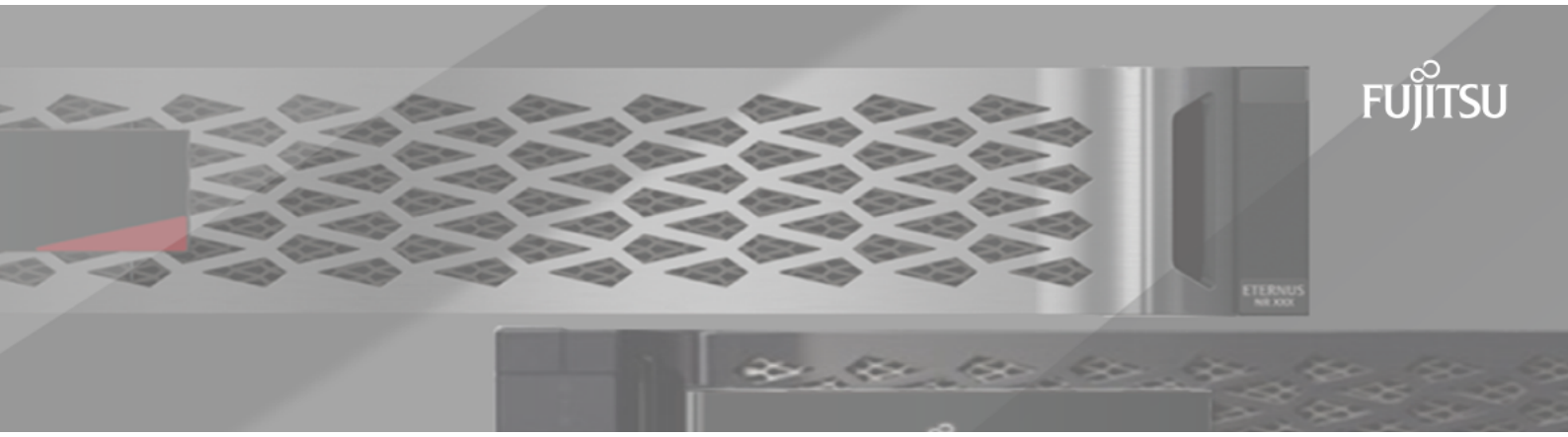
A3CA08733-A616-03



FUJITSU Storage ETERNUS AX/HX Series

データ保護パワー ガイド

目次

このガイドの対象者 4

ローカルSnapshotコピーの管理 5

カスタムSnapshotポリシーの設定 5

カスタムSnapshotポリシーを設定するタイミング 5

Snapshotジョブ スケジュールの作成 5

Snapshotポリシーの作成 6

Snapshotコピー リザーブの管理 6

Snapshotコピー リザーブを増やすタイミング 6

保護対象のファイルを削除することで想定よりもファイル スペースが少なくなる仕組み 7

Snapshotコピーのディスク使用状況の監視 8

Snapshotコピー リザーブの変更 8

Snapshotコピーの自動削除 8

Snapshotコピーからのファイルのリストア 9

NFSクライアントまたはCIFSクライアントにおけるSnapshotコピーからのファイルのリストア 9

Snapshotコピーからの単一ファイルのリストア 9

Snapshotコピーからのファイルの一部のリストア 10

Snapshotコピーからのボリューム内容のリストア 10

SnapMirrorボリューム レプリケーションの概要 12

非同期SnapMirrorディザスタ リカバリの基本 12

SnapMirror Synchronousディザスタ リカバリの基本 14

StrictSyncポリシーとSyncポリシーでサポートされるワークロードの概要 16

SnapVaultアーカイブの基本 16

SnapMirrorユニファイド レプリケーションの基本 18

SnapMirrorのデフォルトの変更（DPからXDP） 19

デスティネーション ボリュームが自動的に拡張される状況 20

ファンアウト構成およびカスケード構成のデータ保護 20

ファンアウト構成の仕組み 21

カスケード構成の仕組み 22

SnapMirrorのライセンス 23

DPOシステムの機能拡張 25

SnapMirrorボリューム レプリケーションの管理 27

SnapMirrorレプリケーションのワークフロー 27

ワンステップでのレプリケーション関係の設定 28

レプリケーション関係の設定時の各手順 29

デスティネーション ボリュームの作成 29

レプリケーション ジョブ スケジュールの作成 29

レプリケーション ポリシーのカスタマイズ 30

レプリケーション関係の作成 33

レプリケーション関係の初期化 36

例：ヴォールト-ヴォールト カスケードの設定 36

既存のDPタイプの関係からXDPへの変換 38

SnapMirror関係のタイプの変換 39

SnapMirror Synchronous関係のモードの変換 40

SnapMirror DRデスティネーション ボリュームからのデータの提供 41

デスティネーション ボリュームを書き込み可能にする 41

データ アクセスのためのデスティネーション ボリュームの設定 42

元のソース ボリュームの再アクティブ化....42
SnapMirrorデスティネーション ボリュームからのファイルのリストア.... 45
SnapMirrorデスティネーションからの単一ファイル、LUN、NVMeネームスペースのリストア.... 45
SnapMirrorデスティネーションからのボリューム内容のリストア....46
手動によるレプリケーション関係の更新.... 48
レプリケーション関係の再同期.... 48
ボリューム レプリケーション関係の削除....49
ストレージ効率の管理.... 49
SnapMirrorグローバル スロットルの使用....50
SnapMirror SVMレプリケーションの概要.... 52
SnapMirror SVMレプリケーションの管理.... 57
SVMの設定のレプリケート....57
SnapMirror SVMレプリケーションのワークフロー.... 57
SVMの設定全体のレプリケート....58
SVMレプリケーション対象からのLIFと関連ネットワークの設定の除外....59
SVMレプリケーション対象からのネットワーク、ネーム サービス、およびその他の設定の除外....60
CIFSのみ：CIFSサーバの作成.... 62
SVMレプリケーション対象からのボリュームの除外....62
初期化後に設定されたジャンクション パスの更新.... 63
SVM DRデスティネーションからのデータの提供....63
SVMのディザスタ リカバリのワークフロー....63
SVMデスティネーション ボリュームを書き込み可能にする.... 64
ソースSVMの再アクティブ化....65
ソースSVMの再アクティブ化ワークフロー....65
元のソースSVMの再アクティブ化....66
元のソースSVMの再アクティブ化（FlexGroupボリュームのみ）.... 68
ボリューム レプリケーション関係からSVMレプリケーション関係への変換.... 71
SVMレプリケーション関係の削除.... 72
SnapMirrorルート ボリューム レプリケーションの管理....74
負荷共有ミラー関係の作成および初期化.... 74
負荷共有ミラー関係の更新....75
負荷共有ミラーの昇格.... 75
SnapMirrorの技術的な詳細.... 77
パス名のパターン マッチングの使用.... 77
拡張クエリを使用した多数のSnapMirror関係の操作....77
ミラー-ヴォールト構成における共通のSnapshotコピーの保持.... 78
SnapMirror関係に対応したONTAPバージョン.... 79
SnapMirrorの制限事項.... 79
詳細情報の入手方法.... 81
著作権に関する情報....82
登録商標....83
マニュアルの更新について.... 84

データ保護パワー ガイドの対象者

このマニュアルでは、ローカルのETERNUS AX/HXシリーズでSnapshotコピーを管理する方法と、SnapMirrorを使用してリモート システムにSnapshotコピーをレプリケートする方法について説明します。ディザスタ リカバリまたは長期保持用のSnapshotコピーをレプリケートできます。

注：このデータ保護パワー ガイドは、*SnapMirror*と*SnapVault*を使用したデータ保護に代わるものです。

このマニュアルは、次のような状況で使用することを想定しています。

- ONTAPのバックアップとリカバリの機能の範囲について理解する必要がある。
- ONTAP System Manager、自動スクリプト ツール、SnapCenter製品ではなく、コマンドライン インターフェイス（CLI）を使用する必要がある。

  ONTAP System Managerを使用する場合は、*ONTAP System Manager*を使用したクラスタ管理を参照してください。

  ONTAP System Managerを使用したクラスタ管理

  SnapCenterを使用する場合は、SnapCenter製品のマニュアルを参照してください。
- ソース クラスタとデスティネーション クラスタ間およびソースSVMとデスティネーションSVM間でピア関係を作成済みである。

  クラスタ / SVMピアリング エクスプレス ガイド
- ボリュームまたはSVMをETERNUS AXまたはETERNUS HXストレージ システムからETERNUS AXまたはETERNUS HXストレージ システムにバックアップする。

ONTAP 9.8ではSnapMirrorビジネス継続性機能が追加されました。目標復旧時間ゼロ（ゼロRTO）または透過的アプリケーション フェイルオーバー（TAF）によって、SAN環境でビジネス クリティカルなアプリケーションを自動的にフェイルオーバーできます。

ベストプラクティスに基づいてボリュームのバックアップとリカバリをすばやく実行する場合は、以下のマニュアルを参照してください。

- ボリュームのディザスタ リカバリの準備

  ボリューム ディザスタ リカバリ設定エクスプレス ガイド
- ボリュームのディザスタ リカバリ

  ボリューム ディザスタ リカバリエクスプレス ガイド
- SnapVaultを使用したボリュームのバックアップ

  SnapVault®によるボリューム バックアップ エクスプレス ガイド
- SnapVaultを使用したボリュームのリストア

  SnapVault®によるボリュームリストア エクスプレス ガイド

構成または概念の詳細な情報が必要な場合は、以下のマニュアルを参照してください。

- ONTAPの背景にある概念

  コンセプト ガイド
- MetroCluster構成での同期ディザスタ リカバリ

  MetroCluster管理およびディザスタ リカバリ ガイド
- SnapLockボリューム内のWORMファイルのデータ保護

  SnapLock®テクノロジを使用したアーカイブとコンプライアンス パワー ガイド
- テープ テクノロジを使用したデータ保護
  - NDMP構成エクスプレス ガイド
  - データ保護：テープ バックアップおよび リカバリ ガイド

ローカルSnapshotコピーの管理

*Snapshot*コピーは、ボリュームのポイントインタイム イメージ（読み取り専用）です。イメージにはSnapshotコピーが最後に作成されたあとに発生したファイルへの変更だけが記録されるため、ストレージ スペースは最小限しか消費せず、パフォーマンスのオーバーヘッドもわずかです。

Snapshotコピーを使用すると、ボリュームの内容全体をリストアしたり、個々のファイルやLUNをリカバリしたりできます。Snapshotコピーは、ボリューム上の`.snapshot`ディレクトリに格納されます。

ONTAP 9.7以降では、1つのボリュームに最大1023個のスナップショット・コピーを含めることができます。

カスタムSnapshotポリシーの設定

*Snapshot*ポリシーは、Snapshotコピーの作成方法を定義します。このポリシーは、Snapshotコピーを作成するタイミング、保持するコピーの数、およびSnapshotコピーに名前を付ける方法を指定します。たとえば、毎日午前12時10分に1つのSnapshotコピーを作成し、直近の2つのコピーを保持して、それに「daily.*timestamp*」という名前を付ける、と指定できます。

ボリュームのデフォルト ポリシーでは、次のスケジュールでSnapshotコピーが自動的に作成されます。新しいコピー用のスペースを確保するために、最も古いSnapshotコピーが削除されます。

- 最大6つの時間単位Snapshotコピーが毎時5分に作成されます。
- 最大2つの日単位Snapshotコピーが月曜日から土曜日の午前0時10分に作成されます。
- 最大2つの週単位Snapshotコピーが毎週日曜日の午前0時15分に作成されます。

ボリュームの作成時にSnapshotポリシーを指定しなかった場合は、そのボリュームを含むStorage Virtual Machine（SVM）に関連付けられているSnapshotポリシーが継承されます。

カスタムSnapshotポリシーを設定するタイミング

デフォルトのSnapshotポリシーがボリュームに適していない場合は、Snapshotコピーの頻度、保持設定、および名前を変更するカスタム ポリシーを設定できます。スケジュールは、主にアクティブ ファイルシステムの変更率によって決まります。

