会触发 Snapshot 副本自动删除操作。  
这是因为 Snapshot 预留是基于卷大小百分比的（默认值为 5%），而该卷大小现已变小，因此可能会导致 Snapshot 副本从预留中溢出并被自动删除。
- 如果启用了 `grow_shrink` 自动调整大小模式，而您手动删除 Snapshot 副本，则可能会触发卷自动缩减操作。

### 如何解决 FlexVol 卷填满度警报和分配过度警报

当 FlexVol 卷即将用尽空间时，Data ONTAP 会发出 EMS 消息，以便您能采取纠正措施，方法是为己满的卷提供更多空间。了解警报类型及其相应的解决方法有助于确保数据可用性。

当卷标记为已满时，表明该卷中可供活动文件系统（用户数据）使用的空间百分比已降至阈值（可配置）以下。当卷变为分配过度时，表明供 Data ONTAP 用于元数据或支持基本数据访问的空间已用尽。有时，正常预留用于其他目的的空间可用于使卷保持工作，但预留或数据可用性会处于风险之中。

分配过度具有逻辑和物理两种性质。逻辑分配过度是指为履行未来的空间承诺（例如文件预留）而预留的空间已被用于其他目的。物理分配过度是指卷即将用尽物理块。处于此状态的卷会遭受如下风险：拒绝写入、进入脱机状态或可能导致控制器中断。

受元数据所使用或预留的空间的影响，卷的填满度可能会超过 100%。但是，填满度超过 100% 的卷不一定存在分配过度问题。

下表介绍了卷填满度警报和分配过度警报、可用于解决相应问题的措施以及不采取措施将带来的风险。

警报类型	EMS 级别	是否可配置	定义	解决方法	不采取措施是否会带来风险
接近已满	调试	Y	文件系统已超过为此警报设置的阈值（默认值是 95%）。此百分比等于已用总数减去 Snapshot 预留的大小。	<ul style="list-style-type: none"> <li>• 增加卷大小</li> <li>• 减少用户数据</li> </ul>	目前不会对写入操作或数据可用性带来风险。

警报类型	EMS 级别	是否可配置	定义	解决方法	不采取措施是否会带来风险
已满	调试	Y	文件系统已超过为此警报设置的阈值（默认值是 98%）。此百分比等于已用总数减去 Snapshot 预留的大小。	<ul style="list-style-type: none"> <li>• 增加卷大小</li> <li>• 减少用户数据</li> </ul>	目前不会对写入操作或数据可用性带来风险，但卷的写入操作遭受风险的可能性逐渐增大。
逻辑分配过度	SVC 错误	N	除了文件系统已满之外，卷中用于元数据的空间也已用尽。	<ul style="list-style-type: none"> <li>• 增加卷大小</li> <li>• 删除 Snapshot 副本</li> <li>• 减少用户数据</li> <li>• 删除文件预留或 LUN 预留</li> </ul>	对未预留文件的写入操作可能会失败。
物理分配过度	节点错误	N	可供卷写入的物理块即将用尽。	<ul style="list-style-type: none"> <li>• 增加卷大小</li> <li>• 删除 Snapshot 副本</li> <li>• 减少用户数据</li> </ul>	写入操作和数据可用性都处于风险之中；卷可能会进入脱机状态。

每当超过卷的阈值时，无论填满度百分比正在上升还是下降，都会生成 EMS 消息。当卷的填满度级别降至阈值以下时，将生成 volume ok EMS 消息。

#### 相关概念

[如何解决聚合填满度警报和分配过度警报](#)（第 30 页）

#### 相关任务

[将卷配置为在已满时自动提供更多空间](#)（第 24 页）

相关信息

《适用于集群管理员的集群模式 Data ONTAP 8.3 系统管理指南》

## 如何解决聚合填满度警报和分配过度警报

当聚合即将用尽空间时，Data ONTAP 会发出 EMS 消息，以便您能采取纠正措施，方法是为已满的聚合提供更多空间。了解警报类型及其相应的解决方法有助于确保数据可用性。

当聚合标记为已满时，表明该聚合中可供卷使用的空间百分比已降至预定义的阈值以下。当聚合变为分配过度时，表明供 Data ONTAP 用于元数据或支持基本数据访问的空间已用尽。有时，正常预留用于其他目的的空间可用于使聚合保持工作，但与该聚合关联的卷的卷保证或数据可用性会处于风险之中。

分配过度具有逻辑和物理两种性质。逻辑分配过度是指为履行未来的空间承诺（例如卷保证）而预留的空间已被用于其他目的。物理分配过度是指聚合即将用尽物理块。处于此状态的聚合会遭受如下风险：拒绝写入、进入脱机状态或可能导致控制器中断。

下表介绍了聚合填满度警报和分配过度警报、可用于解决相应问题的措施以及不采取措施将带来的风险。

警报类型	EMS 级别	是否可配置	定义	解决方法	不采取措施是否会带来风险
接近已满	调试	N	为卷分配的空间量（包括其保证）已超过为此警报设置的阈值 (95%)。 此百分比等于已用总数减去 Snapshot 预留的大小。	<ul style="list-style-type: none"> <li>向聚合添加空间</li> <li>缩减或删除卷</li> <li>将卷移动到有更多空间的聚合中</li> <li>删除卷保证（将其设置为 <b>none</b>）</li> </ul>	目前不会对写入操作或数据可用性带来风险。

警报类型	EMS 级别	是否可配置	定义	解决方法	不采取措施是否会带来风险
已满	调试	N	文件系统已超过为此警报设置的阈值 (98%)。 此百分比等于已用总数减去 Snapshot 预留的大小。	<ul style="list-style-type: none"> <li>向聚合添加空间</li> <li>缩减或删除卷</li> <li>将卷移动到有更多空间的聚合中</li> <li>删除卷保证 (将其设置为 <b>none</b>)</li> </ul>	聚合中的卷的卷保证以及对这些卷的写入操作可能处于风险之中。
逻辑分配过度	SVC 错误	N	除了为卷预留的空间已满之外，聚合中用于元数据的空间也已用尽。	<ul style="list-style-type: none"> <li>向聚合添加空间</li> <li>缩减或删除卷</li> <li>将卷移动到有更多空间的聚合中</li> <li>删除卷保证 (将其设置为 <b>none</b>)</li> </ul>	聚合中的卷的卷保证以及对这些卷的写入操作都处于风险之中。

警报类型	EMS 级别	是否可配置	定义	解决方法	不采取措施是否会带来风险
物理分配过度	节点错误	N	可供聚合写入的物理块即将用尽。	<ul style="list-style-type: none"> <li>向聚合添加空间</li> <li>缩减或删除卷</li> <li>将卷移动到有更多空间的聚合中</li> </ul>	对聚合中的卷的写入操作以及数据可用性都处于风险之中；聚合可能会进入脱机状态。在极端情况下，节点可能会遇到中断。

每当超过聚合的阈值时，无论填满度百分比正在上升还是下降，都会生成 EMS 消息。当聚合的填满度级别降至阈值以下时，将生成 `aggregate ok` EMS 消息。

相关概念

[如何解决 \*FlexVol\* 卷填满度警报和分配过度警报](#)（第 28 页）

相关信息

《适用于集群管理员的集群模式 *Data ONTAP 8.3* 系统管理指南》

## 如何确定卷或聚合中的空间使用量

在 *Data ONTAP* 中启用一项功能可能会在您不知情的情况下占用空间，或者可能会占用比预期更多的空间。*Data ONTAP* 可以通过从以下三个视角来查看空间来帮助确定空间占用情况：卷、卷在聚合中的占用空间和聚合。

卷可能会由于空间被占用或者卷、聚合或这两者空间不足而用尽空间。通过从不同视角查看功能占用空间的详细情况，您可以评估哪些功能需要调整或关闭，或者您也可以采取其他措施（例如增加聚合或卷的大小）。

可以从以下任意视角查看空间使用量的详细信息：

- 卷的空间使用量  
此视角可提供有关卷中空间使用量的详细信息，包括 Snapshot 副本的空间使用量。  
可以使用 `volume show-space` 命令查看卷的空间使用量。
- 卷在聚合中的占用空间  
此视角可提供有关每个卷在所属聚合中使用的空间量详细信息，包括该卷的元数据。  
可以使用 `volume show-footprint` 命令查看卷在聚合中的占用空间。

- 聚合的空间使用量

此视角可提供聚合中所有卷的占用空间、为聚合 Snapshot 副本预留的空间以及其他聚合元数据的总空间使用量。

可以使用 `storage aggregate show-space` 命令查看聚合的空间使用量。

某些功能（例如磁带备份和重复数据删除）会使用卷中的元数据空间，同时还会直接使用聚合中的元数据空间。从卷和卷占用空间这两种视角查看时，这些功能的空间使用量会有所差别。

#### 相关概念

[更改文件容量或目录容量时的警告和注意事项](#)（第 45 页）

[更改 FlexVol 卷上允许的最大文件数的注意事项](#)（第 45 页）

#### 如何确定和控制卷中的空间使用量

您可以查看有关卷的空间使用量的详细信息，以了解哪些 Data ONTAP 功能正在占用空间以及可以执行哪些操作来减少已用空间。

卷的活动文件系统（未在 Snapshot 副本中捕获的卷数据）包含用户数据、文件系统元数据和索引节点。您启用的 Data ONTAP 功能可能会增加元数据的数量；在执行 Snapshot 副本时，这些功能有时可能会溢出到活动文件系统的用户数据部分。

`volume show-space` 命令可显示卷中已用空间的使用情况。例如，您可能想要了解为什么在删除了卷中所有数据后 `df` 命令输出仍显示正在使用大量空间。无限卷成分卷会显示在空间使用情况命令输出中，就好像这些成分卷是 FlexVol 卷一样。在这种情况下，`volume show-space` 命令的输出可能会指出原因在于 Snapshot 副本、索引节点或其他元数据没有缩减。

命令输出将只显示非零值。但是，您可以使用 `-instance` 参数来显示所有可能的功能行，而不管其是否已启用以及是否占用了任何空间。值 `-` 表示没有可显示的数据。

下面几个表说明了 `volume show-space` 命令输出中的一些常见的行，以及可以尝试执行哪些操作来减少相应功能的空间使用量。

此命令的输出包含以下主要类别：

- 用户数据
- 卷元数据
- Snapshot 副本信息
- 已用空间

有关如何减少其他功能（例如重复数据删除）占用的空间的信息，请参见相应的 Data ONTAP 指南。

保证类型为 `None` 的卷的可用空间受聚合中的可用空间限制。

## 用户数据

以下输出行与用户数据相关：

行/功能名称	说明	可用于减少空间使用量的一些方法
用户数据	与用户数据（包括写入到卷的数据）相关的所有信息，包括与用户索引节点关联的间接块和目录块以及卷中预留的空间。	<ul style="list-style-type: none"> <li>删除用户数据</li> <li>禁用文件预留或 LUN 预留 请注意，禁用文件预留或 LUN 预留会使 Data ONTAP 无法保证对这些文件或 LUN 的写入。这样可能会返回空间不足的错误。禁用预留应该作为临时措施，一旦为卷提供了更多的可用空间，就应立即重新启用预留。</li> </ul>

## 卷元数据

以下输出行与卷元数据相关：

行/功能名称	说明	可用于减少空间使用量的一些方法
重复数据删除/重复数据删除百分比	重复数据删除元数据文件使用的空间量。	将重复数据删除节省的空间与所需元数据大小进行比较。如果所需元数据大小超过节省的空间，则可以在卷上禁用重复数据删除。
临时重复数据删除/临时重复数据删除百分比	临时重复数据删除元数据文件使用的空间量。	没有直接的控制方法。重复数据删除扫描程序运行完毕后，临时元数据使用量将减少。
文件系统元数据/文件系统元数据百分比	对 Data ONTAP 所需文件系统进行内部跟踪。	没有直接的控制方法。
SnapMirror 元数据/SnapMirror 元数据百分比	SnapMirror 元数据文件使用的空间量。此行仅与逻辑复制相关。在传输过程中，系统会临时使用一些额外空间。	没有直接的控制方法。 可等待传输完成，以释放临时使用的额外空间。
磁带备份元数据/磁带备份元数据百分比	卷中磁带备份元数据文件使用的空间量。	下次成功运行基线（0 级）备份后会清除磁带备份元数据占用的空间量。您可以启动基线备份，也可以在下次计划的时间运行基线备份。

行/功能名称	说明	可用于减少空间使用量的一些方法
配额元数据/配额元数据百分比	配额元数据文件使用的空间量。	禁用配额。
性能元数据/性能元数据百分比	性能优化活动使用的空间量。	没有直接的控制方法。
索引节点/索引节点百分比	此行与卷中曾经创建过的最大文件数成比例。	没有可用于控制当前使用量的任何直接方法。 您可以通过降低最大公共索引节点设置 (maxfiles) 来减少可用于索引节点分配的最大空间量。但是, 已分配给索引节点的空间不会返回到卷中, 因此, 如果已经使用了这些索引节点, 则此操作将不起作用。

### Snapshot 副本信息

以下输出与 Snapshot 副本相关:

行/功能名称	说明	可用于减少空间使用量的一些方法
Snapshot 预留	当前卷大小的百分比。即使 Snapshot 预留中不存在任何 Snapshot 副本, 该预留也会计入已用空间。当卷未滿时, Snapshot 预留不能由活动文件系统使用。 此行与 <code>df</code> 命令中的 <code>.snapshot</code> 所用总空间行相同。	使用带有 <code>-percent-snapshot-space</code> 参数的 <code>volume modify</code> 命令可减少卷中允许 Snapshot 副本使用的空间。
不可用的 Snapshot 预留	如果活动文件系统要使用的空间超过卷为其分配的空间, 则它就会开始使用先前为 Snapshot 预留分配的空间。此行显示了最初为 Snapshot 预留分配的空间量, 该预留空间由于被活动文件系统占用而无法由 Snapshot 副本使用。 此值显示为负数。	通过删除用户数据或减少卷元数据来减小活动文件系统的大小。

行/功能名称	说明	可用于减少空间使用量的一些方法
Snapshot 溢出	<p>由 Snapshot 副本使用但超出 Snapshot 预留大小的空间量，该空间量将溢出到活动文件系统。除非删除 Snapshot 副本，否则不能使用该空间量向活动文件系统写入。</p> <p>如果此行中的值非零，则表示尚未根据当前配置正确调整 Snapshot 预留大小。</p> <p>卷克隆、SnapMirror 和定期计划的 Snapshot 副本均可能导致 Snapshot 副本溢出。</p>	<ul style="list-style-type: none"> <li>• 增加 Snapshot 预留的大小。</li> <li>• 手动删除或者通过启用 Snapshot 自动删除功能来删除卷的 Snapshot 副本。</li> <li>• 更改 SnapMirror 计划。</li> </ul>

### 已用空间

以下行与卷中已用总空间相关：

行/功能名称	说明	可用于减少空间使用量的一些方法
已用总空间	<p>卷中已用总空间量，包括为整个 Snapshot 预留分配的空间量以及为活动文件系统分配的空间量。此行等于 <code>volume show</code> 命令输出中的 <code>used</code> 字段。</p> <p>由于 Snapshot 空间被视为已用空间，因此，此行的值会大于 <code>df</code> 命令输出的值。在 <code>df</code> 命令中，此行等于 <code>used</code> 列中的卷已用空间加上 <code>total</code> 列中 Snapshot 已用空间 (<code>.snapshot</code>) 行的 Snapshot 总空间。</p> <p>如果存在 Snapshot 溢出，<code>volume show-space</code> 命令只会计入已用空间一次。但是，<code>df</code> 命令会将该空间同时显示为活动文件系统和 <code>.snapshot</code> 行的已用空间。</p>	各个输出行所使用的任何方法。
已用总物理空间	当前正在使用（而不是为将来预留）的总空间量，包括由 Snapshot 副本使用的空间量。	各个输出行所使用的任何方法。

#### 启用了 Snapshot 溢出和重复数据删除时的示例输出

以下示例显示了已启用重复数据删除且 Snapshot 副本数超出 Snapshot 预留大小的 FlexVol 卷的输出：

```
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%
```

### 当 Snapshot 预留不可用时的示例输出

以下示例显示了由于活动文件系统已满而将一些 Snapshot 预留用于活动文件系统的 FlexVol 卷的输出。

```
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%
```

### 如何确定和控制聚合中卷的空间使用量

您可以确定哪些 FlexVol 卷和无限卷成分卷在聚合中使用的空间最大，以及具体是哪些功能。`volume show-footprint` 命令可提供在所属聚合内卷占用空间或空间使用量的相关信息。

`volume show-footprint` 命令可显示聚合中每个卷（包括脱机卷）空间使用量的详细信息。此命令与 `df` 命令的输出并不直接对应，但它正好是 `volume show-space` 与 `aggregate show-space` 命令输出结果之差。所有百分比均以占聚合大小的百分比进行计算。

命令输出将只显示非零值。但是，您可以使用 `-instance` 参数来显示所有可能的功能行，而不管其是否已启用以及是否占用了任何空间。值 `-` 表示没有可显示的数据。

无限卷成分卷会显示在空间使用情况命令输出中，就好像这些成分卷是 FlexVol 卷一样。

以下示例显示了针对 testvol 卷的 volume show-footprint 命令输出：

```
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%
```

下表对 volume show-footprint 命令输出的某些关键行以及如何减少相关功能的空间使用量进行了说明：

行/功能名称	行说明/内容	减少空间使用量的方法
Volume Data Footprint	在所属聚合内由活动文件系统中的卷数据使用的总空间量以及卷 Snapshot 副本使用的空间量。此行不包括预留空间，因此，如果卷包含预留文件，则在 volume show-space 命令输出中该卷的总使用空间可能会超过此行显示的值。	<ul style="list-style-type: none"> <li>删除卷中的数据。</li> <li>删除卷中的 Snapshot 副本。</li> </ul>
Volume Guarantee	聚合中卷为将来的写入操作所预留的空间量。预留的空间量取决于卷的保证类型。	<p>将卷的保证类型更改为 <b>none</b>。此行将变为 0。</p> <p>如果您要将卷的卷保证类型配置为 <b>none</b>，则应参考技术报告 3965 或 3483 来查看有关卷保证类型为 <b>none</b> 会对存储可用性产生怎样的影响。</p>
Flexible Volume Metadata	聚合中由卷的元数据文件使用的总空间量。	没有直接的控制方法。

行/功能名称	行说明/内容	减少空间使用量的方法
Delayed Frees	Data ONTAP 为保持性能而使用的块，不能立即释放。 当 Data ONTAP 释放 FlexVol 卷中的块时，此空间不会立即在聚合中显示为空闲空间，因为为了提升性能，释放聚合中空间的操作是批量执行的。在 FlexVol 卷中声明为空闲空间但在聚合中尚未成为空闲空间的块被称为“延迟空闲块”（在对关联的延迟空闲块进行处理之前）。 对于 SnapMirror 目标，此行值为 0 且不会显示。	没有直接的控制方法。
File Operation Metadata	为文件操作元数据预留的总空间量。 在文件操作元数据使用空间后，此空间不会作为空闲空间归还给聚合，但后续的文件操作会对其进行重用。	没有直接的控制方法。
Total Footprint	卷在聚合中使用的总空间量。该值是所有行的总和。	卷为减少空间用量而使用的任何方法。

#### 相关概念

[在 FlexVol 卷中创建空间的方法](#)（第 42 页）

[在聚合中创建空间的方法](#)（第 44 页）

[什么是卷占用空间](#)（第 39 页）

#### 相关信息

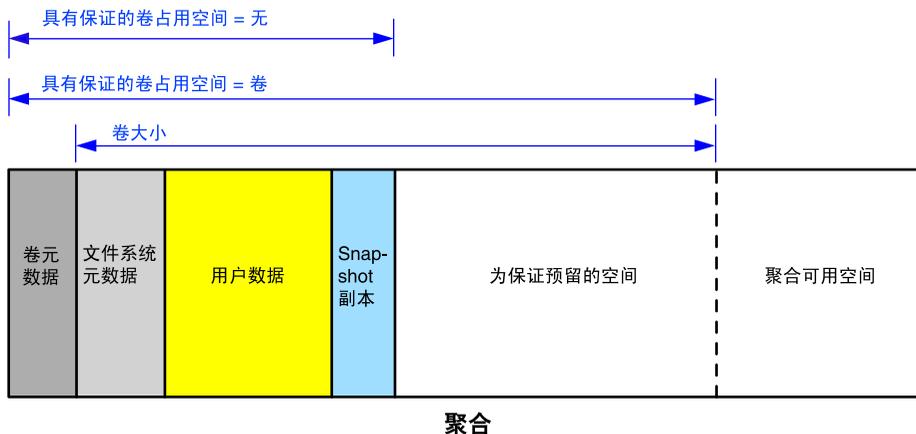
[NetApp 技术报告 3965: 《NetApp 精简配置部署和实施指南（7-模式 Data ONTAP 8.1）》](#)

[NetApp 技术报告 3483: 《NetApp SAN 或 IP SAN 企业环境中的精简配置》](#)

#### 什么是卷占用空间

卷占用空间是卷在聚合中使用的空间量。了解卷占用空间所包含的内容有助于了解卷的空间要求。

卷占用空间包含由用户数据和元数据（包括驻留在聚合中而不是驻留在卷本身之中的元数据）使用的空间。因此，卷占用空间可能会大于卷大小，如下图所示：



### 如何确定聚合中的空间使用量

您可以使用 `aggregate show-space` 命令查看一个或多个聚合中所有卷的空间使用量。这样即可看到哪些卷在所属聚合中占用的空间最大，以便您可以执行相应的操作来释放更多空间。

聚合中的已用空间直接受其包含的 FlexVol 卷和无限卷成分卷所用的空间影响。您为增加卷空间而采取的措施也会影响聚合中的空间。

当聚合处于脱机状态时，不会显示任何值。命令输出将只显示非零值。但是，您可以使用 `-instance` 参数来显示所有可能的功能行，而不管其是否已启用以及是否占用了任何空间。值 `-` 表示没有可显示的数据。

`aggregate show-space` 命令输出中包含以下行：

- **Volume Footprints**  
聚合中所有卷占用空间的总量。该值包括由所属聚合中所有卷的所有数据和元数据占用或预留的所有空间。该值也相当于销毁所属聚合中的所有卷后释放的空间量。无限卷成分卷会显示在空间使用情况命令输出中，就好像这些成分卷是 FlexVol 卷一样。
- **Aggregate Metadata**  
聚合所需的文件系统元数据总量，例如，分配位图和索引节点文件。
- **Snapshot Reserve**  
基于卷大小为聚合 Snapshot 副本预留的空间量。该空间量被视为已用空间，不可用于卷或聚合数据或者元数据。
- **Snapshot Reserve Unusable**  
最初为聚合 Snapshot 预留的空间量，不可用于聚合 Snapshot 副本，因为该空间供与聚合关联的卷使用。仅适用于具有非零聚合 Snapshot 预留的聚合。
- **Total Used**  
聚合中由卷、元数据或 Snapshot 副本所使用或预留的所有空间的总量。

- Total Physical Used  
目前数据所使用的空间量（而不是供将来使用而预留的空间量）。包括聚合 Snapshot 副本所使用的空间。

从来没有快照溢出。

以下示例显示了 Snapshot 预留为 5% 的聚合的 aggregate show-space 命令输出。如果 Snapshot 预留为 0，则不会显示该行。

```
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%
```

## Snapshot 副本和 Snapshot 预留如何使用卷中的空间

了解 FlexVol 卷或无限卷的 Snapshot 预留区以及 Snapshot 溢出的含义有助于正确调整 Snapshot 预留的大小。对于 FlexVol 卷，还有助于确定是否启用 Snapshot 自动删除功能。

如果 Snapshot 副本使用的空间超出 Snapshot 预留的空间，则这些副本会溢出并使用活动文件系统中的空间。卷的 *Snapshot* 预留区是专为 Snapshot 副本预留的空间。该预留区不能由卷的用户数据或元数据区使用。Snapshot 预留大小表示为当前卷大小的指定百分比，与 Snapshot 副本的数量或占用的空间量无关。

如果为 Snapshot 预留分配的所有空间都已用尽，但活动文件系统（用户数据和元数据）未充满，则 Snapshot 副本可以使用比 Snapshot 预留更多的空间并溢出到活动文件系统。此额外空间称为 *Snapshot* 溢出。

下图显示了未发生 Snapshot 溢出的 FlexVol 卷。左侧的两个图块显示了卷的用户数据和元数据已用空间和可用空间。右侧的两个图块显示了 Snapshot 预留中的已用空间和未使用空间。当修改 Snapshot 预留的大小时，右侧的图块将发生变化。



卷

下图显示了发生 Snapshot 溢出的 FlexVol 卷。Snapshot 预留区已满，Snapshot 副本溢出到用户数据和元数据区的可用空间中的溢出区。Snapshot 预留的大小保持不变。



卷

有关 Snapshot 副本的详细信息，请参见《集群模式 Data ONTAP 数据保护指南》。

### 何时应使用 `df` 命令和空间使用量命令

如果要查看有关卷或聚合中的已用空间和可用空间的简洁信息，请使用 `df` 命令。如果要查看卷中的空间使用量按功能统计的详细细分或卷在某个聚合中占用的空间，请使用空间使用量命令。

如果要快速查看每个卷的可用空间量和已用空间量，则 `df` 命令很有用。

可以使用 `df` 命令（或者 `volume show` 命令和 `aggregate show` 命令）查看总空间、可用空间和已用空间。如果需要查看有关卷或聚合中的空间使用情况或使用原因的详细信息，请使用 `show-space` 命令和 `show-footprint` 命令（这两个空间使用量命令分别适用于卷和聚合）。

另一方面，空间使用量命令可提供有关已用空间以及哪项 Data ONTAP 功能正在使用空间的更多详细信息。例如，可以使用这些命令来帮助了解为什么 `df` 命令即使在卷中没有任何数据时也会显示已用空间。

即使对客户端未在访问的系统来说，已用空间也是动态的。因此，不应尝试详细比较两个不同的空间使用量命令的输出，甚至也不应详细比较两次调用同一个命令的输出。

#### 相关参考

[用于显示空间使用量信息的命令](#)（第 50 页）

## 在 FlexVol 卷中创建空间的方法

有多种方法可在 FlexVol 卷中创建空间。了解这些方法的详情及其相应的优点和缺点有助于确定哪种方法最符合要求。

在卷中创建空间的一些常用方法如下所示：

- 增加卷的大小。  
可以手动执行此操作，或者也可以通过启用自动增长功能来自动执行。
- 减小 Snapshot 预留的大小（当 `df` 命令显示 Snapshot 预留的填满度未达到 100% 时）。

此方法可将空间提供给活动文件系统使用。

- 在聚合中创建更多空间。  
此方法可直接或间接为卷创建更多空间，例如：
  - 通过在聚合中创建更多空间，可使卷能够通过自动增长功能自动增加大小。
  - 对于保证类型为 none 的卷，聚合中有限的可用空间会使卷中的可用空间也显得相当有限。
- 启用存储效率技术，例如重复数据删除和压缩。
- 删除卷 Snapshot 副本（当 Snapshot 预留的填满度达到 100% 并且 Snapshot 副本将要溢出到活动文件系统时）。  
可以手动删除 Snapshot 副本，或者也可以通过为该卷启用 Snapshot 自动删除功能来自动删除。
- 手动删除 FlexClone 文件和 FlexClone LUN，或者启用自动删除。
- （临时方法）将预留百分比更改为 0（当卷包含预留的文件或 LUN 并且预留百分比为 100% 时）。  
此方法仅可用作创建空间的临时措施。当预留百分比设置为 0 时，覆盖操作可能会失败，而且某些部署可能无法接受写入失败。
- 删除文件。  
如果卷的填满度已达到 100%，则当该卷参与任何块共享（例如卷 Snapshot 副本或重复数据删除）时，可能无法删除文件，并且无法恢复空间。此外，通过修改目录的方式删除文件可能需要更多空间，因此，删除文件会实际占用空间。在这些情况下，可以执行以下操作之一：
  - 可以使用 `rm` 命令（在高级权限级别下可用）来删除已满且包含 Snapshot 副本的卷中的文件。
  - 可以使用以上列出的其他任意方法在卷和聚合中创建更多空间，以便有足够空间可供文件删除操作使用。

#### 相关概念

[在聚合中创建空间的方法](#)（第 44 页）

[如何确定和控制聚合中卷的空间使用量](#)（第 37 页）

[使用重复数据删除和数据压缩提高存储效率](#)（第 123 页）

[FlexVol 卷如何从 FlexClone 文件和 FlexClone LUN 回收可用空间](#)（第 71 页）

[设置预留百分比的注意事项](#)（第 23 页）

#### 相关任务

[将卷配置为自动增长和缩减其大小](#)（第 27 页）

[自动删除 Snapshot 副本](#)（第 26 页）

## 在聚合中创建空间的方法

如果聚合的可用空间已用完，则会产生丢失数据或禁用卷保证等各种问题。有多种方法可在聚合中创建更多空间。

每种方法都会产生不同的后果。在执行操作之前，您应阅读本文档中的相关章节。

下面按照后果从少到多的顺序列出了在聚合中创建空间的一些常见方法：

- 向聚合中添加磁盘。
- 将一些卷移动到具有可用空间的其他聚合。
- 缩减聚合中具有卷保证的卷的大小。  
此操作可以手动执行，也可以通过自动调整功能的 `autoshrink` 选项来执行。
- 在使用较大空间的卷（预留文件较大的具有卷保证的大型卷）上将卷保证类型更改为 `none`，以减少卷在聚合中占用的空间。  
保证类型为 `none` 的卷在聚合中占用的空间比保证类型为 `volume` 的卷要少。`volume show-footprint` 命令输出的 `Volume Guarantee` 行显示了卷是否因其保证类型而在聚合中预留了大量的空间。
- 如果卷的保证类型为 `none`，则删除不需要的卷 `Snapshot` 副本。
- 删除不需要的卷。
- 启用节省空间功能，例如，重复数据删除或数据压缩。
- （暂时）禁用使用大量元数据的功能（通过 `volume show-footprint` 命令进行查看）。

### 相关概念

- [在 FlexVol 卷中创建空间的方法](#)（第 42 页）
- [移动和复制卷（仅限集群管理员）](#)（第 51 页）
- [如何确定和控制聚合中卷的空间使用量](#)（第 37 页）

### 相关任务

- [将卷配置为自动增长和缩减其大小](#)（第 27 页）
- [自动删除 Snapshot 副本](#)（第 26 页）

### 相关信息

- [NetApp 技术报告 3965: 《NetApp 精简配置部署和实施指南（7-模式 Data ONTAP 8.1）》](#)
- [NetApp 技术报告 3483: 《NetApp SAN 或 IP SAN 企业环境中的精简配置》](#)

## 更改文件容量或目录容量时的警告和注意事项

如果数据需要大量文件或非常大的目录，则可以扩展 Data ONTAP 文件容量或目录容量。但是，在执行此操作之前，您应了解相关限制和注意事项。

### 更改 FlexVol 卷上允许的最大文件数的注意事项

FlexVol 卷可包含的最大文件数会受到限制。可以更改卷的最大文件数，但执行此操作之前，应了解此更改对卷具有哪些影响。

卷可包含的文件数由该卷具有的索引节点数决定。索引节点是一种数据结构，其中包含有关文件的信息。卷同时具有私有索引节点和公共索引节点。公共索引节点用于对用户可见的文件；私有索引节点用于由 Data ONTAP 在内部使用的文件。只能更改卷的最大公共索引节点数；不能更改私有索引节点数。

Data ONTAP 会自动基于新建的卷的大小为该卷设置最大公共索引节点数：每 32 KB 卷大小 1 个索引节点。当卷大小增加时，Data ONTAP 也会根据需要增加最大公共索引节点数（直接由管理员增加或由 Data ONTAP 自动通过自动调整功能增加），以便每 32 KB 卷大小至少有 1 个索引节点，直至卷大小达到大约 1 TB 为止。使卷大小增长到 1 TB 以上不会自动生成更多索引节点，因为 Data ONTAP 不会自动创建超过 33,554,409 个索引节点。如果所需的文件数超过任何卷大小的默认文件数，可以使用 `volume modify` 命令增加该卷的最大索引节点数。

也可以减少最大公共索引节点数。这不会更改当前分配给索引节点的空间量，但肯定会降低公共索引节点文件可占用的最大空间量。然而，为索引节点分配了空间之后，该空间永远不会退回到卷中。因此，即使将最大索引节点数降至当前分配的索引节点数以下，也无法使已分配但未使用的索引节点所使用的空间退回到卷中。

### 为 FlexVol 卷增加最大目录大小时的注意事项

FlexVol 卷的默认最大目录大小与型号无关，并已针对系统内存的大小进行了优化。增加最大目录大小之前，请寻求技术支持。

您可以使用 `volume modify` 命令的 `-maxdir-size` 选项来增加特定 FlexVol 卷的默认最大目录大小，但这样做会影响系统性能。此命令对无限卷不起作用。

## 对 Flash Pool 聚合中的卷使用缓存策略

您可以通过在 `volume create` 命令中使用 `-caching-policy` 参数来更改位于 Flash Pool 聚合上的卷的缓存策略。在 Flash Pool 聚合上创建卷时，将默认为该卷分配 `auto` 缓存策略。

