



# 在 **SnapDrive for UNIX** 中创建和使用 **Snapshot** 副本

## Snapdrive for Unix

NetApp  
October 04, 2023

This PDF was generated from [https://docs.netapp.com/zh-cn/snapdrive-unix/aix/concept\\_considerations\\_when\\_working\\_with\\_snapshot\\_copies.html](https://docs.netapp.com/zh-cn/snapdrive-unix/aix/concept_considerations_when_working_with_snapshot_copies.html) on October 04, 2023. Always check docs.netapp.com for the latest.

# 目录

在 SnapDrive for UNIX 中创建和使用 Snapshot 副本 .....	1
SnapDrive for UNIX 中的 Snapshot 操作是什么 .....	1
Snapshot 副本操作 .....	2
此时将显示 Snapshot 副本信息 .....	8
Snapshot 副本重命名 .....	13
还原 Snapshot 副本 .....	15
连接到 Snapshot 副本 .....	25
拆分卷克隆或 LUN 克隆 .....	36
删除 Snapshot 副本 .....	45
断开 Snapshot 副本的连接 .....	47

# 在 SnapDrive for UNIX 中创建和使用 Snapshot 副本

通过 SnapDrive for UNIX，您可以为主机数据创建 Snapshot 副本。

## SnapDrive for UNIX 中的 Snapshot 操作是什么

通过 SnapDrive for UNIX，您可以使用 Data ONTAP Snapshot 技术为存储系统上存储的主机数据创建映像（Snapshot 副本）。

此 Snapshot 副本可为您提供一份数据副本，您可以稍后还原这些数据。Snapshot 副本中的数据可以位于一个存储系统上，也可以跨越多个存储系统及其卷。这些存储系统可以位于 HA 对或节点本地文件系统中，也可以位于主机集群环境中的磁盘组或 LUN 中。在安装了 SnapDrive for UNIX 的非集群模式 UNIX 主机上，您可以为存储系统上的一个或多个卷组创建 Snapshot 副本。

Snapshot 副本可以包含文件系统，逻辑卷，磁盘组，LUN 和 NFS 目录树。创建 Snapshot 副本后，您可以对其进行重命名，还原或删除。您也可以连接到同一主机上的其他位置或其他主机。连接后，您可以查看和修改 Snapshot 副本的内容，也可以断开 Snapshot 副本的连接。此外，您还可以通过 SnapDrive for UNIX 显示有关已创建的 Snapshot 副本的信息。在安装了 SnapDrive for UNIX 的集群模式 UNIX 主机上，您可以对包含磁盘组和文件系统的主机集群节点执行 Snapshot 操作。Snapshot 操作包括创建，重命名，还原，连接，断开连接，显示和删除。

### 使用 Snapshot 副本时的注意事项

在使用 Snapshot 副本时，应牢记一些注意事项，例如 SnapDrive for UNIX 仅适用于其创建的 Snapshot 副本，或者 Snapshot 副本自动从存储系统复制到主机等。

使用 Snapshot 操作时，应考虑以下事项：

- SnapDrive for UNIX 仅适用于其创建的 Snapshot 副本。

它无法还原未创建的 Snapshot 副本。

- 在目标存储系统上创建 Snapshot 副本时，Snapshot 副本会自动从创建该副本的源存储系统复制到目标存储系统。

通过 SnapDrive for UNIX，您还可以还原目标存储系统上的 Snapshot 副本。

- 如果您使用 SnapDrive snap connect` 命令在上次连接（或仍连接）的主机上的新位置连接到 Snapshot 副本，则会连接到源主机。
- 对跨多个存储系统卷或多个存储系统的存储实体的 Snapshot 支持仅限于不允许在软件堆栈中执行冻结操作的配置。
- 通过 NFS 协议导出卷时，必须将匿名用户 ID 选项设置为 0，SnapDrive for UNIX 命令才能正常运行。
- 如果一个磁盘组中有两个 JFS 文件系统（JFS 和 JFS2），则 SnapDrive for UNIX 仅支持对 JFS2 文件系统执行操作。
- 通过 SnapDrive for UNIX，您可以创建并发卷组的 Snapshot 副本，以及使用相同属性克隆或还原。

- 在创建 Snapshot 副本后立即执行单文件 SnapRestore （SFSR）操作失败。

必须在经过一段时间后重试此操作。有关详细信息，请参见 "《[集群模式 Data ONTAP 逻辑存储管理指南](#)》"。

## Snapshot 副本操作

您可以使用 `SnapDrive snap create`` 命令创建 Snapshot 副本。

### 崩溃状态一致的 Snapshot 副本

您可能需要为文件系统或磁盘组创建崩溃状态一致的 Snapshot 副本。SnapDrive for UNIX 会创建 Snapshot 副本，其中包含实体中指定的所有存储系统卷的映像。

在为文件系统或磁盘组等存储实体创建 Snapshot 副本时，SnapDrive for UNIX 会创建一个 Snapshot 副本，其中包含使用 `file_specs` 参数构成您指定实体的所有存储系统卷的映像。`file_specs` 参数指定存储实体，例如 SnapDrive for UNIX 用于创建 Snapshot 副本的文件系统，LUN 或 NFS 目录树。

SnapDrive for UNIX 可提供一致的存储组件，用于构成您在 Snapshot 副本中请求的实体。这意味着，在 `SnapDrive snap create`` 命令 `file_specs` 参数指定的 LUN 或目录之外使用的 LUN 或目录可能在 Snapshot 副本中没有一致的映像。通过 SnapDrive for UNIX，您可以仅还原由 `file_specs` 参数指定且在 Snapshot 副本中保持一致的实体。

单个存储系统卷上包含的实体的 Snapshot 副本始终保持崩溃状态一致。SnapDrive for UNIX 会采取特殊步骤，以确保跨多个存储系统或存储系统卷的 Snapshot 副本也保持崩溃状态一致。SnapDrive for UNIX 用于确保崩溃一致性的方法取决于 Snapshot 副本中存储实体所在的 Data ONTAP 版本。

#### 与 Data ONTAP 7.2 及更高版本的崩溃一致性

SnapDrive for UNIX 会使用 Data ONTAP 7.2 及更高版本提供的一致性组支持，以便跨多个卷的所有 Snapshot 副本都保持崩溃状态一致。

Data ONTAP 7.2 及更高版本支持一致性组和存储系统隔离。SnapDrive for UNIX 使用这些功能来确保跨多个卷的所有 Snapshot 副本都保持崩溃状态一致。

要在多个卷之间创建崩溃状态一致的 Snapshot 副本，SnapDrive for UNIX 将执行以下操作：

- 将 I/O 隔离（冻结）到包含存储实体的每个卷。
- 为每个卷创建一个 Snapshot 副本。

隔离卷和创建 Snapshot 副本所需的时间有限，并且由 Data ONTAP 控制。

在 `snapdrive.conf` 文件中的 ``* snapcreate-cg-timeout*`` 参数指定在 Data ONTAP 限制范围内您希望允许存储系统隔离的时间量。您可以指定紧急，中等或宽松的间隔。如果存储系统所需时间超过完成隔离操作所允许的时间，则 SnapDrive for UNIX 将使用先前 Data ONTAP 7.2 版本的一致性方法创建 Snapshot 副本。您也可以在创建 Snapshot 副本时使用 ``-nodirfence`` 选项来指定此方法。

如果您请求的存储实体跨 Data ONTAP 7.2 及先前 Data ONTAP 版本的存储系统创建 Snapshot 副本，则 SnapDrive for UNIX 还会对 7.2 之前的 Data ONTAP 版本使用一致性方法创建 Snapshot 副本。

## SnapDrive for UNIX 中的一致性组 Snapshot 副本

一致性组 Snapshot 是一组卷的 Snapshot 副本，该卷跨越集群中的不同 Vserver 或节点，并作为一个实体进行管理。

SnapDrive for UNIX 可捕获跨集群中不同 SVM 或节点的所有卷的崩溃状态一致的 Snapshot 副本。您还可以配置捕获 Snapshot 副本的时间段。

默认情况下，SnapDrive for UNIX 会捕获一致性组 Snapshot 副本。您可以禁用此功能并还原为在尽力服务模式捕获 Snapshot 副本。



SnapDrive for UNIX 5.2 仅在 Data ONTAP 8.2 或更高版本中支持集群模式 Data ONTAP 的一致性组 Snapshot 副本。

### • 相关信息 \*

[创建一致性组 Snapshot 副本](#)

[禁用一致性组 Snapshot 副本](#)

## 创建一致性组 Snapshot 副本

您可以将 SnapDrive for UNIX 配置为创建一致性组 Snapshot 副本。

### 步骤

1. 在主机上输入以下命令：

```
` * SnapDrive snap create -fs /mnt/test -snapname snapshotName -f -noprompts*`
```

` *snapshotName* ` 是为一致性组 Snapshot 副本指定的名称。

### 示例

以下是命令示例：

```
snapdrive snap create -fs /mnt/test -snapname snap_123 -f -noprompt
```

已成功创建一致性组 Snapshot 副本。

## 禁用一致性组 Snapshot 副本

您可以将 SnapDrive for UNIX 配置为禁用一致性组 Snapshot 副本。

### 步骤

1. 输入 ...

```
` * SnapDrive snap create -fs /mnt/fs3 -snapname nfs_snap -nodfiference*`
```

已成功禁用一致性组 Snapshot 副本。

## 应用程序一致的 Snapshot 副本

要创建应用程序一致的 Snapshot 副本，应在执行 Snapshot 操作之前暂停应用程序。

要确保 Snapshot 副本与应用程序一致，您可能需要在创建 Snapshot 副本之前停止应用程序或执行必要的步骤以使其暂停。请注意，数据库热备份工具取决于 DBMS 使用的方法，并不总是暂停数据库文件的 I/O。

如果应用程序尚未完成其事务并将数据写入存储系统，则生成的 Snapshot 副本可能与应用程序不一致。



如果应用程序可以从崩溃状态一致的 Snapshot 副本中恢复，则无需停止它。请参考适用于您的应用程序的文档。有关创建应用程序一致的 Snapshot 副本的详细信息，请参见。

无论何时添加或删除主机卷，LUN 或 NFS 目录树，或者调整主机卷或文件系统的大小，都应创建新的 Snapshot 副本。这样可以确保新配置的磁盘组具有一致的副本，您可以在需要还原磁盘组时使用该副本。

## 跨存储系统或卷的 Snapshot 副本

通过 SnapDrive for UNIX，您可以创建驻留在相同或不同存储系统上的多个存储系统卷中的 Snapshot 副本。

SnapDrive for UNIX 允许您创建跨多个存储系统卷或多个存储系统的 Snapshot 副本。这些卷可以位于同一个存储系统或不同的存储系统上。尽管 SnapDrive snap create` 命令会为构成您请求的实体的所有卷创建一个 Snapshot 副本，但 SnapDrive for UNIX 仅会还原您在 SnapDrive snap create` 命令中指定的实体。

使用 SnapDrive snap create` 命令创建跨多个卷的 Snapshot 副本时，无需在命令提示符处为卷命名。SnapDrive for UNIX 可从您指定的 file\_specs 参数获取此信息。

- 如果您输入的 file\_specs 请求磁盘组，文件系统或主机卷驻留在磁盘组上，则 SnapDrive for UNIX 会自动创建一个 Snapshot 副本，其中包含您指定的磁盘组，卷或文件系统的所有存储系统卷。
- 如果您输入的 file\_specs 请求 LUN，则 SnapDrive for UNIX 将为包含该 LUN 的存储系统卷创建一个 Snapshot 副本。
- 如果您输入的 file\_specs 请求的文件系统直接驻留在 LUN 上，则 SnapDrive for UNIX 将为包含您指定的 LUN 和文件系统的存储系统卷创建一个 Snapshot 副本。
- 如果您输入的 file\_specs 请求 NFS 目录，则 SnapDrive for UNIX 将为包含 NFS 目录树的卷创建一个 Snapshot 副本。

除了使用基于多个存储系统和存储系统卷中的实体构建的 file\_specs 参数之外，您还可以组合使用 file\_specs 参数，其中每个值都基于单个存储系统或存储系统卷。例如，假设您设置了一个磁盘组 dg1 跨越存储系统存储系统 2 和存储系统 3，DG2 位于存储系统 2 上，dg3 位于存储系统 3 上。在这种情况下，以下任一命令都是正确的：

```
SnapDrive snap create -dg dg1 -snapname snapdg1`
```

```
SnapDrive snap create -dg dg2 dg3 -snapname snapdg23`
```

```
SnapDrive snap create -dg dg1 dg2 dg3 -snapname snapdg123`
```

在创建跨存储系统和卷的 Snapshot 副本时，需要注意的一点是，SnapDrive for UNIX 会使用短名称在每个存储系统卷上创建 Snapshot 副本。它不会在名称中包含完整路径名称，即使卷位于不同的存储系统上也是如此。这意味着，如果稍后重命名 Snapshot 副本，则必须转到每个存储系统和卷并在其中重命名它。

## 创建不相关实体的 **Snapshot** 副本

SnapDrive for UNIX 通过维护各个崩溃状态一致的 Snapshot 副本来创建不相关实体的 Snapshot 副本。

除非您另有指定，否则 SnapDrive for UNIX 会假定您在特定 SnapDrive snap create 命令行上指定的所有实体都是相关的；换言之，对一个实体的更新是否有效取决于对指定的其他实体的更新。如果存储实体以这种方式进行依赖写入，则 SnapDrive for UNIX 将采取步骤为所有存储实体创建一个组中崩溃状态一致的 Snapshot 副本。

以下示例显示了 SnapDrive for UNIX 如何为可能具有依赖写入的存储实体创建 Snapshot 副本。在以下示例中，SnapDrive snap create 命令用于指定 LUN 上的文件系统以及磁盘组。磁盘组由驻留在单个存储系统上的 LUN 组成（请参见创建 Snapshot 副本）。LUN 上的文件系统位于不同的存储系统和存储系统卷上。作为一个组，文件系统和磁盘组跨越多个存储系统卷，而每个卷不会跨越多个存储系统卷。

以下命令指定同时包含文件系统 /mnt/fs1 和磁盘组 dg1 的 Snapshot 副本：SnapDrive snap create -fs /mnt/fs1 -dg dg1 -snapname fs1\_dg1`

由于这些存储实体可能具有依赖写入，因此 SnapDrive for UNIX 会尝试创建崩溃状态一致的 Snapshot 副本，并将文件系统 /mnt/fs1 和磁盘组 dg1 视为一个组。这意味着，在创建 Snapshot 副本之前，需要使用 SnapDrive for UNIX 冻结对存储系统卷的 I/O 操作。