使用頻度の高いファイルシステム（データベースなど）は1時間ごとにバックアップする必要がありますが、ほとんど使用されないファイルは1日に1回バックアップすれば十分です。データベースであっても、通常は1日に1～2回フル バックアップを実行しますが、トランザクション ログのバックアップは1時間ごとに行います。

その他の要因としては、組織におけるファイルの重要性、サービス レベル アグリーメント（SLA）、目標復旧時点（RPO）、および目標復旧時間（RTO）があります。通常は、Snapshotコピーを必要な数だけ保持してください。

Snapshotジョブ スケジュールの作成

Snapshotポリシーには、Snapshotコピーのジョブ スケジュールが少なくとも1つ必要です。`job schedule cron create`コマンドを使用すると、ジョブ スケジュールを作成できます。

このタスクについて

デフォルトでは、ONTAPがSnapshotコピーの名前を作成する際には、ジョブ スケジュール名にタイムスタンプを追加します。

日にちと曜日の両方に値を指定すると、それぞれ個別に判断されます。たとえば、曜日に`Friday`が指定され、日にちに`13`が指定されたcronスケジュールは、13日の金曜日だけでなく、各月の毎週金曜日と13日に実行されます。

手順

ジョブ スケジュールを作成します。 `job schedule cron create -name job_name -month month -dayofweek day_of_week -day day_of_month -hour hour -minute minute`

-month、-dayofweek、-hourに`all`を指定すると、ジョブを毎月、特定の曜日、時間にそれぞれ実行できます。

次の例は、土曜日の午前3時に実行する`myweekly`というジョブ スケジュールを作成します。

```
cluster1::> job schedule cron create -name myweekly -dayofweek "Saturday" -
hour 3 -minute 0
```

Snapshotポリシーの作成

Snapshotポリシーは、Snapshotコピーを作成するタイミング、保持するコピーの数、およびSnapshotコピーに名前を付ける方法を指定します。たとえば、毎日午前12時10分に1つのSnapshotコピーを作成し、直近の2つのコピーを保持して、それに「daily.*timestamp*」という名前を付ける、と指定できます。Snapshotポリシーには最大5つのジョブ スケジュールを設定できます。

このタスクについて

デフォルトでは、ONTAPがSnapshotコピーの名前を作成する際には、ジョブ スケジュール名にタイムスタンプを追加します。

```
daily.2017-05-14_0013/              hourly.2017-05-15_1106/
daily.2017-05-15_0012/              hourly.2017-05-15_1206/
hourly.2017-05-15_1006/             hourly.2017-05-15_1306/
```

必要に応じて、プレフィックスをジョブ スケジュール名に置き換えることができます。

snapmirror-labelはSnapMirrorレプリケーション用のオプションです。詳細については、ポリシーのルールの定義を参照してください。

手順

Snapshotポリシーを作成します。 `volume snapshot policy create -vserver SVM -policy policy_name enabled true|false -schedule1 schedule1_name -count1 copies_to_retain -prefix1 snapshot_prefix -snapmirror-label1 snapshot_label ... -schedule1 schedule5_name -count5 copies_to_retain-prefix5 snapshot_prefix -snapmirror-label5 snapshot_label`
次の例は、`daily`スケジュールで実行する`snap_policy_daily`というSnapshotポリシーを作成します。このポリシーには最大5つのSnapshotコピーが含まれます。それぞれの名前は`daily`.***timestamp***になり、SnapMirrorラベルは`daily`です。

```
cluster1::> volume snapshot policy create -vserver vs0 -policy
 snap_policy_daily -schedule1 daily -count1 5 -snapmirror-label1 daily
```

Snapshotコピー リザーブの管理

*Snapshot*コピー リザーブを使用すると、Snapshotコピー用に一定の割合のディスク スペース（デフォルトでは5%）が確保されます。Snapshotコピーでは、Snapshotコピー リザーブを使い切るとアクティブ ファイルシステムのスペースが使用されるため、必要に応じてSnapshotコピー リザーブを増やします。また、リザーブがフルになるとSnapshotコピーを自動削除することもできます。

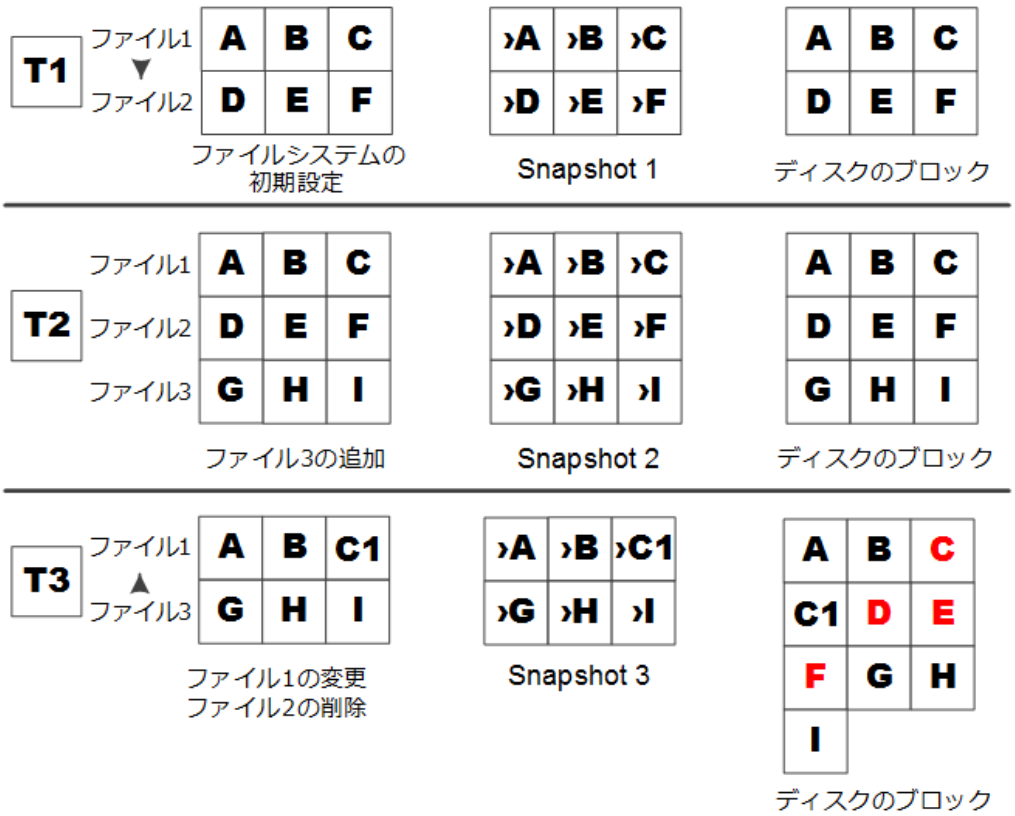
Snapshotコピー リザーブを増やすタイミング

Snapshotリザーブを増やすかどうかを判断する際には、Snapshotコピーには、前回のSnapshotコピー作成後のファイルに対する変更のみが記録されるという点を考慮することが重要です。Snapshotのコピーによってディスク スペースが消費されるのは、アクティブ ファイルシステムのブロックが変更または削除された場合のみです。

つまり、Snapshotコピーで使用されるディスク スペース容量を判断する際の重要な要素はファイルシステムの変更率です。作成するSnapshotコピーの数にかかわらず、アクティブ ファイルシステムが変更されていない場合、Snapshotコピーはディスク スペースを消費しません。

たとえば、データベース トランザクション ログを格納しているFlexVolには、変更率の増加に対応するために20%のSnapshotコピー リザーブが用意されている場合があります。より多くのSnapshotコピーを作成して、データベースに対する高頻度の更新をキャプチャするだけでなく、Snapshotコピー リザーブのサイズを拡張して、Snapshotコピーが消費する追加のディスク スペースを処理することができます。

ヒント： Snapshotコピーは、ブロックのコピーではなくブロックへのポインタで構成されています。ポインタはブロック上の「要求」とみなすことができます。ONTAPでは、Snapshotコピーが削除されるまでブロックを「保持」します。



Snapshotのコピーによってディスク スペースが消費されるのは、アクティブなファイルシステムのブロックが変更または削除された場合のみです。

保護対象のファイルを削除することで想定よりもファイル スペースが少なくなる仕組み

Snapshotコピーは、ブロックを使用していたファイルを削除したあともそのブロックをポイントします。そのため、Snapshotコピー リザーブを使い切ると、期待に反した結果を引き起こす可能性があります。つまり、ファイルシステム全体を削除することで、ファイルシステムが占有するスペースよりも、使用可能なスペースが少なくなります。

次の例を考えてみましょう。ファイルを削除する前の、dfコマンドの出力は次のとおりです。

```
Filesystem           kbytes  used    avail  capacity
/vol/vol0/           3000000 3000000 0       100%
/vol/vol0/.snapshot 1000000 500000  500000   50%
```

ファイルシステム全体を削除してボリュームのSnapshotコピーを作成すると、dfコマンドによって次の出力が生成されます。

```
Filesystem           kbytes  used    avail  capacity
/vol/vol0/           3000000 2500000 500000   83%
/vol/vol0/.snapshot 1000000 3500000 0       350%
```

この出力に示すとおり、アクティブ ファイルシステムで使用されていた3GBは、現在はSnapshotコピーで使用されています（削除前に使用されていた0.5GBを含む）。

Snapshotコピーで使用されるディスク スペースはSnapshotコピー リザーブを超えているため、アクティブ ファイル用にリザーブされたスペースに2.5GBが「オーバーフロー」します。想定していた3GBについては、ファイル用の空きスペースとして0.5GBしか残りません。

Snapshotコピーのディスク使用状況の監視

dfコマンドを使用して、Snapshotコピーのディスク使用状況を監視できます。このコマンドは、アクティブ ファイルシステムとSnapshotコピー リザーブの空きスペースの量を表示します。

手順

Snapshotコピーのディスク使用状況を表示します。 `df`
次の例は、Snapshotコピーのディスク使用状況を示しています。

```
cluster1::> df
Filesystem           kbytes  used    avail  capacity
/vol/vol0/           3000000 3000000 0       100%
/vol/vol0/.snapshot 1000000 500000  500000   50%
```

Snapshotコピー リザーブの変更

Snapshotコピー リザーブのサイズを拡張して、アクティブ ファイルシステム用にリザーブしたスペースがSnapshotコピーによって使用されないようにすることができます。Snapshotコピー用のスペースが不要になったら、Snapshotコピー リザーブのサイズを縮小できます。

手順

Snapshotコピー リザーブを変更します。`volume modify -vserver` *`SVM`* `-volume` *`volume`* `-percent-snapshot-space` *`snap_reserve`*

コマンド構文全体については、マニュアル ページを参照してください。

次の例は、`vol1`のSnapshotコピー リザーブを10%に設定します。

```
cluster1::> volume modify -vserver vs0 -volume vol1 -percent-snapshot-space
 10
```

Snapshotコピーの自動削除

`volume snapshot autodelete modify`コマンドを使用すると、Snapshotリザーブを超えた場合にSnapshotコピーの自動削除をトリガーできます。デフォルトでは、最も古いSnapshotコピーが最初に削除されます。

このタスクについて

LUNクローンとファイル クローンは、削除するSnapshotコピーがなくなると削除されます。

手順

Snapshotコピーを自動削除します。 `volume snapshot autodelete modify -vserver` *`SVM`* `-volume` *`volume`* `-enabled true|false -trigger volume|snap_reserve`

コマンド構文全体については、マニュアル ページを参照してください。

次の例では、Snapshotコピー リザーブを使い切った場合に`vol1`のSnapshotコピーが自動削除されます。

```
cluster1::> volume modify -vserver vs0 -volume vol1 -enabled true -trigger
 snap_reserve
```

Snapshotコピーからのファイルのリストア

Snapshotコピーを使用して、個々のファイルまたはLUNをリカバリしたり、ボリュームの内容全体をリストアしたりできます。NFSクライアントまたはCIFSクライアントから、およびストレージ システムからファイルをリストアできます。この機能にはSnapRestoreライセンスが必要です。このライセンスは、 Premium Bundleソフトウェアを購入すると自動的にインストールされます。