大多数情况下，最好使用默认缓存策略。只有在其他策略可以提升性能时，才应更改卷的缓存策略。

在 Flash Pool 聚合上创建卷时，您可以设置卷的缓存策略。使用 `volume modify` 命令可以修改缓存策略。也可以在 Flash Pool 聚合和非 Flash Pool 聚合之间移动缓存策略。

以下所列为缓存策略、说明以及可根据卷使用情况设置的读写缓存策略组合：

策略名称	说明	读取缓存策略	写入缓存策略	特权
auto	将所有元数据块和随机读取的用户数据块读取到缓存并将所有随机覆盖的用户数据块写入缓存。	random_read	random-write	管理
none	不缓存任何用户数据或元数据块。	none	none	管理
random_read	将所有元数据块和随机读取的用户数据块读取到缓存。	random_read	none	高级
noread-random_write	将所有随机覆盖的用户数据块写入缓存。	none	random-write	高级
meta	只将元数据块读取到缓存。	meta	none	高级
meta-random_write	将所有元数据块读取到缓存并将所有随机覆盖的用户数据块写入缓存。	meta	random-write	高级
random_read_write	将所有元数据、随机读取的用户数据块和随机写入的用户数据块读取到缓存。	random_read_write	none	高级
random_read_write-random_write	将所有元数据、随机读取的用户数据块和随机写入的用户数据块读取到缓存。同时将随机覆盖的用户数据块写入缓存。	random_read_write	random-write	高级

相关信息

[《集群模式 Data ONTAP 8.3 物理存储管理指南》](#)

## 节点根卷和根聚合需要遵循的规则

节点的根卷包含该节点的特殊目录和文件。根聚合包含根卷。节点的根卷和根聚合需要遵循一些规则。

节点的根卷是一个在出厂时安装或由安装软件安装的 FlexVol 卷。它已预留用于保存系统文件、日志文件和核心文件。目录名称为 `/mroot`，只有技术支持才能通过 `systemshell` 对其进行访问。节点的最小根卷大小取决于平台的型号。

- 节点的根卷需要遵循以下规则：
  - 请不要修改根卷的配置或内容，除非技术支持指示您这样做。
  - 不要将用户数据存储存储在根卷中。  
将用户数据存储存储在根卷中会增加 HA 对中节点之间的存储交还时间。
  - 如果需要将其他卷指定为新的根卷或需要将根卷移至另一聚合，请联系技术支持人员。
- 根聚合必须专门用来存放根卷。  
请勿在根聚合中放置或创建数据卷。

相关信息

[NetApp Hardware Universe](#)

## FlexVol 卷基本管理

您可以创建和删除 FlexVol 卷、更改其基本属性以及显示有关其空间使用情况的信息。

### 创建 FlexVol 卷

您可以使用 `volume create` 命令创建 FlexVol 卷并指定其属性。

开始之前

新卷的 Storage Virtual Machine (SVM) 以及将为该卷提供存储的聚合必须已存在。如果该 SVM 具有一个关联聚合列表，则必须将该聚合包含到此列表中。

步骤

1. 使用 `volume create` 命令创建卷。

### 示例

以下命令将在 SVM vs1 和聚合 aggr2 上创建名为 dept\_eng 的新卷。该卷位于 vs1 SVM 命名空间的 /dept/eng 目录下。该卷的大小为 750 GB，并且其卷保证类型是 volume（默认值）。

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

## 删除 FlexVol 卷

您可以删除不再需要的或包含损坏的数据的 FlexVol 卷。

### 开始之前

必须确保没有任何应用程序正在访问要删除的卷中的数据。

注：如果意外删除了某个卷，请联系技术支持寻求帮助。

### 步骤

1. 如果卷已挂载，请通过输入以下命令卸载卷：

```
volume unmount -vserver vserver_name -volume volume_name
```

2. 如果卷是 SnapMirror 关系的一部分，请使用 snapmirror delete 命令删除该关系。

3. 如果卷处于联机状态，请通过输入以下命令使卷脱机：

```
volume offline -vserver vserver_name volume_name
```

4. 通过输入以下命令删除卷：

```
volume delete -vserver vserver_name volume_name
```

### 结果

此时即会删除卷以及任何关联的配额策略和 qtrees。

## 使用存储服务质量 (QoS) 控制和监控 FlexVol 卷的 I/O 性能

您可以通过将 FlexVol 卷分配到存储服务质量 (QoS) 策略组，控制这些卷的输入/输出 (I/O) 性能。通过控制 I/O 性能，您可以确保工作负载实现特定性能目标或限制对其他工作负载产生不利影响的工作负载。

### 关于本任务

策略组会实施最大吞吐量限制（例如，100 MB/秒）。您可以创建未指定最大吞吐量的策略组，这样便能先监控性能，然后再控制工作负载。

您还可以将包含 FlexVol 卷、LUN 和文件的 Storage Virtual Machine (SVM) 分配到策略组。

将卷分配到策略组时，请注意以下要求：

- 卷必须包含在策略组所属的 SVM 中。  
创建策略组时指定 SVM。
- 如果将一个卷分配到策略组，则不能将此卷所属的 SVM 或此卷的任何子 LUN 或文件分配到策略组。

注：可以在最多拥有八个节点的集群上支持存储服务质量 (QoS)。

有关如何使用存储服务质量 (QoS) 的详细信息，请参见《适用于集群管理员的集群模式 *Data ONTAP* 系统管理指南》。

#### 步骤

1. 使用 `qos policy-group create` 命令创建策略组。
2. 使用 `volume create` 命令或带有 `-qos-policy-group` 参数的 `volume modify` 命令，以将卷分配到策略组。
3. 使用 `qos statistics` 命令查看性能数据。
4. 如果需要，请使用 `qos policy-group modify` 命令调节策略组的最大吞吐量限制。

## 显示文件或索引节点使用量

FlexVol 卷可包含的最大文件数会受到限制。了解卷包含的文件数有助于确定是否需要增加卷的（公共）索引节点数，以防止卷达到最大文件数限制。

#### 关于本任务

公共索引节点可以是可用节点（它们未与任何文件关联）或已用节点（它们指向某个文件）。卷的可用索引节点数等于卷的索引节点总数减去已用索引节点数（即文件数）。

#### 步骤

1. 要显示卷的索引节点使用量，请输入以下命令：

```
df -i volume_name
```

卷名称可以忽略；在这种情况下，Data ONTAP 将显示集群中所有卷的索引节点使用量。您还可以指定 Storage Virtual Machine (SVM)，以便只查看该 SVM 上的卷。

## 示例

```
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.
```

## 用于管理 FlexVol 卷的命令

有特定的命令可用于通过 Data ONTAP CLI 管理 FlexVol 卷。

操作	命令
将卷置于联机状态	<code>volume online</code>
更改卷的大小	<code>volume size</code>
确定卷的关联聚合	<code>volume show</code>
确定 Storage Virtual Machine (SVM)	<code>volume show -vserver &lt;vserver_name&gt; -fields aggregate</code>
确定卷的格式	<code>volume show -fields block-type</code>
使用接合将卷挂载到其他卷	<code>volume mount</code>
将卷置入受限状态	<code>volume restrict</code>
重命名卷	<code>volume rename</code>
使卷脱机	<code>volume offline</code>

有关详细信息，请参见各个命令的手册页。

## 用于显示空间使用量信息的命令

您使用 `storage aggregate` 和 `volume` 命令来查看聚合、卷及其 Snapshot 副本的空间使用情况。

显示信息	命令
聚合，包括有关已用空间百分比和可用空间百分比的详细信息、Snapshot 预留大小以及其他空间使用量信息	<code>storage aggregate show</code> <code>storage aggregate show-space -snap-size-total,-used-including-snapshot-reserve</code>

显示信息	命令
聚合中的磁盘和 RAID 组的使用情况以及 RAID 状态	<code>storage aggregate show-status</code>
删除特定 Snapshot 副本后可回收的磁盘空间量	<code>volume snapshot compute-reclaimable</code> (高级)
卷使用的空间量	<code>volume show -fields size,used,available,percent-used volume show-space</code>
卷在所属聚合中使用的空间量	<code>volume show-footprint</code>

#### 相关概念

[如何确定和控制卷中的空间使用量](#) (第 33 页)

[如何确定和控制聚合中卷的空间使用量](#) (第 37 页)

[如何确定聚合中的空间使用量](#) (第 40 页)

[Snapshot 副本和 Snapshot 预留如何使用卷中的空间](#) (第 41 页)

#### 相关信息

[《集群模式 Data ONTAP 8.3 命令：手册页参考》](#)

## 移动和复制卷（仅限集群管理员）

可以通过移动或复制卷来提高容量利用率和性能以及满足服务级别协议的要求。

### FlexVol 卷移动的工作原理

FlexVol 卷可以在同一个 Storage Virtual Machine (SVM) 中从一个聚合或节点移动到另一个聚合或节点，从而提高容量利用率并改善性能，同时满足服务级别协议的要求。

卷移动期间，客户端访问不会中断。

卷的移动分为多个阶段：

- 在目标聚合上创建一个新卷。
- 将数据从原始卷复制到此新卷。  
在此期间，原始卷保持完好，并可供客户端访问。
- 移动过程结束时，系统将暂时阻止客户端访问。  
在此期间，系统将执行从源卷到目标卷的最后复制，交换源卷与目标卷的标识，并将目标卷更改为源卷。

- 完成移动之后，系统将客户端流量路由到新的源卷并恢复客户端访问。

移动操作不会中断客户端访问，因为在客户端发现中断并超时之前，阻止客户端访问的时间已结束。默认情况下，客户端访问会被阻止 45 秒。如果卷移动操作无法在拒绝访问期间完成，则系统将中止卷移动操作的最后阶段，并允许客户端访问。默认情况下，系统将尝试最后阶段三次。在第三次尝试之后，系统将等待一个小时，然后才会再次尝试最后阶段。系统将运行卷移动操作的最后阶段，直至卷移动完成。

如果您认为默认值不够，则可以更改阻止客户端访问的时间长短或运行卷移动操作的最后阶段的次数（转换尝试次数）。此外，还可以确定当卷移动操作在客户端访问被阻止期间无法完成时，系统要执行的操作。`volume move start` 手册页介绍了有关在不中断客户端访问的情况下移动卷的详细信息。

## 用于移动卷的命令

有特定的 Data ONTAP 命令可用于管理卷移动。

操作	命令
中止处于活动状态的卷移动操作。	<code>volume move abort</code>
显示卷从一个聚合移动到另一个聚合的状态。	<code>volume move show</code>
开始将卷从一个聚合移动到另一个聚合。	<code>volume move start</code>
管理卷移动的目标聚合。	<code>volume move target-aggr</code>
触发移动作业的转换。	<code>volume move trigger-cutover</code>

有关详细信息，请参见各个命令的手册页。

## 移动卷时的注意事项和建议

移动卷时，应了解许多注意事项和建议，这些注意事项和建议会受到要移动的卷或系统配置（例如 MetroCluster 配置）的影响。您应了解与移动卷相关的注意事项和建议。

### 一般注意事项和建议

- 不能移动无限卷。
- 如果正在升级集群的版本系列，请勿移动卷，直到升级完该集群中所有节点为止。此建议可防止用户无意中从较新的版本系列移动到较旧的版本系列。
- 源卷必须保持一致。
- 如果已将一个或多个聚合分配给关联的 Storage Virtual Machine (SVM)，则目标聚合必须是已分配的聚合之一。
- 不能将卷移动到接管的 CFO 聚合，也不能从接管的 CFO 聚合移动卷。

- 如果包含 LUN 的卷在移动前未启用 NVFAIL，则会在移动后为其启用 NVFAIL。
- 您可以将卷从一个 Flash Pool 聚合移动到另一个 Flash Pool 聚合。
  - 该卷的缓存策略也会移动。
  - 移动可能会影响卷性能。
- 您可以在 Flash Pool 聚合与非 Flash Pool 聚合之间移动卷。
  - 如果将卷从 Flash Pool 聚合移动到非 Flash Pool 聚合，则 Data ONTAP 会显示一条消息，警告您移动操作可能会影响卷性能并询问您是否要继续。
  - 如果将卷从非 Flash Pool 聚合移动到 Flash Pool 聚合，则 Data ONTAP 会分配 `auto` 缓存策略。

### FlexClone 卷注意事项和建议

- 移动 FlexClone 卷时，这些卷不能处于脱机状态。
- 您可以将 FlexClone 卷从一个聚合移动到同一个 SVM 中相同节点或不同节点上的另一个聚合，而不必进行拆分。
- 移动克隆之后，FlexClone 卷 Snapshot 副本不会丢失。
- 您可以将 FlexClone 父卷从一个聚合移动到另一个聚合。  
移动 FlexClone 父卷时，系统会保留一个临时卷，用作所有 FlexClone 卷的父卷。除了使临时卷脱机或将其删除外，不允许对临时卷执行任何操作。拆分或销毁所有 FlexClone 卷之后，系统会自动清除临时卷。
- 移动 FlexClone 子卷之后，该卷不再是 FlexClone 卷。
- FlexClone 移动操作不能与 FlexClone 复制操作或拆分操作同时进行。
- 如果克隆拆分操作正在进行，则卷移动可能会失败。  
在克隆拆分操作完成之前，您不应移动卷。

### MetroCluster 配置注意事项

- 如果在转换之前发生 MetroCluster 切换，则目标卷会创建一条记录，并成为临时卷（卷类型为 TMP）。  
Data ONTAP 不会删除此临时卷；此卷必须手动删除。  
移动 MetroCluster 配置中的卷时，如果在源集群的目标聚合上创建了与已镜像但未恢复的聚合中的卷对应的临时卷，则会为该临时卷创建一条记录。
- 如果在开始转换阶段之后而移动操作完成之前发生 Metrocluster 切换，则目标聚合上的卷移动操作会运行完毕，但 Data ONTAP 不会删除源聚合上的卷。  
您必须手动删除该卷。

- 如果正在进行任何卷移动操作，则不允许进行强制和非强制的 MetroCluster 切换。
- 如果正在进行任何卷移动操作，则非强制的 Metrocluster 切换会被阻止，而强制的 Metrocluster 切换则不会被阻止。

## 在 SAN 中移动卷的要求

移动包含一个或多个 LUN 的卷之前，应为连接到集群中各个节点的每个 LUN 配置至少 2 个路径。这样可消除单点故障，并使系统能够在发生组件故障时继续工作。

## 移动卷

确定存在存储容量不平衡后，可以将 FlexVol 卷移动到同一个 Storage Virtual Machine (SVM) 的其他聚合或节点（或者也可以移动到其他节点的其他聚合）以平衡存储容量。

关于本任务

管理员之前已确定正在访问卷中的数据的客户端能够承受最多 120 秒的 I/O 超时。

步骤

1. 如果要移动的是数据保护镜像，而且尚未初始化镜像关系，请使用 `snapmirror initialize` 命令初始化镜像关系。  
必须先初始化数据保护镜像关系，然后才能移动其中一个卷。
2. 使用 `volume move target-aggr show` 命令确定可以将卷移动到哪个聚合。  
所选聚合必须具有用于容纳该卷的足够空间；也就是说，其可用大小必须大于要移动的卷。

示例

以下示例显示了可以将卷 vs2 移动到列出的任意聚合：

```
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. 验证是否可以将卷移动到预定聚合，方法是使用 `volume move start -perform-validation-only` 命令运行验证检查。
4. 使用 `volume move start` 命令移动卷。

## 示例

以下命令可将 vs2 SVM 上的 user\_max 卷移动到 node12a\_aggr3 聚合。移动操作作为后台进程运行。

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

## 5. 使用 volume move show 命令确定卷移动操作的状态。

## 示例

以下示例显示了已完成复制阶段并且正处于转换阶段的卷移动的状态。

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

## 复制卷的方法

复制卷会创建该卷的独立副本，此副本可用于测试及其他目的。用来复制卷的方法取决于使用情形。

用来复制卷的方法取决于是要复制到同一个聚合还是复制到不同的聚合，并取决于是否要保留原始卷中的 Snapshot 副本。下表列出了副本的特性以及用于创建该副本的方法。

复制卷	方法
在同一个聚合中复制，并且不想复制原始卷中的 Snapshot 副本。	创建原始卷的 FlexClone 卷。
复制到其他聚合，并且不想复制原始卷中的 Snapshot 副本。	创建原始卷的 FlexClone 卷，然后使用 volume move 命令将创建的卷移动到其他聚合。
复制到其他聚合，并保留原始卷中的所有 Snapshot 副本。	使用 SnapMirror 复制原始卷，然后中断 SnapMirror 关系以使复制的卷成为读写卷副本。

### 相关任务

[创建 FlexClone 卷](#) (第 59 页)

### 相关信息

[《集群模式 Data ONTAP 8.3 数据保护指南》](#)

## 使用 FlexClone 卷创建 FlexVol 卷的高效副本

FlexClone 卷是父 FlexVol 卷的可写时间点副本。FlexClone 卷可节省空间，因为它们与其父 FlexVol 卷共享相同的数据块来存储共用数据。用于创建 FlexClone 卷的 Snapshot 副本也与父卷共享。

可以克隆现有 FlexClone 卷以创建另一个 FlexClone 卷。也可以创建包含 LUN 和 LUN 克隆的 FlexVol 卷的克隆。

自 Data ONTAP 8.2 起，可以创建以下两种类型的 FlexClone 卷：读写 FlexClone 卷和数据保护 FlexClone 卷。可以创建常规 FlexVol 卷的读写 FlexClone 卷，但只能使用 SnapVault 二级卷来创建数据保护 FlexClone 卷。

### 了解 FlexClone 卷

FlexClone 卷可采用与常规 FlexVol 卷类似的方式来管理，但也存在一些重要区别。例如，创建 FlexClone 卷后对父 FlexVol 卷进行的更改不会反映在 FlexClone 卷中。

下面列出了与 FlexClone 卷有关的重要信息：

注：除非另有说明，否则以下陈述既适用于读写 FlexClone 卷又适用于数据保护 FlexClone 卷。

- FlexClone 卷是父 FlexVol 卷的可写时间点副本。
- FlexClone 卷是一种功能完备的 FlexVol 卷，与其父卷类似。
- FlexClone 卷始终与其父卷在同一个聚合中创建。
- FlexClone 卷与其父卷始终创建在同一个 Storage Virtual Machine (SVM) 中。
- 无限卷不能用作 FlexClone 卷的父卷。
- 由于 FlexClone 卷与其父卷会使用相同的磁盘空间来存储通用的数据，因此，创建 FlexClone 卷可即时完成，不需要其他磁盘空间（除非对 FlexClone 卷或其父卷进行更改）。
- FlexClone 卷在创建后与其父卷具有相同的卷保证。  
只有当包含新 FlexClone 卷的聚合具有足够空间时，才可以为该卷强制实施卷保证设置。
- FlexClone 卷在创建后与其父卷具有相同的空间预留及预留百分比设置。
- FlexClone 卷在创建后与其父卷具有相同的 Snapshot 计划。
- FlexClone 卷在创建后与其父卷具有相同的语言设置。
- 如果存在 FlexClone 卷，则 FlexClone 卷与其父卷共享的通用 Snapshot 副本就不能删除。
- 如果存在 FlexClone 卷，则不允许对其父卷执行某些操作，如删除父卷。

- 不能为处于部分交还状态的存储系统中的卷创建克隆。
- 可以断开父卷和读写 FlexClone 卷之间的连接。  
该操作称为拆分 FlexClone 卷。拆分后，会解除父卷上的所有限制，并使 FlexClone 卷使用自己的额外磁盘空间，而不再与其父卷共享空间。  
注：不能将数据保护 FlexClone 卷从其父卷中拆分出来。  
注意：如果将 FlexClone 卷从其父卷中拆分出来，则会删除 FlexClone 卷的所有现有 Snapshot 副本，并在执行拆分操作期间禁止创建新的 Snapshot 副本。  
如果要保留 FlexClone 卷的 Snapshot 副本，则可以使用 volume move 命令将 FlexClone 卷移动到其他聚合中。在卷移动操作期间，如有必要，还可以创建新的 Snapshot 副本。
- 应用于父卷的配额不会自动应用于 FlexClone 卷。
- 创建 FlexClone 卷后，父卷中的所有 LUN 都将出现在 FlexClone 卷中，但是处于未映射和脱机状态。

#### 相关概念

[FlexVol 卷移动的工作原理](#)（第 51 页）

#### 相关任务

[移动卷](#)（第 54 页）

## FlexClone 卷和共享 Snapshot 副本

当卷保证生效时，新 FlexClone 卷使用与其父卷共享的 Snapshot 副本，以最大程度地降低空间需求。如果删除共享 Snapshot 副本，则可能会增加 FlexClone 卷的空间需求。

例如，假设您有一个 100 MB 的 FlexVol 卷，卷保证为卷，已使用 70 MB，剩余 30 MB 可用，您将该 FlexVol 卷用作新 FlexClone 卷的父卷。该新 FlexClone 卷的初始卷保证为卷，但它不需要聚合提供的全部 100 MB 空间（如果您复制了该卷，则需要全部 100 MB 空间）。相反，聚合仅需要向克隆分配 30 MB（100 MB 减去 70 MB）的可用空间。

现在，假设已从 FlexClone 卷删除共享 Snapshot 副本。FlexClone 卷无法再优化其空间需求，需要所在聚合提供全部 100 MB 的空间。

注：如果由于“聚合中空间不足”而无法从 FlexClone 卷删除 Snapshot 副本，这是因为删除该 Snapshot 副本需要分配的空间要多于当前聚合具有的可用空间。您可以增加聚合的大小或更改 FlexClone 卷的卷保证。

## 如何识别 FlexClone 卷中的共享 Snapshot 副本

可以通过以下方法识别共享 Snapshot 副本：使用带有 `-instance` 参数的 `volume snapshot show` 命令列出父卷中的 Snapshot 副本。任何在父卷中标记为繁忙并同时存在于 FlexClone 卷中的 Snapshot 副本都是共享 Snapshot 副本。

## 如何将卷 SnapMirror 复制与 FlexClone 卷配合使用

由于卷 SnapMirror 复制和 FlexClone 卷都依赖 Snapshot 副本，因此存在一些与这两个功能结合使用的方式相关的限制。例如，可以将 FlexClone 卷或其父卷作为源卷来创建卷 SnapMirror 关系，但不能将 FlexClone 卷或其父卷作为目标卷来创建新的卷 SnapMirror 关系。

## 从 SnapMirror 源卷或目标卷创建 FlexClone 卷的注意事项

可以从现有卷 SnapMirror 关系中的源卷或目标卷创建 FlexClone 卷。但是，这样做会妨碍将来的 SnapMirror 复制操作成功完成。

复制可能无法进行，因为在创建 FlexClone 卷时，您可能会锁定 SnapMirror 所使用的 Snapshot 副本。如果发生此问题，SnapMirror 将停止复制到目标卷，直到 FlexClone 卷销毁或与其父卷拆分为止。要解决此问题，可以选择两种方法：

- 如果临时需要 FlexClone 卷，并允许临时停止 SnapMirror 复制，则可以创建 FlexClone 卷，然后在可能的情况下将其删除或将其从父卷拆分。  
在删除 FlexClone 卷或将其从父卷拆分之后，SnapMirror 复制将继续正常进行。
- 如果不允许临时停止 SnapMirror 复制，则可以在 SnapMirror 源卷中创建 Snapshot 副本，然后使用该 Snapshot 副本创建 FlexClone 卷。（如果正在从目标卷创建 FlexClone 卷，则必须等待该 Snapshot 副本复制到 SnapMirror 目标卷。）  
通过在 SnapMirror 源卷中创建 Snapshot 副本，您无需锁定 SnapMirror 正在使用的 Snapshot 副本，即可创建克隆。

## 如何将 FlexClone 卷从其父卷拆分

将读写 FlexClone 卷从其父卷拆分会删除该 FlexClone 卷当前所使用的任何空间优化。拆分后，该 FlexClone 卷和父卷都需要根据其卷保证确定所分配的完整空间。FlexClone 卷将变为普通的 FlexVol 卷。

对于克隆拆分操作，必须注意下列注意事项：

- 只能拆分读写 FlexClone 卷。不能将数据保护 FlexClone 卷从其父卷拆分。
- 在将 FlexClone 卷从其父卷拆分后，该 FlexClone 卷的所有现有 Snapshot 副本都将被删除。如果要保留 FlexClone 卷的 Snapshot 副本，则可以使用 `volume move` 命令将 FlexClone 卷移动到其他聚合中。在卷移动操作期间，如有必要，还可以创建新的 Snapshot 副本。
- 在拆分操作期间，无法为 FlexClone 卷创建新的 Snapshot 副本。

- 由于克隆拆分操作是一种复制操作，可能需要花费大量时间才能完成，因此，Data ONTAP 提供了 `volume clone split stop` 和 `volume clone split status` 命令，用于停止或检查克隆拆分操作的状态。
- 克隆拆分操作继续在后台进行，不会干扰对父卷或克隆卷的数据访问。
- 启动拆分操作时，FlexClone 卷必须处于联机状态。
- 要成功完成拆分操作，父卷必须处于联机状态。
- 如果 FlexClone 卷具有数据保护或负载共享镜像，则不能从其父卷拆分。
- 如果从已启用重复数据删除和数据压缩的 FlexVol 卷拆分 FlexClone 卷，则拆分的卷不会启用重复数据删除和数据压缩。
- FlexClone 卷与其父卷拆分后，不能再重新联结。

## FlexClone 卷和 LUN

可以克隆包含 LUN 和 FlexClone LUN 的 FlexVol 卷。

注：此上下文中的 LUN 是指 Data ONTAP 为客户端提供的 LUN，不是用于存储阵列上的存储的阵列 LUN。

创建 FlexClone 卷时，父卷中的 LUN 将出现在 FlexClone 卷中，但是处于未映射和脱机状态。要使 FlexClone 卷中的 LUN 联机，需要将其映射到启动程序组。

如果父卷包含 FlexClone LUN，则 FlexClone 卷也包含 FlexClone LUN，它们与父卷中的 FlexClone LUN 共享存储。

## 了解数据保护 FlexClone 卷

可以使用 FlexClone 技术为数据保护卷创建节省空间的副本，以用作 SnapVault 二级卷。在主卷和二级卷之间建立 SnapVault 关系的 Snapshot 副本是备用 Snapshot 副本，用于创建数据保护 FlexClone 卷。

数据保护 FlexClone 卷与读写 FlexClone 卷相似，因为它们与其父 FlexVol 卷共享公用数据块。但是，只能从同时也是二级 SnapVault 卷的父 FlexClone 卷创建数据保护 FlexVol 卷。此外，不能将数据保护 FlexClone 卷从其父卷中拆分出来。

有关 SnapVault 关系中的卷的详细信息，请参见《集群模式 Data ONTAP 数据保护指南》。

## 创建 FlexClone 卷

如果要在不使用大量空间的情况下创建数据的即时副本，可以从包含该数据的父 FlexVol 卷创建 FlexClone 卷。根据父卷的类型，可以创建读写 FlexClone 卷或数据保护 FlexClone 卷。

开始之前

- 集群上必须安装 FlexClone 许可证。

- 要克隆的卷必须处于联机状态。

关于本任务

- 您可以从 SnapMirror 目标卷创建数据保护 FlexClone 卷，也可以从父 FlexVol 卷（此卷是 SnapVault 二级卷）创建。
- 创建 FlexClone 卷之后，只要该 FlexClone 卷存在，便无法删除父卷。

步骤

#### 1. 使用 `volume clone create` 命令创建 FlexClone 卷。

注：创建读写 FlexClone 卷时，不必指定基线 Snapshot 副本。如果未指定任何特定 Snapshot 副本以用作克隆的基线 Snapshot 副本，则 Data ONTAP 会创建一个 Snapshot 副本。但是，创建数据保护 FlexClone 卷时，必须指定基线 Snapshot 副本。

示例

- 以下命令将创建父卷 `vol1` 的读写 FlexClone 卷 `vol1_clone`：

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

- 以下命令将使用基线 Snapshot 副本 `snap1` 创建父卷 `dp_vol` 的数据保护 FlexClone 卷 `vol_dp_clone`：

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

## 从父卷中拆分 FlexClone 卷

如果希望读写 FlexClone 卷拥有自己的磁盘空间而不是使用其父磁盘的空间，可以将其从父卷中拆分出来。由于此操作会创建当前在父卷与 FlexClone 卷之间共享的数据的副本，因此可能需要一段时间才能完成。

开始之前

必须确保要从父卷中拆分的 FlexClone 卷是读写 FlexClone 卷。无法从父卷中拆分数据保护 FlexClone 卷。

关于本任务

从父卷中拆分 FlexClone 卷会占用所属聚合中的可用空间。如果没有足够的权限查看聚合中的可用空间，则需要与存储管理员联系以确保能够完成拆分操作。

## 步骤

1. 使用带有 **estimate** 参数的 `volume clone show` 命令，以确定完成拆分操作所需的可用空间。

## 示例

以下示例提供了有关从父卷 `vol1` 中拆分 FlexClone 卷 `clone1` 所需的可用空间的信息：

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

2. 使用 `storage aggregate show` 命令确定包含该 FlexClone 卷及其父卷的聚合中的可用空间。
3. 如果所属聚合没有足够的可用空间，请使用 `storage aggregate add-disks` 命令向该聚合添加存储。
4. 使用 `volume clone split start` 命令开始拆分操作。

## 示例

以下示例显示了启动从父卷 `vol1` 中拆分 FlexClone 卷 `clone1` 的流程：

```
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. 可以使用 `job show` 命令来监控拆分作业的进度。
6. 可以通过使用 `volume show` 命令并将 `fields` 参数设置为 `clone-volume` 来确认已拆分的卷是否不再是 FlexClone 卷。

如果 `clone-volume` 选项的值为 `false`，则表明相关卷不是 FlexClone 卷。

## 示例

以下示例显示了如何验证已从父卷中拆分的卷 `clone1` 是否不是 FlexClone 卷。

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

## 确定 FlexClone 卷所使用的空间

您可以基于 FlexClone 卷的标称大小及其与父 FlexVol 卷共享的空间量来确定该 FlexClone 卷使用的空间。

关于本任务

创建 FlexClone 卷时，该卷与其父卷共享所有数据。因此，虽然 FlexVol 卷的标称大小与其父卷的大小相同，但它只使用聚合中的极少数可用空间。新建的 FlexClone 卷所使用的可用空间大约是其标称大小的 0.5%。此空间用于存储 FlexClone 卷的元数据。

写入到父卷或 FlexClone 卷的新数据不会在这两个卷之间共享。随着写入到 FlexClone 卷的新数据量不断增加，这会导致该 FlexClone 卷需要从其所属的聚合获取更多空间。

步骤

1. 请使用 `volume size` 命令确定 FlexClone 卷的标称大小。

示例

以下示例显示了 FlexClone 卷 `clone1` 的标称大小：

```
cluster1::> volume size -volume clone1
vol size: Volume "vs1:clone1" has size 200m.
```

2. 请使用 `volume clone split estimate` 命令确定在父卷与 FlexClone 卷之间共享的空间量。

示例

以下示例显示了在 FlexClone 卷 `clone1` 与其父卷 `vol1` 之间共享的空间量：

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

3. 将 FlexClone 卷的标称大小减去共享空间的大小，即可确定 FlexClone 卷所使用的可用空间量。

## 使用 FlexClone 文件和 FlexClone LUN 创建文件和 LUN 的高效副本

FlexClone 文件和 FlexClone LUN 是为父文件和父 LUN 创建的可节省空间的 writable 克隆，有助于提高物理聚合空间的利用率。只有 FlexVol 卷才支持 FlexClone 文件和 FlexClone LUN，而无限卷则不支持它们。

FlexClone 文件和 FlexClone LUN 使用自身大小的 0.4% 来存储元数据。克隆与其父文件和父 LUN 共享数据块，并占用极小的存储空间，直到客户端向父文件/父 LUN 或该克隆写入新数据为止。

客户端可以对父实体和克隆实体执行所有文件操作和 LUN 操作。

可以使用多种方法来删除 FlexClone 文件和 FlexClone LUN。

### FlexClone 文件和 FlexClone LUN 的优势

创建 FlexClone 文件或 FlexClone LUN 期间可节省大量空间和时间，因为克隆操作不涉及以物理方式复制任何数据。

在下列情况下，可以通过使用 FlexClone 文件和 FlexClone LUN 来为数据创建节省空间的副本：

- 需要部署、升级或重新部署数千个标准化的虚拟桌面或服务器。
- 需要一个数据库副本以用于应用程序开发目的。
- 需要启动服务器场中的服务器。  
您可为父启动 LUN 创建 FlexClone LUN，然后使用这些 FlexClone LUN 启动服务器场中的服务器。