为卷中的多个存储实体创建崩溃状态一致的 Snapshot 副本需要额外的时间，如果 SnapDrive for UNIX 无法冻结 I/O 操作，则此操作并非始终可行。因此，SnapDrive for UNIX 允许您为不相关的存储实体创建 Snapshot 副本。不相关的存储实体是指在创建 Snapshot 副本时您可以指定的不具有依赖写入的实体。由于这些实体没有依赖写入，因此 SnapDrive for UNIX 不会采取措施使这些实体作为一个组保持一致。相反，SnapDrive for UNIX 会创建一个 Snapshot 副本，其中每个存储实体的崩溃状态一致。

以下命令用于指定 LUN 上文件系统的 Snapshot 副本以及前面所述的磁盘组。由于指定了 -unrelated 选项，因此 SnapDrive for UNIX 会创建一个 Snapshot 副本，其中文件系统 /mnt/fs1 和磁盘组 dg1 作为单个存储实体崩溃状态一致，但不会被视为组。以下命令不要求 SnapDrive for UNIX 冻结存储系统卷上的 I/O 操作：  
SnapDrive snap create -fs /mnt/fs1 -dg dg1 -unrelated -snapname fs1\_dg1`

- 相关信息 \*

## 创建 Snapshot 副本

### 创建 **Snapshot** 副本的准则

使用 SnapDrive for UNIX 创建 Snapshot 副本时，请考虑以下事项：每个卷最多可以保留 255 个 Snapshot 副本，SnapDrive for UNIX 仅支持其创建的 Snapshot 副本，无法创建根磁盘组的 Snapshot 副本以及启动设备或交换设备，SnapDrive for UNIX 需要执行冻结操作以保持崩溃一致性。

输入用于创建 Snapshot 副本的命令时，请遵循以下准则：

- 每个存储系统卷最多可以保留 255 个 Snapshot 副本。此限制由存储系统设置。总数可能因其他工具是否使用这些 Snapshot 副本而异。

当 Snapshot 副本数量达到最大限制时，Snapshot 创建操作将失败。您必须先删除一些旧的 Snapshot 副本，然后才能使用 SnapDrive for UNIX 再使用。

- SnapDrive for UNIX 不支持不创建的 Snapshot 副本。例如，它不支持从存储系统控制台创建的 Snapshot 副本，因为这种做法可能会导致文件系统内部不一致。
- 不能使用 SnapDrive for UNIX 创建以下 Snapshot 副本：
  - 根磁盘组

当您尝试为 LVM 创建根磁盘组的 Snapshot 副本时，Snapshot 创建操作将失败。

- 启动设备或交换设备


SnapDrive for UNIX 不会为系统启动设备或系统交换设备创建 Snapshot 副本。

- 当 Snapshot 副本跨越多个存储系统或存储系统卷时，SnapDrive for UNIX 需要执行冻结操作以确保崩溃一致性。有关在未提供冻结操作的配置上创建 Snapshot 副本的信息。

使用 **SnapDrive snap create** 命令所需的信息

创建 Snapshot 副本时，应确定要捕获的存储实体并指定 Snapshot 副本的名称。

下表提供了使用 SnapDrive snap create` 命令时所需的信息：

要求 / 选项	参数
<p>确定要在 Snapshot 副本中捕获的存储实体的类型。您可以通过一个命令行指定直接在 LUN 上创建的 NFS 实体，LUN，文件系统以及 LVM 实体。</p> <p>请使用相应的参数提供该实体的名称。这是 <code>file_specs</code> 参数的值。</p> <ul style="list-style-type: none"><li>• 如果您指定的磁盘组具有主机卷或文件规范，则参数将转换为存储系统上的一组磁盘组。SnapDrive for UNIX 会创建包含此实体的整个磁盘组，即使此实体是主机卷或文件系统也是如此。</li><li>• 如果指定的文件规范是 NFS 挂载点，则参数将转换为存储系统卷上的目录树。</li><li>• 如果指定 LUN 或具有文件系统的 LUN，则参数将转换为 LUN 或具有文件系统的 LUN。</li></ul> <div><div></div><div>您不能使用存储实体指定特殊字符，例如 <code>`-vg`</code>，<code>`-DG`</code>，<code>`-FS`</code>，<code>`-lvol`</code> 和 <code>`-hostvol`</code>。但是，<code>`-FS`</code>，<code>`-lvol`</code> 和 <code>`-hostvol`</code> 等存储实体允许使用斜线（/）。</div></div>	<p>LUN (<code>`-lun file_specs`</code>)</p>



要求 / 选项	参数
LUN 的名称  您必须包含存储系统，卷和 LUN 的名称。	磁盘组 ( <code>`-dg file_specs`</code> ) 或卷组 ( <code>`-vg file_specs`</code> )
磁盘或卷组的名称	文件系统 ( <code>`-fs file_specs`</code> )
文件系统名称	主机卷 ( <code>`-hostvol file_specs`</code> ) 或逻辑卷 ( <code>`-lvol file_specs`</code> )
主机或逻辑卷的名称  <div>  <div>             您必须同时提供请求的卷及其所在的磁盘组；例如， <code>`-hostvol dg3/Acct_volume`</code> 。           </div> </div>	Snapshot 副本名称 ( <code>`-snapname snap_name`</code> )
Snapshot copy_name	指定 Snapshot 副本的名称。此名称可以是名称的较长版本，其中包含具有 Snapshot 副本名称的存储系统和卷，也可以是 Snapshot 副本名称的较短版本。  <div>  <div>             您必须确保为 snapname 指定的值不超过 200 个字符。           </div> </div>
<code>` 无关`</code>	~
可选：要在创建 Snapshot 副本时为不具有依赖写入的存储实体创建 Snapshot 副本， SnapDrive for UNIX 会为各个存储实体创建崩溃状态一致的 Snapshot 副本，但不会尝试使这些实体彼此保持一致。	<code>` 强制`</code>
~	<code>` -noprop`</code>
~	可选：确定是否要覆盖现有 Snapshot 副本。如果没有此选项，则如果您提供现有 Snapshot 副本的名称，此操作将暂停。提供此选项并指定现有 Snapshot 副本的名称时，此命令将提示您确认是否要覆盖此 Snapshot 副本。要防止 SnapDrive for UNIX 显示提示符，请同时包含 <code>` -noprop`</code> 选项。（如果要使用 <code>` -noprop`</code> 选项，则必须始终包含 <code>` -force`</code> 选项。）
<code>` -fstype`</code>	<code>` _ 类型 _`</code>
<code>` -vmtype`</code>	<code>` _ 类型 _`</code>

## 创建 Snapshot 副本

要创建 Snapshot 副本，请运行 `SnapDrive snap create`` 命令。

在执行此语法之前，您必须了解此命令中提及的选项，关键字和参数。

### 步骤

1. 输入以下命令语法以创建 Snapshot 副本：

```
` * SnapDrive snap create { -lun _ -dg _ -fs | -hostvol } file_spec[file_spec...] ( { -lun dg _ -fs _  
-hostvol } _file_spec [file_spec...] -snapname snap_name [ -force [-noprompt]]-unrelated ] [-  
nodfilerfencet] [-fstype type][[-vmttype type]]*`
```

``file_spec`` 参数表示一个或多个存储系统上的一组存储实体。Snapshot 创建操作会为包含这些实体的存储系统卷创建一个 Snapshot 副本，并为其提供在 `snap_name` 参数中指定的名称。

- 相关信息 \*

### 创建不相关实体的 Snapshot 副本

## 此时将显示 Snapshot 副本信息

您可以显示存储系统，存储系统卷，LUN 和其他存储实体的 Snapshot 副本。使用命令 `SnapDrive snap show``（或 `list`）显示 Snapshot 副本。

### 用于显示 Snapshot 副本信息的命令

使用 `SnapDrive snap show``（或 `list`）命令显示有关 Snapshot 副本的信息。

您可以使用 `SnapDrive snap show``（或 `list`）命令显示有关 SnapDrive for UNIX 创建的每个 Snapshot 副本的信息。您可以使用此命令显示有关以下内容的信息：

- 存储系统
- 存储系统上的卷
- 存储实体，例如 NFS 文件和目录树，卷组，磁盘组，文件系统，逻辑卷，和主机卷
- Snapshot 副本



此命令的 `show` 和 `list` 形式是同义词。对于适用于 UNIX 的 SnapDrive 2.0 及更高版本，在显示有关 Snapshot 副本的信息时，必须使用 Snapshot 副本名称的长格式。

### 显示 Snapshot 副本的准则

您可以使用通配符显示 Snapshot 副本。您可以显示特定对象中的所有 Snapshot 副本。

在使用 Snapshot 副本之前，请记住以下几点：

- 您可以在 Snapshot 副本名称中使用通配符（\*）。使用 Snapshot show 操作可以使用通配符显示与特定模

式匹配的所有 Snapshot 副本名称或特定卷上的所有 Snapshot 副本名称。在 Snapshot 副本名称中使用通配符时，以下规则适用：

- 您只能在名称末尾使用通配符。不能在 Snapshot 副本名称的开头或中间使用通配符。
- 您不能在 Snapshot 副本名称的存储系统或存储系统卷字段中使用通配符。
- 您也可以使用此命令列出特定对象上的所有 Snapshot 副本，包括存储系统及其卷，磁盘组，主机卷组，文件系统，主机卷， 和逻辑卷。
- 如果输入 SnapDrive snap show` 命令，而 SnapDrive for UNIX 未找到任何 Snapshot 副本，则会显示消息 "no matching Snapshot copies 。`" 如果您在命令行上指定了参数，但其中某些部分不存在，则 SnapDrive for UNIX 将返回部分 Snapshot 副本列表。它还会列出无效的参数。
- 如果 SnapDrive snap create` 命令突然中止，则存储系统上的卷会存储一个不完整的`.stoc.xml` 文件。因此，存储系统创建的所有计划 Snapshot 副本都将具有不完整的`.stoc.xml` 文件的副本。要使 SnapDrive snap list` 命令成功运行，请完成以下步骤：

步骤

1. 删除卷中不完整的`.stoc.xml` 文件。
2. 删除存储系统创建的计划 Snapshot 副本，其中包含不完整的`.stoc.xml` 文件。

使用 SnapDrive snap show 或 list 命令所需的信息

您可以使用命令 SnapDrive snap show` 或 list 显示有关存储系统，存储系统卷，磁盘或卷组，文件系统， Snapshot 副本等。

下表提供了使用 SnapDrive snap show | list` 命令时需要提供的信息。



无论输入 SnapDrive snap show` 还是 SnapDrive snap list` 作为命令，都可以使用相同的参数。这些命令是同义词。

要求 / 选项	参数
<p>根据您输入的命令，您可以显示有关以下任一项的信息：</p> <ul style="list-style-type: none"><li>• 存储系统</li><li>• 存储系统卷</li><li>• 磁盘或卷组</li><li>• 文件系统</li><li>• 主机或逻辑卷</li><li>• Snapshot 副本</li></ul> <p>为 file_specs 参数输入的值必须标识要显示其信息的存储实体。此命令假定实体位于当前主机上。</p>	存储系统（`存储器`）
`文件名 _`	存储系统上的卷（`-filervol`）

要求 / 选项	参数
` filervol_`	磁盘组 (` -dg file_spec` ) 或卷组 (` -vg file_spec` )
磁盘或卷组的 ` 名称 _`	文件系统 (` -fs file_spec` )
` 文件系统名称 _`	主机卷 (` -hostvol file_spec` ) 或逻辑卷 (` -lvol file_spec` )
主机或逻辑卷的 ` 名称 _`	Snapshot 副本名称 (` -snapname lor_snap_name` )
` lor_snap_name_`	其他 Snapshot 副本名称
` Snap_name_` (长版或短版)	如果要显示有关 Snapshot 副本的信息，请指定 Snapshot 副本的名称。对于第一个 Snapshot 副本 ` lor_snap_name_`，请输入此名称的长版本，其中包括存储系统名称，卷和 Snapshot 副本名称。如果 Snapshot 副本名称位于同一个存储系统和卷上，则可以使用该名称的简短版本。
` - 详细`	~

## 显示存储系统上的 **Snapshot** 副本

您可以使用 SnapDrive 命令显示存储系统上的 Snapshot 副本。

要显示有关驻留在存储系统上的 Snapshot 副本的信息，请使用以下语法：

```
SnapDrive snap show -filer filename [filename...] 【 -verbose 】`
```

## 显示存储系统卷的 **Snapshot** 副本

您可以使用 SnapDrive 命令显示存储系统卷上的 Snapshot 副本。

要显示有关存储系统卷的 Snapshot 副本的信息，请使用以下语法：

```
SnapDrive snap show -filervol filervol [filervol...] 【 -verbose 】`
```

## 显示 **Snapshot** 副本

命令 SnapDrive snap show` 或 list 可显示创建 Snapshot 副本的存储系统的名称，主机名，日期和时间等。

### 步骤

1. 输入以下命令语法以显示 Snapshot 副本：

```
` * SnapDrive snap show ( -snapname] long_snap_name [snap_name...] 【 -verbose 】 *`
```

此操作至少会显示有关 Snapshot 副本的以下信息：

- 创建 Snapshot 副本的存储系统的名称
- 创建 Snapshot 副本的主机的名称
- 存储系统上 LUN 的路径
- 创建 Snapshot 副本的日期和时间
- Snapshot 副本的名称
- Snapshot 副本中包含的磁盘组的名称
- 示例 \*：以下是 SnapDrive snap show` 命令的示例：

```
# snapdrive snap show -snapname toaster:/vol/vol2:snapA snapX snapY

# snapdrive snap show -verbose toaster:/vol/vol2:snapA /vol/vol3:snapB
snapC

# snapdrive snap show toaster:/vol/vol2:snapA

# snapdrive snap list -dg dg1 dg2
```

- 示例 \*：此示例使用通配符显示有关特定卷上可用 Snapshot 副本的信息：

```
# snapdrive snap show toaster:/vol/vol1:*
snap name host date
snapped
-----
-----
toaster:/vol/vol1:sss1 DBserver Mar 12
16:19 dg1
toaster:/vol/vol1:testdg DBserver Mar 12
15:35 dg1
toaster:/vol/vol1:t1 DBserver Mar 10
18:06 dg1
toaster:/vol/vol1:hp_1 HPserver Mar 8
19:01 vg01
toaster:/vol/vol1:r3 DBserver Mar 8
13:39 rdg1
toaster:/vol/vol1:r1 DBserver Mar 8
13:20 rdg1
toaster:/vol/vol1:snap2 DBserver Mar 8
11:51 rdg1toaster:/vol/vol1:snap_p1 DBserver
Mar 8 10:18 rdg1
```