NFSクライアントまたはCIFSクライアントにおけるSnapshotコピーからのファイルのリストア

NFSクライアントまたはCIFSクライアント上のユーザは、Snapshotコピーからファイルを直接リストアできます。ストレージ システム管理者の操作は必要ありません。

ファイルシステム内のすべてのディレクトリには、NFSユーザとCIFSユーザがアクセスできる`.snapshot`というサブディレクトリが含まれています。`.snapshot`サブディレクトリには、ボリュームのSnapshotコピーに対応するサブディレクトリが含まれています。

```
$ ls .snapshot
daily.2017-05-14_0013/              hourly.2017-05-15_1106/
daily.2017-05-15_0012/              hourly.2017-05-15_1206/
hourly.2017-05-15_1006/             hourly.2017-05-15_1306/
```

各サブディレクトリには、Snapshotコピーが参照するファイルが含まれています。ファイルを誤って削除または上書きしたユーザは、Snapshotサブディレクトリから親の読み取り / 書き込みディレクトリにファイルをコピーすることで、その読み取り / 書き込みディレクトリにファイルをリストアできます。

```
$ ls my.txt
ls: my.txt: No such file or directory
$ ls .snapshot
daily.2017-05-14_0013/              hourly.2017-05-15_1106/
daily.2017-05-15_0012/              hourly.2017-05-15_1206/
hourly.2017-05-15_1006/             hourly.2017-05-15_1306/
$ ls .snapshot/hourly.2017-05-15_1306/my.txt
my.txt
$ cp .snapshot/hourly.2017-05-15_1306/my.txt .
$ ls my.txt
my.txt
```

Snapshotコピーからの単一ファイルのリストア

`volume snapshot restore-file`コマンドを使用すると、Snapshotコピーから単一ファイルまたはLUNをリストアできます。既存のファイルを置き換えない場合は、読み書き可能な親ボリュームの別の場所にファイルをリストアできます。

このタスクについて

既存のLUNをリストアする場合は、LUNクローンが作成され、Snapshotコピーの形でバックアップされます。リストア処理中、そのLUNに対して読み書きを行うことができます。

デフォルトでは、ストリームを含むファイルがリストアされます。

手順

1. ボリューム内のSnapshotコピーの一覧を表示します。`volume snapshot show -vserver` *`SVM`* `-volume` *`volume`*

   コマンド構文全体については、マニュアル ページを参照してください。

   次の例は、`vol1`内のSnapshotコピーを示しています。

```
clus1::> volume snapshot show -vserver vs1 -volume vol1

Vserver Volume Snapshot                State    Size  Total% Used%
------- ------ ---------- ----------- ------   -----  ------ -----
vs1  vol1   hourly.2013-01-25_0005  valid   224KB     0%    0%
               daily.2013-01-25_0010   valid   92KB      0%    0%
               hourly.2013-01-25_0105  valid   228KB     0%    0%
               hourly.2013-01-25_0205  valid   236KB     0%    0%
               hourly.2013-01-25_0305  valid   244KB     0%    0%
               hourly.2013-01-25_0405  valid   244KB     0%    0%
               hourly.2013-01-25_0505  valid   244KB     0%    0%

7 entries were displayed.
```

2. Snapshotコピーからファイルをリストアします。`volume snapshot restore-file -vserver` *`SVM`* `-volume` *`volume`* `-snapshot` *`snapshot`* `-path` *`file_path`* `-restore-path` *`destination_path`*

   コマンド構文全体については、マニュアル ページを参照してください。

   次の例は、`myfile.txt`ファイルをリストアします。

```
cluster1::> volume snapshot restore-file -vserver vs0 -volume vol1 -
snapshot daily.2013-01-25_0010 -path /myfile.txt
```

Snapshotコピーからのファイルの一部のリストア

`volume snapshot partial-restore-file`コマンドを使用すると、一定の範囲のデータをSnapshotコピーからLUNまたはNFS / CIFSコンテナ ファイルにリストアできます。この処理では、データの開始バイト オフセットとバイト数を把握していることを前提とします。このコマンドでは、同じLUN内に複数のデータベースを格納するホスト上のいずれかのデータベースをリストアできます。

手順

1. ボリューム内のSnapshotコピーの一覧を表示します。`volume snapshot show -vserver` *`SVM`* `-volume` *`volume`*

   コマンド構文全体については、マニュアル ページを参照してください。

   次の例は、`vol1`内のSnapshotコピーを示しています。

```
clus1::> volume snapshot show -vserver vs1 -volume vol1

Vserver Volume Snapshot                State    Size  Total% Used%
------- ------ ---------- ----------- ------   -----  ------ -----
vs1  vol1   hourly.2013-01-25_0005  valid   224KB     0%    0%
               daily.2013-01-25_0010   valid   92KB      0%    0%
               hourly.2013-01-25_0105  valid   228KB     0%    0%
               hourly.2013-01-25_0205  valid   236KB     0%    0%
               hourly.2013-01-25_0305  valid   244KB     0%    0%
               hourly.2013-01-25_0405  valid   244KB     0%    0%
               hourly.2013-01-25_0505  valid   244KB     0%    0%

7 entries were displayed.
```

2. Snapshotコピーからファイルの一部をリストアします。`volume snapshot partial-restore-file -vserver` *`SVM`* `-volume` *`volume`* `-snapshot` *`snapshot`* `-path` *`file_path`* `-start-byte` *`starting_byte`* `-byte-count` *`byte_count`*

   開始バイト オフセットとバイト数は4,096の倍数でなければなりません。

   次の例は、`myfile.txt`ファイルの最初の4,096バイトをリストアします。

```
cluster1::> volume snapshot partial-restore-file -vserver vs0 -volume
 vol1 -snapshot daily.2013-01-25_0010 -path /myfile.txt -start-byte 0 -
byte-count 4096
```

Snapshotコピーからのボリューム内容のリストア

`volume snapshot restore`コマンドを使用すると、Snapshotコピーからボリュームの内容をリストアできます。

このタスクについて

ボリュームにSnapMirror関係が設定されている場合は、Snapshotコピーからリストアしたあと、すぐにボリュームのすべてのミラー コピーを手動でレプリケートします。レプリケートしないと、ミラー コピーを使用できなくなり、削除および再作成が必要になることがあります。

手順

1. ボリューム内のSnapshotコピーの一覧を表示します。`volume snapshot show -vserver *SVM* -volume *volume*`

   コマンド構文全体については、マニュアル ページを参照してください。

   次の例は、vol1内のSnapshotコピーを示しています。

```
clus1::> volume snapshot show -vserver vs1 -volume vol1

Vserver Volume Snapshot                State    Size  Total% Used%
------- ------ ---------- ----------- ------   -----  ------ -----
vs1  vol1   hourly.2013-01-25_0005  valid   224KB     0%    0%
               daily.2013-01-25_0010   valid   92KB      0%    0%
               hourly.2013-01-25_0105  valid   228KB     0%    0%
               hourly.2013-01-25_0205  valid   236KB     0%    0%
               hourly.2013-01-25_0305  valid   244KB     0%    0%
               hourly.2013-01-25_0405  valid   244KB     0%    0%
               hourly.2013-01-25_0505  valid   244KB     0%    0%

7 entries were displayed.
```

2. Snapshotコピーからボリュームの内容をリストアします。`volume snapshot restore -vserver *SVM* -volume *volume* -snapshot *snapshot*`

   コマンド構文全体については、マニュアル ページを参照してください。

   次の例は、vol1の内容をリストアします。

```
cluster1::> volume snapshot restore -vserver vs0 -volume vol1 -snapshot
 daily.2013-01-25_0010
```

SnapMirrorボリューム レプリケーションの概要

従来、ONTAPのレプリケーション テクノロジは、ディザスタ リカバリ（DR）とデータ アーカイブのニーズに対応してきました。

関連概念

非同期SnapMirrorディザスタ リカバリの基本（12ページ）
SnapVaultアーカイブの基本（16ページ）
SnapMirrorユニファイド レプリケーションの基本（18ページ）
SnapMirrorのデフォルトの変更（DPからXDP）（19ページ）
ファンアウト構成およびカスケード構成のデータ保護（20ページ）

非同期SnapMirrorディザスタ リカバリの基本

*SnapMirror*は、地理的に離れたサイトのプライマリ ストレージからセカンダリ ストレージへのフェイルオーバー用に設計されたディザスタ リカバリ テクノロジです。名前が示すように、SnapMirrorは本番データの複製（ミラー）をセカンダリ ストレージに作成し、プライマリ サイトに災害が発生しても、セカンダリ ストレージから引き続きデータを提供できるようにします。

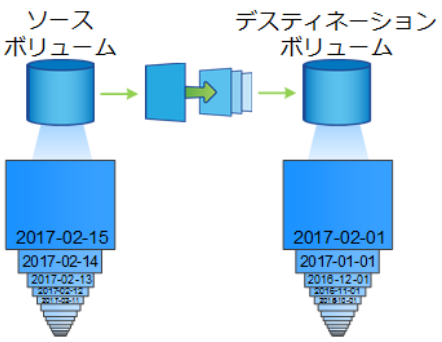
プライマリ サイトから引き続きデータを提供できる場合は、必要なデータをプライマリ サイトに戻すことができます。ミラーからクライアントを提供することはありません。 フェイルオーバーの事例で示すように、ミラーリングされたストレージからデータを効率的に提供するには、セカンダリ システム上のコントローラがプライマリ システム上のコントローラと同じであるか、ほぼ同じである必要があります。

データ保護関係

データのミラーリングはボリューム単位で行われます。プライマリ ストレージのソース ボリュームと、セカンダリ ストレージのデスティネーション ボリュームの関係はデータ保護関係と呼ばれます。ボリュームが含まれるクラスタとボリュームからデータを提供するSVMのピア関係が確立されている必要があります。ピア関係にあることで、クラスタとSVM間でデータをセキュアにやり取りできます。

クラスタ / SVMピアリング エクスプレス ガイド

次の図は、SnapMirrorデータ保護関係を示しています。



通常、SnapMirrorのデータ保護関係は、ソース ボリューム上にあるSnapshotコピーをミラーリングします。

データ保護関係の範囲

ボリューム間またはボリュームを所有するSVM間で直接データ保護関係を作成できます。*SVM*データ保護関係では、SVMのすべてまたは一部の設定がNFSエクスポートおよびSMB共有からRBACにレプリケートされます。また、SVMが所有するボリューム内のデータもレプリケートされます。

次に示す2つの特殊なデータ保護アプリケーションにSnapMirrorを使用することもできます。

- SVMルート ボリュームの負荷共有ミラー コピーを作成すると、ノードに障害やフェイルオーバーが発生したときに引き続きデータにアクセスできます。
- *SnapLock*ボリューム間のデータ保護関係を作成すると、WORMファイルをセカンダリ ストレージにレプリケートできます。

  SnapLock®テクノロジを使用したアーカイブとコンプライアンス パワー ガイド

SnapMirrorデータ保護関係を初期化する方法

SnapMirrorを初めて起動すると、ソース ボリュームからデスティネーション ボリュームへのベースライン転送が実行されます。関係の*SnapMirror*ポリシーでは、ベースラインおよび更新の内容を定義します。

デフォルトのSnapMirrorポリシーである`MirrorAllSnapshots`に基づくベースライン転送では、次の手順を実行します。

- ソース ボリュームのSnapshotコピーを作成します。
- Snapshotコピーおよびコピーが参照するすべてのデータ ブロックをデスティネーション ボリュームに転送します。
- 「アクティブな」ミラーが破損した場合に備えて、ソース ボリューム上の、残りの最新ではないSnapshotコピーをデスティネーション ボリュームに転送します。