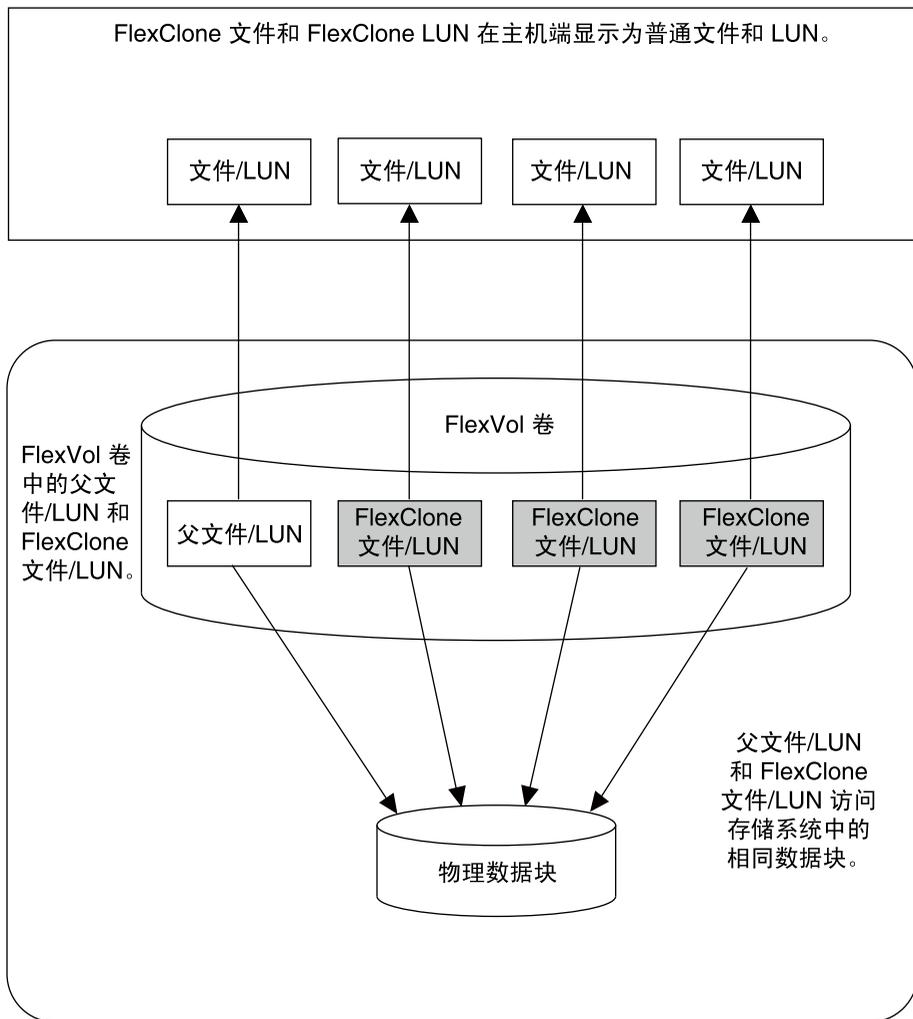
### FlexClone 文件和 FlexClone LUN 的工作原理

FlexClone 文件和 FlexClone LUN 与它们在 FlexVol 或 FlexClone 卷中的父文件和父 LUN 共享相同的物理数据块，并以元数据的形式占用极少的空间。

您可为 NAS 环境中的 FlexVol 卷上的文件创建克隆，也可以克隆 SAN 环境中的 LUN。

由于克隆操作不会复制物理数据块，因此克隆副本可节省大量空间和时间。只有在向父实体或克隆实体写入新数据时，将新数据写入到的实体才会开始占用额外的存储空间。

下图显示了父文件/LUN 和 FlexClone 文件/LUN 会访问存储系统上相同的数据块。在主机端，父文件/LUN 和 FlexClone 文件/LUN 会显示为普通文件和 LUN：



每个节点在达到其最大分摊负载之前都会接受 FlexClone 文件和 FlexClone LUN 创建请求。当节点达到其最大分摊负载时，该节点会暂时停止接受克隆创建请求并发出 `EBUSY` 错误消息。当节点的分摊负载降至最大分摊负载以下时，该节点会再次接受克隆创建请求。

克隆操作不会影响客户端对父文件或父 LUN 的访问。正在访问父文件或父 LUN 的客户端不会出现任何中断或停机。与标准文件和标准 LUN 一样，客户端可以对 FlexClone 文件和 FlexClone LUN 执行所有操作。

在不为父实体创建物理副本的情况下，最多可以从父文件或父 LUN 创建 32,767 个 FlexClone 文件或 FlexClone LUN。如果您试图创建 32,767 个以上的克隆，Data ONTAP 将自动为父文件或父 LUN 创建新的物理副本。

## 相关概念

[分摊负载如何确定用于创建和删除 FlexClone 文件和 FlexClone LUN 的节点容量](#)（第 67 页）

## 处理 FlexClone 文件和 FlexClone LUN 的注意事项

创建、管理和删除 FlexClone 文件和 FlexClone LUN 时，应谨记以下注意事项。

### 创建 FlexClone 文件和 FlexClone LUN 的注意事项

- 只能在包含父文件和父 LUN 的同一个 FlexVol 卷中创建 FlexClone 文件和 FlexClone LUN。
- 您可以克隆整个文件、子文件、LUN 或子 LUN。  
要克隆子文件或子 LUN，必须知道父实体和克隆实体的块范围。
- 首次创建 FlexClone 文件或 FlexClone LUN 时，将向 FlexVol 卷添加 `sis` 属性。
- 如果要在 FlexVol 卷的两个 qtree 之间克隆文件或 LUN，则目标 qtree 不能是 SnapVault 目标的次级。

### 管理 FlexClone 文件和 FlexClone LUN 的注意事项

- 当客户端向 FlexClone 文件或 FlexClone LUN 写入新数据，或者向父文件或父 LUN 写入新数据时，新数据会占用额外存储空间。
- 从 Snapshot 副本创建 FlexClone 文件或 FlexClone LUN 时，克隆过程完成之前将无法创建新的 Snapshot 副本。
- 如果包含 FlexClone 文件和 FlexClone LUN 的 FlexVol 卷的预留百分比设置为零，并且访问该卷的应用程序由于空间不足而遇到错误 (ENOSPC)，请遵守配置限制。  
您还应该为 FlexVol 卷启用自动删除设置，以便自动删除 FlexClone 文件和 FlexClone LUN，以帮助访问 FlexVol 卷的应用程序避免因空间不足而遇到错误。

### 删除 FlexClone 文件和 FlexClone LUN 的注意事项

- 删除 FlexClone 文件或 FlexClone LUN 时，父文件或父 LUN 不受影响。  
删除父文件或父 LUN 不会对 FlexClone 文件或 FlexClone LUN 造成影响。
- 如果要在不使用 NetApp 易管理性 SDK 的情况下删除 FlexClone 文件，可以使用 `volume file clone deletion` 命令对系统进行配置以提高删除速度。  
如果使用 NetApp 易管理性 SDK 来删除 FlexClone 文件和 FlexClone LUN，则不需要对系统进行配置，因为始终会使用速度较快的删除方法。

## 相关概念

[设置预留百分比的注意事项](#)（第 23 页）

## 相关任务

将 *FlexVol* 卷配置为自动删除 *FlexClone* 文件和 *FlexClone LUN* (第 72 页)

## 相关参考

用于配置 *FlexClone* 文件删除的命令 (第 75 页)

## 创建经过空间优化的 **FlexClone** 文件和 **FlexClone LUN** 的示例

以下示例说明了 Data ONTAP 8.3 及更高版本为什么不需要额外空间便能克隆预留百分比设置为零的 *FlexVol* 卷中启用了空间预留的文件和 LUN，同时它还介绍了一些不支持经过空间优化的克隆的场景。

### 创建启用了空间预留的 **FlexClone** 文件的示例

假设要克隆预留百分比设置为零的 *FlexVol* 卷中一个大小为 100 GB 并且没有空洞的文件，而且为该文件启用了空间预留，并将为 *FlexClone* 文件启用空间预留。

以下示例说明了如何克隆 `file1_source` 以创建名为 `file1_clone` 的 *FlexClone* 文件。由于未指定空间预留，因此 *FlexClone* 文件将继承与源文件（此文件启用了空间预留）相同的空间预留设置：

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

在 Data ONTAP 8.3 及更高版本中，不需要额外的临时空间便能创建 *FlexClone* 文件。在先前版本中，需要至少 200 GB 的可用空间才能从预留百分比设置为零的 *FlexVol* 卷中一个大小为 100 GB 并启用了空间预留的文件创建启用了空间预留的 *FlexClone* 文件。现在，上述操作不再需要额外空间，因为 *FlexClone* 文件和 *FlexClone LUN* 已经过空间优化。

### 创建禁用了空间预留的 **FlexClone** 文件的示例

假设要克隆预留百分比设置为零的 *FlexVol* 卷中一个大小为 100 GB 并且没有空洞的文件，而且为该文件启用了空间预留，但将为 *FlexClone* 文件禁用空间预留。

以下示例说明了如何克隆 `file1_source` 以创建名为 `file1_clone` 并禁用了空间预留的 *FlexClone* 文件：

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

在 Data ONTAP 8.3 及更高版本中，不需要额外的临时空间便能创建 *FlexClone* 文件。在先前版本中，需要至少 100 GB 的可用空间才能从预留百分比设置为零的 *FlexVol* 卷中

一个大小为 100 GB 并启用了空间预留的文件创建禁用了空间预留的 FlexClone 文件。现在，此操作不再需要额外空间。

#### 不支持经过空间优化的克隆的场景

以下场景不支持经过空间优化的克隆：

- 无法为 Snapshot 副本中的文件、子文件、LUN 和子 LUN 创建经过空间优化的克隆。  
卷中需要有额外空间才能创建克隆。
- 当现有 FlexClone 文件和 FlexClone LUN 未经过空间优化时，无法为克隆创建经过空间优化的克隆。  
卷中需要有额外空间才能为未经过空间优化的克隆创建克隆。
- 在某些情况下，经过空间优化的克隆会转换为未经过空间优化的克隆，此时卷中需要有额外空间。  
例如，克隆 Snapshot 副本中的子文件或子 LUN 以覆盖卷的活动文件系统中经过空间优化的克隆时，经过空间优化的克隆会转换为未经过空间优化的克隆，此时该克隆将需要额外空间。

#### 相关概念

[设置预留百分比的注意事项](#)（第 23 页）

## 分摊负载如何确定用于创建和删除 FlexClone 文件和 FlexClone LUN 的节点容量

可以查看每个节点的最大分摊负载、当前分摊负载、令牌预留分摊负载和允许的分摊负载，以确定节点何时具有容量可用于创建和删除 FlexClone 文件和 FlexClone LUN。允许的分摊负载可确定节点何时具有容量可用于创建和删除克隆。

下表列出了每个节点具有的分摊负载信息：

分摊负载类型	说明
最大分摊负载	节点上可用于创建和删除 FlexClone 文件和 FlexClone LUN 的最大容量 当达到最大分摊负载时，系统会发出 <code>EBUSY</code> 错误消息并拒绝新请求，直至负载降至最大分摊负载以下为止。
当前分摊负载	节点上当前用于创建和删除 FlexClone 文件和 FlexClone LUN 的容量

分摊负载类型	说明
令牌预留分摊负载	节点上由客户端使用令牌预留用于创建和删除 FlexClone 文件和 FlexClone LUN 的容量 可以将客户端配置为使用 NetApp 易管理性 SDK 来检索在节点上预留了空间用于创建 FlexClone 文件和 FlexClone LUN 的令牌。
允许的分摊负载	节点上可用于接收新的 FlexClone 文件和 FlexClone LUN 创建请求和删除请求的剩余容量。（最大分摊负载减去当前分摊负载再减去令牌预留分摊负载即等于允许的分摊负载。）

#### 相关任务

[查看用于创建和删除 FlexClone 文件和 FlexClone LUN 的节点容量](#)（第 69 页）

## 创建 FlexClone 文件或 FlexClone LUN

您可以使用 `volume file clone create` 命令，以较少的空间和时间在 FlexVol 卷或 FlexClone 卷中存在的文件和 LUN 创建克隆。

#### 开始之前

- 集群上必须安装 FlexClone 许可证。
- 如果将多个块范围用于子 LUN 克隆或子文件克隆，则块编号不能重叠。

#### 关于本任务

根据集群管理员分配的权限，SVM 管理员可以创建 FlexClone 文件和 FlexClone LUN。

创建和修改克隆时，可为 FlexClone 文件和 FlexClone LUN 指定自动删除设置。默认情况下，自动删除设置已禁用。

创建克隆时，可以通过使用带有 `-overwrite-destination` 参数的 `volume file clone create` 命令，以覆盖现有 FlexClone 文件或 FlexClone LUN。

当节点达到其最大分摊负载时，该节点会暂时停止接受 FlexClone 文件和 FlexClone LUN 创建请求并发出 `EBUSY` 错误消息。当节点的分摊负载降至最大分摊负载以下时，该节点会再次接受 FlexClone 文件和 FlexClone LUN 创建请求。再次尝试创建请求之前，应等待节点具有创建克隆所需的容量。

#### 步骤

1. 使用 `volume file clone create` 命令创建 FlexClone 文件或 FlexClone LUN。

#### 示例

以下示例说明了如何为卷 `vol1` 中的父文件 `file1_source` 创建 FlexClone 文件 `file1_clone`：

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

有关使用此命令的详细信息，请参见手册页。

#### 相关概念

[分摊负载如何确定用于创建和删除 \*FlexClone\* 文件和 \*FlexClone LUN\* 的节点容量](#)（第 67 页）

#### 相关任务

[查看用于创建和删除 \*FlexClone\* 文件和 \*FlexClone LUN\* 的节点容量](#)（第 69 页）

#### 相关信息

《适用于 *SVM* 管理员的集群模式 *Data ONTAP 8.3* 系统管理指南》  
《集群模式 *Data ONTAP 8.3* 命令：手册页参考》

## 查看用于创建和删除 **FlexClone** 文件和 **FlexClone LUN** 的节点容量

您可以通过查看节点分摊负载的方式，查看节点是否有容量可用于接收新的 *FlexClone* 文件和 *FlexClone LUN* 创建请求和删除请求。如果已达到最大分摊负载，则在分摊负载降至最大分摊负载以下之前，不会接受任何新请求。

#### 关于本任务

当节点达到最大分摊负载时，会发出 `EBUSY` 错误消息以响应创建请求和删除请求。当节点的分摊负载降至最大分摊负载以下时，该节点会再次接受 *FlexClone* 文件和 *FlexClone LUN* 创建请求。

当“Allowable Split Load”（允许的分摊负载）字段显示容量并且可用容量足以处理创建请求时，节点可以接受新请求。

#### 步骤

1. 使用 `volume file clone split load show` 命令查看节点可用于创建和删除 *FlexClone* 文件和 *FlexClone LUN* 的容量。

#### 示例

以下示例显示了 `cluster1` 中所有节点的分摊负载。该集群中的所有节点都有容量可用于创建和删除 *FlexClone* 文件和 *FlexClone LUN*，如“Allowable Split Load”（允许的分摊负载）字段所示：

```
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.
```

### 相关概念

[分摊负载如何确定用于创建和删除 FlexClone 文件和 FlexClone LUN 的节点容量](#)（第 67 页）

## 查看通过 FlexClone 文件和 FlexClone LUN 节省的空间

可以查看通过在包含 FlexClone 文件和 FlexClone LUN 的卷中共享数据块而节省的空间

### 步骤

1. 通过输入以下命令查看通过 FlexClone 文件和 FlexClone LUN 实现的空间节省：

```
df -s volname
```

*volname* 表示 FlexVol 卷的名称。

注：如果对已启用重复数据删除的 FlexVol 卷运行 `df -s` 命令，则可以同时查看通过重复数据删除以及 FlexClone 文件和 FlexClone LUN 节省的空间。

### 示例

以下示例显示了 FlexClone 卷 test1 上的空间节省：

```
systemA> df -s test1

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

## 用于删除 FlexClone 文件和 FlexClone LUN 的方法

可以使用多种方法来删除 FlexClone 文件和 FlexClone LUN。了解哪些可用的方法有助于规划克隆的管理方式。

可以使用以下方法来删除 FlexClone 文件和 FlexClone LUN：

- 可以将 FlexVol 卷配置为在该 FlexVol 卷中的可用空间降至特定阈值以下时自动删除启用了自动删除的克隆。
- 可以将客户端配置为通过使用 NetApp 易管理性 SDK 来删除克隆。

- 可以将客户端配置为通过使用 NAS 和 SAN 协议来删除克隆。  
默认启用的删除方法速度较慢，因为该方法不使用 NetApp 易管理性 SDK。但是，可以通过使用 `volume file clone deletion` 命令将系统配置为在您删除 FlexClone 文件时使用速度较快的删除方法。

#### 相关概念

[FlexVol 卷如何从 FlexClone 文件和 FlexClone LUN 回收可用空间](#)（第 71 页）

#### 相关参考

[用于配置 FlexClone 文件删除的命令](#)（第 75 页）

## FlexVol 卷如何从 FlexClone 文件和 FlexClone LUN 回收可用空间

您可将 FlexVol 卷的自动删除设置配置为在卷接近已满时自动删除启用了自动删除功能的 FlexClone 文件和 FlexClone LUN，以在该卷中回收指定数量的可用空间。

您可将卷配置为在卷中的可用空间降至特定阈值以下时自动开始删除 FlexClone 文件和 FlexClone LUN，并在卷中回收了指定数量的可用空间后自动停止删除克隆。虽然您无法指定自动开始删除克隆的阈值，但可以指定某个克隆是否符合删除条件，并且可以指定卷通过删除克隆而回收的指定数量可用空间。

如果同时满足以下要求，则卷会在该卷中的可用空间降至特定阈值以下时自动删除 FlexClone 文件和 FlexClone LUN：

- 为包含 FlexClone 文件和 FlexClone LUN 的卷启用了自动删除功能。  
您可以通过使用 `volume snapshot autodelete modify` 命令来为 FlexVol 卷启用自动删除功能。要使卷自动删除 FlexClone 文件和 FlexClone LUN，必须将 `-trigger` 参数设置为 `volume` 或 `snap_reserve`。
- 为 FlexClone 文件和 FlexClone LUN 启用了自动删除功能。  
您可以通过使用带有 `-autodelete` 参数的 `file clone create` 命令为 FlexClone 文件或 FlexClone LUN 启用自动删除功能。因此，通过为克隆禁用自动删除并确保其他卷设置不会覆盖克隆设置，可以保留某些 FlexClone 文件和 FlexClone LUN。

您不但可以指定使用 Data ONTAP 8.2 及更高版本创建的 FlexClone LUN 的自动删除设置，而且可以指定使用 Data ONTAP 8.3 及更高版本创建的 FlexClone 文件的自动删除设置。升级到 Data ONTAP 8.3 之后，将为使用 8.3 之前的 Data ONTAP 版本创建的 FlexClone 文件禁用自动删除。您可以通过使用 `volume file clone autodelete` 命令来为这些 FlexClone 文件启用自动删除。

#### 相关任务

[将 FlexVol 卷配置为自动删除 FlexClone 文件和 FlexClone LUN](#)（第 72 页）

## 将 FlexVol 卷配置为自动删除 FlexClone 文件和 FlexClone LUN

您可以允许 FlexVol 卷在卷中的可用空间减少至特定阈值之下时自动删除启用了自动删除的 FlexClone 文件和 FlexClone LUN。

### 开始之前

- FlexVol 卷必须包含 FlexClone 文件和 FlexClone LUN，而且必须处于联机状态。
- FlexVol 卷不得为只读卷。

### 步骤

1. 使用 `volume snapshot autodelete modify` 命令在 FlexVol 卷中启用 FlexClone 文件和 FlexClone LUN 的自动删除。
  - 对于 `-trigger` 参数，您可以指定 `volume` 或 `snap_reserve`。
  - 对于 `-destroy-list` 参数，您必须始终指定 `lun_clone,file_clone`，而不管您要删除的是否只是一种类型的克隆。

### 示例

下面的示例向您展示如何让 `vol1` 卷触发 FlexClone 文件和 FlexClone LUN 的自动删除以回收空间，直到 25% 的卷空间为可用空间：

```
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
```

注：为 FlexVol 卷启用自动删除时，如果您将 `-commitment` 参数的值设置为 `destroy`，则所有 `-autodelete` 参数设置为 `true` 的 FlexClone 文件和 FlexClone LUN 都会在卷中的可用空间减少至指定的阈值之下时遭到删除。然而，`-autodelete` 参数设置为 `false` 的 FlexClone 文件和 FlexClone LUN 不会遭到删除。

2. 使用 `volume snapshot autodelete show` 命令验证 FlexVol 卷中是否已启用 FlexClone 文件和 FlexClone LUN 的自动删除。

### 示例

下面的示例显示 `vol1` 已启用 FlexClone 文件和 FlexClone LUN 的自动删除：

```
cluster1::> volume snapshot autodelete show -vserver vs1 -volume voll

Vserver Name: vs1
Volume Name: voll
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. 通过执行以下步骤确保为卷中您要删除的 FlexClone 文件和 FlexClone LUN 启用了自动删除:

- a. 使用 `volume file clone autodelete` 命令启用特定 FlexClone 文件或 FlexClone LUN 的自动删除。

您可以通过使用带有 `-force` 参数的 `volume file clone autodelete` 命令强制自动删除特定的 FlexClone 文件或 FlexClone LUN。

示例

下面的示例显示已启用 `voll` 卷包含的 FlexClone LUN `lun1_clone` 的自动删除:

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

您可以在创建 FlexClone 文件和 FlexClone LUN 时启用自动删除。

- b. 使用 `volume file clone show-autodelete` 命令验证 FlexClone 文件或 FlexClone LUN 是否启用了自动删除。

示例

下面的示例显示 FlexClone LUN `lun1_clone` 启用了自动删除:

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

Vserver Name: vs1
Path: vol/voll/lun1_clone
Autodelete Enabled: true
Clone
```

有关使用这些命令的详细信息, 请参见手册页。

## 相关任务

[自动删除 Snapshot 副本](#) (第 26 页)

## 防止自动删除特定 FlexClone 文件或 FlexClone LUN

如果将 FlexVol 卷配置为自动删除 FlexClone 文件和 FlexClone LUN，则可能会删除任何符合所指定标准的克隆。如果要保留特定 FlexClone 文件或 FlexClone LUN，可以将其从 FlexClone 自动删除流程中排除。

### 开始之前

必须安装 FlexClone 许可证。

### 关于本任务

自 Data ONTAP 8.3 起，当创建 FlexClone 文件或 FlexClone LUN 时，默认为克隆禁用自动删除设置。将 FlexVol 卷配置为自动删除克隆以回收该卷上的空间时，会保留禁用了自动删除的 FlexClone 文件和 FlexClone LUN。

注意：如果将卷上的 `commitment` 级别设置为 `try` 或 `disrupt`，可以通过为特定 FlexClone 文件或 FlexClone LUN 禁用自动删除来单独保留这些克隆。但是，如果将卷上的 `commitment` 设置为 `destroy`，并且销毁列表包含 `lun_clone`, `file_clone`，则卷设置会覆盖克隆设置，而且无论为克隆启用还是禁用了自动删除设置，都会删除所有 FlexClone 文件和 FlexClone LUN。

### 步骤

1. 使用 `volume file clone autodelete` 命令防止自动删除特定 FlexClone 文件或 FlexClone LUN。

#### 示例

以下示例显示了如何为包含于 `vol1` 中的 FlexClone LUN `lun1_clone` 禁用自动删除：

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

无法自动删除禁用了自动删除的 FlexClone 文件或 FlexClone LUN 以回收卷上的空间。

2. 使用 `volume file clone show-autodelete` 命令验证是否为该 FlexClone 文件或 FlexClone LUN 禁用了自动删除。

#### 示例

以下示例显示了 FlexClone LUN `lun1_clone` 的自动删除设置为 `false`：

```

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

```

## 用于配置 FlexClone 文件删除的命令

当客户端在不使用 NetApp 易管理性 SDK 的情况下删除 FlexClone 文件时，可以使用 `volume file clone deletion` 命令来以速度较快的方法删除 FlexVol 卷中的 FlexClone 文件。启用速度较快的删除方法时，必须指定 FlexClone 文件的扩展名和最小大小。

可以使用 `volume file clone deletion` 命令为卷中的 FlexClone 文件指定支持的扩展名列表和最小大小要求。速度较快的删除方法仅适用于满足要求的 FlexClone 文件。对于不满足要求的 FlexClone 文件，将使用速度较慢的删除方法。

当客户端通过使用 NetApp 易管理性 SDK 来删除卷中的 FlexClone 文件和 FlexClone LUN 时，扩展名要求和大小要求均不适用，因为这种情况下始终使用速度较快的删除方法。

操作	命令
向卷的支持的扩展名列表添加扩展名	<code>volume file clone deletion add-extension</code>
更改可以使用速度较快的方法从卷中删除的 FlexClone 文件的最小大小	<code>volume file clone deletion modify</code>
从卷的支持的扩展名列表中删除扩展名	<code>volume file clone deletion remove-extension</code>
查看可以使用速度较快的方法从卷中删除的 FlexClone 文件的支持的扩展名列表和最小大小	<code>volume file clone deletion show</code>

有关这些命令的详细信息，请参见相应的手册页。

## 支持与 FlexClone 文件和 FlexClone LUN 配合使用的功能

FlexClone 文件和 FlexClone LUN 可与各种 Data ONTAP 功能配合使用，例如重复数据删除、Snapshot 副本、配额和卷 SnapMirror。

以下功能支持与 FlexClone 文件和 FlexClone LUN 配合使用：

- 重复数据删除
- Snapshot 副本

- 访问控制列表
- 配额
- FlexClone 卷
- NDMP
- 卷 SnapMirror
- volume move 命令
- 空间预留
- HA 配置

### 重复数据删除如何与 **FlexClone** 文件和 **FlexClone LUN** 配合使用

通过为启用了重复数据删除的卷中的父文件和父 LUN 创建 FlexClone 文件或 FlexClone LUN，您将能高效利用数据块的物理存储空间。

重复数据删除与 FlexClone 文件和 LUN 使用相同的块共享机制。通过在 FlexVol 卷上启用重复数据删除，然后再克隆启用了重复数据删除的卷，您将能在该卷中最大限度地节省空间。

注：对启用了重复数据删除的卷执行 `sis undo` 命令时，您将无法为驻留在该卷中的父文件和父 LUN 创建 FlexClone 文件和 FlexClone LUN。

### Snapshot 副本如何与 **FlexClone** 文件和 **FlexClone LUN** 配合使用

可以从包含于 FlexVol 卷中的父文件和父 LUN 的现有 Snapshot 副本创建 FlexClone 文件和 FlexClone LUN。

但在父实体与克隆实体之间的块共享过程完成之前，无法手动删除用来创建 FlexClone 文件或 FlexClone LUN 的 Snapshot 副本。Snapshot 副本会保持锁定直至块共享过程完成，而块共享过程在后台进行。因此，当尝试删除锁定的 Snapshot 副本时，系统会显示一条消息，要求您稍后重试该操作。在这种情况下，如果要删除特定 Snapshot 副本，则必须不断重试该删除操作，以便在块共享完成后删除该 Snapshot 副本。

相关信息

《[集群模式 Data ONTAP 8.3 数据保护指南](#)》

### 访问控制列表如何与 **FlexClone** 文件和 **FlexClone LUN** 配合使用

FlexClone 文件和 FlexClone LUN 会继承其父文件和父 LUN 的访问控制列表。

如果父文件包含 Windows NT 流，FlexClone 文件也会继承流信息。但是，包含超过六个流的父文件无法克隆。

## 配额如何与 FlexClone 文件和 FlexClone LUN 配合使用

配额限制是基于 FlexClone 文件或 FlexClone LUN 的总逻辑大小应用的。自 Data ONTAP 8.1 起，即使块共享导致超过配额，克隆操作也不会使块共享失败。

创建 FlexClone 文件或 FlexClone LUN 时，配额无法识别任何空间节省。例如，如果创建大小为 10 GB 的父文件的 FlexClone 文件，则可供使用的物理空间只有 10 GB，但配额利用率会记录为 20 GB（10 GB 用于父文件和 10 GB 用于 FlexClone 文件）。

当创建 FlexClone 文件或 FlexClone LUN 导致超过组或用户的配额，假如 FlexVol 卷有足够空间可用于保留克隆的元数据，则克隆操作将成功。但是，该用户或组的配额会变为超额预订。

## FlexClone 卷如何与 FlexClone 文件和 FlexClone LUN 配合使用

对于包含 FlexClone 文件和 FlexClone LUN 及其父文件或父 LUN 的 FlexVol 卷，您可以为其创建 FlexClone 卷。

FlexClone 卷中所含的 FlexClone 文件或 FlexClone LUN 及其父文件或父 LUN 将继续共享块，而且共享方式与它们在父 FlexVol 卷中的共享方式相同。事实上，所有 FlexClone 实体及其父实体都共享相同的底层物理数据块，从而最大限度减少物理磁盘空间使用量。

如果 FlexClone 卷是从父卷中拆分而来的，则在 FlexClone 卷的克隆中，FlexClone 文件或 FlexClone LUN 及其父文件或父 LUN 将停止共享块。随后，它们会作为单独的文件或 LUN 存在。这意味着卷的克隆使用的空间会超过拆分前的空间使用量。

## NDMP 如何处理 FlexClone 文件和 FlexClone LUN

NDMP 会在逻辑级别处理 FlexClone 文件和 FlexClone LUN。所有 FlexClone 文件或 LUN 将作为单独的文件或 LUN 进行备份。

使用 NDMP 服务备份 qtree 或包含 FlexClone 文件或 FlexClone LUN 的 FlexVol 卷时，父实体与克隆实体之间的块共享不会保留下来，并且克隆实体会作为单独的文件或 LUN 备份到磁带。此时，节省的空间将不复存在。因此，要将数据备份到的磁带应具有足够空间，可以存储多出的数据。恢复时，所有 FlexClone 文件和 FlexClone LUN 将作为单独的物理文件和 LUN 进行恢复。而且，您可以在卷上启用重复数据删除功能，恢复利用块共享带来的种种优势。

注：从 FlexVol 卷的现有 Snapshot 副本创建 FlexClone 文件和 FlexClone LUN 时，该卷将无法备份到磁带，直至后台运行的块共享过程完成。如果在块共享过程正在运行时在卷上使用 NDMP，系统会显示一条消息，要求您在一段时间后重试该操作。在这种情况下，您必须不断重试磁带备份操作，以便在块共享完成后成功执行该操作。

有关磁带备份的详细信息，请参见《集群模式 Data ONTAP 数据保护磁带备份和恢复指南》

## 卷 SnapMirror 如何与 FlexClone 文件和 FlexClone LUN 配合使用

将 SnapMirror 卷与 FlexClone 文件和 FlexClone LUN 结合使用有助于保留空间节省，因为克隆实体只会复制一次。

如果 FlexVol 卷是 SnapMirror 源卷并包含 FlexClone 文件或 FlexClone LUN，则 SnapMirror 卷只会将共享的物理块和少量元数据传输到 SnapMirror 目标卷。目标卷只会存储一个物理块副本，并且该块会在父实体与克隆实体之间共享。因此，目标卷是源卷的精确副本，并且目标卷上的所有克隆文件或 LUN 共享相同的物理块。

有关 SnapMirror 卷的详细信息，请参见《集群模式 Data ONTAP 数据保护指南》。

## 卷移动对 FlexClone 文件和 FlexClone LUN 具有哪些影响

在卷移动操作的转换阶段，不能创建 FlexVol 卷的 FlexClone 文件或 FlexClone LUN。

## 空间预留如何与 FlexClone 文件和 FlexClone LUN 配合使用

默认情况下，FlexClone 文件和 FlexClone LUN 会从父文件和父 LUN 继承空间预留属性。但是，如果 FlexVol 缺乏空间，可从启用了空间预留的父文件和父 LUN 创建禁用了空间预留的 FlexClone 文件和 FlexClone LUN。

如果 FlexVol 卷没有足够空间可用于创建具有与父文件或父 LUN 相同的空间预留属性的 FlexClone 文件或 FlexClone LUN，则克隆操作会失败。

## HA 配置如何与 FlexClone 文件和 FlexClone LUN 配合使用

HA 配置支持 FlexClone 文件和 FlexClone LUN 操作。

在 HA 对中，当接管或交还操作正在运行时，您不能在配对系统上创建 FlexClone 文件或 FlexClone LUN。接管或交还操作完成后，配对系统上所有待处理的块共享操作将会恢复执行。

## 使用 qtree 对 FlexVol 卷进行分区

qtree 可用于将 FlexVol 卷分区成较小的区块，从而可以逐个管理这些区块。可以使用 qtree 来管理配额、安全模式和 CIFS 机会锁。

Data ONTAP 会为每个卷创建名为 *qtree0* 的默认 qtree。如果未将数据放入某个 qtree 中，则数据会驻留在 *qtree0* 中。

qtree 名称不能包含超过 64 个字符。

不能在两个 qtree 之间移动目录。只能在两个 qtree 之间移动文件。

## 何时使用 qtree

可以利用 qtree 对数据进行分区，期间不会产生与 FlexVol 卷相关的开销。您可以创建 qtree 来组织数据，或者管理下列一项或多项因素：配额、安全模式以及 CIFS 机会锁设置。

以下列表举例说明了 qtree 使用策略：

- **配额**  
可以限制特定项目所使用的数据大小，方法是将该项目的文件放入一个 qtree 并对该 qtree 应用树配额。
- **安全模式**  
如果因为项目的成员使用 Windows 文件和应用程序而需要为该项目使用 NTFS 安全模式，可以将该项目的数据分组到一个 qtree 中，然后将其安全模式设置为 NTFS，其他项目则无需使用相同的安全模式。
- **CIFS 机会锁设置**  
如果项目使用的数据库要求关闭 CIFS 机会锁，则可将该项目的 qtree 的 CIFS 机会锁设置为关闭，而使其他项目保留 CIFS 机会锁。

## qtree 与 FlexVol 卷相比有何不同之处

总体来说，qtree 与 FlexVol 卷十分相似。但是，这两项技术存在一些关键区别。了解这些区别有助于在设计存储架构时在这两项技术之间做出选择。

下表比较了 qtree 与 FlexVol 卷：

功能	qtree	FlexVol 卷
允许组织用户数据	是	是
允许将具有类似需求的用户分成一组	是	是
接受安全模式	是	是
接受机会锁配置	是	是
可以调整大小	是（使用配额限制）	是
支持 Snapshot 副本	否（可以从卷 Snapshot 副本提取 qtree 数据）	是
支持配额	是	是
可进行克隆	否（除非作为 FlexVol 卷的一部分）	是
可以用作 Storage Virtual Machine (SVM)	否	是

功能	qtree	FlexVol 卷
可以用作接合	否	是
可以使用 NFS 导出	是	是

## 获取 qtree 接合路径

可以通过获取 qtree 的接合路径或命名空间路径来挂载单个 qtree。CLI 命令 `qtree show -instance` 所显示的 qtree 路径采用 `/vol/<volume-name>/<qtree-name>` 格式。但是，该路径不是指 qtree 的接合路径或命名空间路径。

关于本任务

需要知道卷的接合路径才能获取 qtree 的接合路径或命名空间路径。

步骤

1. 使用 `vserver volume junction-path` 命令获取卷的接合路径。

示例

以下示例显示了位于名为 `vs0` 的 Storage Virtual Machine (SVM) 上名为 `vol1` 的卷的接合路径：

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

在以上输出中，卷的接合路径是 `/vol1`。由于 qtree 始终以卷作为根目录，因此 qtree 的接合路径或命名空间路径将是 `/vol1/qtree1`。

## qtree 名称限制

qtree 名称的长度不能超过 64 个字符。另外，在 qtree 名称中使用一些特殊字符（例如，逗号和空格）可能会导致其他 Data ONTAP 功能出现问题，应予以避免。

## 使用 qtree 可以对镜像执行哪些操作

可以查看存在于镜像中的 qtree，但不能修改这些 qtree。

例如，可以对镜像使用 `volume qtree statistics` 命令。请注意，所显示的有关 qtree 的信息（包括名称、安全模式、机会锁模式及其他属性）可能未在读写卷与镜像之间保持同步，具体取决于镜像的复制计划。但在读写卷复制到镜像之后，qtree 信息已同步。

然而，不能在镜像上创建、修改或删除 qtree。

## 将目录转换为 qtree

如果要转换为 qtree 的目录位于 FlexVol 卷的根目录下，则必须使用客户端应用程序将该目录包含的数据迁移到具有相同名称的新 qtree 中。

关于本任务

将目录转换为 qtree 的步骤取决于所使用的客户端。以下过程概括了所需完成的一般任务：

步骤

1. 重命名要转换为 qtree 的目录。
2. 创建具有原始目录名称的新 qtree。
3. 使用客户端应用程序将目录的内容移动到新 qtree 中。
4. 删除现已为空的目录。

注：如果目录与现有 CIFS 共享关联，则不能删除该目录。

## 使用 Windows 客户端将目录转换为 qtree

要使用 Windows 客户端将目录转换为 qtree，请重命名该目录，然后在存储系统上创建一个 qtree，最后将目录的内容移动到该 qtree。

关于本任务

必须使用 Windows 资源管理器完成此过程，不能使用 Windows 命令行界面或 DOS 提示符环境。

步骤

1. 打开 Windows 资源管理器。
2. 单击要更改的目录的文件夹表示。  
注：该目录必须驻留在其所属的卷的根目录下。
3. 从文件菜单中选择重命名，以使用其他名称命名该目录。
4. 在存储系统上，使用 `volume qtree create` 命令创建具有该目录的原始名称的新 qtree。
5. 在 Windows 资源管理器中打开已重命名的目录文件夹，然后选择此文件夹内的文件。
6. 将这些文件拖到新 qtree 的文件夹表示中。

注：要移动的文件夹包含越多子文件夹，移动操作所用的时间就越长。

7. 从文件菜单中选择删除，以删除现已为空并已命名的目录文件夹。

### 使用 UNIX 客户端将目录转换为 qtree

要在 UNIX 中将目录转换为 qtree，请重命名该目录，然后在存储系统上创建一个 qtree，最后将目录的内容移动到该 qtree。

#### 步骤

1. 打开 UNIX 客户端窗口。
2. 使用 `mv` 命令重命名目录。

#### 示例