- 示例 \*：在此示例中，AIX 主机的命令行使用了 ` -verbose` 选项：

```
# snapdrive snap list betty:/vol/vol1:testsnap -v
snap name host date
snapped
-----
-----
betty:/vol/vol1:testsnap aix198-75 Jul 31
10:43 test1_SdDg
host OS: AIX 3 5
snapshot name: testsnap
Volume Manager: aixlvm 5.3
disk group: test1_SdDg
host volume: test1_SdHv
file system: test1_SdHv file system type: jfs2
mountpoint:
/mnt/test1
lun path dev paths
-----
betty:/vol/vol1/aix198-75_luntest1_SdLun /dev/hdisk32
```

- 示例 \*：此示例包含有关 AIX 主机上未使用 SnapDrive for UNIX 创建的 Snapshot 副本的消息：

```
# snapdrive snap show -filer toaster
snap name host date
snapped
-----
-----
toaster:/vol/vol1:hourly.0 non-snapdrive snapshot
toaster:/vol/vol1:hourly.0 non-snapdrive snapshot
toaster:/vol/vol1:snap1 DBserver1 Oct 01 13:42 dg1
dg2
toaster:/vol/vol1:snap2 DBserver2 Oct 10 13:40
DBdg/hvol1
toaster:/vol/vol1:snap3 DBserver3 Oct 31 13:45
DBdg
```

此示例使用 SnapDrive snap show` 命令和 ` - verbose` 选项显示 AIX 主机上基于 LVM 的文件系统的 Snapshot 副本：

```
# snapdrive snap show -v -fs /mnt/check_submit/csdg2/hv3_0
snapname host date snapped
-----
-----
toaster:/vol/vol1:mynsnapshot sales-aix Aug 24 10:55 csdg2
host OS: AIX 1 5
snapshot name: mynsnapshot
Volume Manager: aixlvm 5.1
disk group: csdg2
host volume: csdg2_log
host volume: csdg2_hv3_0
file system: csdg2_hv3_0 file system type: jfs2 mountpoint:
/mnt/check_submit/csdg2/hv3_0
lun path dev paths
-----
spinel:/vol/vol1/check_submit_aix-4 /dev/hdisk4
```

- 示例 \*：以下示例使用通配符：

```
# snapdrive snap show toaster:/vol/volX:*
# snapdrive snap show -v toaster:/vol/volX:DB1*
filer1:/vol/volY:DB2*
# snapdrive snap show toaster:/vol/vol2:mynsnap* /vol/vol2:yoursnap*
hersnap*
# snapdrive snap show toaster:/vol/volX:*
```

- 示例 \*：在此示例中，使用通配符无效，因为通配符位于名称中间，而不是置于末尾：

```
# snap show toaster:/vol/vol1:my*snap
```

## 获取 **Snapshot** 副本名称的其他方法

使用 `SnapDrive snap list`` 命令显示 **Snapshot** 副本名称。

另一种获取 **Snapshot** 副本名称的方法是，登录到存储系统并在其中使用 `SnapDrive snap list`` 命令。此命令显示 **Snapshot** 副本的名称。



`SnapDrive snap show`` 命令相当于 `storage system SnapDrive snap list`` 命令。

## **Snapshot** 副本重命名

您可以使用 `snapshot snap rename` 命令更改 **Snapshot** 副本的名称。跨多个存储系统

或存储系统卷的 Snapshot 副本也可以重命名。

## 用于重命名 **Snapshot** 副本的命令

使用 `snapshot snap rename` 命令重命名 Snapshot 副本。

## 重命名跨存储系统或卷的 **Snapshot** 副本

对于跨多个存储系统或存储系统卷的 Snapshot 副本，请重命名所有相关的 Snapshot 副本。

您也可以使用此命令重命名跨多个存储系统或多个存储系统卷的 Snapshot 副本。

如果重命名其中一个 Snapshot 副本，则还必须使用相同名称重命名所有相关的 Snapshot 副本。这是因为 SnapDrive for UNIX 在创建 Snapshot 副本时会使用短名称，即使它跨越多个存储系统或卷也是如此。rename 命令会更改当前 Snapshot 副本的名称，但不会更改其他位置的相关 Snapshot 副本的名称。

## 重命名 **Snapshot** 副本的准则

重命名 Snapshot 副本时，请确保两个 Snapshot 副本的名称不相同。

使用 `SnapDrive snap rename`` 命令时，请遵循以下准则：

- 如果您尝试将 Snapshot 副本重命名为其他存储系统卷，则会出现一条错误消息。
- 如果 Snapshot 副本的新名称已存在，则会出现错误消息。您可以使用 ``-force`` 选项强制 SnapDrive for UNIX 更改名称而不生成错误消息。

## 更改 **Snapshot** 副本名称

您可以使用 `SnapDrive snap rename`` 命令更改 Snapshot 副本的名称。Snapshot 重命名操作会将源 Snapshot 副本的名称更改为目标参数指定的名称。

在执行此语法之前，您必须了解此命令中提及的选项，关键字和参数。

### 步骤

1. 输入以下命令以更改 Snapshot 副本名称：

```
` * SnapDrive snap rename ( -snapname] old_long_snap_name new_snap_name ( -force ( -noprompt] ) *
```

以下是 `SnapDrive snap rename`` 命令的示例。第一个命令行包含 ``-force`` 选项，因为名为 `new snapshot copy` 的 Snapshot 副本已存在。在第二个示例中，两个 Snapshot 副本名称都使用较长的名称形式，但它们都将解析为同一个存储系统卷。

```
snapdrive snap rename -force filer1:/vol/vol1:oldsnap new snapshot
```



```
snapdrive snap rename filer1:/vol/vol1:FridaySnap  
filer1:/vol/vol1:Snap040130
```

## 还原 Snapshot 副本

您可以还原单个存储实体或多个存储实体的 Snapshot 副本。

用于还原 **Snapshot** 副本的命令

使用命令 `SnapDrive snap restore`` 还原 Snapshot 副本。

`SnapDrive snap restore`` 命令可将您在命令提示符处指定的 Snapshot 副本中的数据还原到存储系统。此操作会将您在 `SnapDrive snap restore`` 命令中指定的 `file_specs` 参数（例如，磁盘组，NFS 文件，NFS 目录树，直接在 LUN 上创建的文件系统）的内容替换为指定 Snapshot 副本中的 `file_specs` 参数的内容。

您还可以为不存在的 `file_specs` 参数还原 Snapshot 副本。如果指定的值不再存在于主机上，而是在创建 Snapshot 副本时存在，则会发生这种情况。例如，可能是您现在已卸载的文件系统或已删除的磁盘组。

通常，您可以从创建 Snapshot 副本的主机（即发起主机）还原 Snapshot 副本。



- 在 NFS 实体中，如果同一个卷挂载到两个不同的 IP 上，则一次只能对这些卷的一个文件规范执行快照还原操作。
- 如果您不是 root 用户，则在适用于 UNIX 的 SnapDrive 4.1 中，除了使用 GlobalSDSnapshot 功能可以执行 Snapshot 还原之外，您还应具有全局存储写入功能。

## 在目标存储系统上还原 Snapshot 副本

您可以在创建 Snapshot 副本的存储系统上还原 Snapshot 副本。

在目标存储系统上创建 Snapshot 副本时，Snapshot 副本会自动从创建该副本的源系统复制到目标存储系统。通过 SnapDrive for UNIX，您可以还原源存储系统上的 Snapshot 副本。您还可以在目标存储系统上还原 Snapshot 副本。

如果要在 SnapMirror 关系中执行单个文件快照还原，则源卷名称和目标卷名称不应相同。如果源卷名称和目标卷名称相同，则 SnapDrive for UNIX 将显示以下错误消息：

```
0001-636 Command error: Snapdrive cannot restore LUNs on SnapMirror  
destination filer volumes: <filer-vol-name>
```

在适用于 UNIX 的 SnapDrive 4.0 及更高版本中，如果启用了基于角色的访问控制，则只有在 vFiler 单元具有 Snapshot 还原功能时，才能在 vFiler 单元上执行快照还原。

- 相关信息 \*

[SnapDrive for UNIX 中基于角色的访问控制](#)

## 还原多个存储实体

您可以还原包含多个存储实体的 Snapshot 副本。

要还原包含驻留在多个目标存储系统上的存储实体的 Snapshot 副本，必须满足以下要求：

- 在命令提示符处指定的存储实体必须位于单个存储系统或 HA 对上。
- 源存储系统的卷名称必须与目标存储系统的卷名称匹配。
- 您必须将 `snapdrive.conf` 文件中的 `snapmirror-dest-multily-filervolumes-enabled` 参数设置为 `on`。

您可以使用一个命令还原驻留在单个存储系统或 HA 对上的存储实体。

## 还原 Snapshot 副本的注意事项

在还原 Snapshot 副本之前，请确保您未使用文件系统，并且不会中断还原过程。

在还原 Snapshot 副本之前，请考虑以下重要信息：

- 确保您不在要还原的文件系统上的任何目录中。您可以从任何目录执行 `SnapDrive snap restore`` 命令，但要将信息还原到的文件系统上的目录除外。
- 请勿输入 `Ctrl-C` 中断还原操作这样做可能会使系统处于不可用的配置中。如果发生这种情况，您可能需要联系 NetApp 技术支持才能恢复。
- 将 NFS 实体导出到卷时，请将匿名用户 ID 选项设置为 `"0"`，以便 `SnapDrive snap restore`` 命令能够成功运行。


## 使用 SnapDrive snap restore 命令所需的信息

要还原 Snapshot 副本，请确定要还原的存储实体，指定 Snapshot 副本的名称等。

下表提供了使用 `SnapDrive snap restore`` 命令时需要提供的信息。

要求 / 选项	参数
<p>确定要还原的存储实体的类型，并使用相应的参数输入此实体的名称。</p> <ul style="list-style-type: none"> <li>• 如果指定要还原的主机卷或文件系统，则所提供的参数将转换为包含该卷或文件系统的磁盘组。然后，适用于 UNIX 的 SnapDrive 将还原整个磁盘组。SnapDrive for UNIX 会冻结这些磁盘组中主机卷中的任何文件系统，并为这些磁盘组中包含 LUN 的所有存储系统卷创建一个 Snapshot 副本。</li> <li>• 如果指定的文件规范是 NFS 挂载点，则参数将转换为目录树。SnapDrive for UNIX 仅还原 NFS 目录树或文件。在目录树中， SnapDrive for UNIX 会删除在创建 Snapshot 副本之后创建的任何新 NFS 文件或目录。这样可以确保已还原目录树的状态与创建该树的 Snapshot 副本时相同。</li> <li>• 如果还原 LUN ， SnapDrive for UNIX 将还原您指定的 LUN 。</li> <li>• 如果还原直接在 LUN 上创建的文件系统，则 SnapDrive for UNIX 将还原 LUN 和文件系统。</li> <li>• 如果 Snapshot 副本包含跨越多个存储系统卷的存储实体，则可以还原该 Snapshot 副本中的任何实体。</li> </ul>	LUN ( <code>`-lun file_spec`</code> )
LUN 的名称。您必须包含存储系统，卷和 LUN_ 的名称	磁盘组 ( <code>`-dg file_spec c`</code> ) 或卷组 ( <code>`-vg file_spec`</code> )
磁盘或卷组的名称 _	文件系统 ( <code>`-fs file_spec`</code> )
文件系统的名称	文件 ( <code>`-file file_spec`</code> )
NFS 文件的名称	主机卷 ( <code>`-hostvol file_spec`</code> ) 或逻辑卷 ( <code>`-lvol file_spec`</code> )

要求 / 选项	参数
主机或逻辑卷的名称。您必须同时提供请求的卷及其所在的磁盘组；例如 - hostvol dg3/Acct_volume_	<p>指定 Snapshot 副本的名称。如果您在命令提示符处提供的任何 <code>file_spec_`</code> 参数当前位于本地主机上，则可以使用简短形式的 Snapshot 副本名称。</p> <p>如果主机上不存在 <code>file_spec_`</code> 参数，则必须使用一个长形式的 Snapshot 副本名称，在此输入存储系统名称，卷和 Snapshot 副本名称。如果在命令提示符处为 Snapshot 副本使用较长名称，并且路径名称与存储系统和 / 或存储卷信息不匹配，则 SnapDrive for UNIX 将失败。以下是一个长 Snapshot 副本名称示例：</p> <pre>BIG_filer : /vol/account_vol : snap_20031115</pre> <p>有时，随 <code>file_spec_`</code> 参数提供的值可能在主机上不存在。例如，如果您已卸载文件系统或通过导出，删除或销毁操作删除了磁盘组，则该文件系统或磁盘组仍可作为 <code>file_spec_`</code> 参数的值。但是，该值将被视为不存在的值。SnapDrive for UNIX 可以还原此类不存在的 Snapshot 副本 <code>file_spec_`</code>，但您必须使用较长的 Snapshot 副本名称。</p> <p>当您还原跨越多个存储系统和卷且包含不存在的 <code>file_spec_`</code> 参数的 Snapshot 副本时，SnapDrive for UNIX 会允许命令行出现不一致。它不允许使用现有的 <code>file_spec_`</code> 参数。如果您只想从多个存储系统 Snapshot 副本还原一个存储实体，则您指定的 Snapshot 副本不必与包含该存储实体的存储系统位于同一个存储系统上。</p> <p>相同 Snapshot 副本名称的简短格式将省略存储系统和存储系统卷名称，因此其显示为：<code> Snap_20031415_`</code></p>
Snapshot 副本名称 ( <code>-snapname`</code> )	<code> `snap_name_`</code>
<p>它可以是一个短名称，例如 <code>mysnap1_`</code>，也可以是一个包含存储系统名称，卷和 Snapshot 副本名称的长名称。</p> <p>通常，NetApp 建议您使用短名称。如果 <code>file_spec_`</code> 参数不存在：即，它不再存在于主机上；请参见 <code>file_spec_`</code> 参数的说明。然后，必须使用 Snapshot 副本的长名称。</p>	<code> `预留`</code>
<code>-noreserve`</code>	
可选：希望 SnapDrive for UNIX 在还原 Snapshot 副本时创建空间预留。	<code> `强制`</code>