SnapMirrorデータ保護関係を更新する方法

更新は、設定されたスケジュールに従って非同期に行われます。保持処理によって、ソース上のSnapshotポリシーがミラーリングされます。

`MirrorAllSnapshots`ポリシーに基づく各更新では、SnapMirrorがソース ボリュームのSnapshotコピーを作成し、そのSnapshotコピーと前回の更新後に作成されたすべてのSnapshotコピーを転送します。以下に示す`MirrorAllSnapshots`ポリシーの`snapmirror policy show`コマンドの出力では、次の点に注意してください。

- `Create Snapshot` が「true」です。これは、SnapMirrorが関係を更新するときに、`MirrorAllSnapshots`によってSnapshotコピーが作成されることを示します。
- `MirrorAllSnapshots`には、「sm_created」および「all_source_snapshots」というルールがあります。これは、SnapMirrorが関係を更新するときに、SnapMirrorで作成されたSnapshotコピーと前回の更新後に作成されたすべてのSnapshotコピーが転送されることを示します。

```
cluster_dst::> snapmirror policy show -policy MirrorAllSnapshots -instance

                      Vserver: vs0
      SnapMirror Policy Name: MirrorAllSnapshots
      SnapMirror Policy Type: async-mirror
                Policy Owner: cluster-admin
                 Tries Limit: 8
           Transfer Priority: normal
   Ignore accesstime Enabled: false
     Transfer Restartability: always
 Network Compression Enabled: false
             Create Snapshot: true
                     Comment: Asynchronous SnapMirror policy for mirroring all snapshots
                              and the latest active file system.
       Total Number of Rules: 2
                  Total Keep: 2
                       Rules: SnapMirror Label     Keep  Preserve Warn Schedule Prefix
                              ----------------     ----  -------- ---- -------- ------
                              sm_created              1  false       0 -         -
                              all_source_snapshots    1  false       0 -         -
```

MirrorLatestポリシー

事前設定されたMirrorLatestポリシーは、MirrorAllSnapshotsとまったく同様に機能します。ただし、初期化および更新の際に転送されるのは、SnapMirrorで作成されたSnapshotコピーのみです。

```
                    Rules: SnapMirror Label     Keep  Preserve Warn Schedule Prefix
                           ----------------     ----  -------- ---- -------- ------
                           sm_created              1  false       0 -         -
```

SnapMirror Synchronousディザスタ リカバリの基本

16GB以上のメモリを搭載したすべてのETERNUS HXプラットフォームとETERNUS AXシリーズでSnapMirror Synchronous（SM-S）テクノロジがサポートされます。SnapMirror Synchronousテクノロジはノード単位でライセンスされる機能で、ボリューム レベルの同期データ レプリケーションを提供します。

この機能を使用すると、データ損失ゼロが求められる金融や医療などの業種で、同期レプリケーションに関する規制や国の規定に対応できます。

SnapMirror Synchronousレプリケーションのノードあたりの最大処理数は、コントローラのモデルによって異なります。

表 1 :

プラットフォーム	ノードあたりの最大SnapMirror Synchronous処理数
ETERNUS AX	80
ETERNUS HX	40

サポートされる機能

SnapMirror Synchronousテクノロジは、NFSv3、FC、およびiSCSIの各プロトコルをサポートしており、いずれのネットワークでもレイテンシは10ミリ秒以内に抑えられます。

ONTAP 9.7のSnapMirror Synchronousテクノロジでは、次の機能がサポートされています。

- アプリケーションで作成されたSnapshotコピーのレプリケーション

  SnapMirrorラベルがSyncポリシーまたはStrictSyncポリシーに関連付けられたルールに一致するSnapshotコピーのみが対象です。Snapshotポリシーを使用して作成され、スケジュール設定されたSnapshotコピーはレプリケートされません。
- FC-NVMe
- LUNクローンとNVMeネームスペース クローン

  アプリケーションで作成されたSnapshotコピーに基づくLUNクローンもサポートされます。

ONTAP 9.7のSnapMirror Synchronousテクノロジーでは、次の機能がサポートされています。ソースおよび宛先クラスタ内のすべてのノードがONTAP 9.7を実行している場合：

- NFS V 4.0およびNFS V 4.1
- SMB 2.0以降
- 混合プロトコル・アクセス(NFS V 3およびSMB/CIFS)
- SnapMirrorの同期関係の1次ボリューム上のウイルス対策
- SnapMirror同期関係の1次ボリュームのハード・クォータまたはソフト・クォータ

  クォータのルールはターゲットにレプリケートされない;そのため、quotaデータベースは宛先に複製されません。
- SnapMirror同期関係の1次ボリュームのFPolicy
- SnapMirror同期ミラー・ミラー・カスケード

SnapMirror同期関係のデスティネーションボリュームからの関係は、非同期SnapMirror関係である必要があります。
- 高メタデータ動作周波数制限の除去
- TLS 1.2暗号化を使用した、送信中の機密データのセキュリティ
- クローンの自動削除

サポートされない機能

Synchronous SnapMirror関係では、次の機能はサポートされません。

- SVM DR
- SANアクセスとNASアクセスの混在

  SnapMirror Synchronous関係のプライマリ ボリュームが提供できるのは、NASデータまたはSANデータのどちらか一方です。SnapMirror Synchronous関係のプライマリ ボリュームからSANアクセスとNASアクセスの両方を行うことはできません。
- SANアクセスとNVMeアクセスの混在

  LUNとNVMeネームスペースを同じボリュームまたはSVMで使用することはできません。
- SnapLockボリューム
- FlexGroupボリューム
- FlexCacheボリューム
- SnapRestore
- DP_Optimized（DPO）システム
- デスティネーション ボリュームでのダンプおよびSMTapeを使用したテープ バックアップまたはリストア
- ソース ボリュームへのテープベースのリストア
- ソース ボリュームのしきい値の下限（最小QoS）
- ファンアウト構成で確立できるSnapMirror Synchronous関係は1つだけで、ソース ボリュームからの残りの関係はすべて非同期SnapMirror関係にする必要がある。
- グローバル スロットル

動作モード

SnapMirror Synchronousには、使用するSnapMirrorポリシーに基づいて2つの動作モードがあります。

Syncモード

Syncモードでは、最初にプライマリ ストレージへのI/Oがセカンダリ ストレージにレプリケートされます。その後、プライマリ ストレージにI/Oが書き込まれ、I/Oの発行元のアプリケーションに確認応答が送信されます。何らかの理由でセカンダリ ストレージへの書き込みが完了しない場合、アプリケーションはプライマリ ストレージへの書き込みを継続できます。エラー状態が解消されると自動的にセカンダリ ストレージが再同期され、プライマリ ストレージからセカンダリ ストレージへの同期モードでのレプリケーションが再開されます。

Syncモードでは、セカンダリ レプリケーションに失敗しないかぎり、RPO=0と非常に低いRTOを実現できます。セカンダリ レプリケーションに失敗した場合のRPOとRTOは、正確には予測できませんが、失敗の原因となった問題の解決に要する時間と再同期が完了するまでの時間に相当します。

StrictSyncモード

SnapMirror Synchronousは、必要に応じてStrictSyncモードで実行できます。何らかの理由でセカンダリ ストレージへの書き込みが完了しない場合、アプリケーションI/Oが中断され、プライマリ ストレージとセカンダリ ストレージが同一に保たれます。プライマリへのアプリケーションI/Oは、SnapMirror関係のステー

タスがInSyncに戻るまで再開されません。プライマリ ストレージで障害が発生した場合は、フェイルオーバー後にセカンダリ ストレージでアプリケーションI/Oを再開できます。この場合、データ損失は発生しません。

StrictSyncモードのRPOは常にゼロで、RTOも非常に低く抑えられます。

関係のステータス

SnapMirror Synchronous関係のステータスは、通常の動作時は常にInSyncです。何らかの理由でSnapMirror転送に失敗すると、デスティネーションがソースと同期されていない状態になり、ステータスがOutofSyncになることがあります。

SnapMirror Synchronous関係については、一定の間隔で自動的にステータス（InSyncまたはOutofSync）がチェックされます。関係のステータスがOutofSyncになると、ONTAPは自動的に再同期プロセスを開始してInSyncステータスに戻します。再同期プロセスが実行されるのは、ソースまたはデスティネーションでの計画外のストレージ フェイルオーバーやネットワークの停止などによって転送に失敗した場合のみです。snapmirror quiesceやsnapmirror breakなどの処理をユーザが開始した場合は、自動再同期は実行されません。

StrictSyncモードでは、SnapMirror Synchronous関係のステータスがOutofSyncになると、プライマリ ボリュームへのI/O処理がすべて停止します。Syncモードでは、SnapMirror Synchronous関係がOutofSync状態になってもプライマリは停止せず、I/O処理が継続されます。

StrictSyncポリシーとSyncポリシーでサポートされるワークロードの概要

StrictSyncポリシーとSyncポリシーでは、FC、iSCSI、FC-NVMeの各プロトコルを使用したLUNベースのすべてのアプリケーションに加え、データベース、VMwareクォータ、CIFSなどのエンタープライズ アプリケーションについてはNFSv3およびNFSv4プロトコルもサポートされます。EDA（デザイン オートメーション）、ホーム ディレクトリ、ソフトウェア ビルド ワークロードなどのエンタープライズ ファイルサービスにSnapMirror Synchronousを使用できます。

ONTAP 9.7では、同期ポリシーについて、NFSv 3またはNFSv 4ワークロードの選択時にいくつかの重要な点を考慮する必要があります。Syncポリシーは高い読み取りまたは書き込みI/Oワークロードを処理できるため、ワークロードによるデータの読み取りまたは書き込み操作の量は考慮されません。ONTAP 9.7では、ファイルの作成、ディレクトリの作成、ファイル権限の変更、またはディレクトリ権限の変更が多すぎるワークロードは適切でない場合があります(これらは高メタデータ・ワークロードと呼ばれる)。メタデータ比率の高いワークロードの典型的な例としては、複数のテスト ファイルを作成して自動化の実行後にファイルを削除するDevOpsワークロードがあります。また、コンパイル時に複数の一時ファイルを生成する並列ビルド ワークロードなども含まれます。メタデータの書き込みアクティビティの比率が高いと、ミラー間の同期が一時的に解除され、クライアントからの読み取りや書き込みのIOが停止することがあります。

これらの制限はなくなり、ホーム ディレクトリやソフトウェア ビルド ワークロードなどのマルチユーザ環境を含むエンタープライズ ファイルサービスのワークロードにSnapMirror Synchronousを使用できるようになりました。

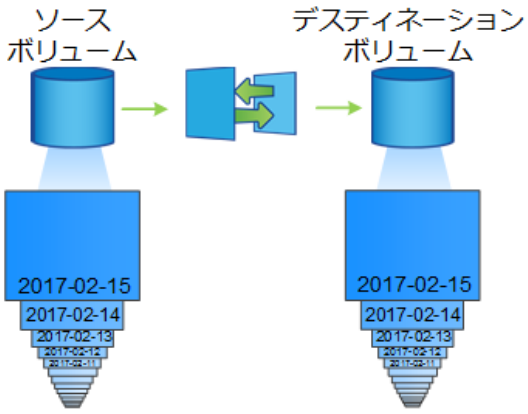
SnapVaultアーカイブの基本

SnapVaultは、基準への準拠およびその他のガバナンス関連の目的で、ディスクツーディスクのSnapshotコピーのレプリケーションを行うために設計されたアーカイブ テクノロジです。SnapMirror関係では、通常、ソース ボリューム内のSnapshotコピーだけがデスティネーションに含まれますが、SnapVaultデスティネーションはより長い期間にわたって作成されたポイントインタイムのSnapshotコピーを保持します。

たとえば、ビジネスに関する政府会計規則に準拠するために、20年の期間にわたってデータのSnapshotコピーを毎月1回保持しなければならない場合があります。バックアップ ストレージからデータを提供するための要件は存在しないため、デスティネーション システムでは低速かつ低コストのディスクを使用できます。

SnapMirrorとSnapVaultは同じコマンド インフラを共有します。SnapMirrorポリシーの作成時に、使用する方法を指定します。どちらの方法にもピア クラスタとピアSVMが必要です。