```
client: mv /n/user1/voll/dir1 /n/user1/voll/olddir
```

3. 从存储系统中，使用 `volume qtree create` 命令创建具有原始名称的 qtree。

#### 示例

```
system1: volume qtree create /n/user1/voll/dir1
```

4. 从客户端中，使用 `mv` 命令将旧目录的内容移动到该 qtree 中。  
注：要移动的目录包含越多子目录，移动操作所用的时间就越长。

#### 示例

```
client: mv /n/user1/voll/olddir/* /n/user1/voll/dir1
```

5. 使用 `rmdir` 命令删除现已为空的旧目录。

#### 示例

```
client: rmdir /n/user1/voll/olddir
```

#### 完成之后

文件所有权和权限可能不会保留下来，具体取决于 UNIX 客户端实施 `mv` 命令的方式。如果发生这种情况，请将文件所有者和权限更新到先前的值。

## 用于管理 qtree 的命令

有特定的 Data ONTAP 命令可用于管理和配置 qtree。

许多 qtree 命令不能在卷移动操作正在进行时执行。如果由于此原因而无法完成 qtree 命令，请等待直至卷移动完成之后再重试该命令。

操作	命令
创建 qtree	<code>volume qtree create</code>
显示经过筛选的 qtree 列表	<code>volume qtree show</code>
删除 qtree	<code>volume qtree delete</code>
修改 qtree 的 UNIX 权限	<code>volume qtree modify -unix-permissions</code>
修改 qtree 的 CIFS 机会锁设置	<code>volume qtree oplocks</code>
修改 qtree 的安全设置	<code>volume qtree security</code>
重命名 qtree	<code>volume qtree rename</code>
显示 qtree 的统计信息	<code>volume qtree statistics</code>
重置 qtree 的统计信息	<code>volume qtree statistics -reset</code>

## 使用配额来限制或跟踪资源使用量

配额提供了一种限制或跟踪用户、组或 qtree 所使用的磁盘空间和文件数量的方式。配额应用到特定的 FlexVol 卷或 qtree。

### 为什么使用配额

可以使用配额限制 FlexVol 卷中资源的使用量、在资源使用量达到特定级别时发出通知或跟踪资源使用情况。

指定配额的原因如下：

- 限制用户或组可使用的磁盘空间量或文件数量，或者限制一个 qtree 可包含的磁盘空间量或文件数量
- 在不施加限制的情况下跟踪用户、组或 qtree 使用的磁盘空间量或文件数量
- 在用户的磁盘或文件使用率较高时向用户发出警告

相关概念

[配额配置示例](#) (第 111 页)

相关任务

[在包含 FlexVol 卷的 SVM 上设置配额](#) (第 117 页)

## 配额流程概述

配额分为软配额和硬配额两种类别。软配额可促使 Data ONTAP 在超过指定阈值时发送通知，而硬配额可在超过指定阈值时使写入操作失败。

当 Data ONTAP 收到对某个 FlexVol 卷的写入请求时，它会检查是否已激活该卷的配额。如果已激活，则 Data ONTAP 将通过执行写入操作来确定是否会超过该卷的任何配额（如果是对某个 qtree 执行写入，则还将确定是否会超过该 qtree 的任何配额）。如果超过任何硬配额，则写入操作将失败，并且会发送配额通知。如果超过任何软配额，则写入操作将成功，并且会发送配额通知。

相关概念

[如何应用配额](#) (第 92 页)

## 硬配额、软配额及阈值配额之间的区别

硬配额禁止操作，软配额则触发通知。

硬配额会对系统资源实施硬限制；任何导致资源使用量超出限制的操作都将失败。以下设置将创建硬配额：

- [Disk Limit 参数](#)
- [Files Limit 参数](#)

软配额会在资源使用量达到特定级别时发送警告消息，但不会影响数据访问操作，以便您能够在超过配额之前采取适当操作。以下设置将创建软配额：

- [Threshold for Disk Limit 参数](#)
- [Soft Disk Limit 参数](#)
- [Soft Files Limit 参数](#)

阈值配额和磁盘软配额可使管理员能够收到多条有关某个配额的通知。管理员通常可将 [Threshold for Disk Limit](#) 设置为稍小于 [Disk Limit](#) 的值，以便阈值配额在写入即将失败之前发送“最终警告”。

## 了解配额通知

配额通知是发送到事件管理系统 (EMS) 的消息，并配置为 SNMP 陷阱。

通知是为响应以下事件而发送的：

- 达到硬配额；换句话说，尝试超过该配额
- 超过软配额
- 不再超过软配额

阈值与其他软配额稍有不同。阈值只在被超过时触发通知，不再被超过时不会触发通知。

硬配额通知可通过使用 `volume quota modify` 命令进行配置。可以完全禁用硬配额通知，也可以更改其频率（例如，为了防止发送冗余消息）。

软配额通知不可配置，因为它们不太可能生成冗余消息，而且通知是其唯一用途。

下表列出了配额发送到 EMS 系统的事件：

条件	发送到 EMS 的事件
达到树配额中的硬限制	<code>waf1.quota.qtree.exceeded</code>
达到卷上用户配额中的硬限制	<code>waf1.quota.user.exceeded</code> （对于 UNIX 用户） <code>waf1.quota.user.exceeded.win</code> （对于 Windows 用户）
达到 qtree 上用户配额中的硬限制	<code>waf1.quota.userQtree.exceeded</code> （对于 UNIX 用户） <code>waf1.quota.userQtree.exceeded.win</code> （对于 Windows 用户）
达到卷上组配额中的硬限制	<code>waf1.quota.group.exceeded</code>
达到 qtree 上组配额中的硬限制	<code>waf1.quota.groupQtree.exceeded</code>
超过软限制（包括阈值）	<code>quota.softlimit.exceeded</code>
不再超过软限制	<code>quota.softlimit.normal</code>

下表列出了配额生成的 SNMP 陷阱：

条件	发送的 SNMP 陷阱
达到硬限制	<code>quotaExceeded</code>
超过软限制（包括阈值）	<code>quotaExceeded</code> 和 <code>softQuotaExceeded</code>
不再超过软限制	<code>quotaNormal</code> 和 <code>softQuotaNormal</code>

有关查看和管理事件与 SNMP 陷阱的详细信息，请参见《适用于集群管理员的集群模式 *Data ONTAP* 系统管理指南》。

注：通知包含的是 qtree ID 号，而不是 qtree 名称。可以使用 `volume qtree show -id` 命令来将 qtree 名称与 ID 号相关联。

## 什么是配额规则、配额策略和配额

配额在特定于 FlexVol 卷的配额规则中进行定义。这些配额规则在 Storage Virtual Machine (SVM) 的配额策略中集合到一起，然后在该 SVM 的每个卷上激活。

一个配额规则始终特定于一个卷。在配额规则中定义的卷上激活之前，配额规则不具有任何效果。

配额策略是 SVM 中所有卷的配额规则的集合。配额策略不在多个 SVM 之间共享。SVM 最多可以具有五个配额策略，这样可为您提供配额策略的备份副本。一个配额策略在任何特定时间都已分配给 SVM。

配额是 Data ONTAP 执行的实际限制，或者是 Data ONTAP 执行的实际跟踪。配额规则始终会生成至少一个配额，并且可能会生成许多附加的派生配额。强制实施的配额的完整列表只显示在配额报告中。

激活指的是促使 Data ONTAP 执行以下操作的过程：从已分配的配额策略中当前的配额规则集创建强制实施的配额。激活逐个卷进行。首次对卷上的配额执行的激活称为初始化。后续的激活称为重新初始化或调整大小，具体取决于更改范围。

注：重新初始化卷上的配额或调整其大小时，将激活当前已分配给 SVM 的配额策略中的配额规则。

## 配额目标和类型

配额包括以下类型：用户配额、组配额或树配额。配额目标指定了对其应用配额限制的用户、组或 qtree。

下表列出了配额目标的类型、与每个配额目标关联的配额类型以及每个配额目标的表示方式：

配额目标	配额类型	目标的表示方式	说明
用户	用户配额	UNIX 用户名 UNIX UID UID 与用户匹配的文件或目录 采用 Windows 2000 之前格式的 Windows 用户名 Windows SID 带有由用户 SID 拥有的 ACL 的文件或目录	可以对特定卷或 qtree 应用用户配额。
组	组配额	UNIX 组名称 UNIX GID GID 与组匹配的文件或目录	可以对特定卷或 qtree 应用组配额。  注：Data ONTAP 不会根据 Windows ID 应用组配额。
qtree	树配额	qtree 名称	树配额应用到某个特定卷，并且不会影响其他卷中的 qtree。
*	用户配额 组配额 树配额	星号字符 (*)	包含 * 的配额目标表示默认配额。对于默认配额，配额类型由类型字段的值确定。

#### 相关概念

[如何对用户和组使用配额](#)（第 93 页）

[如何对 qtree 使用配额](#)（第 97 页）

## 特殊的配额类型

通过使用默认配额、显式配额、派生配额和跟踪配额，能够以最高效的方式管理磁盘使用量。

### 默认配额的工作原理

可以使用默认配额将配额应用于给定配额类型的所有实例。例如，默认用户配额会影响系统上指定 FlexVol 卷或 qtree 的所有用户。另外，还可以使用默认配额轻松地修改配额。

可以使用默认配额自动对大量配额目标应用限制，而无需为每个目标创建单独的配额。例如，如果要将大多数用户的可用磁盘空间限制为 10 GB，则可以指定一个 10 GB 磁盘空间的

默认用户配额，而不必为每个用户创建一个配额。如果要对特定用户应用不同的限制，则可以为其创建显式配额。（显式配额是具有特定目标或目标列表的配额，它会覆盖默认配额。）

另外，使用默认配额，可通过调整大小而不是重新初始化来使配额更改生效。例如，如果要为已具有默认用户配额的卷添加显式用户配额，则可以通过调整大小来激活新配额。

默认配额对所有配额目标类型（用户、组和 qtree 三种）均适用。

默认配额不一定具有指定的限制；默认配额可以是跟踪配额。

配额通过目标来指示，根据使用环境，该目标可能是空字符串 ("") 或星号 (\*)：

- 使用 `volume quota policy rule create` 命令创建配额时，将 `-target` 参数设置为空字符串 ("") 可创建默认配额。
- 在 `volume quota policy rule show` 命令的输出中，默认配额显示时带有作为目标的空字符串 ("")。
- 在 `volume quota report` 命令的输出中，默认配额显示时带有作为 ID 和配额说明符的星号 (\*)。

#### 默认用户配额示例

以下命令将创建一个为 `vol1` 中的每个用户应用 50 MB 限制的默认用户配额：

```
volume quota policy rule create -vserver vs1 -policy-name
quota_policy_vs1_1 -volume vol1 -type user -target ""-disk-limit 50MB
-qtrees ""
```

`volume quota policy rule show` 命令将显示以下输出：

Vserver: vs1			Policy: quota_policy_vs1_1			Volume: vol1		
Type	Target	Qtrees	User Mapping	Disk Limit	Soft Disk Limit	Files Limit	Soft Files Limit	Threshold
user	""	""	off	50MB	-	-	-	-

如果系统上的任何用户执行可能导致该用户的数据在 `vol1` 中所占用空间超过 50 MB 的操作（例如，从编辑器中对文件进行写入），则该命令将失败。

#### 相关概念

[派生配额的工作原理](#)（第 90 页）

## 如何使用显式配额

可以使用显式配额来为特定配额目标指定配额，或覆盖特定目标的默认配额。

显式配额可为特定用户、组或 qtree 指定一个限制。显式配额会取代同一个目标具有的任何默认配额。

当为已具有派生用户配额的用户添加显式用户配额时，必须使用与默认用户配额相同的用户映射设置。否则，当调整配额大小时，该显式用户配额会因为被视为新配额而被拒绝。

显式配额只会影响同一级别（卷或 qtree）下的默认配额。例如，qtree 的显式用户配额不会影响包含该 qtree 的卷的默认用户配额。但是，qtree 的显式用户配额会覆盖该 qtree 的默认用户配额（也就是取代由默认用户配额定义的限制）。

### 显式配额的示例

以下命令会将用户 chen 在卷 vol1 上的可用空间限制为 80 MB：

```
volume quota policy rule create -vserver vs1 -policy-name
quota_policy_vs1_1 -volume vol1 -type user -target corp\chen -disk-
limit 80MB -qtree ""
```

以下命令会将组 eng1 在卷 vol2 的 qtree proj1 中的可用空间限制为 150 MB，并且不限制文件数：

```
volume quota policy rule create -vserver vs1 -policy-name
quota_policy_vs1_1 -volume vol2 -type group -target eng1 -disk-limit
150MB -qtree proj1
```

以下命令会将 qtree proj1 在卷 vol2 上的可用磁盘空间限制为 750 MB，并将文件数限制为 76,800：

```
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 ""
```

volume quota policy rule show 命令将显示以下输出：

```
Vserver: vs1          Policy: quota_policy_vs1_1  Volume:
                               vol1
      Type  Target  Qtree  User      Disk      Soft      Files      Soft
      ----  -
      user  corp\chen  ""      off       80MB      -         -         -
                               vol2
      User      Disk      Soft      Files      Soft
      User      Disk      Disk      Files      Files
```

Type	Target	Qtree	Mapping	Limit	Limit	Limit	Limit	Threshold
group	enjl	proj1	off	150MB	-	-	-	-
tree	proj1	"	off	750MB	-	76800	-	-

## 派生配额的工作原理

执行默认配额时衍生出的配额（而不是具有特定目标的显式配额）称为派生配额。

派生配额的数量和位置取决于配额类型：

- 卷上的默认树配额会为该卷上的每个 qtree 创建派生树配额。
- 默认用户配额或组配额会为在同一个级别（卷或 qtree）下拥有文件的每个用户或组创建派生用户配额或组配额。
- 卷上的默认用户配额或组配额会在同时具有树配额的每个 qtree 上创建默认用户配额或组配额。

派生配额的设置（包括限制和用户映射）与对应默认配额的设置相同。例如，卷上磁盘限制为 20 GB 的默认树配额将在该卷的 qtree 上创建磁盘限制为 20 GB 的派生树配额。如果默认配额是跟踪配额（无限制），则派生配额也是跟踪配额。

要查看派生配额，请生成配额报告。在该报告中，派生用户配额或组配额由配额说明符（为空或者是星号(\*)）指示。但是，派生树配额带有配额说明符；要识别派生树配额，必须查找卷上具有相同限制的默认树配额。由于派生配额不是手动配置的配额规则，因此它们不会显示在 `quota policy rule show` 命令的输出中。

显式配额按下列方式与派生配额交互：

- 当同一个目标已存在显式配额时，不会创建派生配额。
- 为某个目标创建显式配额时，如果派生配额已存在，则可以通过调整大小的方式来激活显式配额，而不必执行整个配额初始化过程。

## 相关概念

[默认配额的工作原理](#)（第 87 页）

[默认用户配额和组配额如何创建派生配额](#)（第 94 页）

[FlexVol 卷上的默认树配额如何创建派生树配额](#)（第 98 页）

[FlexVol 卷上的默认用户配额如何影响该卷中的 qtree 的配额](#)（第 99 页）

## 如何使用跟踪配额

跟踪配额可生成磁盘使用量和文件使用量的报告，并且不会限制资源使用量。当使用跟踪配额时，修改配额的值不会造成太多中断，因为可以调整配额大小而不必先禁用再启用配额。

要创建跟踪配额，请忽略 `Disk Limit` 参数和 `Files Limit` 参数。这样可指示 Data ONTAP 监控相关目标在相应级别（卷或 qtree）下的磁盘使用量和文件使用量，而不实施任何限制。在 `show` 命令的输出和配额报告中，跟踪配额的所有限制以连接号 (-) 指示。

也可以指定适用于所有目标实例的默认跟踪配额。默认跟踪配额可用于跟踪某种配额类型所有实例（例如所有 qtree 或所有用户）的使用量。此外，通过使用默认跟踪配额，可以通过调整大小而不是重新初始化来使配额更改生效。

#### 显式跟踪配额的示例

以下命令将在卷 voll 上对用户 chen 进行跟踪：

```
volume quota policy rule create -vserver vs1 -policy-name
quota_policy_vs1_1 -volume voll -type user -target corp\chen -qtree ""
```

以下命令将在卷 voll 上对组 eng1 进行跟踪：

```
volume quota policy rule create -vserver vs1 -policy-name
quota_policy_vs1_1 -volume voll -type group -target eng1 -qtree ""
```

以下命令将在卷 voll 上对名为 proj1 的 qtree 进行跟踪：

```
volume quota policy rule create -vserver vs1 -policy-name
quota_policy_vs1_1 -volume voll -type tree -target proj1 -qtree ""
```

volume quota policy rule show 命令将显示以下输出：

```
Vserver: vs1          Policy: quota_policy_vs1_1  Volume:
                                voll
```

Type	Target	Qtree	User Mapping	Disk Limit	Soft Disk Limit	Files Limit	Soft Files Limit	Threshold
user	corp\chen	""	off	-	-	-	-	-
group	eng1	""	off	-	-	-	-	-
tree	proj1	""	off	-	-	-	-	-

#### 默认跟踪配额的示例

以下命令将在卷 voll 上对所有用户进行跟踪：

```
volume quota policy rule create -vserver vs1 -policy-name
quota_policy_vs1_1 -volume voll -type user -target "" -qtree ""
```

以下命令将在卷 voll 上对所有组进行跟踪：

```
volume quota policy rule create -vserver vs1 -policy-name
quota_policy_vs1_1 -volume voll -type group -target "" -qtree ""
```

以下命令将在卷 voll 上对所有 qtree 进行跟踪：

```
volume quota policy rule create -vserver vs1 -policy-name
quota_policy_vs1_1 -volume voll -type tree -target "" -qtree ""
```

volume quota policy rule show 命令将显示以下输出：

Vserver: vs1		Policy: quota_policy_vs1_1			Volume: voll				
Type	Target	Qtree	User Mapping	Disk Limit	Soft Disk Limit	Files Limit	Soft Files Limit	Threshold	
user	" "	" "	off	-	-	-	-	-	
group	" "	" "	off	-	-	-	-	-	
tree	" "	" "	off	-	-	-	-	-	

## 如何应用配额

了解配额的应用方式有助于配置配额并设置预期的限制。

每当尝试在 FlexVol 卷中创建文件或向文件写入数据时，系统都会先检查配额限制，然后操作才会继续进行。如果操作超过磁盘限制或文件限制，则系统会阻止该操作。

系统按以下顺序检查配额限制：

1. 该 qtree 的树配额（当在 qtree0 中创建或写入文件时，此检查不相关。）
2. 拥有卷上的文件的用户的用户配额
3. 拥有卷上的文件的组的组配额
4. 拥有 qtree 上的文件的用户的用户配额（当在 qtree0 中创建或写入文件时，此检查不相关。）
5. 拥有 qtree 上的文件的组的组配额（当在 qtree0 中创建或写入文件时，此检查不相关。）

最先超过的可能不是具有最低限制值的配额。例如，如果卷 voll 的用户配额是 100 GB，而包含于卷 voll 中的 qtree q2 的用户配额是 20 GB，则当该用户已在卷 voll 中写入超过 80 GB 数据（但不是写入到 qtree q2 之中）时，最先超过的是卷限制。

## 分配配额策略的注意事项

配额策略是用于 Storage Virtual Machine (SVM) 的所有 FlexVol 卷的一组配额规则。分配配额策略时，必须了解一些注意事项。

- SVM 在任何特定时间都具有一个已分配的配额策略。创建 SVM 时，会创建一个空的配额策略并将其分配给 SVM。此默认配额策略以 default 命名，除非在创建 SVM 时指定了其他名称。
- SVM 最多可以具有五个配额策略。如果 SVM 具有五个配额策略，则在删除一个现有配额策略之前，无法为该 SVM 创建新的配额策略。
- 如果需要创建配额规则或更改配额策略的配额规则，可以选择以下方法之一：
  - 如果要处理已分配给 SVM 的配额策略，则不必将此配额策略分配给该 SVM。

- 如果要处理未分配的配额策略，然后将此配额策略分配给 SVM，则必须备份可供在需要时还原到的配额策略。

例如，您可以创建关联的配额策略的副本，接着更改此副本，然后将此副本分配给 SVM，最后重命名原始配额策略。

- 即使配额策略已分配给 SVM，也可以重命名该配额策略。

## 如何对用户和组使用配额

指定用户或组作为配额目标时，该配额所施加的限制将应用于该用户或组。但是，一些特殊的组和用户会得到不同处理。根据您的环境，可以使用不同方式为用户指定 ID。

### 相关概念

[如何对 \*qtree\* 使用用户和组配额](#) (第 98 页)

## 如何为配额指定 UNIX 用户

可以使用以下三种格式之一为配额指定 UNIX 用户：用户名、UID 或用户拥有的文件或目录。

可以使用下列格式之一为配额指定 UNIX 用户：

- 用户名（如 `jsmith`）。
  - 注：如果 UNIX 用户名包含反斜杠 (\) 或 @ 符号，则无法使用它来指定配额。这是因为 Data ONTAP 将包含这些字符的名称视为 Windows 名称。
- UID（如 `20`）。
- 用户所拥有文件或目录的路径（以便文件的 UID 与用户相匹配）。
  - 注：如果指定文件或目录名称，则只要用户帐户仍在系统上，您选择的文件或目录就必须一直存在。

为 UID 指定文件或目录名称不会导致 Data ONTAP 将配额应用于该文件或目录。

## 如何为配额指定 Windows 用户

可以使用以下三种格式之一为配额指定 Windows 用户：采用 Windows 2000 之前格式的 Windows 名称、SID 或用户 SID 所拥有的文件或目录。

可以使用下列格式之一为配额指定 Windows 用户：

- 采用 Windows 2000 之前格式的 Windows 名称。包括 NetBIOS 形式的域，例如 `corp\Bob`。如果名称包含空格，请将配额目标值放在引号内，例如 `"corp\John Smith"`。
- Windows 以文本格式显示的安全 ID (SID)，如 `S-1-5-32-544`。
- 此用户 SID 所拥有的 ACL 位于其中的文件或目录的名称。

注：如果指定文件或目录名称，则只要用户帐户仍在系统上，您选择的文件或目录就必须一直存在。

Data ONTAP 只能从有效的 ACL 中获取 SID。

如果文件或目录位于 UNIX 模式的 qtree 中，或者存储系统使用 UNIX 模式进行用户身份验证，Data ONTAP 则会将用户配额应用于 *UID*（而非 *SID*）与该文件或目录相匹配的用户。

通过指定文件或目录名称来标识配额用户不会导致 Data ONTAP 将配额应用于该文件或目录。

### 默认用户配额和组配额如何创建派生配额

创建默认用户配额或组配额时，系统会自动为在同一级别下拥有文件的每个用户或组创建相应的派生用户配额或组配额。

派生用户配额和组配额按下列方式创建：

- FlexVol 卷上的默认用户配额会为在该卷上任意位置拥有文件的每个用户创建派生用户配额。
- qtree 上的默认用户配额会为在该 qtree 中拥有文件的每个用户创建派生用户配额。
- FlexVol 卷上的默认组配额会为在该卷上任意位置拥有文件的每个组创建派生组配额。
- qtree 上的默认组配额会为在该 qtree 中拥有文件的每个组创建派生组配额。

如果用户或组在默认用户配额或组配额所在的级别不拥有文件，则不会为该用户或组创建派生配额。例如，如果为 qtree proj1 创建了默认用户配额，而用户 jsmith 拥有的是其他 qtree 上的文件，则不会为 jsmith 创建派生用户配额。

派生配额具有与默认配额相同的设置，包括限制和用户映射。例如，如果默认用户配额具有 50 MB 的磁盘限制并启用了用户映射，则生成的任何派生配额也具有 50 MB 的磁盘限制并启用用户映射。

但是，三个特殊用户和组的派生配额中不存在任何限制。如果以下用户和组在默认用户配额或组配额所在的级别拥有文件，则创建的派生配额会与默认用户配额或组配额具有相同的用户映射设置，但此派生配额只是一个跟踪配额（没有任何限制）：

- UNIX root 用户 (UID 0)
  - UNIX root 组 (GID 0)
  - Windows BUILTIN\Administrators 组
- 由于 Windows 组的配额作为用户配额进行跟踪，因此该组的派生配额是从默认用户配额派生的用户配额，而不是从默认组配额派生而来。

### 派生用户配额示例

如果三个用户（root、jsmith 和 bob）在一个卷上拥有文件，并且该卷上创建了默认用户配额，则 Data ONTAP 会自动创建三个派生用户配额。因此，重新初始化该卷上的配额之后，配额报告中会显示四个新配额：

```
cluster1::> volume quota report
Vserver: vs1
```

Volume	Tree	Type	ID	---Disk---		---Files---		Quota Specifier
				Used	Limit	Used	Limit	
voll		user	*	0B	50MB	0	-	*
voll		user	root	5B	-	1	-	
voll		user	jsmith	30B	50MB	10	-	*
voll		user	bob	40B	50MB	15	-	*

4 entries were displayed.

第一个新行表示创建的默认用户配额，它使用星号(\*)作为ID加以标识。其他新行表示派生用户配额。jsmith 和 bob 的派生配额具有与默认配额相同的 50 MB 磁盘限制。root 用户的派生配额是不具有任何限制的跟踪配额。

#### 相关概念

[派生配额的工作原理](#)（第 90 页）

[FlexVol 卷上的默认用户配额如何影响该卷中的 \*qtree\* 的配额](#)（第 99 页）

#### 如何对 root 用户应用配额

UNIX 客户端上的 root 用户 (UID=0) 受树配额约束，但不受用户配额或组配额约束。这使得 root 用户能够代表受配额影响无法执行操作的其他用户执行操作。

当 root 用户代表权限较低的用户执行文件或目录所有权更改或其他操作（例如 UNIX `chown` 命令）时，Data ONTAP 会检查新所有者的配额，但此时即便新所有者超出了硬配额限制，它也不会报告错误或停止操作。当管理操作（例如恢复丢失的数据）导致暂时超出配额这一情况时，这种做法非常有用。

注：但在执行所有权转移之后，如果用户在配额已超出的情况下尝试分配更多磁盘空间，则客户端系统将报告磁盘空间错误。

#### 配额如何与特殊 Windows 组配合使用

配额应用到 Everyone 组和 BUILTIN\Administrators 组的方式与应用到其他 Windows 组的方式不同。

以下列表介绍了当配额目标是特殊 Windows 组 ID 时发生的情况：

- 当配额目标是 Everyone 组时，如果文件的 ACL 显示所有者是 Everyone，则该文件会计到 Everyone 的 SID 之下。
- 当配额目标是 BUILTIN\Administrators 组时，该配额条目会被视为仅用于跟踪的用户配额。

不能对 BUILTIN\Administrators 组实施限制。

当 BUILTIN\Administrators 组的成员创建一个文件时，该文件由 BUILTIN\Administrators 组拥有并计到 BUILTIN\Administrators 组的 SID 之下，而不是计到该用户的个人 SID 之下。

注：Data ONTAP 不支持基于 Windows 组 ID 的组配额。如果将 Windows 组 ID 指定为配额目标，则该配额会被视为用户配额。

### 如何将配额应用到具有多个 ID 的用户

一个用户可由多个 ID 表示。可以将一个 ID 列表指定为配额目标来为此类用户设置单个用户配额。由该列表中任意 ID 拥有的文件须受用户配额限制。

假设一个用户具有 UNIX UID 20 以及 Windows ID corp\john\_smith 和 engineering\jsmith，则可为该用户指定以包含 UID 和 Windows ID 的列表作为配额目标的配额。当该用户向存储系统写入数据时，无论该写入操作来自 UID 20、corp\john\_smith 还是 engineering\jsmith，都会应用指定的配额。

注：即使多个 ID 属于同一个用户，单独的配额规则也会被视为单独的目标。

例如，即使以下两个 ID 代表同一个用户，也可为同一个用户指定如下两个配额：其中一个配额将 UID 20 的可用磁盘空间限制为 1 GB，另一个配额将 corp\john\_smith 的可用磁盘空间限制为 2 GB。Data ONTAP 会将前述配额分别应用到 UID 20 和 corp\john\_smith。

在这种情况下，即使向同一个用户所使用的其他 ID 应用了限制，也不会向 engineering\jsmith 应用任何限制。

### Data ONTAP 如何在混合环境中确定用户 ID

如果用户通过 Windows 客户端和 UNIX 客户端访问 Data ONTAP 存储，则会同时使用 Windows 安全模式和 UNIX 安全模式来确定文件所有权。多个因素决定了 Data ONTAP 在应用用户配额时使用 UNIX ID 还是 Windows ID。

如果包含相关文件的 qtree 或 FlexVol 卷的安全模式是“仅 NTFS”或“仅 UNIX”，则此安全模式决定了应用用户配额时所使用的 ID 类型。对于具有混合安全模式的 qtree，所使用的 ID 类型由相关文件是否具有 ACL 决定。

下表总结了所使用的 ID 类型：

安全模式	ACL	无 ACL
UNIX	UNIX ID	UNIX ID
混合	Windows ID	UNIX ID
NTFS	Windows ID	Windows ID

### 相关概念

[如何链接配额的 UNIX 名称和 Windows 名称](#)（第 97 页）

## 配额如何与多个用户配合使用

如果在同一个配额目标中指定多个用户，则由该配额定义的配额限制不会应用到每个单独的用户；在这种情况下，配额限制在该配额目标中列出的所有用户之间共享。

注：如果将多个单独的用户配额合并为一个多用户配额，则可以通过调整配额大小来激活更改。但是，如果要从包含多个用户的配额目标中删除用户，或将用户添加到已包含多个用户的目标中，则必须先重新初始化配额才能使更改生效。

### 配额目标包含多个用户的示例

在以下示例中，配额目标中列出了两个用户：