要求 / 选项	参数
~	`-noprop`
~	<p>可选：确定是否要覆盖现有 Snapshot 副本。如果没有此选项，则如果您提供现有 Snapshot 副本的名称，此操作将暂停。提供此选项并指定现有 Snapshot 副本的名称时，系统会提示您确认是否要覆盖此 Snapshot 副本。要防止 SnapDrive for UNIX 显示提示符，请同时包含 `-noprop` 选项。（如果要使用 -noprop 选项，则必须始终包含 `-force` 选项。）</p> <p>如果您尝试还原自上次 Snapshot 副本以来配置发生更改的磁盘组，则必须在命令提示符处包含 `-force` 选项。例如，如果在创建 Snapshot 副本后更改了磁盘上数据的条带化方式，则需要包含 `-force` 选项。如果没有 `-force` 选项，此操作将失败。此选项会要求您确认要继续此操作，除非您在其中包含 `-noprop` 选项。</p> <div>  <p>如果添加或删除了 LUN，则还原操作将失败，即使包含 `-force` 选项也是如此。</p> </div>
mntopts	~

## 还原 Snapshot 副本

您可以使用 SnapDrive snap restore` 命令还原 Snapshot 副本。还原操作可能需要几分钟时间，具体取决于还原的类型和数据量。

### 步骤

1. 输入以下命令以还原 Snapshot 副本：

```
` * SnapDrive snap restore -snapname snap_name { -lun | -dg -fs | - hostvol _-file } file_specs
[file_spec...][ { -lun | -dg | -fs ; -hostvol ; -file } _file_spec 【 file_spec 】 [-force [-noprompt]] [-
mntopts options]] ( { -reserve ; -noreserve } ) *
```

SnapDrive for UNIX 会将您在 SnapDrive snap restore` 命令行中指定的 LUN 内容替换为您指定的 Snapshot 副本中的 LUN 内容。此操作可能需要几分钟时间。操作完成后， SnapDrive for UNIX 将显示一条类似以下内容的消息： snap restore <file 显示器 列表 > succeeded

在以下示例中，文件系统 1（fs1）位于存储系统 1 上，文件系统 2（fs2）位于存储系统 1 上，同时也位于存储系统 2 上，该存储系统是配对存储系统。文件系统 3（FS3）位于存储系统 1，配对存储系统 2 和存储系统 3 上，这不是 HA 对的一部分。另一个文件系统 FS4 完全驻留在存储系统 4 上。

以下命令将创建 fs1， fs2， FS3 和 FS4 的 Snapshot 副本：

```
# snapdrive snap create -fs /mnt/fs1 /mnt/fs2 /mnt/fs3 /mnt/fs4
-snapname fs_all_snap
```

下一个命令将还原目标存储系统上的 fs1 和 fs2。fs1 和 fs2 都驻留在 HA 对上，因此您可以使用一个命令还原它们：

```
# snapdrive snap restore -fs /mnt/fs1 /mnt/fs2 -snapname fs_all_snap
```

以下命令将还原 FS4：

```
snapdrive snap restore -fs /mnt/fs4 -snapname fs_all_snap
```

SnapDrive for UNIX 无法在目标存储系统上还原 FS3，因为此文件系统位于存储系统 1，存储系统 2 和存储系统 3 上。

## 从其他主机还原 **Snapshot** 副本

使用 SnapDrive snap restore` 命令从其他主机还原 Snapshot 副本。

通常，您可以从创建 Snapshot 副本的主机还原 Snapshot 副本。有时，您可能需要使用其他主机或非源主机还原 Snapshot 副本。要使用非源主机还原 Snapshot 副本，请使用通常使用的相同 SnapDrive snap restore` 命令。如果您还原的 Snapshot 副本包含 NFS 实体，则非源主机必须具有访问 NFS 目录的权限。

## 基于卷的 **SnapRestore**

SnapDrive 4.0 for UNIX 及更高版本可在卷级别提供 Snapshot 还原功能。本节介绍了可使用基于卷的 Snapshot 还原执行的各种存储操作。

### 什么是基于卷的 **SnapRestore**

基于卷的 SnapRestore（VBSR）可还原卷及其所有存储对象。基于卷的还原速度快于单独还原的每个存储对象。VBSR 还可与 FlexClone 卷和 vFiler 配置配合使用。适用于 vFiler 的 VBSR 适用于 Data ONTAP 7.3 及更高版本。

SnapDrive 3.0 for UNIX 及更早版本可以从应用程序一致的快照还原主机端实体（例如文件系统，磁盘组和主机卷）的 LUN，或者通过 NFS 创建的正常文件。SnapDrive for UNIX 使用在 Data ONTAP 中实施的单文件快照还原 SFSR。SFSR 的工作原理如下：

- 对于正常文件，在 SFSR 进行期间，尝试更改此文件的任何操作都会暂停，直到 SFSR 完成为止。
- 对于 LUN，当 SFSR 继续运行时，LUN 可用，并且允许 I/O（读取和写入）。正常文件和 LUN 的 SFSR 可能需要很长时间，具体取决于 LUN 的大小或要还原的文件。

因此，对于某些环境，SFSR 是一种障碍。

SnapDrive 4.0 for UNIX 及更高版本可用于创建基于卷的 Snapshot 副本。VBSR 速度更快，所需的 CPU 和存

储资源更少。它会还原活动文件系统上的所有数据。如果用户希望从同一 Snapshot 副本还原卷上的所有 LUN 或正常文件，则可以使用此功能。

应谨慎使用基于卷的 Snapshot 副本，因为在用于还原操作的卷 Snapshot 副本之后创建的所有 Snapshot 副本都将被删除。必须删除在此卷上创建的所有新文件和新 LUN。

### 使用基于卷的 **SnapRestore** 的注意事项

您可以在使用基于卷的 SnapRestore（VBSR）时牢记几点。牢记这些要点有助于您安全地使用 VBSR 功能。

您必须牢记以下几点：

- VBSR 会将整个卷还原到创建 Snapshot 副本时的状态，此状态用于 VBSR。VBSR 包括以下内容：
  - 在 `snap create` 操作期间，指定主机文件规范的所有文件和 LUN。
  - 在 `snap create` 操作期间，应用程序一致的 Snapshot 副本中的所有文件和 LUN。
- VBSR 将删除在用于还原的 Snapshot 副本之后在当前卷上创建的所有较新文件和 LUN。
- VBSR 将删除在用于还原的 Snapshot 副本之后创建的所有较新 Snapshot 副本。
- 建议在使用 `-vbsr execute` 命令之前运行 `-vbsr preview` 命令。

### 基于卷的 **SnapRestore** 的强制检查

在执行基于卷的 SnapRestore 之前，SnapDrive for UNIX 会对系统执行一些强制检查。要安全地使用基于卷的 SnapRestore，必须执行这些检查。用户不能覆盖强制检查。

以下是在实施基于卷的 SnapRestore 之前 SnapDrive for UNIX 进行的强制检查：

- 基于卷的 SnapRestore 仅适用于使用 SnapDrive for UNIX 创建的快照。
- 基于卷的 SnapRestore 中的卷不应是存储系统的根卷。
- SnapDrive for UNIX 会检查卷克隆。如果新 Snapshot 副本中存在任何卷克隆，则不允许执行卷还原操作。这是 Data ONTAP 施加的一项限制。
- 基于卷的 SnapRestore 中的卷不应具有除指定 LUN（文件系统，磁盘组或主机卷中存在的原始 LUN 或 LUN）以外的任何映射 LUN 进行还原。
- SnapDrive for UNIX 会检查卷是否位于 SnapMirror 关系中。
- SnapDrive for UNIX 会检查卷是否位于 SnapVault 关系中。

如果 SnapDrive for UNIX 正在使用 Operations Manager for RBAC 且用户在卷上具有 `sd.snapshot.DisruptionBaseline` 功能，则可以覆盖 SnapMirror 和 SnapVault 检查。有关用户覆盖这些检查的特定 RBAC 功能的详细信息，请参见 SnapDrive for UNIX 中的基于角色的访问控制。

- 相关信息 \*

### [SnapDrive for UNIX 中基于角色的访问控制](#)

可由用户覆盖的检查

在执行基于卷的 SnapRestore 之前， SnapDrive for UNIX 会执行一些检查，用户可以使用 `-force` 选项覆盖这些检查。要安全地使用基于卷的 SnapRestore，必须执行这些检查。

建议您按照系统执行的各种检查进行操作，但您可以使用 `-force` 选项来覆盖这些检查。

对于基于卷的 SnapRestore 检查，您可以覆盖以下 SnapDrive for UNIX：

- SnapDrive for UNIX 会检查 Snapshot 副本中的 LUN 是否崩溃不一致。如果它在 Snapshot 副本中发现应用程序不一致的 LUN，则会向您发出警告。
- SnapDrive for UNIX 会检查活动卷中是否存在创建 Snapshot 副本后创建的其他 LUN。如果 SnapDrive for UNIX 发现其他 LUN，则会警告您活动卷中的这些额外 LUN 将丢失。
- SnapDrive for UNIX 会检查是否存在新的 Snapshot 副本。这些新 Snapshot 副本不会还原，并且会丢失。
- SnapDrive for UNIX 会检查卷中的普通文件（可从本地主机查看的文件）。
- SnapDrive for UNIX 会检查 NFS 导出。
- SnapDrive for UNIX 会检查 CIFS 共享。

如果 Snapshot 副本是使用适用于 UNIX 的 SnapDrive 3.0 或更早版本创建的，则基于卷的 SnapRestore 预览将无法执行上述列表中的前两项检查。如果您提供了 `-force` 选项，则在基于卷的 SnapRestore 执行期间，将显示一条提示消息，以覆盖这些检查并继续操作。

## 基于卷的 SnapRestore 命令

本节介绍使用基于卷的 SnapRestore 的命令和选项。

在 `snap restore` 命令行界面中添加了 `-vbsr` 选项，用于选择基于卷的 SnapRestore。使用以下命令语法使用基于卷的 SnapRestore 执行还原：

```
snapdrive snap restore {-lun | -dg | -vg | -hostvol |  
-lvol | -fs |-file} file_spec [file_spec ...] [{-lun | -dg |  
-vg | -hostvol | -lvol | -fs -file} file_spec [file_spec ...]  
...] -snapname snap_name [-force [-noprompt]][{-reserve |  
-noreserve}]  
[-vbsr [execute | preview]]
```

如果在 `-vbsr` 中未提供任何参数，则默认输出为 `preview` 选项的输出。使用 `-verbose` 选项，可以详细输出用户可能会覆盖的所有必需检查。未提供 `-verbose` 选项时的默认输出将显示失败的检查的结果。

如果您不希望在执行基于卷的 SnapRestore 时出现任何确认消息，则可以在执行 `snap restore -vbsr execute` 命令时使用 `-noprop` 和 `-force` 选项。下表根据您提供的选项介绍了 SnapDrive for UNIX 的行为。



序列号	-vbsr execute	【强制】	-noproPrompt	结果
1.	否	不适用	不适用	预览模式为默认模式。完成所有检查，并为每个检查生成报告。
2.	是的。	否	否	所有检查均已完成。如果用户可以覆盖的任何强制检查失败，SnapDrive for UNIX 将显示一条错误消息。
3.	是的。	是的。	否	所有检查均已完成。如果任何强制检查失败，SnapDrive for UNIX 将显示一条错误消息。如果用户可以覆盖的任何检查失败，SnapDrive for UNIX 将提示您。
4.	是的。	是的。	是的。	所有检查均已完成。如果任何强制检查失败，SnapDrive for UNIX 将显示一条错误消息。如果用户可以覆盖的任何检查失败，SnapDrive for UNIX 不会提示您。

有关映射到本地或远程主机的 **LUN** 的信息

在基于卷的 SnapRestore 期间，从卷到 SnapDrive for UNIX 的唯一映射信息是 LUN 的启动程序组信息。如果您使用的启动程序组始终由适用于 UNIX 的 SnapDrive 创建，则主机的完全限定域名属于启动程序组名称。

如果 SnapDrive for UNIX 管理员指定了 `-igroup`` CLI 选项，或者您使用手动创建的启动程序组，则 `igroup name` 不一定需要主机名。由于上述所有原因，SnapDrive for UNIX 无法可靠地检测 LUN 的本地或远程绑定。因此，SnapDrive for UNIX 会在基于卷的 SnapRestore 中显示完整的 LUN 启动程序组和启动程序信息。

特定卷的主机文件规范信息

在卷还原预览报告中，SnapDrive for UNIX 会显示 LUN 映射信息。显示的信息与检查相关，正常文件将还原。查找基于特定卷上的 LUN 的所有主机文件名非常耗时，并且会减慢卷还原过程的速度。

如果您希望知道本地主机的主机文件规范信息已映射到特定存储系统卷，则可以使用 `SnapDrive storage show -filervol <full-volume-name>`。下面显示了一个示例。

```
#snapdrive storage show -filervol bart:/vol/volusecase2
```

Connected LUNs and devices:

device filename	adapter path	size	proto	state	clone
lun path	backing snapshot				
-----	-----	----	-----	-----	-----
-----	-----	-----	-----	-----	-----
/dev/sdg	- P	100m	iscsi	online	No
bart:/vol/volusecase2/lun5	-				

Host devices and file systems:

```
dg: vbsrfs_1_SdDg          dgtype lvm
hostvol: /dev/mapper/vbsrfs_1_SdDg-vbsrfs_1_SdHv      state: AVAIL
fs: /dev/mapper/vbsrfs_1_SdDg-vbsrfs_1_SdHv      mount point: /mnt/vbsrfs_1
(persistent) fstype jfs2
```

device filename	adapter path	size	proto	state	clone	lun
path	backing snapshot					
-----	-----	----	-----	-----	-----	-----
-----	-----	-----	-----	-----	-----	-----
/dev/sdbe	- P	100m	iscsi	online	No	
bart:/vol/volusecase1/lun9_0	-					
/dev/sdbf	- P	100m	iscsi	online	No	
bart:/vol/volusecase2/lun4_0	-					

```
raw device: /dev/sdbr1  mount point: /mnt/fs11 (persistent) fstype jfs2
```

device filename	adapter path	size	proto	state	clone	lun
path	backing snapshot					
-----	-----	----	-----	-----	-----	-----
-----	-----	-----	-----	-----	-----	-----
/dev/sdbr	- P	200m	iscsi	online	No	
bart:/vol/volusecase2/fs11_SdLun	-					

```
NFS device: bart:/vol/volusecase1      mount point: /mnt/volusecase1
(non-persistent)
```

LUNs not connected to this host:

lun path	size	state
-----	-----	-----
bart:/vol/volusecase2/lunotherhost	20m	online


要使用基于卷的快照预留预留空间，您需要在`snapdrive.conf`文件中设置`\_space-reservations-volume-enabled\_sn`选项。