次の図は、SnapVaultデータ保護関係を示しています。



通常、SnapVaultのデータ保護関係ではソース ボリュームのSnapshotコピーではなく、長期間にわたって作成されたポイントインタイムのSnapshotコピーが保持されます。

SnapVaultデータ保護関係を初期化する方法

SnapMirrorと同様に、SnapVaultを初めて起動すると、ベースライン転送が実行されます。関係のSnapMirrorポリシーでは、ベースラインおよび更新の内容を定義します。

デフォルトのSnapVaultポリシーである`XDPDefault`に基づくベースライン転送では、ソース ボリュームのSnapshotコピーが作成され、そのコピーおよびコピーが参照するデータ ブロックがデスティネーション ボリュームに転送されます。SnapMirrorとは異なり、SnapVaultにはベースラインの古いSnapshotコピーが含まれません。

SnapVaultデータ保護関係を更新する方法

更新は、設定されたスケジュールに従って非同期に行われます。関係のポリシーで定義するルールによって、更新に含める新しいSnapshotコピーおよび保持するコピーの数が特定されます。ポリシーで定義されたラベル（「monthly」など）は、ソース上のSnapshotポリシーで定義された1つ以上のラベルと一致する必要があります。一致しない場合は、レプリケーションが失敗します。

`XDPDefault`ポリシーに基づく各更新では、SnapMirrorが前回の更新後に作成されたSnapshotコピーを転送します（Snapshotコピーのラベルがポリシーのルールで定義されたラベルに一致する場合）。以下に示す`XDPDefault`ポリシーの`snapmirror policy show`コマンドの出力では、次の点に注意してください。

- `Create Snapshot`が「false」です。これは、SnapMirrorが関係を更新するときに、`XDPDefault`によってSnapshotコピーが作成されないことを示します。
- `XDPDefault`には、「daily」および「weekly」というルールがあります。これは、SnapMirrorが関係を更新するときに、ソース上のラベルが一致するすべてのSnapshotコピーが転送されることを示します。

```
cluster_dst::> snapmirror policy show -policy XDPDefault -instance

                       Vserver: vs0
      SnapMirror Policy Name: XDPDefault
      SnapMirror Policy Type: vault
                 Policy Owner: cluster-admin
                  Tries Limit: 8
            Transfer Priority: normal
   Ignore accesstime Enabled: false
     Transfer Restartability: always
 Network Compression Enabled: false
             Create Snapshot: false
                     Comment: Default policy for XDP relationships with daily and weekly
                              rules.
       Total Number of Rules: 2
                  Total Keep: 59
```

```
                        Rules: SnapMirror Label     Keep  Preserve Warn Schedule Prefix
                               ----------------     ----  -------- ---- -------- ------
                               daily                   7  false       0 -        -
                               weekly                 52  false       0 -        -
```

SnapMirrorユニファイド レプリケーションの基本

SnapMirrorユニファイド レプリケーションを使用すると、同じデスティネーション ボリュームでディザスタ リカバリとアーカイブを設定できます。ユニファイド レプリケーションでは、必要なセカンダリ ストレージ容量の削減、ベースライン転送の回数の制限、およびネットワーク トラフィックの減少というメリットが得られます。

一元化されたデータ保護関係を初期化する方法

SnapMirrorと同様に、一元化されたデータ保護機能の初回起動時に、ベースライン転送が実行されます。関係のSnapMirrorポリシーでは、ベースラインおよび更新の内容を定義します。

デフォルトの一元化されたデータ保護ポリシーである`MirrorAndVault`に基づくベースライン転送では、ソース ボリュームのSnapshotコピーが作成され、そのコピーおよびコピーが参照するデータ ブロックがデスティネーション ボリュームに転送されます。SnapVaultと同様に、一元化されたデータ保護にはベースラインの古いSnapshotコピーが含まれません。

一元化されたデータ保護関係を更新する方法

`MirrorAndVault`ポリシーに基づく各更新では、SnapMirrorがソース ボリュームのSnapshotコピーを作成し、そのSnapshotコピーと前回の更新後に作成されたすべてのSnapshotコピーを転送します（SnapshotコピーのラベルがSnapshotポリシーのルールで定義されたラベルに一致する場合）。以下に示す`MirrorAndVault`ポリシーの`snapmirror policy show`コマンドの出力では、次の点に注意してください。

- `Create Snapshot`が「true」です。これは、SnapMirrorが関係を更新するときに、`MirrorAndVault`によってSnapshotコピーが作成されることを示します。
- `MirrorAndVault`には、「sm_created」、「daily」、および「weekly」というルールがあります。これは、SnapMirrorが関係を更新するときに、SnapMirrorで作成されたSnapshotコピーとソース上のラベルが一致するSnapshotコピーが転送されることを示します。

```
cluster_dst::> snapmirror policy show -policy MirrorAndVault -instance

                      Vserver: vs0
      SnapMirror Policy Name: MirrorAndVault
      SnapMirror Policy Type: mirror-vault
                Policy Owner: cluster-admin
                 Tries Limit: 8
            Transfer Priority: normal
   Ignore accesstime Enabled: false
     Transfer Restartability: always
 Network Compression Enabled: false
             Create Snapshot: true
                     Comment: A unified Synchronous SnapMirror and SnapVault policy for
                              mirroring the latest file system and daily and weekly snapshots.
       Total Number of Rules: 3
                  Total Keep: 59
                        Rules: SnapMirror Label     Keep  Preserve Warn Schedule Prefix
                               ----------------     ----  -------- ---- -------- ------
                               sm_created              1  false       0 -        -
                               daily                   7  false       0 -        -
                               weekly                 52  false       0 -        -
```

Unified7yearポリシー

事前設定された`Unified7year`ポリシーは、`MirrorAndVault`とまったく同様に機能します。ただし、4つ目のルールは、月単位Snapshotコピーを転送して7年間保持します。

```
                        Rules: SnapMirror Label     Keep  Preserve Warn Schedule Prefix
                               ----------------     ----  -------- ---- -------- ------
                               sm_created              1  false       0 -        -
                               daily                   7  false       0 -        -
                               weekly                 52  false       0 -        -
                               monthly                84  false       0 -        -
```

データ破損の可能性を防ぐ方法

ユニファイド レプリケーションは、ベースライン転送の内容を、初期化時にSnapMirrorで作成されたSnapshotコピーに制限します。各更新で、SnapMirrorはソースのSnapshotコピーをもう1つ作成して、そのSnapshotコピーおよびSnapshotポリシーのルールで定義されたラベルと一致するラベルを持つ新しいすべてのSnapshotコピーを転送します。

最後に転送されたSnapshotコピーのコピーをデスティネーションで作成することによって、更新したSnapshotコピーが破損する可能性を防ぐことができます。この「ローカル コピー」はソース上の保持ルールに関係なく保持されるため、元はSnapMirrorによって転送されたSnapshotがソースで使用できなくなった場合でも、そのコピーをデスティネーションで使用できます。

ユニファイド データ レプリケーション使用の妥当性

完全なミラーを維持するメリットと、ユニファイド レプリケーションが提供するメリット（セカンダリ ストレージ容量の削減、ベースライン転送の回数の制限、およびネットワーク トラフィックの減少）を比較検討する必要があります。

ユニファイド レプリケーションの妥当性を判断する際の重要な要素は、アクティブ ファイルシステムの変更率です。たとえば、データベース トランザクション ログの時間単位Snapshotコピーを保持するボリュームには、従来のミラーの方が適している場合があります。

SnapMirrorのデフォルトの変更（DPからXDP）

SnapMirror拡張データ保護（XDP）モードがSnapMirrorデータ保護（DP）モードに代わってSnapMirrorのデフォルトになりました。

- DPモードで起動するSnapMirrorでは、バージョンに依存するレプリケーション エンジンを使用していました。そのため、プライマリ ストレージとセカンダリ ストレージのONTAPバージョンが同じである必要がありました。

```
cluster_dst::>  snapmirror create -type DP -source-path ... -
destination-path ...
```

- XDPモードで起動するSnapMirrorでは、バージョンに依存しないレプリケーション エンジンを使用していました。そのため、プライマリ ストレージとセカンダリ ストレージのONTAPバージョンが異なっていてもかまいませんでした。

```
cluster_dst::>  snapmirror create -type XDP -source-path ... -
destination-path ...
```

パフォーマンスの改善に伴い、レプリケーション スループットではバージョンに依存するモードの方がわずかに優れてはいるものの、バージョンに依存しないSnapMirrorの方がはるかに大きなメリットが得られるようになりました。そのため、XDPモードが新しいデフォルト値となり、コマンドラインまたは新規 / 既存のスクリプトにおけるDPモードの起動は自動的にXDPモードに変換されます。

既存の関係には影響しません。DPタイプの既存の関係は引き続きDPタイプになります。データ保護モードを指定しなかった場合、およびXDPモードを関係のタイプとして指定した場合、デフォルト ポリシーがMirrorAndVaultに変更になりました。次の表は、想定される動作を示しています。

表 2 :

指定するモード	タイプ	デフォルト ポリシー（ポリシーを指定しない場合）
DP	XDP	MirrorAllSnapshots（SnapMirror DR）
なし	XDP	MirrorAndVault（ユニファイド レプリケーション）
XDP	XDP	MirrorAndVault（ユニファイド レプリケーション）

次の表に示すとおり、それぞれの状況でXDPに割り当てられるデフォルト ポリシーでは、変換後も古いタイプと同等の機能が保証されます。もちろん、必要に応じて別のポリシー（ユニファイド レプリケーションのポリシーなど）を使用できます。

表 3 :

指定するモード	ポリシー	結果
DP	MirrorAllSnapshots	SnapMirror DR
DP	XDPDefault	SnapVault
DP	MirrorAndVault	ユニファイド レプリケーション
XDP	MirrorAllSnapshots	SnapMirror DR
XDP	XDPDefault	SnapVault
XDP	MirrorAndVault	ユニファイド レプリケーション

以下は例外です。

- SVMデータ保護関係のデフォルトはXDPモードです。
- ルートボリュームの負荷分散データ保護関係は、引き続きデフォルトでDPモードになります。
- SnapLockデータ保護関係のデフォルトはXDPモードです。
- 次のクラスタ全体のオプションを設定した場合、DPを明示的に指定した場合のデフォルトは引き続きDPモードです。

  ```
  options replication.create_data_protection_rels.enable on
  ```

  DPを明示的に指定しない場合、このオプションは無視されます。

デスティネーション ボリュームが自動的に拡張される状況

ソース ボリュームのサイズが増大していた場合、デスティネーション ボリュームを含むアグリゲートに空きスペースがあれば、データ保護ミラー転送の実行時にデスティネーション ボリュームのサイズが自動的に拡張されます。

この処理は、デスティネーションの自動拡張の設定には関係なく行われます。ボリュームの拡張量を制限したり拡張処理を無効にしたりすることはできません。

デフォルトでは、データ保護ボリュームのオートサイズ モードはgrow_shrinkに設定され、使用済みスペースの量に応じてボリュームが拡張または縮小されます。データ保護ボリュームのmax-autosizeはFlexVolの最大サイズと同じで、プラットフォームによって異なります。