```
volume quota policy rule create -vserver vs0 -policy-name
quota_policy_0 -volume vol0 -type user -target corp\jsmith,corp\chen -
disk-limit 80MB
```

这两个用户合计最多可以使用 80 MB 空间。如果其中一个用户使用了 75 MB，则另一个用户只能使用 5 MB。

## 如何链接配额的 UNIX 名称和 Windows 名称

在混合环境中，用户能够以 Windows 用户身份或 UNIX 用户身份登录。可对配额进行配置，以识别用户的 UNIX ID 以及代表该用户的 Windows ID。

当同时满足以下两个条件时，Windows 用户名的配额和 UNIX 用户名的配额会映射到彼此：

- 在适用于用户的配额规则中 `user-mapping` 参数设置为 `on`。
- 已使用 `vserver name-mapping` 命令映射了用户名。

如果 UNIX 名称和 Windows 名称映射到一起，则当确定配额使用量时，这些名称会被视为同一个用户。

### 相关概念

[Data ONTAP 如何在混合环境中确定用户 ID](#)（第 96 页）

## 如何对 qtree 使用配额

可以创建以 qtree 作为目标的配额，此类配额称为树配额。也可为特定 qtree 创建用户配额和组配额。另外，FlexVol 卷包含的 qtree 有时会继承该卷的配额。

### 树配额的工作原理

可以创建以 qtree 作为目标的配额，以限制目标 qtree 的大小。此类配额也称为树配额。

将配额应用于 qtree 的效果与磁盘分区相似，不同之处是可以随时通过更改配额来更改 qtree 的大小上限。应用树配额时，Data ONTAP 会限制 qtree 中的磁盘空间和文件数，而不管其所

有者是谁。如果写入操作导致超出了树配额，则所有用户（包括 BUILTIN\Administrators 组的 root 用户和成员）都不能对 qtree 执行写入操作。

注：配额大小并非一定与特定数量的可用空间相对应。配额大小可能大于 qtree 的可用空间量。可以使用 `volume quota report` 命令确定 qtree 中的实际可用空间量。

### 如何对 qtree 使用用户和组配额

树配额限制 qtree 的总体大小。为防止具体用户或组占用整个 qtree，可以为此 qtree 指定用户或组配额。

#### qtree 中的用户配额示例

假设 vol2 上没有用户配额。您注意到，一个特定用户 `corp\kjones` 在关键 qtree (`qt1`，位于 `vol2` 中) 上占用的空间过多。可以使用以下命令限制该用户在这一关键 qtree 上占用的空间：

```
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
```

#### 相关概念

[如何对用户和组使用配额](#)（第 93 页）

### FlexVol 卷上的默认树配额如何创建派生树配额

在 FlexVol 卷上创建默认树配额时，系统会自动为该卷中的每个 qtree 创建相应的派生树配额。

这些派生树配额与默认树配额具有相同的限制。如果不存在其他配额，则这些限制具有以下作用：

- 用户在 qtree 中可以使用的空间最大可达到为他们在整个卷上分配的空间（前提是他们通过使用根目录或其他 qtree 中的空间来避免超过卷的限制）。
- 每个 qtree 的空间都可以增长，直至占用整个卷的空间。

卷上的默认树配额将继续影响添加到该卷中的所有新 qtree。每次创建新的 qtree 时，都会同时创建派生树配额。

与所有派生配额一样，派生树配额也会表现出以下行为：

- 仅在目标尚未具有显式配额时才会创建。
- 显示在配额报告中，但在通过 `volume quota policy rule show` 命令显示配额规则时不会显示。

### 派生树配额示例

假设一个卷包含三个 qtree (proj1、proj2 和 proj3)，而仅有的树配额是一个位于 proj1 qtree 上的显式配额，该配额将磁盘大小限制为 10 GB。如果在该卷上创建默认树配额并重新初始化该卷上的配额，则配额报告现在包含四个树配额：

Volume	Tree	Type	ID	---Disk---		---Files---		Quota Specifier
				Used	Limit	Used	Limit	
voll	proj1	tree	1	0B	10GB	1	-	proj1
voll		tree	*	0B	20GB	0	-	*
voll	proj2	tree	2	0B	20GB	1	-	proj2
voll	proj3	tree	3	0B	20GB	1	-	proj3
...								

第一行显示了 proj1 qtree 上最初的显式配额。此配额保持不变。

第二行显示了卷上新的默认树配额。星号 (\*) 配额说明符指示该配额是默认配额。此配额由您创建的配额规则生成。

最后两行显示了 proj2 qtree 和 proj3 qtree 的新的派生树配额。Data ONTAP 基于卷上的默认树配额自动创建了这些配额。与卷上的默认树配额相同，这些派生树配额也具有 20 GB 的磁盘限制。Data ONTAP 没有为 proj1 qtree 创建派生树配额，因为 proj1 qtree 已具有一个显式配额。

#### 相关概念

[派生配额的工作原理](#) (第 90 页)

### FlexVol 卷上的默认用户配额如何影响该卷中的 qtree 的配额

如果为 FlexVol 卷定义了默认用户配额，则系统会自动为该卷中所含的每个 qtree (但这些 qtree 上要存在显式树配额或派生树配额) 创建默认用户配额。

如果 qtree 上已存在默认用户配额，则在卷上创建默认用户配额时，qtree 上的默认用户配额不受影响。

在 qtree 上自动创建的默认用户配额与为卷创建的默认用户配额具有相同限制。

qtree 上的显式用户配额将覆盖自动创建的默认用户配额 (并替换默认用户配额应用的限制)，具体方式与它覆盖由管理员创建的 qtree 默认用户配额的方式一样。

#### 相关概念

[默认用户配额和组配额如何创建派生配额](#) (第 94 页)

## qtree 更改如何影响配额

删除、重命名或更改 qtree 的安全模式时，由 Data ONTAP 应用的配额可能会发生变化，具体视所应用的当前配额而定。

### 删除 qtree 如何影响树配额

删除 qtree 时，Data ONTAP 不再应用适用于该 qtree 的所有配额，无论是显式配额还是派生配额。

配额规则是否保留取决于您删除 qtree 的位置：

- 如果使用 Data ONTAP 删除 qtree，则该 qtree 的配额规则会自动删除，其中包括树配额规则以及为该 qtree 配置的任何用户和组配额规则。
- 如果使用 CIFS 或 NFS 客户端删除 qtree，则必须删除该 qtree 的所有配额规则，以避免在重新初始化配额时出错。如果创建了一个与已删除 qtree 同名的新 qtree，在重新初始化配额之前，现有配额规则不会应用于这一新 qtree。

### 重命名 qtree 如何影响配额

在使用 Data ONTAP 重命名 qtree 时，该 qtree 的配额规则会自动更新。如果使用 CIFS 或 NFS 客户端重命名 qtree，则必须更新该 qtree 的所有配额规则。

注：如果使用 CIFS 或 NFS 客户端重命名 qtree，但没有在重新初始化配额之前使用新名称更新该 qtree 的配额规则，则配额不会应用于该 qtree，并且该 qtree 的显式配额（包括树配额和 qtree 的用户配额或组配额）可能会转变成派生配额。

### 更改 qtree 安全模式如何影响用户配额

可以使用 NTFS 或混合安全模式在 qtree 上应用访问控制列表 (ACL)，但不能使用 UNIX 安全模式来实现。因此，更改 qtree 的安全模式可能会影响配额的计算方式。更改 qtree 的安全模式后，请务必重新初始化配额。

如果将 qtree 的安全模式从 NTFS 或混合更改为 UNIX，则将忽略该 qtree 中所有的文件 ACL，并将文件使用量计到 UNIX 用户 ID 名下。

如果将 qtree 的安全模式从 UNIX 更改为混合或 NTFS，则会显示先前隐藏的 ACL。另外，之前忽略的所有 ACL 将重新生效，NFS 用户信息则会被忽略掉。如果之前没有 ACL，则配额计算过程中将继续使用 NFS 信息。

注：更改 qtree 的安全模式后，为了确保正确计算 UNIX 和 Windows 用户的配额使用量，必须重新初始化该 qtree 所在卷的配额。

#### 示例

以下示例介绍了 qtree 安全模式的更改如何造成该特定 qtree 中文件使用量计到其他用户名下。

假设 qtree A 采用 NTFS 安全模式，ACL 向 Windows 用户 corp\joe 分配了一个 5 MB 的文件。qtree A 的 5 MB 磁盘空间用量将计到用户 corp\joe 名下。

现在将 qtree A 的安全模式从 NTFS 更改为 UNIX。重新初始化配额后，此文件不再计到 Windows 用户 corp\joe 名下，而是计到该文件 UID 所对应的 UNIX 用户名下。UID 可以是映射到 corp\joe 的 UNIX 用户或 root 用户。

## 如何激活配额

激活之前，新建的配额以及对配额所作的更改都不会生效。了解配额的激活方法有助于减少管理配额时造成的中断。

您可以在卷级别激活配额。

配额可以通过初始化（启用）或调整大小的方式来激活。先禁用配额再启用的过程称为重新初始化。

激活过程的时间长度及其对配额实施操作产生的影响取决于激活类型：

- 初始化过程包括以下两个部分，分别涉及 `quota on` 作业以及对卷上的整个文件系统执行的配额扫描操作。扫描操作在 `quota on` 作业成功完成之后开始。配额扫描操作可能会持续一段时间；卷包含的文件数越多，所用时间就越长。扫描结束之前，配额激活操作无法完成，而且配额也无法实施。
- 调整大小过程仅涉及 `quota resize` 作业。由于不涉及到配额扫描操作，因此调整大小所用的时间要少于配额初始化操作。在调整大小过程中，配额会一直实施。

默认情况下，`quota on` 和 `quota resize` 作业在后台运行，这允许您在同一时间使用其他命令。

注：如果配额更改操作是在当前未分配的配额策略中执行的，则必须先将该配额策略分配给卷，然后再对配额执行调整大小或重新初始化操作。

激活过程出现的错误和警告将发送到事件管理系统。如果将 `-foreground` 参数与 `volume quota on` 或 `volume quota resize` 命令结合使用，则命令在作业完成之前不会返回结果；通过脚本执行重新初始化操作时，这种做法非常有用。要在稍后显示错误和警告，可使用带有 `-instance` 参数的 `volume quota show` 命令。

暂停和重新启动后，配额激活过程仍然有效。配额激活过程不会影响存储系统数据的可用性。

### 相关概念

[何时可以使用调整大小操作](#)（第 102 页）

[需要完整配额重新初始化的情况](#)（第 103 页）

## 何时可以使用调整大小操作

由于配额调整大小比配额初始化速度更快，因此，应尽可能使用调整大小操作。然而，调整大小操作仅适用于某些类型的配额更改。

对配额规则进行以下类型的更改时，可以调整配额大小：

- 更改现有配额。  
例如，更改现有配额的限制。
- 为具有默认配额或默认跟踪配额的配额目标添加一个配额。
- 删除已指定默认配额或默认跟踪配额条目的配额。
- 将各个用户配额合并为一个多用户配额。

注意：在进行大量配额更改之后，应执行完整的重新初始化过程，以确保所有更改生效。

注：如果尝试调整大小，而调整大小操作并不能使所有配额更改生效，则 Data ONTAP 会发出警告。

您可以通过配额报告确定存储系统是否正在跟踪特定用户、组或 qtree 的磁盘使用量。如果配额报告显示有配额，则表明存储系统正在跟踪配额目标所拥有的磁盘空间和文件数。

## 可以通过调整大小生效的配额更改示例

调整大小操作可使部分配额规则更改生效。假设有以下配额：

```
cluster1::>volume quota policy rule show
Vserver: vs1 Policy: quota_policy_0 Volume:

```

Type	Target	Qtree	User Mapping	Disk Limit	Soft Disk Limit	Files Limit	vol2 Soft Files Limit	Threshold
user	" "	" "	-	50MB	-	15360	-	-
group	" "	" "	-	750MB	-	87040	-	-
tree	" "	" "	-	-	-	-	-	-
user	corp\jdoe	" "	-	100MB	-	76800	-	-
user	corp\kbuck	" "	-	100MB	-	76800	-	-

假设进行以下更改：

- 增加默认用户目标的文件数。
- 为新用户 boris 添加新的用户配额，该配额所需的磁盘限制比默认用户配额更大。
- 删除用户 kbuck 的显式配额条目；新用户现在只需要默认配额限制。

以上更改将生成以下配额：

```
cluster1::>volume quota policy rule show
Vserver: vs1 Policy: quota_policy_0 Volume:

```

Type	Target	Qtree	User Mapping	Disk Limit	Soft Disk Limit	Files Limit	vol2 Soft Files Limit	Threshold
user	" "	" "	-	50MB	-	15360	-	-
group	" "	" "	-	750MB	-	87040	-	-
tree	" "	" "	-	-	-	-	-	-
user	corp\jdoe	" "	-	100MB	-	76800	-	-
user	corp\kbuck	" "	-	100MB	-	76800	-	-
user	corp\boris	" "	-	100MB	-	76800	-	-

Type	Target	Qtree	User Mapping	Disk Limit	Disk Limit	Files Limit	Files Limit	Threshold
user	" "	" "	-	50MB	-	25600	-	-
group	" "	" "	-	750MB	-	87040	-	-
tree	" "	" "	-	-	-	-	-	-
user	corp\jdoe	" "	-	100MB	-	76800	-	-
user	corp\boris	" "	-	100MB	-	76800	-	-

调整大小操作会激活以上所有更改，不需要进行完整的配额重新初始化过程。

## 相关概念

[如何激活配额](#)（第 101 页）

## 需要完整配额重新初始化的情况

虽然调整配额大小速度较快，但如果您对配额进行特定更改或大量更改，则必须执行完整配额重新初始化。

在以下情况下必须执行完整配额重新初始化：

- 为之前没有配额的目标创建配额。
- 使用 `vserver name-mapping` 命令更改已启用 `user-mapping` 参数的配额规则目标用户的用户映射。
- 将 `qtree` 的安全模式从 UNIX 更改为混合或 NTFS。
- 将 `qtree` 的安全模式从混合或 NTFS 更改为 UNIX。
- 从包含多个用户的配额目标中删除用户，或将用户添加到已包含多个用户的目标中。
- 对配额进行大量更改。

### 需要初始化的配额更改示例

假设您有一个包含三个 `qtree` 的卷，其中只有三个树配额。您决定进行以下更改：

- 添加新 `qtree` 并为其新建树配额。
- 为该卷添加默认用户配额。

这两项更改都需要完整配额初始化。调整大小不会使配额生效。

## 相关概念

[如何激活配额](#)（第 101 页）

## 如何查看配额信息

可以通过配额报告查看如下详细信息：配额规则和策略的配置、已执行配额和已配置配额以及在配额调整大小和重新初始化期间发生的错误。

在下列情况下，查看配额信息很有帮助：

- 配置配额，例如配置配额和验证配置
- 对有关即将达到或已达到磁盘空间限制或文件限制的通知做出响应
- 对更多空间响应请求

## 如何通过配额报告查看已生效的配额

由于配额交互方式多种多样，因此，正在执行的配额数量比显式创建的配额数量要多。要查看正在执行的配额，可以查看配额报告。

以下示例显示了在 FlexVol 卷 vol1 及其包含的 qtree q1 中应用的各种配额类型的配额报告：

### 没有为 qtree 指定用户配额的示例

此示例使用一个 qtree q1，该 qtree 包含在卷 vol1 中。管理员创建了以下三个配额：

- 卷 vol1 上的默认树配额限制 (400 MB)
- 卷 vol1 上的默认用户配额限制 (100 MB)
- 卷 vol1 上针对用户 jsmith 的显式用户配额限制 (200 MB)

这些配额的配额报告摘录如下：

```
cluster1::> volume quota report
Vserver: vs1
```

Volume	Tree	Type	ID	---Disk---		---Files---		Quota Specifier
				Used	Limit	Used	Limit	
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	-	

此配额报告前三行显示了由管理员指定的三个配额。由于其中两个配额是默认配额，因此 Data ONTAP 会自动创建派生配额。

第四行显示了由卷 vol1 上每个 qtree（在此示例中，只有 q1）的默认树配额派生的三个配额。

第五行显示了由于卷上存在默认用户配额和 qtree 配额而为该 qtree 创建的默认用户配额。

第六行显示了由于该 qtree（第 5 行）存在默认用户配额并且用户 jsmith 拥有该 qtree 上的文件而在该 qtree 上为用户 jsmith 创建的派生用户配额。请注意，在 qtree q1 上为用户 jsmith 应用的限制并不由显式用户配额限制 (200 MB) 决定。这是因为显式用户配额限制适用于卷，因此不会影响该 qtree 的限制。相反，该 qtree 的派生用户配额限制由其默认用户配额 (100 MB) 决定。

最后两行显示了由卷和 qtree 上的默认用户配额派生的其他用户配额。由于 root 用户同时拥有卷和 qtree 上的文件，因此在该卷和 qtree 上为 root 用户创建了派生用户配额。由于 root 用户在配额方面需要特殊对待，因此其派生配额仅用于跟踪配额。

为 **qtree** 指定了用户配额的示例

此示例与上一个示例类似，只是管理员在 qtree 上添加了两个配额。

此示例仍然使用卷 vol1 和 qtree q1。管理员创建了以下配额：

- 卷 vol1 上的默认树配额限制 (400 MB)
- 卷 vol1 上的默认用户配额限制 (100 MB)
- 卷 vol1 上针对用户 jsmith 的显式用户配额限制 (200 MB)
- qtree q1 上的默认用户配额限制 (50 MB)
- qtree q1 上针对用户 jsmith 的显式用户配额限制 (75 MB)

这些配额的配额报告如下所示：

```
cluster1::> volume quota report
Vserver: vs1
```

Volume	Tree	Type	ID	---Disk---		---Files---		Quota Specifier
				Used	Limit	Used	Limit	
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	-	-

此配额报告前五行显示了由管理员创建的五個配额。由于部分配额是默认配额，因此 Data ONTAP 会自动创建派生配额。

第六行显示了由卷 vol1 中每个 qtree（在此示例中，只有 q1）的默认树配额派生的三个配额。

最后两行显示了由卷和 qtree 上的默认用户配额派生的用户配额。由于 root 用户同时拥有卷和 qtree 上的文件，因此在该卷和 qtree 上为 root 用户创建了派生用户配额。由于 root 用户在配额方面需要特殊对待，因此其派生配额仅用于跟踪配额。

由于以下原因，没有创建其他默认配额或派生配额：

- 虽然用户 jsmith 同时拥有卷和 qtree 上的文件，但也不会为该用户创建派生用户配额，因为该用户在这两个级别下都已具有显式配额。
- 没有为其他用户创建派生用户配额，因为其他用户都不拥有该卷或 qtree 上的文件。
- 该卷上的默认用户配额未在 qtree 上创建默认用户配额，因为该 qtree 已具有默认用户配额。

#### 相关概念

[强制实施的配额为什么与配置的配额不同](#)（第 106 页）

#### 强制实施的配额为什么与配置的配额不同

强制实施的配额与配置的配额不同的原因是，派生配额在未经配置的情况下实施，而配置的配额只在成功初始化之后实施。了解这些区别有助于将配额报告中显示的强制实施的配额与配置的配额相比较。

配额报告中显示的强制实施的配额与配置的配额规则不同的可能原因如下：

- 派生配额在未配置为配额规则的情况下实施；受默认配额影响，Data ONTAP 会自动创建派生配额。
- 配置了配额规则后可能未重新初始化卷上的配额。
- 初始化卷上的配额时可能发生了错误。

#### 通过配额报告确定哪些配额会限制对特定文件的写入

可以使用 `volume quota report` 命令并指定特定的文件路径以确定哪些配额限制会影响对文件的写入操作。这有助于了解哪个配额正在阻止写入操作。

#### 步骤

1. 使用带有 `-path` 参数的 `volume quota report` 命令。

#### 显示正在影响特定文件的配额的示例

以下示例显示了用于确定哪些配额正在影响对文件 `file1`（此文件驻留在 FlexVol 卷 `vol2` 的 qtree `q1` 中）的写入的命令和输出：

```

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     q1                 group  eng          1MB  700MB       6    70000  *
vol2     q1                 user   corp\jsmith  1MB  50MB        1     -    *
vol2     q1                 user   corp\jsmith  1MB  50MB        1     -

5 entries were displayed.

```

### 用于显示有关配额的信息的命令

可以使用命令来显示配额报告（其中包含强制实施的配额和资源使用量），也可以使用命令来显示有关配额状态和错误的信息或有关配额策略和配额规则的信息。

注：以下命令只能对 FlexVol 卷运行。

操作	命令
显示有关强制实施的配额的信息	volume quota report
查看配额目标的资源使用量（磁盘空间和文件数）	volume quota report
确定允许对文件进行写入时会影响哪些配额限制	volume quota report 命令与 -path 参数结合使用
显示配额状态，例如打开、关闭和正在初始化	volume quota show
查看有关配额消息日志记录的信息	volume quota show 命令与 -logmsg 参数结合使用
查看在配额初始化和调整大小期间发生的错误	volume quota show 命令与 -instance 参数结合使用
查看有关配额策略的信息	volume quota policy show
查看有关配额规则的信息	volume quota policy rule show
查看分配给 Storage Virtual Machine (SVM, 以前称为 Vserver) 的配额策略的名称	vserver show 命令与 -instance 参数结合使用

有关详细信息，请参见各个命令的手册页。

相关概念

[何时应使用 `volume quota policy rule show` 命令和 `volume quota report` 命令](#)（第 108 页）

何时应使用 `volume quota policy rule show` 命令和 `volume quota report` 命令

虽然这两个命令都能显示有关配额的信息，但 `volume quota policy rule show` 命令可快速显示已配置的配额规则，而 `volume quota report` 命令会耗费更多时间和资源并显示强制实施的配额和资源使用量。

`volume quota policy rule show` 命令适用于以下目的：

- 在激活配额规则之前检查其配置  
此命令可显示所有已配置的配额规则，而无论这些配额是否已初始化或调整大小。
- 快速查看配额规则而不影响系统资源  
由于不会显示磁盘使用量和文件使用量，所以此命令不会像配额报告那样占用大量资源。
- 显示未分配给 Storage Virtual Machine (SVM) 的配额策略中的配额规则

`volume quota report` 命令适用于以下目的：

- 查看强制实施的配额，包括派生配额
- 查看由每个强制实施的配额使用的磁盘空间和文件数，包括受派生配额影响的目标（对于默认配额，使用量显示为 0，因为该使用量是按照生成的派生配额进行跟踪的。）
- 确定哪些配额限制会影响何时允许对文件进行写入  
将 `-path` 参数添加到 `volume quota report` 命令中。

注：配额报告是资源密集型操作。如果对集群中的许多 FlexVol 卷运行该操作，可能需要很长时间才能完成。更高效的方法是查看 SVM 中某个特定卷的配额报告。

相关参考

[用于显示有关配额的信息的命令](#)（第 107 页）

配额报告所显示的空间使用量与 UNIX 客户端所显示的空间使用量之间的差别

配额报告中显示的某个 FlexVol 卷或 `qtree` 的已用磁盘空间值可能与 UNIX 客户端显示的同一个 FlexVol 卷或 `qtree` 的相应值不同。使用量值产生差别的原因是，配额报告和 UNIX 命令在计算卷或 `qtree` 中的数据块时遵循不同的方法。

例如，如果卷包含某个文件，而此文件含有空数据块（未向此文件写入任何数据），则该卷的配额报告在报告空间使用量时不会将空数据块计算在内。但是，当该卷挂载到 UNIX 客户端并且该文件显示在 `ls` 命令的输出中时，空数据块也会包含到空间使用量中。因此，`ls` 命令所显示的文件大小比配额报告所显示的空间使用量更大。

同样，配额报告中显示的空间使用量值还可能与 UNIX 命令（例如 `df` 和 `du`）的输出中显示的相应值不同。

### 配额报告如何计算磁盘空间使用量和文件使用量

FlexVol 卷或 `qtree` 的配额报告中指定的已用文件数和磁盘空间量取决于与该卷或 `qtree` 中每个索引节点对应的已用数据块数。

块数包括用于常规文件和流文件的直接块和间接块。用于目录、访问控制列表 (ACL)、流目录和元文件的块不会计入到配额报告中。对于 UNIX 稀疏文件，空数据块不会包含到配额报告中。

#### 相关概念

[ls 命令如何计算空间使用量](#)（第 109 页）

[df 命令如何计算文件大小](#)（第 110 页）

[du 命令如何计算空间使用量](#)（第 111 页）

### ls 命令如何计算空间使用量

当使用 `ls` 命令查看挂载到 UNIX 客户端的 FlexVol 卷的内容时，输出中显示的文件大小可能与该卷的配额报告中显示的空间使用量不相等，具体取决于文件的数据块类型。

`ls` 命令的输出仅显示某个文件的大小，并且不包括由该文件使用的间接块。该文件的任何空数据块也会包含到命令的输出中。

因此，如果文件不包含空数据块，则由 `ls` 命令显示的大小可能小于由配额报告指定的磁盘使用量，因为配额报告中包括间接块。相反，如果文件包含空数据块，则由 `ls` 命令显示的大小可能大于由配额报告指定的磁盘使用量。

`ls` 命令的输出仅显示某个文件的大小，并且不包括由该文件使用的间接块。该文件的任何空数据块也会包含到命令的输出中。

#### ls 命令计算的空间使用量与配额报告计算的空间使用量之间的差别的示例

以下配额报告显示了为 `qtree q1` 实施的磁盘空间限制是 10 MB：

Volume	Tree	Type	ID	---Disk---		---Files---		Quota
				Used	Limit	Used	Limit	Specifier
voll	q1	tree	user1	10MB	10MB	1	-	q1
...								

从 UNIX 客户端使用 `ls` 命令进行查看时，存在于同一个 `qtree` 中的文件的大小可能会超过配额限制，如以下示例所示：

```
[user1@lin-sys1 q1]$ ls -lh
-rwxr-xr-x  1 user1 nfsuser  27M Apr 09  2013 file1
```

#### 相关概念

[配额报告如何计算磁盘空间使用量和文件使用量](#) (第 109 页)

[df 命令如何计算文件大小](#) (第 110 页)

[du 命令如何计算空间使用量](#) (第 111 页)

#### df 命令如何计算文件大小

从配置了配额规则的 qtree 的挂载点运行 df 命令时，命令输出显示的空间使用量与配额报告指定的值相同。

如果为包含 qtree 的卷启用了配额，则 df 命令报告的空间使用量不包括由目录、ACL、流目录和元文件使用的块。因此，报告的空间使用量与配额报告指定的值完全匹配。

但是，如果没有为该 qtree 配置配额规则或没有为 FlexVol 卷启用配额，则报告的空间使用量包括由整个卷的目录、访问控制列表 (ACL)、流目录和元文件（包括卷中的其他 qtree）占用的块。在这种情况下，df 命令报告的空间使用量大于配额报告指定的值。

#### 由 df 命令和配额报告计算的空间使用量的示例

以下配额报告显示了为 qtree q1 实施的磁盘空间限制是 10 MB：

Volume	Tree	Type	ID	---Disk---		---Files---		Quota Specifier
				Used	Limit	Used	Limit	
vol1	q1	tree	user1	10MB	10MB	1	-	q1
...								

在以下示例中，df 命令输出显示的空间使用量与 10 MB 的磁盘空间限制相同（就 1K 块而言），因为已为该 qtree 配置了配额规则：

```
[user1@lin-sys1 q1]$ df -k
192.0.2.245:/vol/vol1/q1
                10240  10240  0  100% /q1
```

#### 相关概念

[配额报告如何计算磁盘空间使用量和文件使用量](#) (第 109 页)

[ls 命令如何计算空间使用量](#) (第 109 页)

[du 命令如何计算空间使用量](#) (第 111 页)

## du 命令如何计算空间使用量

当运行 `du` 命令检查 `qtree` 或挂载到 UNIX 客户端的 FlexVol 卷的磁盘空间使用量时，命令输出中显示的使用量值可能高于该 `qtree` 或卷的配额报告显示的值。

`du` 命令的输出包含目录树中从发出命令的目录级别起所有文件的合计空间使用量。由于 `du` 命令显示的使用量值还包括目录的数据块，因此该值高于配额报告显示的值。

### du 命令计算的空间使用量与配额报告计算的空间使用量之间的差别的示例

以下配额报告显示了为 `qtree q1` 实施的磁盘空间限制是 10 MB：

Volume	Tree	Type	ID	---Disk---		---Files---		Quota Specifier
				Used	Limit	Used	Limit	
vol1	q1	tree	user1	10MB	10MB	1	-	q1
...								

在以下示例中，`du` 命令输出显示的磁盘空间使用量超过了配额限制：

```
[user1@lin-sys1 q1]$ du -sh
11M    q1
```

## 相关概念

[配额报告如何计算磁盘空间使用量和文件使用量](#)（第 109 页）

[ls 命令如何计算空间使用量](#)（第 109 页）

[df 命令如何计算文件大小](#)（第 110 页）

## 配额配置示例

以下示例将帮助您了解如何配置配额和阅读配额报告。

以下示例假定您使用的存储系统包含名为 `vs1` 的 Storage Virtual Machine (SVM，以前称为 Vserver) 和一个名为 `vol1` 的卷。要开始设置配额，请使用以下命令为 SVM 创建新的配额策略：

```
cluster1::>volume quota policy create -vserver vs1 -policy-name
quota_policy_vs1_1
```

由于配额策略是新的，因此请通过输入以下命令将其分配给 SVM：

```
cluster1::>vserver modify -vserver vs1 -quota-policy quota_policy_vs1_1
```

**示例 1：默认用户配额**

您决定通过输入以下命令对 vol1 中每位用户实施 50 MB 的硬限制：

```
cluster1::>volume quota policy rule create -vserver vs1 -policy-
name quota_policy_vs1_1 -volume vol1 -type user -target "" -disk-
limit 50MB -qtree ""
```

要激活新规则，请通过输入以下命令初始化卷上的配额：

```
cluster1::>volume quota on -vserver vs1 -volume vol1 -foreground
```

要查看配额报告，请输入以下命令：

```
cluster1::>volume quota report
```

生成的配额报告类似于以下报告：

```
Vserver: vs1
```

Volume	Tree	Type	ID	---Disk---		---Files---		Quota Specifier
				Used	Limit	Used	Limit	
vol1		user	*	0B	50MB	0	-	*
vol1		user	jsmith	49MB	50MB	37	-	*
vol1		user	root	0B	-	1	-	

第一行显示了已创建的默认用户配额（包括磁盘限制）。与所有默认配额一样，此默认用户配额不显示有关磁盘或文件使用量的信息。除已创建的配额之外，它还显示了另外两个配额：当前在 vol1 上拥有文件的每位用户各一个配额。这些附加配额是自动根据默认用户配额派生的用户配额。与默认用户配额相同，用户 jsmith 的派生用户配额也具有 50 MB 的磁盘限制。root 用户的派生用户配额是跟踪配额（无限制）。

如果系统上任何用户（root 用户除外）尝试执行的操作将在 vol1 中占用超过 50 MB 的空间（例如，从编辑器对文件进行写入），该操作将失败。

**示例 2：显式用户配额覆盖默认用户配额**

如果需要在卷 vol1 中向用户 jsmith 提供更多空间，请输入以下命令：

```
cluster1::>volume quota policy rule create -vserver vs1 -policy-
name quota_policy_vs1_1 -volume vol1 -type user -target jsmith -
disk-limit 80MB -qtree ""
```

这是一个显式用户配额，因为其中明确将用户列为配额规则的目标。

这是对现有配额限制进行的更改，因为它更改了卷上用户 jsmith 的派生用户配额的磁盘限制。因此，您无需重新初始化卷上的配额即可激活更改。您可以通过输入以下命令来调整配额大小：

```
cluster1::>volume quota resize -vserver vs1 -volume voll -foreground
```

调整大小时，配额保持有效；而且调整大小过程耗时很短。

生成的配额报告类似于以下报告：