`\_space-reservations volume-enabled\_sn`选项用于在卷上设置空间保证策略，并且可以采用以下三个值。

- \* 快照 \*：这是默认值。卷上的空间保证不变。
- \* 卷 \*：卷上的空间保证处于卷级别。
- \* 无 \*：空间保证设置为无。

下表介绍了基于卷的快照预留在空间预留中的行为。

未使用空间预留 CLI 选项；指定了`-vbsr execute`	space-reservations-volume-enable=	结果
无	Snapshot	卷上的空间保证不变。
无	无	尝试将卷的空间保证设置为`"none"`。
`_预留`	配置值已被覆盖	尝试将卷的空间保证设置为`"卷"`
`-noreserve`	配置值已被覆盖	尝试将卷的空间保证设置为`"无"`
无	volume	尝试将卷的空间保证设置为`"卷"`



`-vbsr preview`不会检查任何空间预留选项。

## 连接到 **Snapshot** 副本

您可以将 Snapshot 副本从一台主机连接到另一台主机。

通过 SnapDrive for UNIX，您可以将主机从主机上的其他位置连接到 Snapshot 副本。此新位置可以位于创建 Snapshot 副本的主机（发起主机）上，也可以位于其他主机（非发起主机）上。

能够在新位置设置 Snapshot 副本意味着您可以将 Snapshot 副本备份到另一个介质，对磁盘组执行维护或访问 Snapshot 副本数据，而不会中断数据的原始副本。

使用此命令，您可以将主机连接到包含以下任一内容的 Snapshot 副本：

- LUN
- 直接在 LUN 上创建的文件系统
- 在 LUN 上创建的磁盘组，主机卷和文件系统
- NFS 目录树

- 共享存储系统上的磁盘组，主机卷和文件系统

## SnapDrive snap connect 命令的工作原理

使用 SnapDrive snap connect 命令时，它会克隆指定实体的存储并将其导入到主机：

- 如果您指定的 Snapshot 副本包含 LUN（`-lun`），则 SnapDrive for UNIX 会将 LUN 的新副本映射到主机。您不能使用 SnapDrive snap connect 命令在与其它存储实体（`-vg`，`-DG`，`-FS`，`-lvol` 或 `-hostvol`）相同的命令行上指定 LUN。
- 如果您指定的文件系统直接驻留在 LUN 上，则 SnapDrive for UNIX 会将 LUN 映射到主机并挂载文件系统。
- 如果在 snap connect 命令中将源挂载点指定为相对路径名，则 SnapDrive for UNIX 将忽略在命令行界面中指定的目标挂载点，并使用格式为 `ssource_mount_point_<N>` 的内部命名约定来命名目标挂载点。
- 如果您指定的 Snapshot 副本包含磁盘组，或者主机卷或文件系统属于磁盘组，则 SnapDrive snap connect 命令会连接整个目标磁盘组。为了建立连接，SnapDrive for UNIX 会重新激活目标磁盘组的所有逻辑卷，并将所有文件系统挂载到逻辑卷上。
- 如果使用 snap connect 命令指定 autorename 选项，则主机卷和文件系统始终会重命名。只有当磁盘组已位于主机上时，才会对其进行重命名。
- 如果您指定的 Snapshot 副本包含 NFS 目录树，则 SnapDrive for UNIX 将为包含 NFS 目录树的 FlexVol 卷创建一个克隆。然后，SnapDrive for UNIX 会将卷连接到主机并挂载 NFS 文件系统。在目录树中，SnapDrive for UNIX 会删除您在创建 Snapshot 副本后创建的任何新 NFS 文件或目录。如果将 `snapconnect-nfs-removedirectories` 配置选项设置为 on，则 SnapDrive for UNIX 将从 FlexVol 卷中删除您连接的 NFS 目录以外的任何文件或目录。
- 如果使用 `readonly` 选项连接包含 NFS 目录树的 Snapshot 副本，则 SnapDrive for UNIX 将直接挂载该目录的 Snapshot 副本，而不创建克隆。您不能使用 SnapDrive snap connect 命令在与非 NFS 实体相同的命令行上指定 NFS 挂载点，即使用选项 `-vg`，`-DG`，`-FS`，`-lvol` 或 `-hostvol`。



Data ONTAP 7.3 及更高版本支持在 vFiler 环境中使用 `-split` 选项执行 snap connect 操作。

## 在镜像存储系统上连接 Snapshot 副本

如果在镜像存储系统上创建了 Snapshot 副本，则可以将源存储系统上的 Snapshot 副本连接到目标系统。

在镜像存储系统上创建 Snapshot 副本时，Snapshot 副本会自动从创建该副本的源系统复制到目标（镜像）存储系统。SnapDrive for UNIX 允许您在源存储系统上连接 Snapshot 副本。您还可以在目标存储系统上连接 Snapshot 副本。

### 连接多个存储实体

您可以连接包含多个存储实体的 Snapshot 副本。

要连接包含驻留在多个目标存储系统上的存储实体的 Snapshot 副本，必须满足以下要求：

- 在命令提示符处指定的存储实体必须位于单个存储系统或 HA 对上。
- 源存储系统的卷名称必须与目标存储系统的卷名称匹配。

- 您必须将 SnapDrive.conf 文件中的 `snapmirror-dest-multity-filervolumes-enabled\_` 变量设置为 "on"。

您可以使用一个命令来连接驻留在单个存储系统或 HA 对上的存储实体。

## Snapshot 连接和 Snapshot 还原操作

连接到 Snapshot 副本时，Snapshot 副本会克隆信息。

与 Snapshot 还原操作不同，Snapshot 连接操作不会将构成主机实体的 LUN 的现有内容替换为 Snapshot 副本内容。它会克隆信息。

建立连接后，Snapshot 连接和 Snapshot 还原操作都会执行类似的活动：

- Snapshot 连接操作会激活存储实体的逻辑卷，挂载文件系统，并可选择向主机文件系统表添加一个条目。
- Snapshot 还原操作会激活存储实体的逻辑卷，挂载文件系统并应用 Snapshot 副本中保留的主机文件系统挂载条目。

## 连接 Snapshot 副本的准则

连接到 Snapshot 副本时，请遵循相关准则。

- SnapDrive snap connect 命令仅适用于在适用于 UNIX 的 SnapDrive 4.2 中创建的 Snapshot 副本。
- 在源主机上，SnapDrive for UNIX 支持连接和还原由先前版本的 SnapDrive for UNIX 创建的 Snapshot 副本。
- 要对 NFS 目录树进行读写访问，SnapDrive snap connect 命令使用 Data ONTAP FlexVol 卷功能，因此需要 Data ONTAP 7.3 或更高版本。使用 Data ONTAP 7.1 的配置可以连接 NFS 文件或目录树，但提供只读访问权限。
- 如果您在 Snapshot 连接操作期间将 `启用拆分克隆` 配置变量值设置为 "on" 或 "sync"，并在 Snapshot 断开连接操作期间将 "off"，则 SnapDrive for UNIX 不会删除 Snapshot 副本中的原始卷或 LUN。
- 您必须设置 Data ONTAP 7.2.2 配置选项 vFiler 的值。vol\_clone\_zapi\_allow to "on" 以连接到 vFiler 单元中卷或 LUN 的 Snapshot 副本。
- 具有不同主机配置的主机不支持 Snapshot 连接操作。
- 用于连接到物理存储系统或 vFiler 单元的根卷的 SnapDrive snap connect 命令失败，因为 Data ONTAP 不允许克隆根卷。

## 使用 SnapDrive snap connect 命令所需的信息

要连接到 Snapshot 副本，请确定存储实体的类型，将具有 NFS 目录树的 Snapshot 副本连接到 Data ONTAP 7.3 配置等。

下表提供了使用 SnapDrive snap connect 命令时需要提供的信息。



snap connect 操作要求在 AIX 上使用 Veritas 为 vximport 提供 Storage Foundation Enterprise 许可证。

要求 / 选项	参数
<p>确定要用于附加 Snapshot 副本的存储实体的类型，并使用相应的参数提供此实体的名称。这是 `src_fs_spec` 参数的值。</p> <ul style="list-style-type: none"> <li>如果连接 LUN 的 Snapshot 副本，则 SnapDrive for UNIX 将连接您指定的 LUN。您不能在同一命令行上使用 `-lun` 选项和 `-vg`，`-dg`，`-fs`，`-lvol` 或 `-hostvol` 选项。您可以使用 <code>lun_name</code> 或 <code>qtree_name/lun_name</code> 格式指定 LUN 的短名称。</li> <li>如果您连接的是直接在 LUN 上创建的文件系统的 Snapshot 副本，则 SnapDrive for UNIX 将连接具有该文件系统的 LUN。</li> <li>如果连接的磁盘组的 Snapshot 副本具有主机卷或文件规范，则参数将转换为存储系统上的一组磁盘组。SnapDrive for UNIX 将连接包含此实体的整个磁盘组，即使此实体是主机卷或文件系统也是如此。</li> <li>如果连接 NFS 文件系统的 Snapshot 副本，则参数将转换为 NFS 目录树。SnapDrive for UNIX 将创建卷的 FlexClone，删除 Snapshot 副本中未指定的目录树，然后连接并挂载 NFS 目录树。如果指定 NFS 挂载点，则不能在同一命令行上指定非 NFS 实体（`-vg`，`-DG`，`-FS`，`-lvol` 或 `-hostvol`）。</li> </ul> <div>  <p>SnapDrive for UNIX 在挂载点级别不支持符号链接。</p> </div>	<p>LUN（`-lun file_spec`）</p>
<p>LUN 的缩写名称。 _</p>	<p>`s_lun_name` 指定存在于 `-snapname long_snap_name` 中的 LUN。lun_name 短称为必填项。不能包含存储系统或存储系统卷名称。`d_d_lun_name` 指定 LUN 的连接名称。短 `lun_name` 为必填项。不能包含存储系统或存储系统卷名称。您必须指定 `d_lun_name`</p>
<p>磁盘组（`-dg file_spec`）或卷组（`-vg file_spec`）</p>	<p>磁盘或卷组的名称 _</p>
<p>文件系统（`-fs file_spec`）</p>	<p>文件系统的名称</p>
<p>主机卷（`-hostvol file_spec`）或逻辑卷（`-lvol file_spec`）</p>	<p>主机或逻辑卷的名称 _</p>

要求 / 选项	参数
<p>将具有 NFS 目录树的 Snapshot 副本连接到 Data ONTAP 7.3 配置。</p> <ul style="list-style-type: none"> <li>如果您的配置使用 Data ONTAP 7.3 或更高版本的 Data ONTAP 与传统（而不是 FlexVol）卷，则必须指定此选项来连接具有只读访问权限的 Snapshot 副本（必需）。</li> <li>如果您的配置使用 Data ONTAP 7.3 及更高版本以及 FlexVol 卷，则 SnapDrive for UNIX 会自动提供读写访问。仅当要将访问限制为只读（可选）时，才指定此选项。</li> </ul>	-readonly
设置只读权限 _	<p>可选：提供在连接存储实体后可访问目标实体的名称。SnapDrive for UNIX 使用此名称连接目标实体。这是 <code>`dest_file_spec_`</code> 参数。</p> <p>如果省略此名称，则 <code>snap connect</code> 命令将使用您为 <code>`src_fs_spec_`</code> 提供的值。</p>
目标实体的名称	<code>`dest_file_spec_`</code>
<p>可选：指定目标存储实体的名称。如果您在 <code>`dest_fs_spec/src_fs_spec_`</code> 对中包含此信息，则无需在此处输入。</p> <p>如果目标存储实体的名称不是 <code>`destination_fs_spec/src_fs_spec_`</code> 对的一部分，则可以使用 <code>`-destxx`</code> 选项来指定此信息。例如，<code>`-fs`</code> 选项仅命名目标挂载点，因此您可以使用 <code>`-destdg`</code> 选项指定目标磁盘组。</p> <p>如果未指定连接目标磁盘组中的实体所需的名称，则 SnapDrive <code>snap connect`</code> 命令将从源磁盘组中获取该名称。</p> <p>如果未指定连接目标磁盘组中的实体所需的名称，则 <code>snap connect</code> 命令将从源磁盘组中获取该名称。如果不能使用该名称，则操作将失败，除非您在命令提示符处包含 <code>`-autorename`</code>。</p>	目标磁盘组（ <code>`destdg`</code> ）或目标卷组（ <code>`-destvg`</code> ）
<code>`dgname`</code>	目标逻辑卷（ <code>`-destlv`</code> ）或目标主机卷（ <code>`-desthv`</code> ）
<code>`lvname_`</code>	指定 Snapshot 副本的名称。请使用名称的长格式输入存储系统名称，卷和 Snapshot 副本名称。
Snapshot 副本名称（ <code>`-snapname`</code> ）	<code>`lor_snap_name_`</code>

要求 / 选项	参数
`- noperit`	~
<p>可选：将 Snapshot 副本连接到新位置，而不在主机文件系统表中创建条目。</p> <ul style="list-style-type: none"> <li>使用 `-noperit` 选项，您可以将 Snapshot 副本连接到新位置，而无需在主机文件系统表中创建条目。默认情况下， SnapDrive for UNIX 会创建永久性挂载。这意味着： <ul style="list-style-type: none"> <li>在 AIX 主机上连接 Snapshot 副本时， SnapDrive for UNIX 会挂载文件系统，然后在主机的文件系统表中为构成文件系统的 LUN 放置一个条目。</li> <li>您不能使用 `-noperit` 连接包含 NFS 目录树的 Snapshot 副本。</li> </ul> </li> </ul>	` 预留
`-noreserve`	~
<p>可选：无论是否创建空间预留，都将 Snapshot 副本连接到新位置。</p>	igroup 名称（`-igroup`）
`ig_name_`	可选： NetApp 建议您使用主机的默认 igroup，而不是提供 igroup 名称。
` 自动扩展`	~