- HX2x00のデフォルトDPボリュームmax-autosizeは60 TB
- HX6100のデフォルトDPボリュームmax-autosizeは100 TB

ファンアウト構成およびカスケード構成のデータ保護

ファンアウト構成を使用すると、データ保護を複数のセカンダリ システムに拡張できます。カスケード構成を使用すると、データ保護を3次システムに拡張できます。

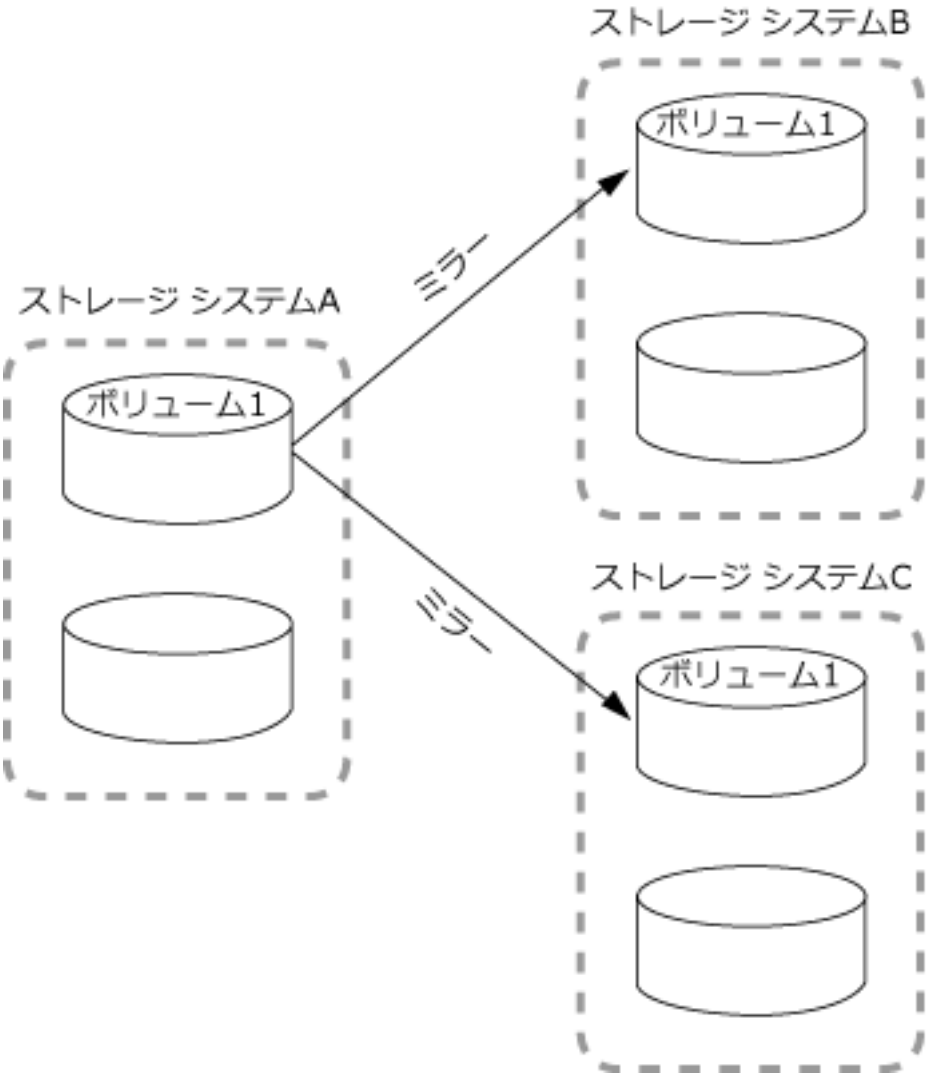
ファンアウトとカスケードのどちらの構成でも、SnapMirror DR、SnapVault、ユニファイド レプリケーションを任意に組み合わせることができます。ただし、SnapMirror Synchronous関係では非同期SnapMirror関係を使用したファンアウト構成のみがサポートされ、カスケード構成はサポートされません。ファンアウト構成で確立できるSnapMirror Synchronous関係は1つだけで、ソース ボリュームからの残りの関係はすべて非同期SnapMirror関係にする必要があります。

注： ファンイン構成を使用すると、複数のプライマリ システムと単一のセカンダリ システムの間のデータ保護関係を作成できます。各関係では、セカンダリ システム上の異なるボリュームを使用する必要があります。

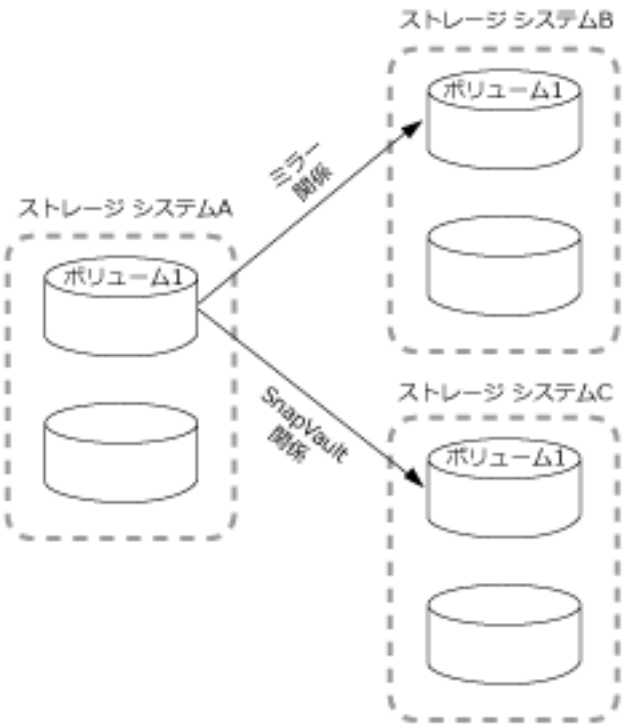
ファンアウト構成の仕組み

SnapMirrorは、複数ミラー ファンアウト構成とミラー-ヴォールト ファンアウト構成をサポートします。

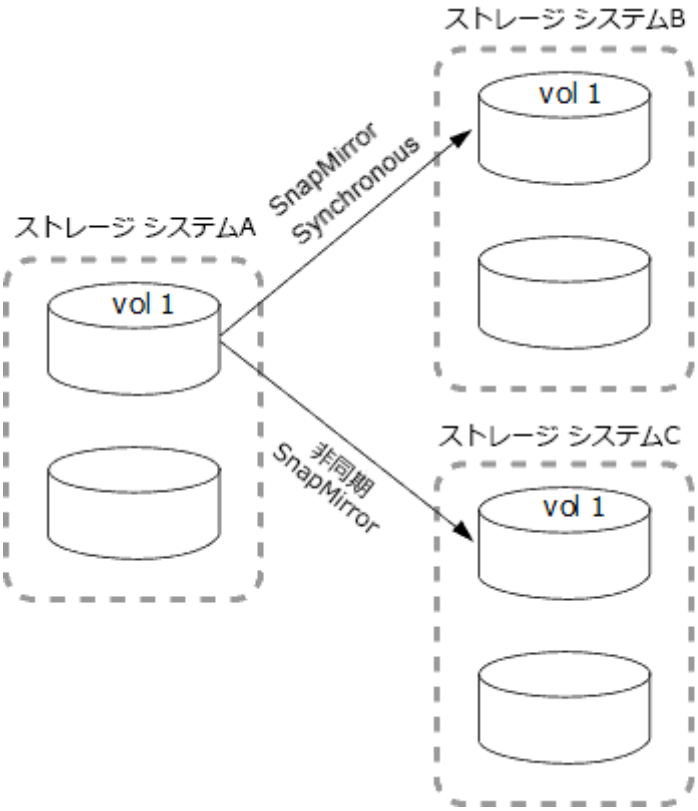
複数ミラー ファンアウト構成では、ソース ボリュームから複数のセカンダリ ボリュームへのミラー関係が確立されます。



ミラー-ヴォールト ファンアウト構成では、ソース ボリュームからセカンダリ ボリュームへのミラー関係と、別のセカンダリ ボリュームへのSnapVault関係が確立されます。



ファンアウト構成でSnapMirror Synchronous関係を確立できます。ただし、ファンアウト構成で確立できるSnapMirror Synchronous関係は1つだけで、ソース ボリュームからの残りの関係はすべて非同期SnapMirror関係にする必要があります。



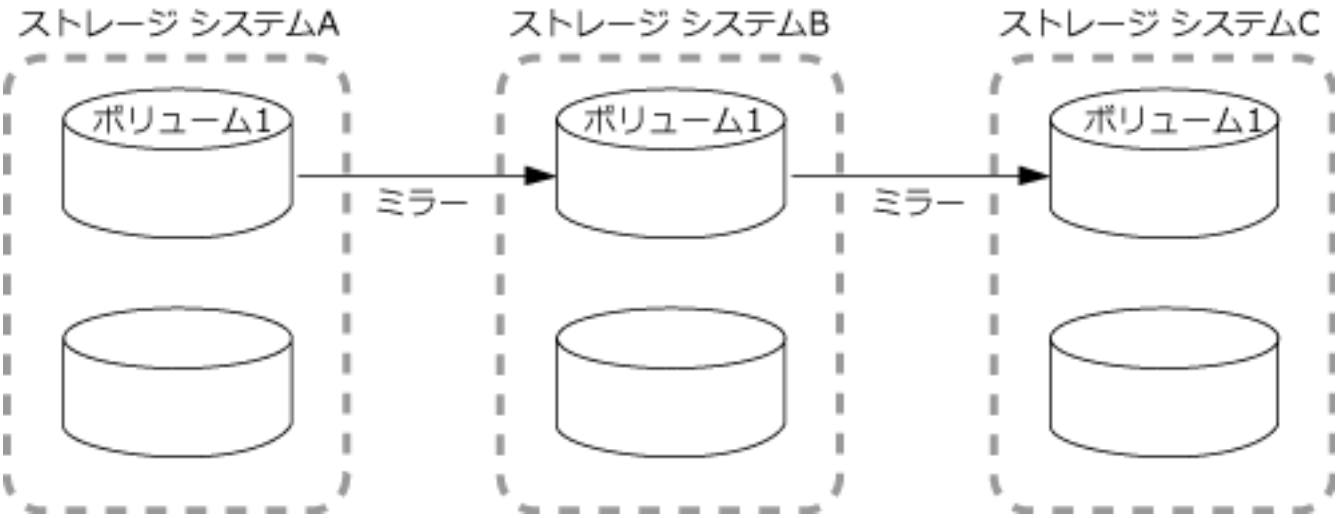
カスケード構成の仕組み

SnapMirrorは、ミラー-ミラー、ミラー-ヴォールト、ヴォールト-ミラー、ヴォールト-ヴォールトの各カスケード構成をサポートします。

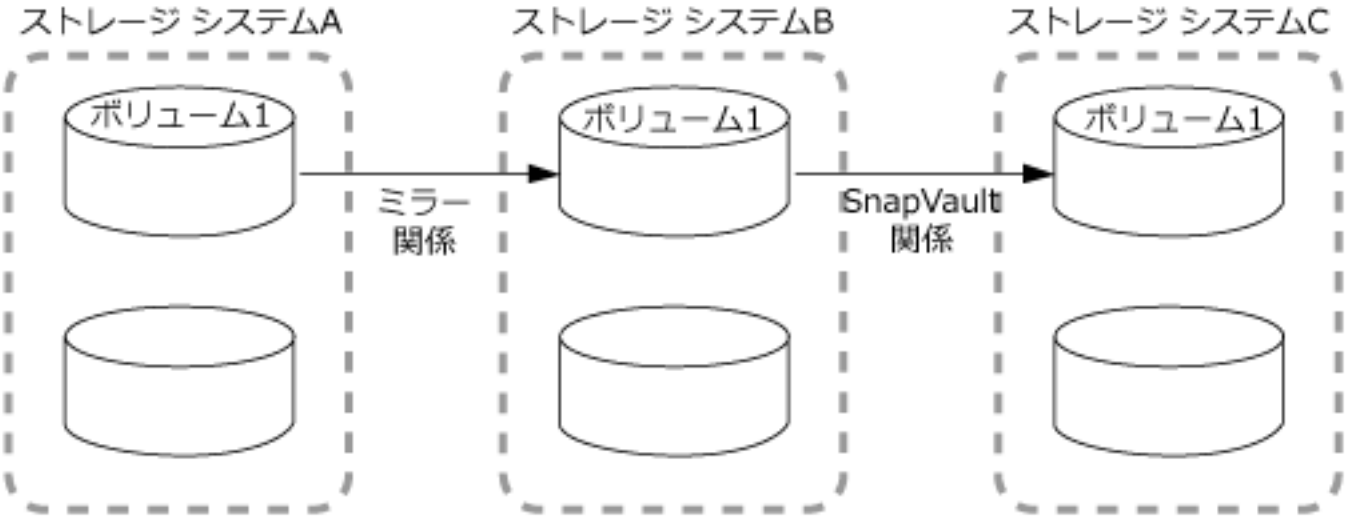
ミラー-ミラー カスケード構成の関係のチェーンでは、ソース ボリュームがセカンダリ ボリュームにミラーリングされ、そのセカンダリ ボリュームが3番目のボリュームにミラーリングされます。セカ