```
cluster1::> volume quota report
Vserver: vs1
```

Volume	Tree	Type	ID	---Disk---		---Files---		Quota Specifier
				Used	Limit	Used	Limit	
voll		user	*	0B	50MB	0	-	*
voll		user	jsmith	50MB	80MB	37	-	jsmith
voll		user	root	0B	-	1	-	

3 entries were displayed.

现在，第二行显示了 80 MB 的磁盘限制和配额说明符 jsmith。

因此，jsmith 最多可以使用 voll 上 80 MB 的空间，而其他所有用户的可用空间量仍限制为 50 MB。

### 示例 3：阈值

假设您希望在用户使用的空间量距磁盘限制还有 5 MB 时收到通知，那么请通过输入以下命令更改现有配额规则，以便为所有用户创建 45 MB 的阈值并为 jsmith 创建 75 MB 的阈值：

```
cluster1::>volume quota policy rule modify -vserver vs1 -policy
quota_policy_vs1_1 -volume voll -type user -target "" -qtree "" -
threshold 45MB
cluster1::>volume quota policy rule modify -vserver vs1 -policy
quota_policy_vs1_1 -volume voll -type user -target jsmith -qtree ""
-threshold 75MB
```

由于现有规则的大小已更改，因此必须调整卷上的配额大小以便激活更改。请等待调整大小过程完成。

要查看包含阈值的配额报告，请将 `-thresholds` 参数添加到 `volume quota report` 命令：

```
cluster1::>volume quota report -thresholds
Vserver: vs1
```

Volume	Tree	Type	ID	---Disk---		---Files---		Quota Specifier
				Used	Limit (Thold)	Used	Limit	
voll		user	*	0B	50MB (45MB)	0	-	*
voll		user	jsmith	59MB	80MB	55	-	jsmith



已创建的默认树配额将显示在第一个新行中，并且 ID 列会显示星号 (\*)。受卷上创建了默认树配额影响，Data ONTAP 会自动为卷中每个 qtree 创建派生树配额。这些派生树配额显示在树列中的 proj1 和 proj2 所在的行中。

以下新行用于派生用户配额：

Volume	Tree	Type	ID	---Disk---		---Files---		Quota Specifier
				Used	Limit	Used	Limit	
...								
voll	proj1	user	*	0B	50MB	0	-	
voll	proj1	user	root	0B	-	1	-	
voll	proj2	user	*	0B	50MB	0	-	
voll	proj2	user	root	0B	-	1	-	
...								

如果为 qtree 启用了配额，则卷上的默认用户配额会自动为该卷所含的所有 qtree 所继承。添加第一个 qtree 配额时，将在所有 qtree 上启用配额。因此，系统将为每个 qtree 创建派生的默认用户配额。这些派生用户配额显示在 ID 为星号 (\*) 的行中。

由于 root 用户是文件的所有者，因此，为每个 qtree 创建默认用户配额时，还会为每个 qtree 上的 root 用户创建特殊的跟踪配额。这些跟踪配额显示在 ID 为 root 的行中。

### 示例 5: qtree 上的用户配额

假设您决定限制 proj1 qtree 中用户的可用空间，使其低于在整个卷中为用户分配的空间，并想阻止用户在 proj1 qtree 中使用 10 MB 以上的空间，那么，请通过输入以下命令为该 qtree 创建默认用户配额：

```
cluster1::>volume quota policy rule create -vserver vs1 -policy-name quota_policy_vs1_1 -volume voll -type user -target "" -disk-limit 10MB -qtree proj1
```

这是对现有配额进行的更改，因为它更改了 proj1 qtree 从卷上的默认用户配额派生而来的默认用户配额。因此，请通过调整配额大小来激活更改。您可以在调整大小过程完成之后查看配额报告。

以下新行会显示在配额报告中，说明为 qtree 指定的新的显式用户配额：

Volume	Tree	Type	ID	---Disk---		---Files---		Quota Specifier
				Used	Limit	Used	Limit	
voll	proj1	user	*	0B	10MB	0	-	*

但是，系统将阻止用户 jsmith 向 proj1 qtree 写入更多数据，因为创建用于覆盖默认用户配额（以便提供更多空间）的配额位于卷上。随着您在 proj1 qtree 上添加了默认用户配额，该配额将会得到应用，并限制所有用户（包括 jsmith）在该 qtree 中的可用空间。

要向用户 jsmith 提供更多空间，请通过输入以下命令为该 qtree 添加磁盘限制为 80 MB 的显式用户配额规则，以覆盖该 qtree 的默认用户配额规则：

```
cluster1::>volume quota policy rule create -vserver vs1 -policy-
name quota_policy_vs1_1 -volume voll1 -type user -target jsmith -
disk-limit 80MB -qtree proj1
```

由于这是一个显式配额，而其默认配额已存在，因此，请通过调整配额大小来激活更改。您可以在调整大小过程完成之后查看配额报告。

以下新行显示在配额报告中：

Volume	Tree	Type	ID	---Disk---		---Files---		Quota Specifier
				Used	Limit	Used	Limit	
voll1	proj1	user	jsmith	61MB	80MB	57	-	jsmith

最终配额报告类似于以下报告：

```
cluster1::>volume quota report
Vserver: vs1
```

Volume	Tree	Type	ID	---Disk---		---Files---		Quota Specifier
				Used	Limit	Used	Limit	
voll1		tree	*	0B	20GB	0	-	*
voll1		user	*	0B	50MB	0	-	*
voll1		user	jsmith	70MB	80MB	65	-	jsmith
voll1	proj1	tree	1	0B	20GB	1	-	proj1
voll1	proj1	user	*	0B	10MB	0	-	*
voll1	proj1	user	root	0B	-	1	-	
voll1	proj2	tree	2	0B	20GB	1	-	proj2
voll1	proj2	user	*	0B	50MB	0	-	
voll1	proj2	user	root	0B	-	1	-	
voll1	proj2	user	root	0B	-	3	-	
voll1	proj1	user	jsmith	61MB	80MB	57	-	jsmith

11 entries were displayed.

用户 jsmith 必须满足以下配额限制才能向 proj1 中的文件写入数据：

1. proj1 qtree 的树配额。
2. proj1 qtree 上的用户配额。
3. 卷上的用户配额。

## 相关任务

[在包含 FlexVol 卷的 SVM 上设置配额](#)（第 117 页）

## 在包含 FlexVol 卷的 SVM 上设置配额

要在包含 FlexVol 卷的新 Storage Virtual Machine (SVM, 以前称为 Vserver) 上设置配额, 必须先创建配额策略, 并向此策略添加配额策略规则, 然后将此策略分配给 SVM, 最后在该 SVM 的每个 FlexVol 卷上初始化配额。

### 步骤

1. 使用带有 `-instance` 选项的 `vserver show` 命令显示创建 SVM 时自动创建的默认配额策略的名称。

如果未在创建 SVM 时指定名称, 则名称为 `default`。也可以使用 `vserver quota policy rename` 命令来为默认策略指定一个名称。

注: 也可以使用 `volume quota policy create` 命令来创建新策略。

2. 使用 `volume quota policy rule create` 命令为 SVM 上的每个卷创建以下任意配额规则:

- 适用于所有用户的默认配额规则
- 适用于特定用户的显式配额规则
- 适用于所有组的默认配额规则
- 适用于特定组的显式配额规则
- 适用于所有 qtree 的默认配额规则
- 适用于特定 qtree 的显式配额规则

3. 使用 `volume quota policy rule show` 命令检查配额规则是否配置正确。
4. 如果正在处理新策略, 请使用 `vserver modify` 命令向 SVM 分配此新策略。
5. 使用 `volume quota on` 命令在 SVM 的每个卷上初始化配额。

可以通过以下方法监控初始化过程:

- 当使用 `volume quota on` 命令时, 可以添加 `-foreground` 参数以在前台运行配额启用作业。(默认情况下, 此作业在后台运行。) 当该作业在后台运行时, 可以使用 `job show` 命令来监控其进度。
- 可以使用 `volume quota show` 命令来监控配额初始化的状态。

6. 使用 `volume quota show -instance` 命令检查初始化错误 (例如无法初始化的配额规则)。
7. 使用 `volume quota report` 命令显示配额报告, 以确保强制实施的配额符合要求。

### 相关概念

[什么是配额规则、配额策略和配额](#)（第 86 页）

[配额目标和类型](#)（第 86 页）

[特殊的配额类型](#)（第 87 页）

[如何激活配额](#)（第 101 页）

[如何查看配额信息](#)（第 104 页）

## 修改配额限制（或调整大小）

如果要对现有配额的大小进行更改，可以在所有受影响的卷上调整配额大小，因为此方法比在这些卷上重新初始化配额更快。

### 关于本任务

您的一个 Storage Virtual Machine (SVM, 以前称为 Vserver) 包含强制实施的配额，并且您想要更改现有配额的大小限制或者为已具有派生配额的目标添加或删除配额。

### 步骤

1. 使用带有 `-instance` 参数的 `vserver show` 命令，以确定当前分配给该 SVM 的策略的名称。
2. 通过执行以下任意操作来修改配额规则：
  - 使用 `volume quota policy rule modify` 命令修改现有配额规则的磁盘限制或文件限制。
  - 使用 `volume quota policy rule create` 命令为当前已具有派生配额的目标（用户、组或 `qtree`）创建显式配额规则。
  - 使用 `volume quota policy rule delete` 命令为同时具有默认配额的目标（用户、组或 `qtree`）删除显式配额规则。
3. 使用 `volume quota policy rule show` 命令检查配额规则是否配置正确。
4. 在每个更改了配额的卷上使用 `volume quota resize` 命令来为这些卷激活相关更改。  
可以通过以下任一方法监控调整大小过程：
  - 使用 `volume quota resize` 命令时，可以添加 `-foreground` 参数以在前台运行调整大小作业。（默认情况下，此作业在后台运行。）  
当该作业在后台运行时，可以使用 `job show` 命令来监控其进度。
  - 可以使用 `volume quota show` 命令来监控调整大小过程的状态。
5. 使用 `volume quota show -instance` 命令检查调整大小错误（例如调整大小失败的配额规则）。

尤其是要检查“新定义”错误，当您在为尚未具有派生配额的目标添加显式配额之后调整配额大小时，将发生此类错误。

6. 使用 `volume quota report` 命令显示配额报告，以确保强制实施的配额符合要求。

相关任务

[执行大量更改后重新初始化配额](#)（第 119 页）

## 执行大量更改后重新初始化配额

对现有配额进行大量更改（例如为不包含已执行配额的目标添加或删除配额）时，必须在所有受影响的卷上进行相应更改并重新初始化配额。

关于本任务

您的一个 Storage Virtual Machine (SVM) 包含已执行配额，并且您想要进行的更改要求执行配额完整重新初始化。

步骤

1. 使用带有 `-instance` 参数的 `vserver show` 命令，以确定当前分配给该 SVM 的策略的名称。
2. 通过执行以下任意操作来修改配额规则：

操作	命令
创建新的配额规则	使用 <code>volume quota policy rule create</code> 命令
修改现有配额规则的设置	使用 <code>volume quota policy rule modify</code> 命令
删除现有配额规则	使用 <code>volume quota policy rule delete</code> 命令

3. 使用 `volume quota policy rule show` 命令检查配额规则是否配置正确。
4. 在每个更改了配额的卷上重新初始化配额，方法是先禁用再启用这些卷的配额。
  - a. 在每个受影响的卷上使用 `volume quota off` 命令，以停用该卷上的配额。
  - b. 在每个受影响的卷上使用 `volume quota on` 命令，以激活该卷上的配额。

注：重新激活每个受影响卷上的配额之前，务必等待大约五分钟，因为在运行 `volume quota off` 命令之后立即尝试激活这些配额可能会出错。

或者，也可以从包含特定卷的节点运行命令以重新激活该卷的配额。

可以通过以下任一方法监控初始化流程：

- 当使用 `volume quota on` 命令时，可以添加 `-foreground` 参数以在前台运行配额启用作业。（默认情况下，此作业在后台运行。）  
当该作业在后台运行时，可以使用 `job show` 命令来监控其进度。

- 可以使用 `volume quota show` 命令来监控配额初始化的状态。
5. 使用 `volume quota show -instance` 命令检查初始化错误（例如无法初始化的配额规则）。
  6. 使用 `volume quota report` 命令显示配额报告，以确保强制实施的配额符合要求。

#### 相关概念

[需要完整配额重新初始化的情况](#)（第 103 页）

[如何查看配额信息](#)（第 104 页）

## 验证配额升级的状态

对包含配额的 FlexVol 卷启动从 Data ONTAP 7.3.x 到集群模式 Data ONTAP 版本（例如 Data ONTAP 8.2 或更高版本）的过渡时，可以验证配额是否已升级到集群模式 Data ONTAP 环境。

#### 开始之前

必须已对包含配额的 FlexVol 卷启动从 7-模式 Data ONTAP 到集群模式 Data ONTAP 的过渡。

#### 步骤

1. 通过使用 `volume quota show -instance` 命令来查看 7-模式卷中的数据将过渡到的特定集群模式 Data ONTAP 卷的配额详细信息。

以下示例显示了卷 `vol3` 的配额详细信息。这些详细信息还包括配额状态，该配额状态设置为正在初始化。

```
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: -
```

注：即将激活配额的卷也会显示正在初始化状态。但是，此类卷的子配额状态显示为正在扫描。

2. 通过使用 `job show` 命令来监控配额升级过程的进度。

3. 通过使用 `volume quota show -instance` 命令来验证集群模式 Data ONTAP 卷的配额升级过程的完成情况。

以下示例显示了卷 `vol3` 在升级完成之后的配额详细信息。配额状态为打开。

```
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: -
```

注：也可以通过 `quota.upgrade` 事件跟踪配额升级过程的开始和完成。

## 用于管理配额规则和配额策略的命令

可以使用 `volume quota policy rule` 命令来配置配额规则，也可以使用 `volume quota policy` 命令和一些 `vserver` 命令来配置配额策略。

注：以下命令只能对 FlexVol 卷运行。

### 用于管理配额规则的命令

操作	命令
创建新的配额规则	<code>volume quota policy rule create</code>
删除现有配额规则	<code>volume quota policy rule delete</code>
修改现有配额规则	<code>volume quota policy rule modify</code>
显示有关已配置的配额规则的信息	<code>volume quota policy rule show</code>

### 用于管理配额策略的命令

操作	命令
重复配额策略及其包含的配额规则	<code>volume quota policy copy</code>
新建空的配额策略	<code>volume quota policy create</code>
删除当前未分配给 Storage Virtual Machine (SVM) 的现有配额策略	<code>volume quota policy delete</code>
重命名配额策略	<code>volume quota policy rename</code>
显示有关配额策略的信息	<code>volume quota policy show</code>

操作	命令
将配额策略分配给 SVM	<code>vserver modify</code>
显示分配给 SVM 的配额策略的名称	<code>vserver show</code>

有关详细信息，请参见各个命令的手册页。

#### 相关概念

[什么是配额规则、配额策略和配额](#)（第 86 页）

[分配配额策略的注意事项](#)（第 92 页）

[如何查看配额信息](#)（第 104 页）

## 用于激活和修改配额的命令

可以使用 `volume quota` 命令来更改配额的状态和配置配额的消息日志记录。

操作	命令
启用配额（也称为初始化）	<code>volume quota on</code>
调整现有配额的大小	<code>volume quota resize</code>
禁用配额	<code>volume quota off</code>
更改配额的消息日志记录、启用配额、禁用配额或调整现有配额的大小	<code>volume quota modify</code>

有关详细信息，请参见各个命令的手册页。

#### 相关概念

[如何激活配额](#)（第 101 页）

[强制实施的配额为什么与配置的配额不同](#)（第 106 页）

#### 相关任务

[修改配额限制（或调整大小）](#)（第 118 页）

[执行大量更改后重新初始化配额](#)（第 119 页）

## 使用重复数据删除和数据压缩提高存储效率

---

可以在 FlexVol 卷或无限卷上同时运行或单独运行重复数据删除和数据压缩，以实现最高空间节省。重复数据删除可清除重复的数据块，而数据压缩可对数据块进行压缩以减少所需的物理存储量。

### 如何设置效率操作

根据存储环境设置，可以先评估能够实现的空间节省，然后再配置重复数据删除和数据压缩（或者只配置重复数据删除）。可以使用计划或策略在卷上运行效率操作。

可以使用空间节省评估工具来评估在现有环境中能够实现的空间节省。空间节省评估工具最多可对 2 TB 数据进行评估。可从以下网址下载空间节省评估工具：[communities.netapp.com/docs/DOC-18699](https://communities.netapp.com/docs/DOC-18699)。

### 配置重复数据删除

重复数据删除是一项 Data ONTAP 功能，通过删除 FlexVol 卷或无限卷中重复的数据块减少所需的物理存储空间量。在根卷上不应启用重复数据删除。

可以决定在启用重复数据删除之后仅删除写入卷的新数据中的重复数据，或者在启用重复数据删除之前删除新数据和卷中现有数据中的重复数据。

相关任务

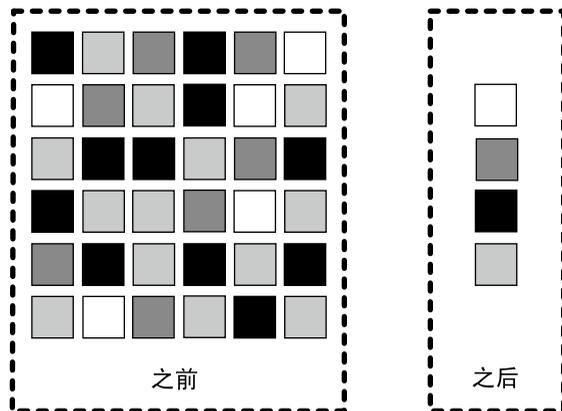
[在卷上启用重复数据删除](#)（第 126 页）

### 重复数据删除工作原理

在整个 FlexVol 卷或无限卷中，重复数据删除在块级别下运行，这样不但能够删除重复数据块，而且只会存储唯一的数据块。

每个数据块都具有一个数字签名，此签名将与数据卷中的所有其他签名进行比较。如果存在完全匹配的块签名，则会对块中的所有字节执行逐字节比较。仅当所有字节都匹配时，才会丢弃重复块并回收其磁盘空间，从而避免造成任何数据丢失。

重复数据删除将消除数据冗余，如下图所示：



Data ONTAP 以 4 KB 的块为单位将所有数据写入到存储系统。首次对已经存在数据的卷运行重复数据删除时，此操作会扫描该卷中的所有块并为每个块创建数字指纹。每个指纹将与卷中的所有其他指纹进行比较。如果发现两个指纹完全相同，则会对块中的所有数据执行逐字节比较。如果逐字节比较检测到完全相同的数据，则会更新指向该数据的指针，并删除重复块。

注：对已经存在数据的卷运行重复数据删除时，最好将重复数据删除配置为扫描该卷中的所有块，以便节省更多空间。

重复数据删除在活动文件系统中运行。因此，随着更多数据写入到已进行重复数据删除的卷，将为每个新块创建指纹并将其写入到更改日志文件。对于后续的重复数据删除操作，系统将对更改日志进行排序并将其与指纹文件合并，然后重复数据删除操作将继续执行如前所述的指纹比较。

有关对无限卷运行重复数据删除的详细信息，请参见《集群模式 *Data ONTAP* 无限卷管理指南》。

## 什么是重复数据删除元数据

重复数据删除元数据包含指纹文件和更改日志。指纹是指 FlexVol 卷或无限卷中每个 4 KB 数据块使用的数字签名。

重复数据删除元数据包含两个更改日志文件。在运行重复数据删除时，其中一个更改日志文件中的新数据块指纹会合并到指纹文件中，而另一个更改日志文件会存储在运行重复数据删除操作期间写入到卷中的新数据的指纹。下次运行重复数据删除操作时，这两个更改日志文件的角色将发生反转。

在 Data ONTAP 8.0.1 中，重复数据删除元数据位于聚合中。自 Data ONTAP 8.1 起，每个卷会保留两份重复数据删除元数据。其中一份重复数据删除元数据会驻留在卷中，而另一份则驻留在聚合中。聚合中的重复数据删除元数据用作所有重复数据删除操作的工作副本。卷中会驻留一份额外的重复数据删除元数据。

移动卷时，重复数据删除元数据也会与该卷一起传输。如果卷所有权发生变化，则当下次运行重复数据删除时，会自动使用卷中的重复数据删除元数据副本在聚合中创建重复数据删除元数据。此方法比创建新的指纹文件速度更快。

自 Data ONTAP 8.2 起，每个物理块都存储有指纹，这样可减少存储重复数据删除元数据所需的空容量。

重复数据删除元数据最多可占卷中总物理数据量的 7%，如下所示：

- 在一个卷中，重复数据删除元数据最多可占该卷中总数据量的 4%。  
对于无限卷，每个数据成分卷中的重复数据删除元数据最多可占该数据成分卷中总数据量的 4%。
- 在一个聚合中，重复数据删除元数据最多可占卷中总物理数据量的 3%。

可以使用 `storage aggregate show` 命令检查聚合中的可用空间，还可以使用 `volume show` 命令检查卷中的可用空间。有关这些命令的详细信息，请参见手册页。

#### 示例

一个 2 TB 的聚合包含四个卷，每个卷的大小为 400 GB。您需要以不同的节省百分比对三个卷进行重复数据删除。

各个卷需要用于重复数据删除元数据的空容量如下所示：

- 2 GB [4% × 100 GB 的 50%]（逻辑数据量为 100 GB 且节省百分比为 50%）
- 6 GB [4% × 200 GB 的 75%]（逻辑数据量为 200 GB 且节省百分比为 25%）
- 3 GB [4% × 300 GB 的 25%]（逻辑数据量为 300 GB 且节省百分比为 75%）

聚合需要用于重复数据删除元数据的总可用空容量是 8.25 GB（3% × 100 GB 的 50% + 3% × 200 GB 的 75% + 3% × 300 GB 的 25% = 1.5 + 4.5 + 2.25 = 8.25 GB）。

## 使用重复数据删除的准则

当重复数据删除操作正在 FlexVol 卷或无限卷上运行时，重复数据删除将作为系统操作运行并占用系统资源。

如果卷中的数据更改不频繁，最好不要频繁运行重复数据删除。如果在存储系统上运行多个重复数据删除并发操作，这些操作将占用大量系统资源。最开始时，重复数据删除并发操作数最好控制在较少的水平。随后逐渐增加重复数据删除并发操作数，从而更好地了解它对系统的影响。

注：最好不要让多个卷的卷大小接近启用了重复数据删除的卷的逻辑数据限制。

## 与重复数据删除相关的性能注意事项

有多种因素会影响重复数据删除的性能。在易受性能影响的环境或生产环境中部署重复数据删除之前，应通过测试设置（包括规模估算注意事项）来检查对重复数据删除的性能造成的影响。

以下因素会影响重复数据删除的性能：

- 数据访问模式（例如顺序访问与随机访问以及输入和输出的大小和模式）
- 重复数据量、数据总量和平均文件大小
- 卷中数据布局的性质
- 两次重复数据删除操作之间更改的数据量
- 并发重复数据删除操作的数量
- 硬件平台（系统内存和 CPU 模块）
- 系统负载
- 磁盘类型（例如，ATA/FC 和磁盘每分钟的转数）

有关重复数据删除的性能注意事项的详细信息，请参见技术报告 3966：《适用于集群模式 Data ONTAP 的数据压缩和重复数据删除部署和实施指南》。

相关信息

[《适用于集群模式 Data ONTAP 的数据压缩和重复数据删除部署和实施指南》](http://media.netapp.com/documents/tr-3966.pdf)：  
[media.netapp.com/documents/tr-3966.pdf](http://media.netapp.com/documents/tr-3966.pdf)

## 在卷上启用重复数据删除

您可以通过使用 `volume efficiency on` 命令在 FlexVol 卷或无限卷上启用重复数据删除，以便提高存储效率。

开始之前

对于 FlexVol 卷，必须已验证卷和聚合中有足够的可用空间供重复数据删除元数据使用。

步骤

1. 使用 `volume efficiency on` 命令启用重复数据删除。

示例

以下命令将在卷 VolA 上启用重复数据删除：

```
volume efficiency on -vserver vs1 -volume VolA
```

相关概念

[使用策略管理卷效率操作](#)（第 130 页）

[使用计划管理卷效率操作](#)（第 137 页）

## 在卷上禁用重复数据删除

您可以通过使用 `volume efficiency off` 命令在卷上禁用重复数据删除。

关于本任务

如果已在卷上启用数据压缩，则运行 `volume efficiency off` 命令会禁用数据压缩。

步骤

1. 使用 `volume efficiency stop` 命令可停止卷上当前处于活动状态的任何卷效率操作。
2. 使用 `volume efficiency off` 命令可禁用重复数据删除操作。

示例

以下命令将在卷 VolA 上禁用重复数据删除：

```
volume efficiency off -vserver vs1 -volume VolA
```

## 配置数据压缩

数据压缩是 Data ONTAP 的一项功能，通过压缩 FlexVol 卷或无限卷内的数据块，可以减少在存储系统上存储数据所需的物理容量。

在 HDD 聚合及 Flash Pool 聚合上托管的所有 FlexVol 卷都支持数据压缩。

您可以在主存储层、二级存储层和三级存储层上使用数据压缩功能。

相关任务

[在卷上启用数据压缩](#)（第 128 页）

## 数据压缩工作原理

通过使用数据压缩，您不但可以用更少空间存储更多数据，而且还能减少在卷 SnapMirror 传输期间复制数据所需的时间和带宽。数据压缩能够节省常规文件或 LUN 占用的空间。

但是，存储系统内部文件、Windows NT 流和卷元数据不会压缩。

数据压缩的工作方式是压缩一小组连续数据块（称为压缩组）。数据压缩可以用下列方式执行：

- 实时压缩  
如果在某个卷上启用了实时压缩，则在后续的数据写入期间，会压缩可压缩数据并将其写入到卷中。但是，无法压缩的数据或实时压缩绕过的数据将以未压缩格式写入到卷中。

如果将某个卷配置为使用仅实时效率策略，则可以在该卷上运行实时数据压缩，而无需执行任何计划的或手动启动的后台效率操作。

- 后处理压缩

如果在某个卷上启用了后处理压缩，则对该卷执行的新的数据写入操作最初不会进行压缩（如果已启用实时压缩），而是在运行后处理压缩时作为压缩数据重新写入到该卷中。后处理压缩操作会作为低优先级后台进程运行。

如果同时启用了实时压缩和后处理压缩，则后处理压缩只会压缩未运行实时压缩的块。这包括实时压缩绕过的块，例如少量、部分压缩组覆盖。

有关对无限卷执行压缩的详细信息，请参见《集群模式 *Data ONTAP* 无限卷管理指南》。

## 数据压缩如何检测不可压缩数据并节省系统资源

不可压缩数据检测功能可用于检查文件是否可压缩，对于大文件，它还可用于检查文件内的压缩组是否可压缩。启用不可压缩数据检测功能可节省实时压缩在压缩不可压缩的数据或压缩组时使用的系统资源。

默认情况下，对于小于 500 MB（此值可以更改）的文件，实时压缩会检查压缩组是否可压缩。如果在压缩组内检测到不可压缩数据，则会为包含该压缩组的文件设置一个标志，以指示该文件不可压缩。在以后尝试执行压缩期间，实时压缩会先检查文件是否设置了不可压缩数据标志。如果设置了该标志，则不会尝试对该文件执行实时压缩。

对于等于或大于 500 MB（此值可以更改）的文件，实时压缩会快速检查每个压缩组的前 4 KB 块，以确定该块是否可压缩。如果无法压缩该 4 KB 块，则不会压缩该压缩组。但是，如果成功压缩该 4 KB 块，则会尝试压缩整个压缩组。

无论文件是否可压缩，后处理压缩都会对所有文件运行。如果后处理压缩至少压缩了不可压缩文件中的一个压缩组，则会清除该文件的不可压缩数据标志。在下次尝试执行压缩期间，实时压缩便可对该文件运行以节省空间。

有关启用或禁用不可压缩数据检测功能以及修改尝试在某个文件上快速检查的最小文件大小的详细信息，请参见 `volume efficiency modify` 命令手册页。

## 在卷上启用数据压缩

您可以通过使用 `volume efficiency modify` 命令在 FlexVol 卷或无限卷上启用数据压缩，以便节省空间。

开始之前

必须已经在卷上启用了重复数据删除。

注：要在卷上启用数据压缩，只需要在该卷上启用重复数据删除，而不需要运行重复数据删除。

[在卷上启用重复数据删除](#)（第 126 页）

关于本任务

您可以在卷上同时启用实时压缩和后处理压缩，也可以只启用后处理压缩。启用实时压缩之前，必须在卷上启用后处理压缩。

步骤

1. 使用 `volume efficiency modify` 命令可启用数据压缩。

示例

以下命令将在卷 VolA 上启用后处理压缩：

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

以下命令将在卷 VolA 上同时启用后处理压缩和实时压缩：

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

相关概念

[使用策略管理卷效率操作](#)（第 130 页）

[使用计划管理卷效率操作](#)（第 137 页）

## 在卷上禁用数据压缩

可以通过使用 `volume efficiency modify` 命令在 FlexVol 卷或无限卷上禁用数据压缩。

关于本任务

如果要禁用后处理压缩，必须先要在卷上禁用实时压缩。

步骤

1. 使用 `volume efficiency stop` 命令可停止卷上当前处于活动状态的任何卷效率操作。
2. 使用 `volume efficiency modify` 命令可禁用数据压缩。

示例

以下命令将在卷 VolA 上禁用实时压缩：

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

以下命令将在卷 VolA 上同时禁用后处理压缩和实时压缩：

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

## 使用策略管理卷效率操作

可以通过以下方式在 FlexVol 卷或无限卷上运行重复数据删除或数据压缩：将操作计划为在特定时间启动；或指定一个阈值百分比以在超过该阈值百分比后触发操作。

可以通过以下方法计划重复数据删除操作或数据压缩操作：创建封装于效率策略内的作业计划；或者指定一个阈值百分比（此时系统会等新数据超过指定的百分比之后再触发重复数据删除操作或数据压缩操作）。可以使用 `volume efficiency modify` 命令来解除效率策略关联，以停止在卷上运行其他任何基于计划的重复数据删除操作和数据压缩操作。

卷效率策略存在于 Storage Virtual Machine (SVM) 的环境中。

卷效率策略仅支持 cron 类型的作业计划。有关创建类型为 cron 的作业计划的详细信息，请参见《适用于集群管理员的集群模式 Data ONTAP 系统管理指南》。

## 使用卷效率优先级确定效率操作的优先顺序

服务质量 (QoS) 策略的功能可用于确定正在卷上作为 `best-effort` 操作或 `background` 操作运行的卷效率操作的优先顺序。

将卷效率操作计划为 `best-effort` 操作或 `background` 操作可使存储系统上系统资源及其他系统操作的利用率最大化。无论是否为卷分配了效率策略，都可以指定每个卷的卷效率优先级。

有关为卷效率操作指定优先级的详细信息，请参见 `volume efficiency policy modify` 命令手册页和技术报告 *TR-3966*：《适用于集群模式 Data ONTAP 的数据压缩和重复数据删除部署和实施指南》。

相关信息

《适用于集群模式 Data ONTAP 的数据压缩和重复数据删除部署和实施指南》：  
[media.netapp.com/documents/tr-3966.pdf](http://media.netapp.com/documents/tr-3966.pdf)

## 了解预定义的效率策略

从 Data ONTAP 8.3 开始，可以在没有对卷配置计划或手动启动的后台效率操作的情况下运行内联压缩。创建 Storage Virtual Machine (SVM) 时，将自动创建两个预定义的效率策略：仅内联和默认。

您可以使用默认效率策略来配置卷，以在卷上运行计划的重复数据删除操作。

您可以使用仅内联效率策略来配置卷，在启用内联数据压缩后，可以在卷上运行内联数据压缩，无需执行任何计划的或手动启动的后台效率操作。

注：您不能删除仅内联和默认效率策略。

有关仅内联和默认效率策略的详细信息，请参见手册页。

## 创建卷效率策略以运行效率操作

您可以使用 `volume efficiency policy create` 命令创建执行如下任务的卷效率策略并指定作业计划：运行重复数据删除或数据压缩，然后在特定持续时间内对 FlexVol 卷或无限卷运行重复数据删除。

### 开始之前

您必须已经使用 `job schedule cron create` 命令创建了 cron 计划。有关管理 cron 计划的详细信息，请参见《适用于集群管理员的集群模式 *Data ONTAP* 系统管理指南》。

### 关于本任务

具有默认的预定义角色的 SVM 管理员无法管理重复数据删除策略。但是，集群管理员可以使用任何自定义角色来修改分配给 SVM 管理员的权限。有关 SVM 管理员功能的详细信息，请参见《适用于 SVM 管理员的集群模式 *Data ONTAP* 系统管理指南》。

注：您可以通过以下方法，在指定的计划时间运行重复数据删除操作或数据压缩操作：创建具有特定持续时间的计划；或者指定一个阈值百分比，此时系统会等到新数据超过指定的百分比之后，触发重复数据删除操作或数据压缩操作。

### 步骤

1. 使用 `volume efficiency policy create` 命令创建卷效率策略。

#### 示例

以下命令将创建名为 `pol1` 并在每天触发效率操作的卷效率策略：

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

以下命令将创建名为 `pol2` 并在阈值百分比达到 20% 时触发效率操作的卷效率策略：

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

## 将卷效率策略分配给卷

您可以使用 `volume efficiency modify` 命令，将效率策略分配给卷，以便运行重复数据删除操作或数据压缩操作。

### 关于本任务

如果为 SnapVault 二级卷分配了效率策略，则运行卷效率操作时只会考虑卷效率优先级属性。对 SnapVault 二级卷进行增量更新之后，将会忽略作业计划并运行重复数据删除操作。

## 步骤

1. 使用 `volume efficiency modify` 命令为卷分配策略。

## 示例

以下命令将为卷 VoLA 分配名为 `new_policy` 的卷效率策略：