要求 / 选项	参数
<p>要缩短连接到卷组时必须提供的信息量，请在命令提示符处包括 <code>`-AutoExpand`</code> 选项。使用此选项，您只能为卷组中的一部分逻辑卷或文件系统命名。然后，它会扩展与磁盘组中其余逻辑卷或文件系统的连接。通过这种方式，您无需指定每个逻辑卷或文件系统。SnapDrive for UNIX 使用此信息生成目标实体的名称。</p> <p>此选项用于对命令提示符处指定的每个磁盘组以及组中的所有主机 LVM 实体执行适用场景操作。如果不使用 <code>`-AutoExpand`</code> 选项（默认），则必须指定该磁盘组中包含的所有受影响主机卷和文件系统来连接整个磁盘组。</p> <div>  <p>如果输入的值是磁盘组，则无需输入所有主机卷或文件系统，因为 SnapDrive for UNIX 知道磁盘组连接到的内容。</p> </div> <p>NetApp 建议，如果包括此选项，则还应包括 <code>`-autorename`</code> 选项。如果需要使用 <code>`-AutoExpand`</code> 选项连接 LVM 实体的目标副本，但此名称已在使用中，则命令将失败，除非 <code>`-autosename`</code> 选项位于命令提示符处。</p>	<p>如果您不包括 <code>-AutoExpand</code>，也不指定在命令提示符处引用的所有磁盘组中的所有 LVM 主机卷（通过指定主机卷本身或文件系统），则此命令将失败。</p>
<code>` 自动名称`</code>	~
<p>如果在不使用 <code>`-autosename`</code> 选项的情况下使用 <code>`-AutoExpand`</code> 选项，则在使用 LVM 实体的目标副本的默认名称时，<code>snap connect</code> 命令将失败。如果包含 <code>`-autorename`</code> 选项，则 SnapDrive for UNIX 会在使用默认名称时重命名实体。这意味着，在命令提示符处使用 <code>-autorename</code> 选项时，无论所有必要名称是否可用，Snapshot 连接操作都会继续进行。</p> <p>此选项用于适用场景命令提示符处指定的所有主机端实体。</p> <p>如果在命令提示符处包括 <code>`-autosename`</code> 选项，则表示 <code>-AutoExpand</code> 选项，即使不包含该选项也是如此。</p>	<code>` 拆分`</code>
~	用于在 Snapshot 连接和 Snapshot 断开操作期间拆分克隆的卷或 LUN。
<code>mntopts</code>	~

## 连接到包含 LUN 的 Snapshot 副本

您可以使用 SnapDrive snap connect` 命令连接到包含 LUN 的 Snapshot 副本。

### 步骤

1. 输入以下命令语法以连接到包含 LUN 的 Snapshot 副本：

```
` * SnapDrive snap connect -lun s_lun_name d_lun_name [-lun] s_lun_name d_lun_name... -snapname long_snap_name "-igroup ig_name [ig_name...]` 【拆分】 ` *
```



*s\_lun\_name* 和 *d\_lun\_name* 的格式应为 *lun\_name* 或 *qtree\_name/lun\_name*。

SnapDrive for UNIX 会克隆您指定的 LUN 并将其连接到新位置。

以下示例将 `hornet/vol/vol1/tuesdaysnapshot` 中的 LUN `mylun1` 连接到 `mylun1copy`：

```
# ./snapdrive snap connect -lun mylun1 mylun1copy -snapname
hornet:/vol/vol1:tuesdaysnapshot
connecting hornet:/vol/vol1/mylun1:
LUN copy mylun1copy ... created
(original: hornet:/vol/vol1/mylun1) mapping new lun(s) ... done
discovering new lun(s) ... done
```

以下示例将两个 LUN（`mylun1` 和 `mylun2`）分别连接到 `mylun1copy` 和 `mylun2copy`：

```
# ./snapdrive snap connect -lun mylun1 mylun1copy -lun mylun2
mylun2copy -snapname hornet:/vol/vol1:tuesdaysnapshot
connecting hornet:/vol/vol1/mylun1:
LUN copy mylun1copy ... created
(original: hornet:/vol/vol1/mylun1)
mapping new lun(s) ... done
connecting hornet:/vol/vol1/mylun2:
LUN copy mylun2copy ... created
(original: hornet:/vol/vol1/mylun2)
mapping new lun(s) ... done
discovering new lun(s) ... done
```

## 连接到 LUN 以外存储实体的 Snapshot 副本

您可以使用 SnapDrive snap connect` 命令连接到包含 LUN 以外存储实体的 Snapshot 副本。如果您提供的目标名称当前正在使用中，或者存在用作挂载点的文件系统名称，则无法使用此命令。

从非源主机连接到包含使用默认挂载 `QIO` 选项挂载的 VxFS 文件系统的 Snapshot 副本时，您应安装 Veritas File Device Driver（VxFDD）许可证。

## 步骤

### 1. 输入以下命令：

```
` * SnapDrive snap connect -snapname fspec_set [fspec_set...] -snapname long_snap_name "-igroup ig_name [ig_name...] [-AutoExpand] [-autosename] [-noperesit] [ { -reserve | -noreserve } ] [-readonly] [-splone]*`
```

在上述用法中，`fspec\_set` 的格式如下：

```
` * { -dg _ -fs _ -hostvol } src_file_spec [dest_file_spec] ( { -destdg _ -destvg } dgname) ( { -destlv _ -desthv } lvname)*`
```

此命令必须始终以要连接的存储实体的名称开头（例如，`-DG`，`-hostvol` 或 `-FS`）。如果指定 NFS 挂载点，则不能在同一命令行上指定非 NFS 实体（`-vg`，`-DG`，`-FS`，`-lvol` 或 `-hostvol`）。

SnapDrive for UNIX 会克隆您指定的 LUN 并将其连接到新位置。

以下命令行将连接一个磁盘组，并使用默认名称作为目标名称（即，从源名称创建这些名称）：

```
# snapdrive snap connect -vg vg1 -snapname
filer1:/vol/vol1:vg1snapshot
connecting vg1:
LUN copy vg1_lun1_0 ... created
(original: filer1:/vol/vol1/vg1_lun1)
mapping new lun(s) ... done
discovering new lun(s) ... done
Importing vg1
```

以下命令行将磁盘组与单个主机卷连接起来。它还会指定目标主机卷和磁盘组的名称：

```
# snapdrive snap connect -lvol vg1/vol1 vg1copy/vol1copy -snapname
filer1:/vol/vol1:vg1snapshot
connecting vg1:
LUN copy vg1_lun1_0 ... created
(original: filer1:/vol/vol1/vg1_lun1)
mapping new lun(s) ... done
discovering new lun(s) ... done
Importing vg1copy
```

以下命令行会将一个磁盘组连接到两个 LUN 和两个文件系统。它指定每个文件系统的目标名称，其中一个文件系统的主机卷以及两个文件系统的磁盘组：

```
# snapdrive snap connect -fs mnt/fs1 /mnt/fs1copy -destvg vg1copy \
-fs /mnt/fs2 /mnt/fs2copy -destlv vg1copy/vol2copy -destvg vg1copy
\ -snapname filer1:/vol/vol1:vg1snapshot
connecting vg1:
LUN copy vg1_lun1_0 ... created
(original: filer1:/vol/vol1/vg1_lun1)
LUN copy vg1_lun2_0 ... created
(original: filer1:/vol/vol1/vg1_lun2)
mapping new lun(s) ... done
discovering new lun(s) ... done
Importing vg1copy
```

以下命令行包括 `-AutoExpand` 选项，因为它会将一个磁盘组连接到两个文件系统。它使用默认名称作为目标名称（即，从源名称创建目标名称）：

```
# snapdrive snap connect -lvol mnt/fs1 -snapname
filer1:/vol/vol1:vg1snapshot \
-autoexpand
connecting vg1:
LUN copy vg1_lun1_0 ... created
(original: filer1:/vol/vol1/vg1_lun1)
LUN copy vg1_lun2_0 ... created
(original: filer1:/vol/vol1/vg1_lun2)
mapping new lun(s) ... done
discovering new lun(s) ... done
Importing vg1
```

以下命令行包括 `-autorename` 选项，因为它会将一个磁盘组连接到两个文件系统和两个 LUN：

```
# snapdrive snap connect -fs mnt/fs1 -snapname
filer1:/vol/vol1:vg1snapshot \
-autorename
connecting vg1:
LUN copy vg1_lun1_0 ... created
(original: filer1:/vol/vol1/vg1_lun1)
LUN copy vg1_lun2_0 ... created
(original: filer1:/vol/vol1/vg1_lun2)
mapping new lun(s) ... done
discovering new lun(s) ... done
Importing vg1_0
```

以下示例将连接到文件系统 Snapshot 副本，即 LVM 堆栈上创建的磁盘组：

```
# snapdrive snap connect -fs /mnt/jfs1 /mnt/jfs1_clone -snapname
snoopy:/vol/vol1:snapLvm -autorename
connecting lvml:
LUN copy lunLvml_0 ... created
(original: snoopy:/vol/vol1/lunLvml)
mapping new lun(s) ... done
discovering new lun(s) ... done
Importing lvml_0
Successfully connected to snapshot snoopy:/vol/vol1:snapLvm
disk group lvml_0 containing host volumes
jfs1_SdHv_0 (filesystem: /mnt/jfs1_clone)
```

在以下示例中，文件系统 1（fs1）位于存储系统 1 上，文件系统 2（fs2）位于存储系统 1 上，同时也位于存储系统 2 上，该存储系统是配对存储系统。文件系统 3（FS3）位于存储系统 1，配对存储系统 2 和存储系统 3 上，这不是 HA 对的一部分。另一个文件系统 FS4 完全驻留在存储系统 4 上。

以下命令将创建 fs1，fs2，FS3 和 FS4 的 Snapshot 副本：

```
snapdrive snap create -fs /mnt/fs1 /mnt/fs2 /mnt/fs3 /mnt/fs4
-snapname fs_all_snap
```

下一个命令用于连接目标存储系统上的 fs1 和 fs2。fs1 和 fs2 都驻留在 HA 对上，因此您可以使用一个命令还原它们：

```
snapdrive snap connect -fs /mnt/fs1 /mnt/fs2 -snapname fs_all_snap
```

以下命令将还原 FS4：

```
snapdrive snap connect -fs /mnt/fs4 -snapname fs_all_snap
```

SnapDrive for UNIX 无法在目标存储系统上连接 FS3，因为此文件系统位于存储系统 1，存储系统 2 和存储系统 3 上。

## 连接到 LUN 以外的共享存储实体的 Snapshot 副本

您可以使用 SnapDrive snap connect` 命令连接到 LUN 以外的共享存储实体的 Snapshot 副本。

### 步骤

1. 输入以下命令：

```
` * SnapDrive snap connect fspec_set [fspec_set...] -snapname long_snap_name`
```

```
` * fspec_set [fspec_set...] -snapname long_snap_name [-DeviceType shared] [-sparate]*`
```

在此语法中，`fspec\_set\_` 为：` { -dg \_ -fs } src\_file\_spec [dest\_file\_spec ] [-destdg dname]`

## 拆分卷克隆或 LUN 克隆

通过 SnapDrive for UNIX，您可以拆分卷克隆或 LUN 克隆。克隆拆分完成后，父卷与克隆之间的关系将被销毁，两个实体彼此独立，并具有各自的存储空间。

以下是克隆拆分操作：

- 估算卷克隆或 LUN 克隆的磁盘空间（以 MB 为单位）。
- 拆分卷克隆或 LUN 克隆。
- 停止卷克隆或 LUN 克隆拆分。
- 查看正在进行，已完成或失败的克隆拆分的状态。



- 如果拆分卷克隆，则克隆卷中的所有 Snapshot 副本都将被删除。
- 在拆分文件规范之前，必须运行 clone split estimate 命令，以确定克隆的卷中是否已创建任何 Snapshot 副本。
- 对于所有克隆拆分命令，只能使用 `-lun` 选项指定长 LUN 名称。您不能在同一命令行上指定 `-lun` 选项以及其他存储实体（-vg，-dg，-fs，-lvol 或 -hostvol 选项）。
- 在 clone split 命令中，始终必须使用文件规范的绝对路径名。
- 只有从适用于 UNIX 的 SnapDrive 4.2 及更高版本创建的 Snapshot 副本才可以使用 Snapshot 进行 LUN 克隆拆分评估。

### 估算用于拆分卷克隆的存储空间

克隆拆分估计有助于估算拆分卷克隆所需的存储空间（以 MB 为单位）。根据 SnapDrive for UNIX 提供的克隆拆分估计值，您可以确定用于拆分卷克隆的空间可用性。

#### 步骤

1. 输入以下命令以估计拆分卷克隆所需的存储空间。

```
` * SnapDrive 克隆拆分估计值 ( -lun ) lun_name 【lon_lun_name...】 | [ { -DG _ -vg _ -fs _ -hostvol _ -lvol } file_spec [file_spec...] | "-snapname lort_snap_name" { -volclone]-v | -verbose ) 【 -dump | -dumpall]*`
```

此操作将显示以下信息：

- 资源名称
- Container — FlexClone 的聚合
- Required Space — 拆分卷克隆所需的空間
- Available space — 容器上的可用空间
- 存储状态—指示卷克隆拆分的可用空间

- 自有空间—卷克隆占用的空间
- 共享空间—卷克隆与父卷一起占用的空间

使用 ` -verbose ` 选项时，将显示 自有空间 和 s共享空间。

以下示例估计了用于拆分卷克隆的存储空间。

```
# snapdrive clone split estimate -fs /mnt/my_mnt1 -fs /mnt/my_mnt2
Resource      Container  Required Space (MB)  Available Space (MB)  Storage Status
/mnt/my_mnt1  f3050-220  400                61500                AVAILABLE
               -111:aggr0
/mnt/my_mnt2  f3050-220  3292              1129                 NOT AVAILABLE
               -112:aggr1
```

对于每个文件规范，SnapDrive for UNIX 将确定在存储系统中拆分卷克隆所需的可用空间。此处，` /mnt/my\_mnt1 ` 文件规范具有所需的拆分空间，因此存储状态显示为可用。` `，` /mnt/my\_mnt2 ` 文件规范没有所需的拆分空间，因此存储状态显示为不可用。

以下是使用 ` -verbose ` 选项的示例。或者，您也可以使用 ` -v ` 选项。

```
# snapdrive clone split estimate -fs /mnt/my_mnt1 -verbose
Resource      Container  Owned   Shared   Required Space (MB)  Available Space (MB)  Storage Status
/mnt/my_mnt1  f3050-220  32365   403      403      55875  AVAILABLE
               -111:aggr0
```

## 估算用于拆分 LUN 克隆的存储空间

克隆拆分估计有助于估算拆分 LUN 克隆所需的存储空间（以 MB 为单位）。根据 SnapDrive for UNIX 提供的克隆拆分估计值，您可以确定用于拆分 LUN 克隆的空间可用性。

### 步骤

1. 输入以下命令以估计拆分 LUN 克隆所需的存储空间。

```
` * SnapDrive 克隆拆分估计值 ( -lun]long_lun_name_lun_name...][ { -DG _ -vg _ -fs _ -hostvol _ -lvol
} file_spec [file_spec...] | "-snapname long_snap_name" { -volclone]-lunclone } ] ( -v | -verbose ) *`
```