ンダリ ボリュームが使用できなくなった場合は、プライマリ ボリュームと3番目のボリュームの間の関係を同期できます。ベースライン転送を新たに実行する必要はありません。

ミラー-ミラー カスケード構成でSnapMirror Synchronous関係がサポートされます。SnapMirror Synchronous関係に含めることができるのは、プライマリ ボリュームとセカンダリ ボリュームだけです。セカンダリ ボリュームと3番目のボリュームの関係は非同期でなければなりません。



ミラー-ヴォールト カスケード構成の関係のチェーンでは、ソース ボリュームがセカンダリ ボリュームにミラーリングされ、そのセカンダリ ボリュームが3番目のボリュームに保存されます。



ヴォールト-ミラー カスケード構成とヴォールト-ヴォールト カスケード構成もサポートされます。

- ヴォールト-ミラー カスケード構成の関係のチェーンでは、ソース ボリュームがセカンダリ ボリュームに保存され、そのセカンダリ ボリュームが3番目のボリュームにミラーリングされます。
- ヴォールト-ヴォールト カスケード構成の関係のチェーンでは、ソース ボリュームがセカンダリ ボリュームに保存され、そのセカンダリ ボリュームが3番目のボリュームに保存されます。

SnapMirrorのライセンス

ONTAPインスタンス間のレプリケーションに関するライセンスが簡素化されました。SnapMirrorライセンスでバックアップ関係とミラー関係の両方がサポートされます。SnapMirrorライセンスを購入すれば、バックアップとディザスタ リカバリ両方のユースケースでONTAPレプリケーションがサポートされるようになりました。

ONTAPインスタンス間にバックアップ関係を設定するには、SnapVaultライセンスが必要でした。この場合、保持期間が長いバックアップのユースケースに対応するために、DPインスタンスに多数のSnapshotコピーが保持される可能性があります。ONTAPインスタンス間にミラー関係を設定するには、SnapMirrorライセンスが必要でした。この場合、クラスタ フェイルオーバーが発生する可能性があるディザスタ リカバリのユースケースに対応するために、各ONTAPインスタンスに同じ数のSnapshotコピー（ミラー イメージ）が保持されます。SnapMirrorライセンスとSnapVaultライセンスは、どちらも9.xリリースで引き続き使用でき、サポートされます。

SnapVaultライセンスは9.xリリースでも引き続き機能し、サポートされますが、販売は終了しました。SnapMirrorライセンスは引き続き購入可能で、SnapVaultの代わりに使用でき、ミラー設定とバックアップ設定の両方に使用できます。

ONTAP非同期レプリケーションでは単一のユニファイド レプリケーション エンジンを使用して拡張データ保護モード（XDP）ポリシーが設定されます。この場合、ミラー ポリシー、バックアップ ポリシー、またはミラー バックアップ ポリシーにSnapMirrorライセンスを設定できます。SnapMirrorライセンスは、ソースとデスティネーションの両方のクラスタに必要です。SnapVaultライセンスは、SnapMirrorライセンスがすでにインストールされていれば必要ありません。

データ保護設定の制限は、ONTAPバージョン、ハードウェア プラットフォーム、インストールされているライセンスなど、いくつかの要因で決まります。

SnapMirror Synchronousライセンス

SnapMirror Synchronous関係がサポートされます。SnapMirror Synchronous関係を作成するには、次のライセンスが必要です。

- ソース クラスタとデスティネーション クラスタの両方にSnapMirror Synchronousライセンスが必要です。

  2019年6月より前にPremium BundleまたはFlash Bundleでシステムを購入した場合は、製品添付のDVDから富士通マスター キーをダウンロードして必要なSnapMirror Synchronousライセンスを入手できます。
- ソース クラスタとデスティネーション クラスタの両方にSnapMirrorライセンスが必要です。

Data Protection Optimized（DPO）ライセンス

スタンドアロンのDPOライセンスはETERNUS HXとETERNUS AX両方のプラットフォームでサポートされ、新しいクラスタに事前設定する形で購入できるほか、導入済みのクラスタにソフトウェアアップグレードとして追加することも可能です。この新しいDPOライセンスは、ハードウェアとソフトウェアのソリューション バンドルに組み込まれなかったために価格が低く抑えられ、プライマリ クラスタ用の無償SnapMirrorライセンスは提供されませんでした。スタンドアロンのDPOライセンスが設定されたセカンダリ クラスタにはSnapMirrorライセンスも購入する必要があり、DPOセカンダリ クラスタにレプリケートするすべてのプライマリ クラスタにはSnapMirrorライセンスを購入する必要があります。

DPOは、複数のONTAPリリース対して追加のONTAP機能を提供します。

表 4 :

機能	9.7+
ノードあたりの最大ボリューム数	1500/2500
最大同時レプリケーション セッション数	200
ワークロード バイアス*	SnapMirror
HDDでのボリューム間アグリゲート重複排除	○

* SnapMirrorバックオフ（ワークロード バイアス）機能の優先度：

- クライアント：クラスタI/Oの優先度は、SnapMirrorトラフィックではなくクライアント ワークロード（本番アプリケーション）に設定されます。
- 同等：SnapMirrorレプリケーション要求の優先度は、本番アプリケーションのI/Oと同等です。
- SnapMirror：すべてのSnapMirror I/O要求が本番アプリケーションのI/Oよりも優先されます。

表 5 : 各ONTAPリリースのノードあたりの最大FlexVol数

	9.7、DPOなし	9.7、DPOあり
HX2100	1000	1500
HX2200	1000	1500
A200	1000	1500
A200	1000	1500
HX6100	1000	2500
A300	2500	2500
A400	2500	2500
A700	2500	2500
A700s	2500	2500
A800	2500	2500

DPOを新たにインストールする場合の考慮事項

- 一度有効にしたDPOライセンス機能は、無効にしたり元に戻したりすることができません。
- インストールしたDPOライセンスを有効にするには、ONTAPのリブートまたはフェイルオーバーが必要です。
- DPOソリューションはセカンダリ ストレージのワークロードを対象としているため、DPOクラスタではアプリケーション ワークロードのパフォーマンスに影響が生じる可能性があります
- DPOライセンスは、特定のETERNUS AX/HXシリーズプラットフォーム モデルでサポートされます。
- DPOの機能はONTAPのリリースによって異なります。詳細については富士通のサポートにお問い合わせください。

DPOシステムの機能拡張

DP_Optimized（DPO）ライセンスをインストールすると、サポートされるFlexVolの最大数が増加します。DPOライセンスのあるシステムでSnapMirrorバックオフ、ボリューム間のバックグラウンド重複排除、ボリューム間のインライン重複排除、ドナーとしてのSnapshotブロックの使用、およびコンパクションがサポートされます。

セカンダリ システムまたはデータ保護システムでサポートされるFlexVolの最大数が増加し、FlexVolをノードあたり最大2,500個まで、フェイルオーバー モードでは最大5,000個まで拡張できるようになりました。このFlexVolの増加は、DP_Optimized（DPO）ライセンスで有効になります。この場合も、ソースとデスティネーションの両方のノードにSnapMirrorライセンスが必要です。

DPOシステムの次の機能が強化されています。

- SnapMirrorバックオフ：DPOシステムでは、レプリケーション トラフィックにクライアントのワークロードと同じ優先順位が割り当てられます。

  DPOシステムでは、SnapMirrorバックオフはデフォルトでは無効になります。
- ボリュームのバックグラウンド重複排除とボリューム間のバックグラウンド重複排除：DPOシステムでは、ボリュームのバックグラウンド重複排除とボリューム間のバックグラウンド重複排除が有効になります。

既存データを重複排除するには、`storage aggregate efficiency cross-volume-dedupe start -aggregate` *`aggregate_name`* `-scan-old-data true`コマンドを実行します。パフォーマンスへの影響を少なくするために、このコマンドはピーク時を避けて実行することを推奨します。

- Snapshotブロックをドナーとして使用して削減効果を向上：アクティブなファイルシステムでは使用できないがSnapshotコピーに含まれるデータ ブロックをボリューム重複排除のドナーとして使用します。

  Snapshotコピーに含まれるデータと照合して新しいデータを重複排除することができるため、結果としてSnapshotブロックが共有されることになります。ドナー スペースが多いほど削減効果が高まり、特にボリュームに多数のSnapshotコピーがある場合に効果的です。
- コンパクション：DPOシステムでは、データ コンパクションはデフォルトで有効になります。

SnapMirrorボリューム レプリケーションの管理

SnapMirrorは、3つのタイプのデータ保護関係（SnapMirror DR、SnapVault、ユニファイド レプリケーション）を提供します。各タイプの関係を同一の基本的なワークフローに従って設定できます。

ONTAP 9.9.1の正式版以降では、SnapMirrorビジネス継続性（SM-BC）で提供される目標復旧時間ゼロ（ゼロRTO）または透過的アプリケーション フェイルオーバー（TAF）によって、SAN環境でビジネスクリティカルなアプリケーションを自動的にフェイルオーバーすることができます。SM-BCは、2つのETERNUS AXクラスタの構成でサポートされます。

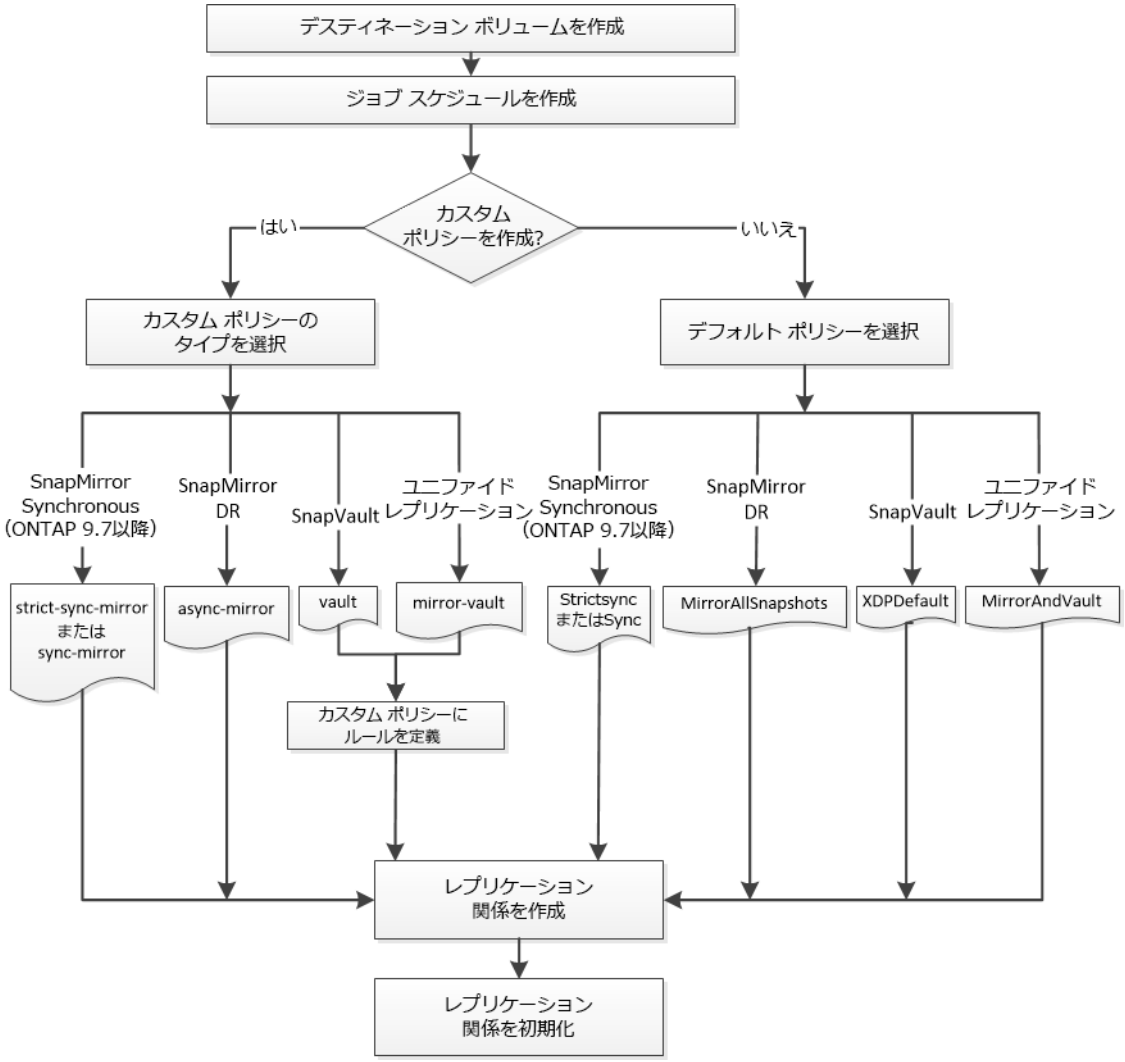
関連概念

SnapMirrorボリューム レプリケーションの概要（12ページ）

SnapMirrorレプリケーションのワークフロー

SnapMirrorデータ保護関係の各タイプのワークフロー（デスティネーション ボリュームの作成、ジョブ スケジュールの作成、ポリシーの指定、関係の作成と初期化）は同じです。