```
volume efficiency modify -vserver vs1 -volume VoLA -policy new_policy
```

## 修改卷效率策略

可以通过使用 `volume efficiency policy modify` 命令来将卷效率策略修改为在不同的持续时间内运行重复数据删除和数据压缩，或者更改作业计划。

## 步骤

1. 使用 `volume efficiency policy modify` 命令修改卷效率策略。

## 示例

以下命令会将名为 `policy1` 的卷效率策略修改为每小时运行一次：

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

以下命令会将名为 `pol2` 的卷效率策略的阈值修改为 30%：

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

## 查看卷效率策略

可以通过使用 `volume efficiency policy show` 命令来查看卷效率策略的名称、计划、持续时间和说明。

## 关于本任务

在集群范围内运行 `volume efficiency policy show` 命令时，不会显示集群范围内的策略。但是，可以在 Storage Virtual Machine (SVM) 上下文中查看集群范围内的策略。

## 步骤

1. 使用 `volume efficiency policy show` 命令查看有关卷效率策略的信息。

输出取决于所指定的参数。有关显示详细视图及其他参数的详细信息，请参见此命令的手册页。

**示例**

以下命令将显示为 SVM vs1 创建的策略的相关信息：

```
volume efficiency policy show -vserver vs1
```

以下命令将显示持续时间设置为 10 小时的策略：

```
volume efficiency policy show -duration 10
```

## 解除卷效率策略与卷的关联

您可以解除卷效率策略与卷的关联，以停止在该卷上运行任何基于计划的后继重复数据删除操作和数据压缩操作。解除卷效率策略关联后，可以手动触发该策略。

**步骤**

1. 使用 `volume efficiency modify` 命令可解除卷效率策略与卷的关联。

**示例**

以下命令将解除卷效率策略与卷 VolA 的关联：

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

## 删除卷效率策略

您可以通过使用 `volume efficiency policy delete` 命令来删除卷效率策略。

**开始之前**

您必须已确认要删除的策略未与任何 FlexVol 卷或无限卷关联。

注：仅实时和默认预定义效率策略无法删除。

**步骤**

1. 使用 `volume efficiency policy delete` 命令删除卷效率策略。

**示例**

以下命令将删除名为 policy1 的卷效率策略：

```
volume efficiency policy delete -vserver vs1 -policy policy1
```

## 手动管理卷效率操作

可以通过手动运行效率操作来控制效率操作在 FlexVol 卷或无限卷上的运行方式。

还可以基于以下条件控制效率操作的运行方式：

- 是否使用检查点
- 对现有数据运行效率操作还是仅对新数据运行
- 根据需要停止效率操作

可以使用 `volume efficiency show` 命令并将 `schedule` 指定为 `-fields` 选项的值，以查看分配给卷的计划。

### 手动运行效率操作

可以使用 `volume efficiency start` 命令在 FlexVol 卷或无限卷上手动运行效率操作。

#### 开始之前

根据要手动运行的效率操作的不同，必须先要在卷上启用重复数据删除或同时启用数据压缩和重复数据删除。

#### 关于本任务

如果在卷上启用了重复数据删除和数据压缩，则会先运行数据压缩，然后再运行重复数据删除。

重复数据删除是一个在运行期间会消耗系统资源的后台进程。如果卷中的数据更改不频繁，最好不要频繁运行重复数据删除。在存储系统中并发运行多个重复数据删除操作会消耗更多系统资源。

每个节点最多可以运行八个并发重复数据删除操作或数据压缩操作。如果计划运行更多效率操作，这些操作将排队等待。

在无限卷上运行重复数据删除或数据压缩时，单独的操作会在卷中的每个数据成分卷上运行，而且每个节点最多可以运行八个并发操作。

#### 步骤

1. 使用 `volume efficiency start` 命令在卷上启动效率操作。

#### 示例

以下命令可用于在卷 VolA 上手动单独启动重复数据删除或先启动数据压缩再启动重复数据删除：

```
volume efficiency start -vserver vs1 -volume VolA
```

## 使用检查点来恢复效率操作

检查点用来在内部记录效率操作的执行过程。当由于任何原因（例如系统暂停、系统中断、重新启动或者上次效率操作失败或停止）而停止效率操作，并且存在检查点数据时，该效率操作可从最新检查点文件恢复。

对于无限卷，检查点会在每个数据成分卷上创建。无法查看无限卷上的检查点，但可以恢复操作。

检查点在下列情况下创建：

- 相关操作的每个阶段或子阶段
- 运行 `sis stop` 命令时
- 持续时间到期后

## 使用检查点选项恢复效率操作

可以通过将 `volume efficiency start` 命令与检查点选项结合使用来恢复效率操作。

关于本任务

如果在卷上只启用了重复数据删除，则会对数据运行重复数据删除。但是，如果在卷上启用了重复数据删除和数据压缩，则会先运行数据压缩，然后再运行重复数据删除。

可以使用 `volume efficiency show` 命令来查看卷检查点的详细信息。

默认情况下，效率操作会从检查点恢复。但是，如果与上次效率操作（运行 `volume efficiency start -scan-old-data` 命令的阶段）对应的检查点超过 24 小时，则效率操作不会自动从上一个检查点恢复。在这种情况下，效率操作会从头开始。但是，如果知道卷自上次扫描以来未发生重大更改，则可以使用 `-use-checkpoint` 选项来强制从上一个检查点继续。

步骤

1. 将 `volume efficiency start` 命令与检查点选项结合使用来恢复效率操作。

示例

以下命令在 VolA 卷上使用了检查点选项，可用来恢复效率操作：

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

以下命令在 VolA 卷上使用了检查点选项，可用来在现有数据上恢复效率操作：

```
volume efficiency start -vserver vs1 -volume VolA -scan-old-data true
-use-checkpoint true
```

## 对现有数据手动运行效率操作

启用重复数据删除或数据压缩之前，可以对 FlexVol 卷或无限卷中的数据手动运行效率操作。可以使用 `volume efficiency start -scan-old-data` 命令来运行重复数据删除，或者先运行数据压缩，再运行重复数据删除。

关于本任务

如果在卷上只启用了重复数据删除，则会对数据运行重复数据删除。但是，如果在卷上启用了重复数据删除和数据压缩，则会先运行数据压缩，然后再运行重复数据删除。

对现有数据运行数据压缩时，数据压缩操作会默认跳过由重复数据删除共享的数据块以及由 Snapshot 副本锁定的数据块。如果选择对共享块运行数据压缩，则会关闭优化功能，并捕获指纹信息，以便在再次共享时使用。可以更改数据压缩在压缩现有数据时的默认行为。有关详细信息，请参见 *TR-3966: 《适用于集群模式 Data ONTAP 的数据压缩和重复数据删除部署和实施指南》*。

在无限卷上运行重复数据删除或数据压缩时，系统会分别在该卷中的每个数据成分卷上单独运行操作。

每个节点最多可以同时运行八个重复数据删除操作或数据压缩操作，剩余操作会排队等待。

步骤

1. 使用 `volume efficiency start -scan-old-data` 命令对现有数据手动运行重复数据删除，或者先运行数据压缩，再运行重复数据删除。

示例

以下命令可用于对卷 VolA 中的现有数据手动运行重复数据删除，或者先运行数据压缩，再运行重复数据删除：

```
volume efficiency start -vserver vs1 -volume VolA -scan-old-data true
```

相关信息

《适用于集群模式 Data ONTAP 的数据压缩和重复数据删除部署和实施指南》：  
[media.netapp.com/documents/tr-3966.pdf](http://media.netapp.com/documents/tr-3966.pdf)

## 使用计划管理卷效率操作

可以通过使用计划或根据写入到 FlexVol 卷或无限卷的新数据量来管理效率操作在 FlexVol 卷或无限卷上的运行方式。

### 根据写入的新数据量运行效率操作

您可以修改效率操作计划，以便在上次执行效率操作（手动执行或按计划执行）之后写入到卷中的新数据块数超过指定阈值百分比时运行重复数据删除或数据压缩。

关于本任务

如果 `schedule` 选项设置为 `auto`，则会在新数据量超过指定百分比时运行计划的效率操作。默认阈值是 20%。此阈值是指占效率操作已处理的总数据块数的百分比。

步骤

1. 使用 `volume efficiency modify` 命令与 `auto@num` 选项修改阈值百分比值。  
`num` 是一个用来指定百分比的两位数。

示例

以下命令会将卷 VolA 的阈值百分比值修改为 30%：

```
volume efficiency modify -vserver vs1 -volume -VolA -schedule auto@30
```

### 使用计划来运行效率操作

可以使用 `volume efficiency modify` 命令来修改 FlexVol 卷或无限卷上的重复数据删除操作或数据压缩操作的计划。计划和卷效率策略使用的是互相排斥的配置选项。

步骤

1. 使用 `volume efficiency modify` 命令修改卷上的重复数据删除操作或数据压缩操作的计划。

示例

以下命令会将 VolA 的效率操作的计划修改为在星期一到星期五晚上 11 点运行：

```
volume efficiency modify -vserver vs1 -volume VolA -schedule mon-fri@23
```

## 监控卷效率操作

可以通过以下方法监控 FlexVol 卷或无限卷上的效率操作的进度：查看该 FlexVol 卷或无限卷上的效率操作的状态和实现的空间节省。

有关无限卷上的效率操作和空间节省的详细信息，请参见《集群模式 *Data ONTAP* 无限卷管理指南》。

### 查看效率操作的状态

您可以使用 `volume efficiency show` 命令查看 FlexVol 卷或无限卷上是否启用了重复数据删除或数据压缩，并查看 FlexVol 卷或无限卷中各个数据成分卷上效率操作的状态和进度。

#### 步骤

1. 使用 `volume efficiency show` 命令查看卷上的效率操作的状态。

#### 示例

以下命令将显示卷 VolA 上的效率操作的状态：

```
volume efficiency show -vserver vs1 -volume VolA
```

如果在卷 VolA 上启用了效率操作，而该操作处于闲置状态，则会在系统输出中显示以下内容：

```
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
.....
.....
.....
```

### 查看效率空间节省

可以使用 `volume show` 命令查看通过重复数据删除和数据压缩在卷上实现的空间节省量。

#### 关于本任务

计算卷上实现的空间节省时，Snapshot 副本中的空间节省不包括在内。使用重复数据删除不会影响卷配额。配额是在逻辑级别下报告的，并且保持不变。

## 步骤

1. 使用 `volume show` 命令查看通过重复数据删除和数据压缩在卷上实现的空间节省。

## 示例

以下命令可用于查看通过重复数据删除和数据压缩在卷 VolA 上实现的空间节省：

```
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%

.....

.....
```

## 查看 FlexVol 卷的效率统计信息

可以使用 `volume efficiency stat` 命令查看在 FlexVol 卷上运行的效率操作的详细信息。

## 步骤

1. 使用 `volume efficiency stat` 命令查看 FlexVol 卷上的效率操作的统计信息。

## 示例

以下命令可用于查看卷 VolA 上的效率操作的统计信息：

```
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
```

## 停止卷效率操作

可以使用 `volume efficiency stop` 命令来停止重复数据删除操作或后处理压缩操作。此命令会自动生成检查点。

### 步骤

1. 使用 `volume efficiency stop` 命令停止处于活动状态的重复数据删除操作或后处理压缩操作。

如果指定了 `-all` 选项，则会中止处于活动状态的和排队等待的效率操作。

### 示例

以下命令将停止卷 VolA 上当前处于活动状态的重复数据删除操作或后处理压缩操作：

```
volume efficiency stop -vserver vs1 -volume VolA
```

以下命令将同时中止卷 VolA 上处于活动状态的和排队等待的重复数据删除操作或后处理压缩操作：