此操作将显示以下信息：

- 资源名称
- 容器— LUN 克隆的卷
- Required Space —拆分 LUN 克隆所需的空间

- Available space —容器上的可用空间
- 存储状态—指示 LUN 克隆拆分的可用空间
- 已拥有空间— LUN 克隆占用的空间
- 共享空间— LUN 克隆与父级一起占用的空间

使用 ` -verbose ` 选项时，将显示 自有空间 和 s共享空间。

以下示例估计了用于拆分 LUN 克隆的存储空间。

```
# snapdrive clone split estimate -fs /mnt/my_mnt1
Resource      Container Required Available Storage
                Space (MB) Space (MB) Status
/mnt/my_mnt1 f3050-220  5120    9986    AVAILABLE
              -112:/vol/vol_1
```

以下是使用 ` -verbose ` 选项的示例。或者，您也可以使用 ` -v ` 选项。

```
# snapdrive clone split estimate -fs /mnt/my_mnt1 -verbose
Resource      Container Owned    Shared    Required Available Storage
                Space (MB) Space (MB) Space (MB) Space (MB) Status
/mnt/my_mnt1 f3050-220  365    403      5120    9986 AVAILABLE
              -112:/vol/vol_1
```

## 使用 **Snapshot** 副本估算存储空间

如果存储系统中没有可用于 Snapshot 副本的克隆，则克隆拆分估计有助于使用 Snapshot 副本估计所需的存储空间（以 MB 为单位）。

### 步骤

1. 输入以下命令以估计所需的存储空间。

```
` * SnapDrive 克隆拆分估计值 -snapname [long_snap_name] { -volclone]-v` -verbose *
```

以下示例估计了使用 Snapshot 副本拆分 LUN 克隆所需的存储空间。

```
snapdrive clone split estimate -snapname f3050-220-112:/vol/vol_1:snap_1
-lunclone
Resource      Container Required Available Storage
                Space (MB) Space (MB) Status
f3050-220-112: f3050-220  5120    14078    AVAILABLE
/vol/vol_1:snap_1 -112:/vol/vol_1
```



以下示例使用带有 ` -FS` 选项的 Snapshot 副本估算用于拆分 LUN 克隆的存储空间。

```
# snapdrive clone split estimate -fs /mnt/my_mnt1 -snapname f3050-220-112:/vol/vol_1:snap_1 -lunclone
Resource          Container Required Available Storage
                  Space (MB) Space (MB) Status
f3050-220-112:    f3050-220 4120      14078  AVAILABLE
/vol/vol_1:snap_1 -112:/vol/vol_1
```

以下示例使用带有 ` -FS` 选项的 Snapshot 副本估算用于拆分卷克隆的存储空间。

```
# snapdrive clone split estimate -fs /mnt/fs1 -snapname f3050-220-112:/vol/vol_1:snap_1 -volclone
Resource          Container Required Available Storage
                  Space (MB) Space (MB) Status
f3050-220-112:    f3050-220 54019     54517  AVAILABLE
/vol/vol0:snap_1 112:aggr0
```

以下示例估计了使用 Snapshot 副本拆分卷克隆所需的存储空间。

```
# snapdrive clone split estimate -snapname f3050-220-112:/vol/vol_1:snap_1 -volclone
Resource          Container Required Available Storage
                  Space (MB) Space (MB) Status
f3050-220-112:    f3050-220 54019     54517  AVAILABLE
/vol/vol0:snap_1 112:aggr0
```



- 如果对 Snapshot 副本执行克隆拆分估计，则 "Resource" 字段包含 Snapshot 副本名称。
- 如果您提供任何失效文件规范以及带有 ` -lunclone` 选项的 Snapshot 副本，则 "所需空间" 将显示为 0。
- 只有从适用于 UNIX 的 SnapDrive 4.2 及更高版本创建的 Snapshot 副本才可以使用 Snapshot 进行 LUN 克隆拆分评估。

## 启动卷克隆或 LUN 克隆拆分

您可以启动卷克隆或 LUN 克隆拆分操作。

### 步骤

1. 输入以下命令以启动卷克隆或 LUN 克隆拆分。

```
` * # SnapDrive 克隆拆分起始值 ( -lun ) lun_name [lon lun_name...] | [ { -DG _ -vg _ -fs _ -hostvol _ -lvol } file_spec [file_spec ...] ] [-force][-noprompt] [-dump | -dumpall]*`
```

当存储状态显示为不可用时，可以使用以下选项。

- 您可以使用 `-force` 选项强制启动克隆拆分操作，并收到确认消息，指出操作已启动。
- 您可以使用 `-noprop` 以及 `-force` 选项启动克隆拆分启动操作，而不会收到任何确认消息。



如果在停止正在进行的克隆拆分操作后不久再启动另一个克隆拆分操作，则该操作可能会失败。如果启动和停止克隆拆分操作之间的延迟不足以使存储系统同步停止操作，则可能会出现此问题描述。

以下示例显示了如何拆分卷克隆：

```
# snapdrive clone split start -fs /mnt/my_mnt4_0 /mnt/my_mnt3_0
Resource      Container  Required Available Storage
                Space (MB) Space (MB) Status
-----
/mnt/my_mnt4_0 f3050-220 3295    66033 AVAILABLE
                -111:aggr0
/mnt/my_mnt3_0 f3050-220 293     37707 AVAILABLE
                -112:aggr1

Job ID: B265Dbv8gh
Clone-Split for "/mnt/my_mnt4_0" is started
Clone-Split for "/mnt/my_mnt3_0" is started
```

以下示例显示了如何使用 `-force` 选项拆分克隆：

```
# snapdrive clone split start -fs /mnt/my_mnt5 /mnt/my_mnt6 -force
Resource      Container  Required Available Storage
                Space (MB) Space (MB) Status
-----
/mnt/my_mnt5 f3050-220 1198    20033  AVAILABLE
                -111:aggr0
/mnt/my_mnt6 f3050-220 3294    2196  NOT AVAILABLE
                -112:aggr1
Not enough space available for Clone-Split. Do you want to continue
(y/n)?y
Clone-Split for "/mnt/my_mnt5" is started
Clone-Split for "/mnt/my_mnt6" is started
```

以下示例显示了如何使用 `-nopromptf` 选项直接启动克隆，这意味着没有确认消息：

```
# snapdrive clone split start -fs /mnt/my_mnt5 /mnt/my_mnt6 -force
-noprompt
Resource      Container  Required  Available Storage
              Space (MB) Space (MB) Status
-----
/mnt/my_mnt5  f3050-220  1198      20033    AVAILABLE
              -111:aggr0
/mnt/my_mnt6  f3050-220  3294      2196    NOT AVAILABLE
              -112:aggr1
Clone-Split for "/mnt/my_mnt5" is started
Clone-Split for "/mnt/my_mnt6" is started
```

## 查看卷克隆或 LUN 克隆拆分的状态

您可以使用作业 ID 或文件规范查询克隆拆分状态。SnapDrive for UNIX 将克隆拆分的当前状态指示为 in-progress , failed 或 complete 。

### 步骤

1. 输入以下命令，使用作业 ID 或文件规范查询克隆拆分状态。

```
` * SnapDrive 克隆拆分状态 ( -lun ) lun_name ( lon_lun_name... ) [ { -DG _ -vg _ -fs _ -hostvol _
-lvol } file_spec [file_spec...] ] [ -job <jobid> ] [ -all]*`
```

以下示例显示了使用作业 ID 的克隆拆分状态。

```
# snapdrive clone split status -job SVE2oxKXzH
Clone-Split-Status for /fs1-1_3 is 1% Complete
Clone-Split-Status for /fs1_0 is 73% Complete
Clone-Split-Status for /fs1_1 is 73% Complete
Clone-Split-Status for /fs1_2 is 74% Complete
Clone-Split-Status for /fs1_3 is 1% Complete
```

您可以通过以下任一方式检查克隆拆分进度的状态：

- 您可以使用验证克隆

```
` * SnapDrive storage show -fs /mnt/my_mn`
```

或

```
` * SnapDrive storage show -lun long_lun_pathname`
```

命令无论哪种情况，如果未完成拆分，克隆类型都会显示为 FlexClone 或 LUN 克隆。

- 您可以通过登录到存储系统并在存储系统命令行界面中使用以下命令来验证克隆拆分进度状态：

```
` * 卷克隆拆分状态 vol_name` ` * LUN 克隆拆分状态 lun_name`
```

以下示例显示了使用文件规范进行的克隆拆分状态查询：

```
# snapdrive clone split status -fs /mnt/my_mnt3 -fs /mnt/my_mnt4
Clone-Split-Status for /mnt/my_mnt3 is 14% Complete
Clone-Split-Status for /mnt/my_mnt4 is 17% Complete
```

以下示例显示了正在运行的克隆拆分状态查询：

```
# snapdrive clone split status -all
Job ID: SVE2oxKXzH:
Clone-Split-Status for /fs1-1_3 is 100% Complete
Clone-Split-Status for /fs1_0 is 100% Complete
Clone-Split-Status for /fs1_1 is 100% Complete
Clone-Split-Status for /fs1_2 is 100% Complete
Clone-Split-Status for /fs1_3 is 100% Complete
```

- 从作业集中删除作业后，如果您使用文件规范查询克隆拆分状态，则 SnapDrive for UNIX 将错误消息显示为

当前未对给定资源执行拆分

- 从作业集中删除作业后，如果您使用作业 ID 查询克隆拆分的状态，则 SnapDrive for UNIX 会将错误消息显示为

作业 ID 无效

- 从作业中删除所有文件规范后，如果您使用作业 ID 查询克隆拆分的状态，则 SnapDrive for UNIX 将显示为

作业 ID 无效

因为作业已从作业集中删除。

- 如果由于存储系统中的空间不足而导致任何文件规范失败，则作业仍会继续拆分以满足其余文件规范。这意味着不会从作业队列中删除作业，并且作业状态会一直保留，直到您查询整个作业结果为止。

## 停止卷克隆或 LUN 克隆拆分操作

您可以使用作业 ID 或文件规范停止卷克隆或 LUN 克隆的克隆拆分。

## 步骤

1. 输入以下命令:

\* SnapDrive 克隆拆分停止符 (-lun) lun\_name (lon\_lun\_name...) [ {-DG\_ -vg\_ -fs\_ -hostvol\_ -lvol} \_file\_spec [file\_spec...]] 我们提供了一个或一个或一个或一个或一个或一个或一个或一个或两个

SnapDrive for UNIX 将停止正在进行的克隆拆分停止操作。

以下示例显示了使用文件规范停止的克隆拆分操作。

```
# snapdrive clone split stop -fs /mnt/my_mnt4 /mnt/my_mnt3
Clone-Split for "/mnt/my_mnt4" is 0% Completed and Stopped.
Clone-Split for "/mnt/my_mnt3" is 0% Completed and Stopped.
```

以下示例显示了使用作业 ID 停止的克隆拆分操作。

```
# snapdrive clone split stop -job B265Dbv8gh
Clone-Split for "/mnt/my_mnt3" is 14% Completed and Stopped.
Clone-Split for "/mnt/my_mnt4" is 17% Completed and Stopped.
```

以下示例是一个典型输出，其中显示了已停止的文件规范的克隆拆分停止操作。

```
# snapdrive clone split stop -fs /mnt/my_mnt4 /mnt/my_mnt3
Clone-Split for "/mnt/my_mnt3" is not stopped : No split is in progress
for this resource
Clone-Split for "/mnt/my_mnt4" is not stopped : No split is in progress
for this resource
```



- 如果由于作业 ID 中的特定文件规范而停止克隆拆分，并且克隆拆分停止成功，则会从作业中删除文件规范。
- 如果作业的克隆拆分停止，而作业中所有文件规范的克隆拆分停止成功，则该作业将从作业集中删除。

使用作业 ID 或文件规范查看克隆拆分操作的结果

您可以使用作业 ID 或文件规范查看已完成克隆拆分操作的结果。

## 步骤

1. 输入以下命令以使用文件规范查看克隆拆分结果：

\* SnapDrive 克隆拆分结果 ( -lun) *lun\_name* [*lon\_lun\_name...*] [ { -DG \_-vg \_-fs \_-hostvol \_-lvol }  
file\_spec< 文件 规范 ...]] 我们提供了一个或`一个或一个或一个或一个或一个或一个或一个或两个

SnapDrive for UNIX 将显示文件规范已完成或失败的克隆拆分的结果，然后从作业中删除文件规范并从作业队列中删除作业。

以下示例显示了已成功完成的作业 ID 的克隆拆分结果。

```
# snapdrive clone split result -job VT1ov6Q8vU
Clone-Split for "/mnt/my_mnt3" is 100% completed and succeeded
Clone-Split for "/mnt/my_mnt4" is 100% completed and succeeded
```

如果存在两个文件规范，其中一个文件规范因存储系统中空间不足而失败，则克隆拆分操作的结果将显示为一个文件规范失败，另一个文件规范已成功完成。

以下示例显示了已成功完成的文件规范的克隆拆分结果。

```
# snapdrive clone split result -fs /mnt/my_mnt3 /mnt/my_mnt4
Clone-Split for "/mnt/my_mnt3" is 100% completed and succeeded
Clone-Split for "/mnt/my_mnt4" is 100% completed and succeeded
```

以下示例显示了克隆拆分操作仍在进行且尚未完成时的克隆拆分结果。

```
# snapdrive clone split result -job R57aCzUaeG
Clone-Split for "/mnt/my mnt3" is 0% completed and Split in progress
```

以下示例显示了从作业集中永久删除的作业，当您尝试使用文件规范查看结果时，SnapDrive for UNIX 将遇到错误消息 "不属于任何作业"。

```
# snapdrive clone split result -fs /mnt/my_mnt2
Storage resource /mnt/my_mnt2 does not belong to any job
```

以下示例显示了从作业集中永久删除的作业，当您尝试使用作业 ID 查看结果时，SnapDrive for UNIX 将遇到错误消息 "作业 ID 无效"。

```
# snapdrive clone split result -job T59aCzUaeG
Job ID is not valid
```

以下示例显示了克隆拆分结果，其中一个克隆拆分正在进行中，另一个克隆拆分失败。

```
# snapdrive clone split result -job qJrG8U59mg
Clone-Split for "/mnt/my_mnt4" is 100% completed and succeeded
Clone-Split for "/mnt/my_mnt5" is 0% completed and split failed
```