```
volume efficiency stop -vserver vs1 -volume VolA -all true
```

## 有关从卷中移除空间节省的信息

在卷上运行效率操作就能选择移除已实现的空间节省。移除或撤消卷上的空间节省之前，务必与技术支持联系。

有关从卷中移除空间节省的详细信息，请参见技术报告 *TR-3966*：《适用于集群模式 *Data ONTAP* 的数据压缩和重复数据删除部署和实施指南》。

### 相关信息

《适用于集群模式 *Data ONTAP* 的数据压缩和重复数据删除部署和实施指南》：  
[media.netapp.com/documents/tr-3966.pdf](http://media.netapp.com/documents/tr-3966.pdf)

## 重复数据删除与 Data ONTAP 功能的互操作性

使用重复数据删除时，应了解重复数据删除所支持的功能以及这些功能如何与重复数据删除配合使用。

重复数据删除支持以下功能：

- Snapshot 副本
- 卷 SnapMirror
- SnapRestore
- OnCommand Unified Manager 服务器
- 卷副本
- 数据压缩
- FlexClone 卷
- HA 对
- DataMotion for Volumes
- SnapVault 备份

您可以在已删除重复数据的卷上启用块区，也可以执行读取重新分配，以改善已删除重复数据的卷的文件布局和顺序读取性能。

### 相关概念

[预留百分比如何与重复数据删除配合使用](#)（第 142 页）

[Snapshot 副本如何与重复数据删除配合使用](#)（第 142 页）

[卷 SnapMirror 如何与重复数据删除配合使用](#)（第 142 页）

[SnapRestore 如何与重复数据删除配合使用](#)（第 143 页）

[OnCommand Unified Manager 服务器如何与重复数据删除配合使用](#)（第 143 页）

[重复数据删除如何与数据压缩配合使用](#)（第 144 页）

[FlexClone 卷如何与重复数据删除配合使用](#)（第 144 页）

[HA 对如何与重复数据删除配合使用](#)（第 144 页）

[DataMotion for Volumes 如何与重复数据删除配合使用](#)（第 144 页）

[SnapVault 备份如何与重复数据删除配合使用](#)（第 145 页）

## 预留百分比如何与重复数据删除配合使用

如果要为预留百分比设置为 0 的卷使用重复数据删除，则需要满足其他一些配置要求才能确保应用程序永不收到 ENOSPC（空间不足）错误。有关详细信息，请参见与设置预留百分比相关的文档。

相关概念

[设置预留百分比的注意事项](#)（第 23 页）

## Snapshot 副本如何与重复数据删除配合使用

可以在活动文件系统中单独运行重复数据删除。但是，数据会在运行重复数据删除之前创建的 Snapshot 副本中锁定，减少空间节省。

要避免重复数据删除与 Snapshot 副本之间发生冲突，应遵循以下准则：

- 创建新 Snapshot 副本之前运行重复数据删除。
- 删除存储在已进行重复数据删除的卷中不需要的 Snapshot 副本。
- 缩短存储在已进行重复数据删除的卷中的 Snapshot 副本的保留时间。
- 在向卷写入了大量新数据之后，方可运行重复数据删除计划。
- 为 Snapshot 副本配置相应的预留空间。
- 如果快照预留为 0，则应禁用自动创建 Snapshot 副本的计划（大多数 LUN 部署都已禁用该计划）。

## 卷 SnapMirror 如何与重复数据删除配合使用

无论卷的大小和卷中包含多少逻辑数据，都可以使用卷 SnapMirror 来复制已进行重复数据删除的卷。

将卷 SnapMirror 与重复数据删除结合使用时，必须考虑以下信息：

- 可以在源系统和（或）目标系统上启用重复数据删除。
- 共享块仅传输一次。  
因此，重复数据删除也能降低网络带宽使用量。
- 中断卷 SnapMirror 关系后，在目标存储系统中实施默认的重复数据删除计划。

当配置卷 SnapMirror 和重复数据删除时，应将重复数据删除计划与卷 SnapMirror 计划相协调。应在重复数据删除操作完成之后再对已进行重复数据删除的卷启动卷 SnapMirror 传输。此计划可防止通过网络发送未进行重复数据删除的数据和其他临时元数据文件。如果源卷中的临时元数据文件锁定在 Snapshot 副本中，则这些文件会在源卷和目标卷中占用额外空间。

## SnapRestore 如何与重复数据删除配合使用

在运行重复数据删除操作期间创建的元数据同时位于 FlexVol 卷和聚合中。因此，在卷上启动 SnapRestore 操作时，元数据会恢复到卷中，并且恢复后的数据将保留原始空间节省。

如果在卷上启用了重复数据删除，则会在 SnapRestore 操作完成后继续对写入此卷的任何新数据进行重复数据删除。

## OnCommand Unified Manager 服务器如何与重复数据删除配合使用

重复数据删除可与 OnCommand Unified Manager 服务器中的 NetApp 管理控制台数据保护功能、NetApp 管理控制台配置功能和 Operations Manager 配合使用。

重复数据删除与 **OnCommand Unified Manager** 服务器中的 **NetApp** 管理控制台数据保护功能

在 OnCommand Unified Manager 服务器 5.2R1 之前的版本中，NetApp 管理控制台数据保护功能会等待处于活动状态的重复数据删除操作完成，然后再重命名 Snapshot 副本。当 NetApp 管理控制台数据保护功能正在等待时，它不允许客户端列出 Snapshot 副本或从这些副本进行恢复。因此，在 OnCommand Unified Manager 服务器 5.2R1 之前的版本中，将重复数据删除与 NetApp 管理控制台数据保护功能结合使用并不是最佳做法。

但是，OnCommand Unified Manager 服务器 5.2R1 已解决此限制。

有关将重复数据删除与 NetApp 管理控制台数据保护功能结合使用的详细信息，请参见《*OnCommand Unified Manager* 管理指南》。

重复数据删除与 **OnCommand Unified Manager** 服务器中的 **NetApp** 管理控制台配置功能

通过使用 OnCommand Unified Manager 服务器中的 NetApp 管理控制台配置功能，您可使配置策略支持全部三种重复数据删除模式（也就是按需重复数据删除、自动重复数据删除和计划的重复数据删除）。

有关将重复数据删除与 NetApp 管理控制台配置功能结合使用的详细信息，请参见《*OnCommand Unified Manager* 管理指南》。

重复数据删除与 **OnCommand Unified Manager** 服务器

您可以通过 OnCommand Unified Manager 服务器中的 Operations Manager 执行重复数据删除操作。

您可以生成报告或图形，汇总文件克隆或 LUN 克隆节省的空间。

有关将重复数据删除与 Operations Manager 配合使用的详细信息，请参见《*OnCommand Unified Manager* 管理指南》。

相关信息

[NetApp 支持站点 \(mysupport.netapp.com\)](http://mysupport.netapp.com) 上的文档

## 重复数据删除如何与数据压缩配合使用

当 FlexVol 卷上同时启用了数据压缩和重复数据删除时，将先进行数据压缩，然后再进行重复数据删除。将这两项操作结合使用可能会比单独运行重复数据删除节省更多空间，具体取决于数据类型。

## FlexClone 卷如何与重复数据删除配合使用

FlexClone 卷支持重复数据删除。已删除重复数据的卷的 FlexClone 卷也是已删除重复数据的卷。克隆卷会继承父卷的重复数据删除配置（例如重复数据删除计划）。

执行重复数据删除操作期间创建的元数据（指纹文件和更改日志文件）会克隆下来。此元数据位于 FlexVol 卷和聚合中。

如果对克隆卷运行重复数据删除，则会删除该克隆的重复数据，但不会删除父卷的重复数据。

要手动为克隆卷中的所有新数据运行重复数据删除，则应使用 `volume efficiency start` 命令。

如果从父卷中拆分克隆卷，则会在执行卷拆分操作后删除克隆中属于父卷的所有数据的重复数据删除信息。但是，如果对克隆卷运行重复数据删除，则会在后续的重复数据删除操作期间对这部分数据执行重复数据删除。

## HA 对如何与重复数据删除配合使用

自 Data ONTAP 8.1 起，如果 HA 对上发生了接管，则接管期间可从任一节点对卷运行重复数据删除操作。HA 对上每个节点允许的重复数据删除最大并发操作数为八个。

如果一个节点发生故障，另一个节点将接管由故障节点管理的重复数据删除操作。在接管模式下，正常工作的节点将继续执行重复数据删除操作。正常工作的节点可对属于故障节点的卷启动重复数据删除操作。当正常工作的节点管理同属两个节点的卷的重复数据删除操作时，重复数据删除最大并发操作数仍为八个。

## DataMotion for Volumes 如何与重复数据删除配合使用

使用 DataMotion for Volumes（卷移动）操作移动 FlexVol 卷时，该卷上通过重复数据删除而节省的空间将会保留。卷移动操作处于活动状态时，如果重复数据删除操作正在运行，这些操作会短暂停止，直至最终转换完成。

卷移动完成后，效率操作无法从上一个检查点继续进行，而会从头开始。

如果尝试以无中断方式移动正在运行重复数据删除操作的 FlexVol 卷，则会中止重复数据删除操作。

## SnapVault 备份如何与重复数据删除配合使用

可以从逻辑上将启用了重复数据删除的主卷复制到 SnapVault 二级卷。复制过程中，可以保留在主卷上实现的空间节省，或者也可以在 SnapVault 二级卷上单独启用重复数据删除。

进行复制以及向 SnapVault 二级卷写入数据期间，通过重复数据删除在主卷上实现的空间节省会通过网络保留下来。

在以下情况下，建议在 SnapVault 二级卷上启用重复数据删除：

- 无法在主卷上配置重复数据删除，但 SnapVault 二级卷需要实现空间节省，而这些二级卷中的 Snapshot 副本的保留时间较长。
- 无法在主卷到 SnapVault 二级卷的复制传输开始之前完成对主卷的重复数据删除。

将 SnapVault 备份与重复数据删除结合使用时，必须遵循以下准则：

- 不能在 SnapVault 二级卷上手动运行效率操作，但可以对二级卷运行 `volume efficiency start -scan-old-data` 命令。
- 如果已将重复数据删除配置为在 SnapVault 二级卷上运行，则会在每次从主卷到 SnapVault 二级卷的复制传输完成后触发重复数据删除。
- 从主卷到 SnapVault 二级卷进行数据传输过程中，无法修改 SnapVault 二级卷上的重复数据删除操作的状态。
- 在 SnapVault 二级卷上启动卷复制操作或克隆操作时，不会复制自创建上一个 Snapshot 副本以来对卷进行的任何更改。

有关 SnapVault 关系中的卷的详细信息，请参见《集群模式 *Data ONTAP* 数据保护指南》。

## 虚拟机对齐如何与重复数据删除配合使用

可以将虚拟机对齐功能与适用于 VMware vSphere 的 Virtual Storage Console (VSC) 结合使用，从而以无中断方式修复未对齐的虚拟机磁盘 (VMDK)。对于存在大量未对齐 VM 的用户，此功能有助于提高系统性能并增加通过重复数据删除实现的空间节省。

虚拟机对齐是需要创建卷时指定的卷级别设置。虚拟机对齐属性可确保在卷中创建的虚拟机磁盘与存储系统的基础层对齐。

注：虚拟机对齐功能只能在基于 NFS 的存储系统上使用。

有关安装和管理适用于 VMware vSphere 的 Virtual Storage Console 的详细信息，请参见《适用于 *VMware vSphere* 的 *Virtual Storage Console* 安装和管理指南》。

## MetroCluster 配置如何与重复数据删除配合使用

在 MetroCluster 切回过程中，如果卷上所含的聚合正在切回，则这些卷上处于活动状态的所有重复数据删除操作都会停止，而且系统会为停止的每个重复数据删除操作记录一个检查点。

当 MetroCluster 切回过程完成后，如果卷上含有关联的计划或策略，则重复数据删除操作会从记录的检查点重新启动。如果卷上没有关联的计划或策略，则可以从记录的检查点手动重新启动重复数据删除操作。

## 数据压缩与 Data ONTAP 功能的互操作性

使用数据压缩时，应了解数据压缩所支持的功能以及这些功能如何与数据压缩配合使用。

数据压缩支持以下功能：

- Snapshot 副本
- 卷 SnapMirror
- 磁带备份
- 基于卷的 SnapRestore
- 单一文件 SnapRestore
- 卷副本
- 重复数据删除
- FlexClone 卷
- FlexClone 文件
- HA 对
- 闪存卡
- DataMotion for Volumes
- Flash Pool 聚合
- SnapVault 备份

而且，经过压缩的卷上也不支持读取重新分配和块区。

相关概念

[预留百分比如何与数据压缩配合使用](#)（第 147 页）

- [Snapshot 副本如何与数据压缩配合使用](#) (第 147 页)
- [卷 SnapMirror 如何与数据压缩配合使用](#) (第 147 页)
- [磁带备份如何与数据压缩配合使用](#) (第 148 页)
- [基于卷的 SnapRestore 如何与数据压缩配合使用](#) (第 148 页)
- [单文件 SnapRestore 如何与数据压缩配合使用](#) (第 148 页)
- [重复数据删除如何与数据压缩配合使用](#) (第 149 页)
- [FlexClone 卷如何与数据压缩配合使用](#) (第 149 页)
- [FlexClone 文件如何与数据压缩配合使用](#) (第 149 页)
- [HA 对如何与数据压缩配合使用](#) (第 149 页)
- [闪存卡如何与数据压缩配合使用](#) (第 149 页)
- [DataMotion for Volumes 如何与数据压缩配合使用](#) (第 150 页)
- [Flash Pool 聚合如何与数据压缩配合使用](#) (第 150 页)
- [SnapVault 备份如何与数据压缩配合使用](#) (第 150 页)

## 预留百分比如何与数据压缩配合使用

如果要为预留百分比设置为 0 的卷使用数据压缩，则需要满足其他一些配置要求才能确保应用程序永不收到 ENOSPC（空间不足）错误。有关详细信息，请参见与设置预留百分比相关的文档。

相关概念

- [设置预留百分比的注意事项](#) (第 23 页)

## Snapshot 副本如何与数据压缩配合使用

如果在创建 Snapshot 副本之后以默认模式运行数据压缩，则会对该 Snapshot 副本锁定的数据进行压缩。

Snapshot 副本将锁定数据块，直到 Snapshot 副本过期或被删除才能将其释放。在任何启用了数据压缩的卷上，如果创建数据的 Snapshot 副本，则后续对该数据进行的任何更改都需要使用额外的磁盘空间，直至 Snapshot 被删除或过期为止。

## 卷 SnapMirror 如何与数据压缩配合使用

由于卷 SnapMirror 在物理块级别下运行，因此，如果在源存储系统上启用数据压缩，则将数据复制到目标存储系统时，该数据仍处于压缩状态。此操作可大幅减少复制过程所需的网络带宽。

将卷 SnapMirror 与数据压缩结合使用时，必须遵循以下准则：

- 为使 SnapMirror 传输按预期进行，目标存储系统必须正在运行相同版本或更高版本的 Data ONTAP。  
如果源存储系统正在运行 Data ONTAP 8.1，则目标存储系统必须正在运行 Data ONTAP 8.1 或更高版本。

- 只能从主存储系统启用、运行和管理数据压缩。  
但是，二级存储系统中的 FlexVol 卷会继承所有数据压缩属性以及通过卷 SnapMirror 传输实现的存储节省。
- 如果计划使用 `-shared-blocks` 或 `-snapshot-blocks` 选项将磁盘中的现有数据压缩到某个 FlexVol 卷，而此卷包含由 Snapshot 副本锁定的数据块并且已有卷 SnapMirror 关系，则此操作可能会导致传输大量数据块。  
只能在高级权限级别下指定这些选项。  
数据压缩操作会将数据作为经过压缩的新块重新写入，并且在下次进行增量传输时传输这些新块。

有关卷 SnapMirror 的详细信息，请参见《集群模式 *Data ONTAP* 数据保护指南》。

相关信息

[NetApp 支持站点 \(mysupport.netapp.com\) 上的文档](https://mysupport.netapp.com)

## 磁带备份如何与数据压缩配合使用

当使用 NDMP 备份压缩数据时，源卷中的数据会以未压缩格式写入到磁带上。因此，如果在从磁带恢复的卷上重新获取空间节省，必须于启动恢复之前在该卷上启用数据压缩。

可以从磁带将压缩数据恢复到目标卷，并且恢复过程会保留空间节省。仅当希望在恢复后的卷上压缩从客户端写入的新数据时，才可以启用实时压缩。

## 基于卷的 SnapRestore 如何与数据压缩配合使用

在包含压缩数据的 FlexVol 卷上启动基于卷的 SnapRestore 操作时，压缩设置将恢复为 Snapshot 副本的压缩设置，并且恢复后的数据将保留 Snapshot 副本的原始空间节省。

有关基于卷的 SnapRestore 的详细信息，请参见《集群模式 *Data ONTAP* 数据保护指南》。

相关信息

[NetApp 支持站点 \(mysupport.netapp.com\) 上的文档](https://mysupport.netapp.com)

## 单文件 SnapRestore 如何与数据压缩配合使用

启动单文件 SnapRestore 操作时，数据会从 Snapshot 副本恢复到活动文件系统中，并且会恢复原始空间节省。

如果在卷上启用了不可压缩数据检测，则不可压缩数据标志会从 Snapshot 副本恢复到活动文件系统中。

有关单文件 SnapRestore 的详细信息，请参见《集群模式 *Data ONTAP* 数据保护指南》。

相关信息

[NetApp 支持站点 \(mysupport.netapp.com\) 上的文档](https://mysupport.netapp.com)

## 重复数据删除如何与数据压缩配合使用

当 FlexVol 卷上同时启用了数据压缩和重复数据删除时，将先进行数据压缩，然后再进行重复数据删除。将这两项操作结合使用可能会比单独运行重复数据删除节省更多空间，具体取决于数据类型。

## FlexClone 卷如何与数据压缩配合使用

如果从父卷中拆分 FlexClone 卷，则新卷会从父卷继承数据压缩属性。继承的属性会指示重复数据删除、后处理压缩和实时压缩是否已启用。新卷还会继承在父卷中节省的空间。

如果在父卷上的解压缩操作处于活动状态时创建 FlexClone 卷，则解压缩操作不会在克隆卷上运行。

## FlexClone 文件如何与数据压缩配合使用

您可以在包含 FlexClone 文件的 FlexVol 卷上运行数据压缩。在启用了数据压缩的 FlexVol 卷上，只能创建完全克隆的文件。在启用了数据压缩的 FlexVol 卷上，不能创建不完全克隆的文件。

如果在卷上启用了不可压缩数据检测，则克隆的文件会继承不可压缩标志。

## HA 对如何与数据压缩配合使用

您可以在 HA 对中启用数据压缩。如果某个节点发生故障，另一个节点将接管故障节点的操作。在接管模式下，正常工作的节点将继续执行数据压缩操作。

有关 HA 对的详细信息，请参见《集群模式 *Data ONTAP* 高可用性配置指南》。

相关信息

[NetApp 支持站点 \(mysupport.netapp.com\)](https://mysupport.netapp.com) 上的文档

## 闪存卡如何与数据压缩配合使用

虽然数据压缩和闪存卡在同一个存储系统中工作，但无论是否使用闪存卡，压缩数据的读取性能始终保持不变。

## DataMotion for Volumes 如何与数据压缩配合使用

使用 DataMotion for Volumes（卷移动）操作移动 FlexVol 卷时，该卷上通过数据压缩而节省的空间将会保留。要使卷移动操作成功完成，源卷和目标卷必须运行相同版本的 Data ONTAP。

## Flash Pool 聚合如何与数据压缩配合使用

Flash Pool 聚合支持数据压缩，但经过压缩的块在 Flash Pool 聚合的固态硬盘 (SSD) 缓存中只能用于读取缓存，而不能用于写入缓存。对于启用了数据压缩的卷，只有未经压缩的块才会进行写入缓存。

## SnapVault 备份如何与数据压缩配合使用

可以从逻辑上将 SnapVault 主卷复制到 SnapVault 二级卷。复制过程中，如果未在 SnapVault 二级卷上启用数据压缩，则在传输数据以及向二级卷写入数据期间，通过数据压缩在主卷上实现的空间节省会通过网络保留下来。

但是，如果将 SnapVault 二级卷配置为运行压缩，则未压缩数据会通过网络传输并写入到二级卷。即使在增量更新过程中，未压缩数据也会通过网络传输。

将 SnapVault 备份与数据压缩结合使用时，必须遵循以下准则：

- 如果 SnapVault 主卷与二级卷之间正在进行数据传输，则无法修改 SnapVault 二级卷上的数据压缩操作的状态。
- 虽然在 SnapVault 二级卷上禁用了数据压缩，但可以在 SnapVault 二级卷上启用重复数据删除。  
重复数据删除会在每次从 SnapVault 主卷到二级卷的复制传输完成后触发。
- 如果需要在 SnapVault 二级卷上运行数据压缩以实现空间节省，但不需要在主卷上运行数据压缩，则可以在 SnapVault 二级卷上使用实时压缩。  
但是，如果要在 SnapVault 主卷和二级卷上运行数据压缩，则在传输数据以及向二级卷写入数据期间，在主卷上实现的空间节省不会保留下来。
- 如果只需要在 SnapVault 二级卷上运行实时压缩而不必运行重复数据删除，则必须将 SnapVault 二级卷的计划更改为 `manual`，或将效率策略设置为 `inline-only`。
- 如果在 SnapVault 二级卷上禁用数据压缩，则会先对二级卷执行 SnapRestore 操作，然后再启动后续从主卷到二级卷的传输。  
但是，如果创建上一个 Snapshot 副本时在二级卷上启用了数据压缩，则不会执行 SnapRestore 操作。虽然现已在 SnapVault 二级卷上禁用了数据压缩，但未压缩数据仍会写入到二级卷。
- 在 SnapVault 二级卷上启动卷复制或克隆操作时，不会复制自创建上一个 Snapshot 副本以来对卷进行的任何更改。

不能在 SnapVault 二级卷上手动运行效率操作，但可以对二级卷运行 `volume efficiency start -scan-old-data` 命令。

有关 SnapVault 关系中的卷的详细信息，请参见《集群模式 *Data ONTAP* 数据保护指南》。

## 存储限制

规划和管理存储架构时，应考虑存储对象的限制。

限制详情请参见以下各节：

- [卷限制](#)
- [FlexClone 文件和 FlexClone LUN 限制](#)

### 卷限制

存储对象	限制	本机存储	存储阵列	虚拟存储 (Data ONTAP-v)
阵列 LUN	根卷的最大大小 <sup>1</sup>	不适用	因型号而异	不适用
文件	最大大小	16 TB	16 TB	16 TB
	每个卷的上限 <sup>2</sup>	因卷大小而异，最多 20 亿个	因卷大小而异，最多 20 亿个	因卷大小而异，最多 20 亿个
<b>FlexClone 卷</b>	分层克隆深度 <sup>3</sup>	499	499	499
<b>FlexVol 卷</b>	每个节点的上限 <sup>1</sup>	因型号而异	因型号而异	200
	每个 SVM 每个节点的上限 <sup>4</sup>	因型号而异	因型号而异	200
	最小大小	20 MB	20 MB	20 MB
	最小大小 <sup>1</sup>	因型号而异	因型号而异	因型号而异
<b>FlexVol 根卷</b>	最小大小 <sup>1</sup>	因型号而异	因型号而异	因型号而异
<b>LUN</b>	每个节点的上限 <sup>4</sup>	因型号而异	因型号而异	1,024
	每个集群的上限 <sup>4</sup>	因型号而异	因型号而异	1,024
	每个卷的上限 <sup>4</sup>	因型号而异	因型号而异	512
	大小上限	16 TB	16 TB	16 TB
<b>qtree</b>	每个 FlexVol 卷的上限	4,995	4,995	4,995

存储对象	限制	本机存储	存储阵列	虚拟存储 (Data ONTAP-v)
Snapshot 副本	每个 FlexVol 卷或无限卷的上限 <sup>5</sup>	255	255	255
卷	每个 NAS 集群的上限 <sup>6</sup>	12,000	12,000	200
	配置了 SAN 协议的每个集群的上限 <sup>7</sup>	因型号而异	因型号而异	200

注意：

1. 请参见 [Hardware Universe](#)。
2. 20 亿 =  $2 \times 10^9$  的 9 次方。
3. 可以从单个 FlexVol 卷创建的 FlexClone 卷嵌套层次结构的最大深度。
4. 此限制仅适用于 SAN 环境。  
请参见《[集群模式 Data ONTAP SAN 配置指南](#)》。
5. 使用某些 Data ONTAP 功能可以减小此限制。  
请参见《[集群模式 Data ONTAP 数据保护指南](#)》。
6. 无限卷不算在此限制内，但其成分卷算在此限制内。
7. 无限卷不算在此限制内，但其成分卷算在此限制内。  
请参见《[集群模式 Data ONTAP SAN 配置指南](#)》。

### FlexClone 文件和 FlexClone LUN 限制

限制	本机存储	存储阵列	虚拟存储 (Data ONTAP-v)
每个文件或 LUN 的上限 <sup>1</sup>	32,767	32,767	32,767
每个 FlexVol 卷的总共享数据上限	640 TB	640 TB	100 TB

注意：

1. 如果您试图创建 32,767 个以上的克隆，Data ONTAP 将自动为父文件或父 LUN 创建新的物理副本。  
对于使用重复数据删除的 FlexVol 卷，此限制可能更低。

## 版权信息

---

版权所有 © 1994 - 2015 NetApp, Inc.。保留所有权利。中国印刷。

未经版权所有者事先书面许可，本文档中受版权保护的任何部分不得以任何形式或通过任何手段（图片、电子或机械方式，包括影印、录音、录像或存储在电子检索系统中）进行复制。

从受版权保护的 NetApp 资料派生的软件受以下许可和免责声明的约束：

本软件由 NETAPP 按“原样”提供，不含任何明示或暗示担保，包括但不限于适销性以及针对特定用途的适用性的隐含担保，特此声明不承担任何责任。在任何情况下，对于因使用本软件而以任何方式造成的任何直接性、间接性、偶然性、特殊性、惩罚性或后果性损失（包括但不限于购买替代商品或服务；使用、数据或利润方面的损失；或者业务中断），无论原因如何以及基于何种责任理论，无论出于合同、严格责任或侵权行为（包括疏忽或其他行为），NETAPP 均不承担责任，即使已被告知存在上述损失的可能性。

NetApp 保留在不另行通知的情况下随时对本文档所述的任何产品进行更改的权利。除非 NetApp 以书面形式明确同意，否则 NetApp 不承担因使用本文档所述产品而产生的任何责任或义务。使用或购买本产品不表示获得 NetApp 的任何专利权、商标权或任何其他知识产权许可。

本手册中描述的产品可能受一项或多项美国专利、外国专利或正在申请的专利的保护。

有限权利说明：美国政府使用、复制或公开本文档受 DFARS 252.277-7103（1988 年 10 月）和 FAR 52-227-19（1987 年 6 月）中“技术数据和计算机软件权利”条款第 (c)(1)(ii) 条规定的限制条件的约束。

## 商标信息

---

NetApp、NetApp 标识、Go Further, Faster、ASUP、AutoSupport、Campaign Express、Cloud ONTAP、clustered Data ONTAP、Customer Fitness、Data ONTAP、DataMotion、Fitness、Flash Accel、Flash Cache、Flash Pool、FlashRay、FlexArray、FlexCache、FlexClone、FlexPod、FlexScale、FlexShare、FlexVol、FPolicy、GetSuccessful、LockVault、Manage ONTAP、Mars、MetroCluster、MultiStore、NetApp Insight、OnCommand、ONTAP、ONTAPI、RAID DP、SANtricity、SecureShare、Simplicity、Simulate ONTAP、Snap Creator、SnapCopy、SnapDrive、SnapIntegrator、SnapLock、SnapManager、SnapMirror、SnapMover、SnapProtect、SnapRestore、Snapshot、SnapValidator、SnapVault、StorageGRID、Tech OnTap、Unbound Cloud 和 WAFL 是 NetApp, Inc. 在美国和/或其他国家或地区的商标或注册商标。有关最新的 NetApp 商标列表，请参见 <http://www.netapp.com/cn/legal/netapptmlist.aspx>。

Cisco 和 Cisco 标识是 Cisco 在美国和/或其他国家或地区的商标或注册商标。所有其他品牌或产品均为其各自所有者的商标或注册商标，应予同样对待。

## 意见反馈

---

您可以向我们发送反馈意见，帮助我们提高文档质量。

您的反馈意见对于帮助我们提高信息的准确度和质量而言至关重要。如果您对改进本文档有任何建议，请发送电子邮件至 [ng-gpso-china-documents@netapp.com](mailto:ng-gpso-china-documents@netapp.com)。为了方便我们将您的意见或建议转发给相关主管部门，请在主题行写明产品名称、版本和操作系统。

您也可以通过以下方式与我们联系：

- 北京市朝阳区东大桥路9号侨福芳草地C座6层606室100020
- 电话：86-10-59293000
- 传真：86-10-59293099
- 支持电话：86-10-59293008

# 索引

- ## A
- aggregate show-space 命令
    - 如何确定聚合空间使用量 [40](#)
- ## B
- 保证
    - FlexVol 卷 [20](#)
    - 卷, 如何使用 FlexVol 卷 [19](#)
  - 备份
    - 重复数据删除如何与 SnapVault 配合使用 [145](#)
  - 不允许执行重复数据删除操作
    - 无中断卷移动期间 [144](#)
- ## C
- 拆分
    - FlexClone 卷从其父卷 [58](#)
  - 策略
    - 配额 [86](#)
  - 传统机会锁
    - 提高 SMB 客户端性能 [15](#)
  - 创建
    - FlexClone 文件和 FlexClone LUN [68](#)
    - 卷效率策略 [131](#)
  - 磁盘空间使用量 [108](#)
  - 存储限制
    - FlexClone 文件和 LUN [152](#)
    - RAID 组 [152](#)
    - 聚合 [152](#)
    - 卷 [152](#)
  - 存储效率
    - 重复数据删除 [123](#)
    - 如何设置 [123](#)
    - 使用数据压缩 [123](#)
    - 使用重复数据删除 [123](#)
    - 数据压缩 [127](#)
- ## D
- DataMotion for Volumes
    - 比较 [10](#)
  - df 命令
    - 何时不应使用空间使用量命令 [42](#)
    - 如何计算磁盘空间使用量 [110](#)
  - du 命令
    - 如何计算磁盘空间使用量 [111](#)
  - 调整配额大小
    - 更改配额限制 [118](#)
    - 命令 [118](#)
  - 读写 FlexClone 卷
    - 创建 [59](#)
- ## F
- 反馈
    - 如何发送对文档的意见 [156](#)
  - Flash Pool 聚合
    - 缓存策略 [45](#)
    - 它们如何与数据压缩配合使用 [150](#)
    - 移动卷时的注意事项和建议 [52](#)
  - FlexClone LUN
    - 工作原理 [63](#)
    - 另请参阅 FlexClone 文件和 FlexClone LUN
  - FlexClone 卷
    - 共享 Snapshot 副本和 [57](#)
    - 关于 [56](#)
    - 使用 LUN 和 FlexClone LUN 的方式 [59](#)
    - 比较 [10](#)
    - 创建 [59](#)
    - 从父卷中拆分 [60](#)
    - 分层克隆深度 [152](#)
    - 共享 Snapshot 副本, 识别 [58](#)
    - 何时应创建 FlexClone 卷以用作复制聚合卷的方法 [55](#)
    - 确定 [62](#)
    - 如何与卷 SnapMirror 结合使用 [58](#)
    - 它们是指什么 [56](#)
    - 移动卷时的注意事项和建议 [52](#)
    - 与重复数据删除配合使用 [144](#)
  - FlexClone 文件
    - FlexClone LUN
      - 卷大小上限 [153](#)
      - 每个卷的共享数据上限 [153](#)
      - 每个文件或 LUN 的上限 [153](#)
    - 工作原理 [63](#)
    - 卷大小上限 [153](#)
    - 每个卷的共享数据上限 [153](#)
    - 每个文件或 LUN 的上限 [153](#)
    - 另请参阅 FlexClone 文件和 FlexClone LUN

## FlexClone 文件和 FlexClone LUN

- FlexClone 卷如何与 77
- HA 对 78
- Snapshot 副本如何与 76
- volume file clone deletion 命令 75
- 查看节点的分摊负载 69
- 查看空间节省 70
- 查看用于创建和删除的节点容量 69
- 创建 68
- 创建 FlexClone 卷 77
- 对令牌的支持 67
- 分摊负载的定义 67
- 关于 63
- 回收可用空间 71
- 禁用自动删除 74
- 卷移动 78
- 空间预留 78
- 配额如何 77
- 启用自动删除 72
- 删除方法 70
- 优势 63
- 与 Data ONTAP 功能的互操作性 75
- 预留百分比设置为零 66
- 重复数据删除如何与 76
- 注意事项 65

## FlexClone 文件和 LUN

- 零预留百分比的影响 23

## FlexVol 卷

- SVM 对聚合选择的影响 16
- 创建 47
- 创建 FlexClone 文件和 LUN 68
- 创建空间的方法 42
- 从父卷创建 FlexClone 卷 59
- 分配过度警报, 如何解决 28
- 功能比较 10
- 管理简介 47
- 厚配置 19
- 精简配置 19
- 控制 I/O 性能 48
- 了解 Snapshot 副本使用的空间 41
- 每个 SVM 的上限 152
- 配置 FlexClone 文件和 LUN 的自动删除 72
- 配置为在已满时自动提供更多空间 24
- 配置自动更改大小 27
- 启用保证 20
- 如何从 FlexClone 文件和 FlexClone LUN 回收可用空间 71
- 如何确定空间使用量 32, 33
- 如何使用卷保证 19
- 如何限制 SVM 16

## 删除 48

- 设置预留百分比的注意事项 23
- 使用 qtree 进行分区的简介 78
- 使用简介 16
- 使用精简配置的注意事项 22
- 它们如何与 SVM 配合使用 16
- 提供的功能 9
- 填满度警报, 如何解决 28
- 同时启用自动缩减和 Snapshot 副本自动删除的要求 27
- 移动的工作原理 51
- 已定义 9
- 用于管理的命令 50
- 用于显示空间使用量信息的命令 50
- 与 qtree 相比 79
- 与无限卷的对比 12
- 与无限卷共享的聚合 14
- 语言受到 SVM 17
- 在同一个 SVM 中移动 54
- 重复数据删除准则 125
- 自动缩减功能与 Snapshot 副本自动删除功能交互 28
- 最大和最小大小 152
- 最大目录大小, 注意事项 45
- 最大文件数, 注意事项 45

## 父 FlexVol 卷

- 拆分 FlexClone 卷 58

## 访问控制列表

- 它们如何与 FlexClone 文件和 LUN 配合使用 76

## 分配配额策略

- 注意事项 92

## 分区

- 使用 qtree 对 FlexVol 卷进行分区的简介 78

## 分摊负载

- 查看节点 69
- 为 FlexClone 文件和 FlexClone LUN 定义的 67

## G

## 根聚合

- 节点需要遵循的规则 47

## 根卷

- 节点需要遵循的规则 47

## 跟踪配额 90

## 功能

- FlexVol 卷, 提供的功能 9
- 无限卷 11

## 规则

- 配额 86

- ## H
- HA 对
    - 和重复数据删除 [144](#)
  - 厚配置
    - 如何使用 FlexVol 卷 [19](#)
  - 缓存策略
    - 使用 Flash Pool 聚合 [45](#)
- ## J
- 建议
    - 如何发送文档反馈 [156](#)
  - 卷
    - 定义的无限卷 [11](#)
    - FlexVol 功能比较 [10](#)
    - FlexVol 卷移动的工作原理 [51](#)
    - FlexVol 卷与无限卷的对比 [12](#)
    - FlexVol, 配置为在已满时自动提供更多空间 [24](#)
    - FlexVol, 使用简介 [16](#)
    - FlexVol, 提供的功能 [9](#)
    - FlexVol, 移动的工作原理 [51](#)
    - FlexVol, 已定义 [9](#)
    - 工作原理 [9](#)
    - qtree 与 FlexVol 相比 [79](#)
    - 创建 FlexClone [59](#)
    - 创建 FlexVol [47](#)
    - 分配过度警报, 如何解决 [28](#)
    - 复制方法 [55](#)
    - 将精简配置与 FlexVol 结合使用的注意事项 [22](#)
    - 节点根卷需要遵循的规则 [47](#)
    - 聚合的共享 [14](#)
    - 配置 FlexVol 卷的自动更改大小功能 [27](#)
    - 如何确定空间使用量 [32, 33](#)
    - 如何确定空间使用量, 在聚合中 [37](#)
    - 删除 FlexVol [48](#)
    - 设置 FlexVol 预留百分比的注意事项 [23](#)
    - 填满度警报, 如何解决 [28](#)
    - 为 FlexVol 卷同时启用自动缩减和 Snapshot 副本自动删除的要求 [27](#)
    - 系统卷, 定义 [15](#)
    - 移动卷时的注意事项和建议 [52](#)
    - 移动要求 [54](#)
    - 用于管理 FlexVol 卷的命令 [50](#)
    - 在 FlexVol 卷中创建空间的方法 [42](#)
    - 在同一个 SVM 中移动 [54](#)
    - 自动缩减功能与 Snapshot 副本自动删除功能交互 [28](#)
  - 机会锁
    - 提高 SMB 客户端性能 [15](#)
  - 节点
    - 查看用于创建和删除 FlexClone 文件和 FlexClone LUN 的容量 [69](#)
    - 分摊负载如何确定用于创建和删除 FlexClone 文件和 FlexClone LUN 的容量 [67](#)
    - 将卷移动到, 在同一个 SVM 中 [54](#)
  - 节点根聚合
    - 需要遵循的规则 [47](#)
  - 节点根卷
    - 需要遵循的规则 [47](#)
  - 解除
    - 卷效率策略 [133](#)
  - 精简配置
    - 如何使用 FlexVol 卷 [19](#)
    - 与 FlexVol 卷结合使用的注意事项 [22](#)
  - 警报
    - 聚合的填满度和分配过度, 如何解决 [30](#)
    - 卷的填满度和分配过度, 如何解决 [28](#)
  - 镜像
    - 负载共享, 比较 [10](#)
    - 数据保护, 比较 [10](#)
  - 聚合
    - SVM 对选择的影响 [16](#)
    - 创建空间的方法 [44](#)
    - 分配过度警报, 如何解决 [30](#)
    - 复制卷的方法 [55](#)
    - 将卷移动到, 在同一个 SVM 中 [54](#)
    - 节点根卷需要遵循的规则 [47](#)
    - 如何确定空间使用量 [32, 40](#)
    - 填满度警报, 如何解决 [30](#)
    - 移动 Flash Pool 中的卷时的注意事项和建议 [52](#)
    - 用于显示空间使用量信息的命令 [50](#)
    - 在 FlexVol 卷和无限卷之间共享 [14](#)
  - 聚合过量
    - 与 FlexVol 卷结合使用的注意事项 [22](#)
  - 卷 SnapMirror
    - 它如何与 FlexClone 文件和 FlexClone LUN 配合使用 [78](#)
  - 卷 SnapMirror 与重复数据删除 [142](#)
  - 卷保证
    - FlexVol [20](#)
    - 对最大 FlexVol 卷大小的影响 [19](#)
    - 如何使用 FlexVol 卷 [19](#)
  - 卷副本
    - 比较 [10](#)
  - 卷联合
    - 使用规则 [17](#)
    - 已定义 [17](#)
  - 卷效率策略

- 查看 132
- 创建 131
- 解除与卷的关联 133
- 删除 133
- 修改 132

- 卷效率优先级
  - 使用卷效率优先级确定效率操作的优先顺序 130
- 卷移动
  - 不允许执行重复数据删除操作 144
- 卷占用空间
  - 已说明 39

## K

- 可用空间
  - FlexVol 卷如何从 FlexClone 文件和 FlexClone LUN 回收 71
  - 将已满的 FlexVol 卷配置为自动提供 24
- 空间
  - 方法, 在 FlexVol 卷中 42
  - 将已满的 FlexVol 卷配置为自动提供 24
  - 用于显示使用量信息的命令 50
  - 在聚合中创建的方法 44

- 空间保证
  - 请参阅卷保证
- 空间不足错误
  - 可能预留百分比为零 23

- 空间管理
  - 如何使用 18
- 空间使用量
  - 确定卷或聚合中的空间使用量的命令 32
  - 如何确定 33
  - 如何确定和控制卷, 在聚合中 37
  - 如何确定聚合中的 40
  - 如何确定卷或聚合中的 32

- 空间使用量命令
  - 何时不应使用 df 命令 42
- 空间预留
  - 它如何与 FlexClone 文件和 FlexClone LUN 配合使用 78
  - 另请参阅预留

## L

- ls 命令
  - 如何计算磁盘空间使用量 109
- LUN
  - 防止单点故障 54
  - 每个集群的上限 152

- 每个节点和卷的上限 152
- LUN 预留
  - 工作原理 19
- 了解 130
- 联合
  - 使用规则 17
  - 已定义 17
- 令牌
  - 用于创建 FlexClone 文件和 FlexClone LUN 67
- 逻辑存储
  - 已定义 8

## M

- mapping
  - 配额中的用户名 97
- maxfiles
  - 更改注意事项 45
- MetroCluster 配置
  - 移动卷时的注意事项和建议 52
  - 与重复数据删除配合使用 146
- 名称限制
  - qtree 80
- 默认配额
  - 工作原理 87
- 命令
  - FlexVol 卷管理 50
  - storage aggregate 50
  - volume file clone deletion 75
  - volume show-footprint 50
  - volume show-space 50
  - volume snapshot 50
  - 按扩展名和最小大小删除 FlexClone 文件 75
  - 何时应使用 df 命令而不应使用空间使用量命令 42
  - 确定卷或聚合中的空间使用量 32
  - 用于管理卷移动 52
  - 用于显示 FlexVol 卷空间使用量信息 50
  - 用于显示聚合空间使用量信息 50
- 默认 qtree
  - 已定义 78
- 默认用户配额
  - 影响 qtree 的配额 99
- 目录
  - 使用 UNIX 转换为 qtree 82
  - 使用 Windows 转换为 qtree 81
  - 转换为 qtree 81
- 目录大小
  - 增加最大目录大小时的注意事项 45
- 目录容量

更改时的注意事项简介 45

## N

### NDMP

它如何与 FlexClone 文件和 FlexClone LUN 配合使用 77

## O

### OnCommand Unified Manager 服务器

它如何与重复数据删除配合使用 143

## P

### 配额的 UNIX 用户

如何指定它们 93

### 配额的 Windows 用户

如何指定它们 93

### 派生配额

根据默认用户配额和组配额创建 94

### 派生树配额

关于 98

### 配额

安全模式更改 100

重新初始化 103, 119

default 87

FlexClone 文件和 FlexClone LUN 77

qtree 重命名 100

qtree 删除, 以及 100

root 用户和 95

如何对 qtree 使用配额 97

树 97

SNMP 陷阱 85

为什么使用 83

用户和组, 使用 qtree 98

策略 86

调整大小 122

调整大小, 何时可以使用 102

多个用户 97

跟踪 90

工作原理 84

关于 83

规则 86

激活 122

链接 UNIX 名称和 Windows 名称 97

派生 90

配置的 106

强制实施的 86, 106

确定用户 ID 96

如何激活 101

软 84

示例 111

它们如何与特殊 Windows 组配合使用 95

停用 122

通知 85

显示 108

显示信息 104

修改 122

验证升级状态 120

硬 84

用户映射 97

有多个 ID 的用户 96

阈值 84

正在初始化 122

### 配额报告

何时使用 108

它们如何统计已用空间 109

限制对文件的写入 106

要查看正在执行的配额 104

配额报告与 UNIX 客户端之间的差别 108

### 配额策略

创建 121

分配 121

复制 121

删除 121

显示 121

重命名 121

### 配额规则

创建 121

更改 121

删除 121

显示 121

### 配额目标

与配额类型的关联 86

### 配额限制

顺序 92

### 配置文件

节点根卷和根聚合需要遵循的规则 47

## Q

### qtree

重命名, 配额 100

何时使用 79

名称限制 80

删除, 配额以及 100

对 FlexVol 卷进行分区的简介 78

将目录转换, 使用 UNIX 82

将目录转换为 81

将目录转换为, 使用 Windows 81

每个卷的上限 152

默认 qtree, 已定义 78

与 FlexVol 卷相比 79

qtree 接合路径

获取 80

qtree 命名空间路径

获取 80

qtree0

已定义 78

强制实施的配额 106

确定

FlexClone 卷使用的空间 62

## S

树配额 97

SMB 客户端

使用机会锁提高性能 15

SnapMirror

何时可以用作复制聚合卷的方法 55

SnapMirror 卷

FlexClone 卷的注意事项 58

SnapRestore

使用重复数据删除 143

Snapshot 副本

FlexClone 文件和 FlexClone LUN 如何与 76

了解 Snapshot 溢出 41

了解 Snapshot 预留 41

每个卷的上限 152

如何使用卷中的空间 41

同时启用自动缩减和 Snapshot 副本自动删除的要求 27

自动删除 26

自动缩减功能与 Snapshot 副本自动删除功能的交互 28

Snapshot 溢出

已定义 41

Snapshot 预留

了解已用空间和未使用空间 41

Snapshot 预留空间

用于显示大小的命令 50

SnapVault 备份

它们如何与重复数据删除配合使用 145

storage aggregate 命令

用于显示空间信息 50

SVM

volume quota policy show 107

对 FlexVol 卷语言产生的影响 17

对聚合选择的影响 16

分配配额策略 121

如何限制 FlexVol 卷数 16

设置配额 117

它们如何与 FlexVol 卷配合使用 16

显示已分配的配额策略 107, 121

在同一个 SVM 中移动 54

删除

FlexClone 文件和 LUN 自动 72

FlexClone 文件和 LUN, 方法 70

FlexVol 卷 48

为 FlexClone 文件和 FlexClone LUN 禁用自动删除 74

以速度较快的方法删除 FlexClone 文件 75

自动删除 Snapshot 副本 26

使用情况

显示文件或索引节点 49

数据保护 FlexClone 卷

创建 59

它们是指什么 59

数据访问

安全模式的影响简介 14

数据压缩

Flash Pool 聚合如何与 150

SnapVault 备份 150

不可压缩数据

检测以节省系统资源 128

查看进度 138

查看空间节省 138

查看统计信息 139

查看状态 138

撤消空间节省 140

磁带备份如何与 148

对现有数据运行 136

工作原理 127

管理操作 137

基于新数据量运行 137

监控操作 138

检测不可压缩数据以节省系统资源 128

检查点 135

禁用 129

卷效率优先级 130

零预留百分比的影响 23

启用 128

使用卷效率优先级确定效率操作的优先顺序 130

手动管理操作 134

手动启动 134

提高存储效率 127

- 停止操作 [140](#)
- 修改计划 [137](#)
- 与 Data ONTAP 功能的互操作性 [146](#)
- 与 HA 对的互操作性 [149](#)
- 与闪存卡的互操作性 [149](#)
- 预留百分比如何与 [147](#)
- 运行, 使用策略 [131](#)
- 运行, 使用检查点 [135](#)
- 数据压缩的互操作性
  - 使用重复数据删除 [144, 149](#)
  - 与 DataMotion for Volumes [150](#)
  - 与 FlexClone LUN [149](#)
  - 与 FlexClone 卷 [149](#)
  - 与 FlexClone 文件 [149](#)
  - 与 Snapshot 副本 [147](#)
  - 与单文件 SnapRestore [148](#)
  - 与基于卷的 SnapRestore [148](#)
  - 与卷 SnapMirror [147](#)
- 缩减 FlexVol 卷大小
  - 自动缩减功能与 Snapshot 副本自动删除功能交互 [28](#)
- 索引节点
  - 显示使用量 [49](#)

## U

### UNIX

- 将目录转换为 qtree [82](#)

## V

- volume quota policy rule show 命令
  - 何时使用 [108](#)
- volume quota policy rule 命令
  - volume quota policy rule create [121](#)
  - volume quota policy rule delete [121](#)
  - volume quota policy rule modify [121](#)
  - volume quota policy rule show [121](#)
- volume quota policy 命令
  - volume quota policy copy [121](#)
  - volume quota policy create [121](#)
  - volume quota policy delete [121](#)
  - volume quota policy rename [121](#)
  - volume quota policy show [121](#)
- volume quota report 命令
  - 何时使用 [108](#)
- volume quota 命令
  - volume quota modify [122](#)
  - volume quota off [122](#)

- volume quota on [122](#)
- volume quota policy [107](#)
- volume quota report [107](#)
- volume quota resize [122](#)
- volume quota show [107](#)
- volume show-footprint 命令
  - 了解输出 [37](#)
- volume show-space 命令
  - 用于确定卷的空间使用量 [33](#)
- volume 命令
  - 用于显示空间信息 [50](#)
- Vserver
  - 请参阅 SVM

## W

### 文档

- 如何发送反馈 [156](#)

### Windows

- 将目录转换为 qtree [81](#)

### 无限卷

- 定义 [11](#)
- 功能 [11](#)
- 了解 Snapshot 副本使用的空间 [41](#)
- 如何确定成分卷的空间使用量 [37](#)
- 如何确定空间使用量 [33](#)
- 与 FlexVol 卷相比 [12](#)

### 文件

- 每个卷的上限 [152](#)
- 显示使用量 [49](#)
- 允许的最大文件数, 注意事项 [45](#)
- 最大大小 [152](#)

### 文件和 LUN

- 创建可节省空间的副本 [63](#)
- 另请参阅 FlexClone 文件和 FlexClone LUN

### 文件容量

- 更改时的注意事项简介 [45](#)

### 文件预留

- 工作原理 [19](#)

## X

### 限制

- FlexClone 文件和 LUN 存储 [152](#)
- qtree 名称 [80](#)
- RAID 组存储和大小 [152](#)
- 聚合存储 [152](#)
- 卷存储 [152](#)

### 信息

如何发送有关提高文档质量的反馈 [156](#)

## 系统卷

定义 [15](#)

## 显示

索引节点或文件使用量 [49](#)

## 显式配额

如何使用 [89](#)

## 效率操作

基于阈值计划 [130](#)

计划为在特定时间启动 [130](#)

通过创建作业计划 [130](#)

用于管理的策略 [130](#)

## 性能

使用机会锁提高 SMB 客户端性能 [15](#)

重复数据删除, 影响因素 [125](#)

## 虚拟机对齐

工作原理 [145](#)

与重复数据删除配合使用 [145](#)

## Y

### 意见

如何发送文档反馈 [156](#)

### 移动

卷 [54](#)

### 已用空间

了解, Snapshot 预留 [41](#)

如何确定 [33](#)

如何确定和控制聚合, 卷 [37](#)

如何确定聚合中的 [40](#)

如何确定卷或聚合中的 [32](#)

### 用户名

mapping [97](#)

### 用户映射

配额 [97](#)

### 语言

SVM 对 FlexVol 卷有哪些影响 [17](#)

### 预定义的效率策略 [130](#)

### 预留

克隆文件和 LUN 的示例 [66](#)

它们如何用于 LUN [19](#)

### 预留百分比

FlexVol 卷设置的注意事项 [23](#)

与经过空间优化的 FlexClone 文件和 FlexClone LUN 的关系 [66](#)

与数据压缩配合使用 [147](#)

与重复数据删除配合使用 [142](#)

## Z

### 占用空间

卷, 已说明 [39](#)

### 支持的功能

与 FlexClone 文件和 FlexClone LUN [75](#)

### 重复数据删除

MetroCluster 切回操作 [146](#)

SnapVault 备份如何与 [145](#)

提高存储效率 [123](#)

查看进度 [138](#)

查看空间节省 [138](#)

查看统计信息 [139](#)

查看状态 [138](#)

撤消空间节省 [140](#)

对现有数据运行 [136](#)

工作原理 [123](#)

管理操作 [137](#)

和 HA 对 [144](#)

会影响性能的因素 [125](#)

基于新数据量运行 [137](#)

监控操作 [138](#)

检查点 [135](#)

禁用 [127](#)

卷效率优先级 [130](#)

零预留百分比的影响 [23](#)

启用 [126](#)

使用卷效率优先级确定效率操作的优先顺序 [130](#)

手动管理操作 [134](#)

手动启动 [134](#)

它如何与 FlexClone 文件和 FlexClone LUN 配合使用 [76](#)

它如何与 MetroCluster 配置配合使用 [146](#)

它如何与 OnCommand Unified Manager 服务器配合使用 [143](#)

它如何与虚拟机对齐配合使用 [145](#)

停止操作 [140](#)

修改计划 [137](#)

与 Data ONTAP 功能的互操作性 [141](#)

与 FlexClone 卷配合使用 [144](#)

与 SnapRestore [143](#)

与 Snapshot 副本 [142](#)

与卷 SnapMirror [142](#)

预留百分比如何与 [142](#)

元数据已重新定位 [124](#)

运行, 使用策略 [131](#)

运行, 使用检查点 [135](#)

准则 [125](#)

重复数据删除与 SnapRestore [143](#)

注意事项

- 处理 FlexClone 文件和 FlexClone LUN 时 [65](#)
- 准则
  - 运行重复数据删除 [125](#)
- 自动更改大小
  - 配置 FlexVol 卷 [27](#)
- 自动删除
  - FlexClone 文件和 FlexClone LUN [72](#)
  - Snapshot 副本 [26](#)
  - 为 FlexClone 文件和 FlexClone LUN 禁用自动删除 [74](#)
- 自动缩减
  - 配置 FlexVol 卷大小 [27](#)
  - 与 Snapshot 副本自动删除结合使用的要求 [27](#)
- 自动缩减功能
  - 与卷 Snapshot 副本自动删除功能交互 [28](#)
- 租用机会锁
  - 提高 SMB 客户端性能 [15](#)
- 最大目录大小
  - 注意事项 [45](#)