## 删除 Snapshot 副本

您可以使用 `SnapDrive snap delete`` 命令删除存储系统上的 Snapshot 副本。

### 用于删除 Snapshot 副本的命令

使用 `SnapDrive snap delete`` 命令删除 Snapshot 副本。

`SnapDrive snap delete`` 命令可从存储系统中删除您指定的 Snapshot 副本。此命令不会对主机执行任何操作。只有当您有权从存储系统中删除 Snapshot 副本时，它才会删除此副本。（如果要保留 LUN 和映射。）

### 删除 Snapshot 副本的原因

删除旧的 Snapshot 副本以释放存储系统卷上的空间，或者保留较少的 Snapshot 副本。

由于以下原因，您可能会删除较早的 Snapshot 副本：

- 在存储系统卷上保留的已存储 Snapshot 副本数少于硬限制 255。达到限制后，创建新 Snapshot 副本的尝试将失败。
- 以释放存储系统卷上的空间。即使在 Snapshot 副本达到限制之前，如果磁盘没有足够的预留空间，Snapshot 副本也会失败。
- 您也可以使用通配符（\*）来删除 Snapshot 副本。使用 `SnapDrive snap show`` 操作可以显示与特定模式匹配的所有 Snapshot 副本名称。以下规则适用于在 Snapshot 副本名称中使用通配符：
  - 您只能在名称末尾使用通配符。不能在 Snapshot 副本名称的开头或中间使用通配符。
  - 您不能在 Snapshot 副本名称的存储系统或存储系统卷字段中使用通配符。

### 删除 Snapshot 副本的准则

如果 Snapshot 副本位于多个存储系统卷上，则不能删除正在使用的 Snapshot 副本。

使用 `SnapDrive snap delete`` 命令时，请遵循以下准则：

- 如果要删除的任何 Snapshot 副本正在使用中或未由 SnapDrive for UNIX 创建，则 Snapshot 删除操作将失败。您可以通过将 `-force`` 选项与 `SnapDrive snap delete`` 命令结合使用来覆盖此行为。
- 如果您的 Snapshot 副本跨越多个存储系统卷，则必须手动删除每个卷上的 Snapshot 副本。

### 使用 SnapDrive snap delete 命令所需的信息

要删除 Snapshot 副本，请指定要删除的 Snapshot 副本的名称。使用 `SnapDrive snap delete`` 命令，您可以查看已删除的 Snapshot 副本列表。

下表提供了有关 SnapDrive snap delete` 命令的信息。

要求 / 选项	参数
指定 Snapshot 副本的名称。请使用 Snapshot 副本名称的长格式，在其中输入存储系统名称，卷和 Snapshot 副本名称。以下是一个长 Snapshot 副本名称示例：BIG_filer : /vol/account_vol : snap_20031115  如果要指定其他 Snapshot 副本，则可以使用名称的简短形式，前提是这些副本与第一个 Snapshot 副本位于同一存储系统和卷上。否则，请再次使用名称的长格式。	Snapshot 副本名称（`-snapname`）
`lor_Snapshot copy_name_`	其他 Snapshot 副本
`Snapshot copy_name`（长格式或短格式）	`-详细`
~	要显示已删除的 Snapshot 副本的列表，请包括 -verbose 选项。如果您使用的是 Snapshot 副本名称的简短格式，则此选项将填充缺失的存储系统和卷信息。
`强制`	~
`-noprop`	~

## 删除 Snapshot 副本

您可以使用 SnapDrive snap delete` 命令删除 Snapshot 副本。

### 步骤

1. 输入以下命令：

```
* SnapDrive snap delete （-snapname）long_snap_name [snap_name ...] （-verbose）[-force （-noprompt）]*
```



如果指定的 Snapshot 副本正在使用中，则此操作将失败。SnapDrive for UNIX 仅在删除所有 Snapshot 副本后才报告此操作已成功完成。

SnapDrive for UNIX 会删除您在 snap delete 命令行中指定的 LUN 的现有内容，并将其替换为您指定的 Snapshot 副本中的 LUN 内容。

此示例显示了要删除的内容的列表：



```
# snapdrive snap delete -v filer1:/vol/vol1/snap1 snap2 snap3
snapdrive: deleting
filer1:/vol/vol1/snap1
filer1:/vol/vol1/snap2
filer1:/vol/vol1/snap3
```

## 断开 Snapshot 副本的连接

您可以将 Snapshot 副本与 LUN，LUN 上的文件系统，磁盘组，NFS 目录或 LUN 上的共享磁盘组，主机卷和文件系统断开连接。



您可以将拆分的克隆卷与 SnapDrive for UNIX 4.2 及更高版本断开连接。

### 使用 Snapshot 断开连接操作

使用 SnapDrive snap disconnect` 命令断开多个存储系统或存储系统卷上的 Snapshot 副本的连接。

您可以使用 SnapDrive snap disconnect` 命令删除副本中 LUN，存储实体和底层 LUN 或 NFS 目录的映射。

您可以使用此命令断开跨多个存储系统卷或多个存储系统的 Snapshot 副本的连接。存储实体和卷可以位于同一存储系统或不同存储系统上。

使用此命令可断开以下任何连接：

- LUN
- 直接在 LUN 上创建的文件系统
- 在 LUN 上创建的磁盘组，主机卷和文件系统
- NFS 目录树
- 在 LUN 上创建的共享磁盘组，主机卷和文件系统

断开连接操作不会修改已连接的 Snapshot 副本。但是，默认情况下，此操作会删除由相应连接操作创建的任何临时 LUN 或克隆。



对于 LUN，LUN 上的文件系统和 LVM 实体，此命令相当于 SnapDrive storage delete`。

### 断开 Snapshot 副本连接的准则

以下是断开 LUN，存储实体或 NFS 目录的 Snapshot 副本连接的准则。

- 断开文件系统连接时，SnapDrive for UNIX 始终会删除挂载点。
- 要撤消 Snapshot 连接操作的影响，请使用 Snapshot disconnect 命令。
- 如果在 Snapshot 连接操作期间将 `enable-split-clon\_` 配置变量值设置为 on 或 sync，并在 Snapshot 断

开操作期间将其设置为 `off`，则 SnapDrive for UNIX 不会删除 Snapshot 副本中的原始卷或 LUN。

断开 **NFS** 实体的 **Snapshot** 副本连接的准则

可以从主机集群环境中的任何节点执行 Snapshot 副本的断开连接操作。使用命令选项将存储实体与特定节点断开连接。

断开包含 NFS 实体的 Snapshot 副本的连接时，请遵循以下准则：

- 如果断开使用只读权限连接的 NFS 目录树的连接，则 SnapDrive for UNIX 将执行以下操作：
  - 卸载文件系统。
  - 删除文件系统表文件中的挂载条目。
  - 删除挂载点。
  - 不会删除 Snapshot 副本目录的导出规则，该目录是在从二级主机（对父卷没有导出权限的主机）连接 NFS 文件规范时创建的。
- 如果断开使用读写权限连接的 NFS 目录树的连接，则 SnapDrive for UNIX 将执行以下操作：
  - 卸载文件系统。
  - 删除文件系统表文件中的挂载条目。
  - 删除与 FlexVol 卷克隆中的文件系统对应的 NFS 目录树。
  - 销毁底层 FlexVol 卷克隆（如果为空）。
  - 删除挂载点。

使用 **SnapDrive snap disconnect** 命令所需的信息

要断开 Snapshot 副本的连接，请指定要使用的存储实体类型，例如 LUN，磁盘组，文件系统，或主机卷。

下表提供了使用 SnapDrive snap disconnect 命令时需要提供的信息。

要求 / 选项	参数
LUN ( <code>`-lun file_spec`</code> )	LUN 的名称。包括存储器，卷和 LUN_ 的名称
磁盘组 ( <code>`-dg file_spec`</code> ) 或卷组 ( <code>`-vg file_spec`</code> )	磁盘或卷组的名称 _
文件系统 ( <code>`-fs file_spec`</code> )	` 文件系统名称 _`
主机卷 ( <code>-hostvol` file_spec`</code> ) 或逻辑卷 ( <code>-lvol` file_spec`</code> )	主机或逻辑卷的名称 _

要求 / 选项	参数
指定要用于断开 Snapshot 副本连接的存储实体的类型，并使用适当的参数提供该实体的名称。  这是 <code>file_spec</code> 参数的值。	<code>`全满`</code>
<code>~</code>	如果您希望 SnapDrive for UNIX 从 Snapshot 副本断开对象连接，即使命令行上的主机端实体包含其他实体（例如具有一个或多个主机卷的磁盘组），也可以在命令行中包含 <code>-full</code> 选项。如果不包含此选项，则只能指定空的主机端实体。
<code>`-fstype`</code>	<code>`_类型_`</code>
<code>`-vmtype`</code>	<code>`_类型_`</code>
可选：指定要使用的文件系统和卷管理器的类型。	<code>`拆分`</code>
<code>~</code>	用于在 Snapshot 连接和 Snapshot 断开操作期间拆分克隆的卷或 LUN。

## 断开 Snapshot 副本与 LUN 的连接，而不断开存储实体的连接

您可以使用 `SnapDrive snap disconnect` 命令断开包含无存储实体的 LUN 的 Snapshot 副本的连接。

### 步骤

1. 输入以下命令语法：

```
` * SnapDrive snap disconnect -lun long_lun_name [lun_name...]*`
```

SnapDrive for UNIX 将删除命令行中指定的存储实体的映射。

以下命令将删除存储系统面包机上的 luna 和 lunb 映射：

```
# snapdrive snap disconnect -lun toaster:/vol/vol1/luna lunb
```

## 断开 Snapshot 副本与存储实体的连接

您可以使用 `SnapDrive snap disconnect` 命令断开包含存储实体的 Snapshot 副本的连接。

### 步骤

1. 输入以下命令：

```
` * SnapDrive snap disconnect { -dg _ -fs _ -hostvol } file_spec [file_spec...] { -dg _ FS _ -hostvol }
file_spect [file_spec...] 【 -full】 【 -fstype type】 【 -vmtype 类型】 【 -ssplit】 *`
```

此命令必须始终以存储实体开头，例如，`-lun`，`-dg`，`-hostvol`，或`-FS`。

- 如果指定 LUN（`-lun`），则必须输入长 LUN 名称。您不能在与其它存储实体（`-vg`，`-dg`，`-fs`，`-lvol` 或`-hostvol` options）相同的命令行上使用`-lun`选项指定 LUN。
- 如果指定 NFS 挂载点，则不能在同一命令行上指定非 NFS 实体（`-vg`，`-DG`，`-FS`，`-lvol` 或`-hostvol`）。您必须使用单独的命令来指定 NFS 挂载点。



如果主机实体使用的 LUN 不属于 Snapshot 副本，则会出现错误消息。如果指定每个目标磁盘组中包含的一个子集主机卷和 / 或文件系统，则也会发生错误。

SnapDrive for UNIX 将删除命令行中指定的存储实体的映射。

此命令行会删除与主机卷 dg5/myvolume 下的所有 LUN 的映射。它会删除通过 Snapshot 连接操作创建的任何临时 LUN：

```
# snapdrive snap disconnect -hostvol dg5/myvolume
```

此命令行会删除与主机卷 dg5/myvolume 下的所有 LUN 的映射。它会删除通过 Snapshot 连接操作创建的任何临时 LUN：

```
# snapdrive snap disconnect -hostvol dg5/myvolume
```

此命令将断开与磁盘组 1（dg1）和底层 LUN 的映射。它还会删除通过 Snapshot 连接操作创建的任何临时 LUN：

```
# snapdrive snap disconnect -lun toaster:/vol/vol1/luna -dg dg1
```

此命令行将删除与文件系统 fs1 及其所在 LUN 的映射。它还会删除通过 Snapshot 连接操作创建的任何临时 LUN：

```
# snapdrive snap disconnect -fs mnt/fs1
```

此命令行将删除磁盘组 dg1，dg2 和 dg3 的映射。它会删除通过 Snapshot 连接操作创建的任何临时 LUN：

```
# snapdrive snap disconnect -dg dg1 dg2 dg3
```

以下示例将断开 Snapshot 副本与 Veritas 堆栈上的文件系统磁盘组的连接：

```
# snapdrive snap disconnect -fs /mnt/fs1_clone -fstype vxfs
delete file system /mnt/fs1_clone
- fs /mnt/fs1_clone ... deleted
- hostvol vxvm1_0/fs1_SdHv_0 ... deleted
- dg vxvm1_0 ... deleted
- LUN snoop:/vol/vol1/lunVxvm1_0 ... deleted
```

以下示例将断开 LVM 堆栈上文件系统磁盘组的 Snapshot 副本的连接：

```
# snapdrive snap disconnect -fs /mnt/fs1_clone -fstype jfs2

delete file system /mnt/fs1_clone
- fs /mnt/fs1_clone ... deleted
- hostvol lvm1_0/fs1_SdHv_0 ... deleted
- dg lvm1_0 ... deleted
- LUN snoop:/vol/vol1/lunLvm1_0 ... deleted
```

## 断开 Snapshot 副本与共享存储实体的连接

您可以使用 SnapDrive snap disconnect 命令断开包含共享存储实体的 Snapshot 副本的连接。

### 步骤

1. 输入以下命令语法：

```
` * SnapDrive snap disconnect { -dg _ -fs } file_spec [file_spec...] { -dg _ -fs } file_spec [file_spec...
...] long_snap_name [-full] [-fstype type] [-vmtype type] [-split]*`
```

此示例将断开共享文件系统的连接：

```
# snapdrive snap disconnect -fs /mnt/oracle
```

## 版权信息

版权所有 © 2023 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：政府使用、复制或公开本文档受 DFARS 252.227-7013（2014 年 2 月）和 FAR 52.227-19（2007 年 12 月）中“技术数据权利 — 非商用”条款第 (b)(3) 条规定的限制条件的约束。

本文档中所含数据与商业产品和/或商业服务（定义见 FAR 2.101）相关，属于 NetApp, Inc. 的专有信息。根据本协议提供的所有 NetApp 技术数据和计算机软件具有商业性质，并完全由私人出资开发。美国政府对这些数据的使用权具有非排他性、全球性、受限且不可撤销的许可，该许可既不可转让，也不可再许可，但仅限在与交付数据所依据的美国政府合同有关且受合同支持的情况下使用。除本文档规定的情形外，未经 NetApp, Inc. 事先书面批准，不得使用、披露、复制、修改、操作或显示这些数据。美国政府对国防部的授权仅限于 DFARS 的第 252.227-7015(b)（2014 年 2 月）条款中明确的权利。

## 商标信息

NetApp、NetApp 标识和 <http://www.netapp.com/TM> 上所列的商标是 NetApp, Inc. 的商标。其他公司和产品名称可能是其各自所有者的商标。