`snapmirror protect`コマンドを使用してデータ保護関係をワンステップで設定できます。`snapmirror protect`を使用する場合でも、ワークフローの各手順について理解しておく必要があります。



ワンステップでのレプリケーション関係の設定

snapmirror protectコマンドを使用してデータ保護関係をワンステップで設定できます。レプリケートするボリュームのリスト、デスティネーション クラスタ上のSVM、ジョブ スケジュール、およびSnapMirrorポリシーを指定します。snapmirror protectによってその他の処理を行います。

始める前に

- ソース / デスティネーション クラスタとSVMのピア関係が確立されている必要があります。

  クラスタ / SVMピアリング エクスプレス ガイド
- デスティネーション ボリューム上の言語は、ソース ボリューム上の言語と同じである必要があります。

このタスクについて

snapmirror protectコマンドは、指定したSVMに関連付けられているアグリゲートを選択します。SVMにアグリゲートが関連付けられていない場合は、クラスタ内のすべてのアグリゲートから選択されます。アグリゲートの選択は、空きスペースの量およびアグリゲート上のボリュームの数に基づいて行われます。

snapmirror protectコマンドは次に以下の処理を実行します。

- レプリケートするボリュームのリスト内の各ボリュームについて、適切なタイプとリザーブされたスペースを持つデスティネーション ボリュームを作成します。
- 指定したポリシーに適したレプリケーション関係を設定します。
- 関係を初期化します。

デスティネーション ボリュームの名前の形式は、*source_volume_name*_dstです。既存の名前と競合する場合は、コマンドによってボリューム名に数字が追加されます。コマンド オプションでプレフィックスまたはサフィックスを指定できます。このサフィックスは、システムが指定したサフィックスであるdstを置き換えます。

ONTAP 9.7以降では、デスティネーションボリュームに最大1019個のスナップショットコピーを含めることができます。

注： 初期化には時間がかかる場合があります。snapmirror protectは初期化が完了するのを待たずにジョブを終了します。そのため、初期化が完了するタイミングを確認するには、job showコマンドではなくsnapmirror showコマンドを使用する必要があります。

snapmirror protectコマンドを使用してSnapMirror Synchronous関係を作成できます。

手順

レプリケーション関係をワンステップで作成して初期化します。snapmirror protect -path-list *SVM*:*volume*|*cluster*://*SVM*/*volume*, ... -destination-vserver *destination_SVM* -policy *policy* -schedule *schedule* -auto-initialize true|false -destination-volume-prefix *prefix* -destination-volume-suffix *suffix*

注： このコマンドはデスティネーションSVMまたはデスティネーション クラスタから実行する必要があります。-auto-initializeオプションのデフォルト値は「true」です。

次の例は、デフォルトのMirrorAllSnapshotsポリシーを使用して、SnapMirror DR関係を作成して初期化します。

```
cluster_dst::> snapmirror protect -path-list svm1:volA, svm1:volB -
destination-vserver svm_backup -policy MirrorAllSnapshots -schedule
 replication_daily
```

注： 必要に応じて、カスタム ポリシーを使用できます。詳細については、カスタム レプリケーション ポリシーの作成を参照してください。

次の例は、デフォルトのXDPDefaultポリシーを使用して、SnapVault関係を作成して初期化します。

```
cluster_dst::> snapmirror protect -path-list svm1:volA, svm1:volB
 -destination-vserver svm_backup -policy XDPDefault -schedule
 replication_daily
```

次の例は、デフォルトのMirrorAndVaultポリシーを使用して、ユニファイド レプリケーション関係を作成して初期化します。

```
cluster_dst::> snapmirror protect -path-list svm1:volA, svm1:volB -
destination-vserver svm_backup -policy MirrorAndVault
```

次の例は、デフォルトのSyncポリシーを使用して、SnapMirror Synchronous関係を作成して初期化します。

```
cluster_dst::> snapmirror protect -path-list svm1:volA, svm1:volB -
destination-vserver svm_sync -policy Sync
```

注： SnapVaultポリシーとユニファイド レプリケーション ポリシーの場合は、デスティネーションで最後に転送されたSnapshotコピーのコピーを作成するスケジュールを定義すると便利です。詳細については、デスティネーションでローカル コピーを作成するスケジュールの定義を参照してください。

次のタスク

`snapmirror show`コマンドを使用して、SnapMirror関係が作成されたことを確認します。コマンド構文全体については、マニュアル ページを参照してください。

レプリケーション関係の設定時の各手順

データ保護関係のタイプに応じて、デスティネーション ボリュームの作成、ジョブ スケジュールの設定、ポリシーの指定、関係の作成と初期化を行う必要があります。デフォルト ポリシーまたはカスタム ポリシーを使用できます。

デスティネーション ボリュームの作成

デスティネーションで`volume create`コマンドを使用すると、デスティネーション ボリュームを作成できます。デスティネーション ボリュームのサイズは、ソース ボリュームと同じかそれ以上である必要があります。

手順

デスティネーション ボリュームを作成します。`volume create -vserver` *`SVM`* `-volume` *`volume`* `-aggregate` *`aggregate`* `-type DP -size` *`size`*

コマンド構文全体については、マニュアル ページを参照してください。

次の例は、`volA_dst`という名前の2GBのデスティネーション ボリュームを作成します。

```
cluster_dst::> volume create -vserver SVM_backup -volume volA_dst -aggregate
 node01_aggr -type DP -size 2GB
```

レプリケーション ジョブ スケジュールの作成

`job schedule cron create`コマンドを使用すると、レプリケーション ジョブ スケジュールを作成できます。ジョブ スケジュールでは、スケジュールの割り当て先のデータ保護関係がSnapMirrorによって自動的に更新されるタイミングを決定します。

このタスクについて

ジョブ スケジュールはデータ保護関係の作成時に割り当てます。ジョブ スケジュールを割り当てない場合は、関係を手動で更新する必要があります。

手順

ジョブ スケジュールを作成します。 `job schedule cron create -name job_name -month month -dayofweek day_of_week -day day_of_month -hour hour -minute minute`

-month、-dayofweek、-hourに`all`を指定すると、ジョブを毎月、特定の曜日、時間にそれぞれ実行できます。

次の例は、土曜日の午前3時に実行する`my_weekly`というジョブ スケジュールを作成します。

```
cluster_dst::> job schedule cron create -name my_weekly -dayofweek
 "Saturday" -hour 3 -minute 0
```

レプリケーション ポリシーのカスタマイズ

レプリケーション関係の作成時には、デフォルトまたはカスタムのポリシーを使用できます。カスタムのSnapVaultまたはユニファイド レプリケーション ポリシーの場合は、初期化と更新の際に転送するSnapshotコピーを決定する1つ以上のルールを定義する必要があります。デスティネーションでローカルSnapshotコピーを作成するスケジュールを定義することもできます。

関連概念

SnapMirror Synchronousディザスタ リカバリの基本（14ページ）

関連タスク

カスタム レプリケーション ポリシーの作成（30ページ）
ポリシーのルールの定義（31ページ）
デスティネーションでローカル コピーを作成するスケジュールの定義（33ページ）

カスタム レプリケーション ポリシーの作成

関係のデフォルト ポリシーが適切でない場合は、カスタム レプリケーション ポリシーを作成できます。たとえば、ネットワーク転送の際にデータを圧縮したり、Snapshotコピーを転送するためのSnapMirrorの試行回数を変更したりできます。

このタスクについて

レプリケーション ポリシーのポリシー タイプによって、そのポリシーがサポートする関係のタイプが決まります。次の表は、使用可能なポリシー タイプを示しています。

表 6 :

ポリシー タイプ	関係タイプ
async-mirror	SnapMirror DR
vault	SnapVault
mirror-vault	ユニファイド レプリケーション
strict-sync-mirror	StrictSyncモードのSnapMirror Synchronous
sync-mirror	SyncモードのSnapMirror Synchronous

ヒント : カスタム レプリケーション ポリシーを作成する場合は、デフォルト ポリシーをモデルとすることを推奨します。

手順

カスタム レプリケーション ポリシーを作成します。`snapmirror policy create -vserver SVM -policy policy -type async-mirror|vault|mirror-vault|strict-sync-mirror|sync-mirror -comment comment -tries transfer_tries -transfer-priority low|normal -is-network-compression-enabled true|false`

コマンド構文全体については、マニュアル ページを参照してください。

-common-snapshot-scheduleパラメータを使用して、SnapMirror Synchronous関係の共通のSnapshotコピー スケジュールを作成するスケジュールを指定できます。デフォルトでは、SnapMirror Synchronous関係の共通のSnapshotコピー スケジュールは1時間です。SnapMirror Synchronous関係のSnapshotコピー スケジュールの値は、30分から2時間までの範囲で指定できます。

次の例は、データ転送のためにネットワーク圧縮を有効にする、SnapMirror DR用のカスタム レプリケーション ポリシーを作成します。

```
cluster_dst::> snapmirror policy create -vserver svm1 -policy DR_compressed
 -type async-mirror -comment 「DR with network compression enabled」 -is-
network-compression-enabled true
```

次の例は、SnapVault用のカスタム レプリケーション ポリシーを作成します。

```
cluster_dst::> snapmirror policy create -vserver svm1 -policy my_snapvault -
type vault
```

次の例は、ユニファイド レプリケーション用のカスタム レプリケーション ポリシーを作成します。

```
cluster_dst::> snapmirror policy create -vserver svm1 -policy my_unified -
type mirror-vault
```

次の例は、StrictSyncモードのSnapMirror Synchronous関係用のカスタム レプリケーション ポリシーを作成します。

```
cluster_dst::> snapmirror policy create -vserver svm1 -policy my_strictsync
 -type strict-sync-mirror -common-snapshot-schedule my_sync_schedule
```

次のタスク

ポリシー タイプが「vault」および「mirror-vault」の場合は、初期化と更新の際に転送するSnapshotコピーを決定するルールを定義する必要があります。

`snapmirror policy show`コマンドを使用して、SnapMirrorポリシーが作成されたことを確認します。コマンド構文全体については、マニュアル ページを参照してください。

ポリシーのルールの定義

ポリシー タイプが「vault」または「mirror-vault」のカスタム ポリシーの場合は、初期化と更新の際に転送するSnapshotコピーを決定するルールを定義する必要があります。また、ポリシー タイプが「vault」または「mirror-vault」のデフォルト ポリシーのルールを定義することもできます。

このタスクについて

ポリシー タイプが「vault」または「mirror-vault」のすべてのポリシーには、レプリケートするSnapshotコピーを指定するルールが必要です。たとえば、「bi-monthly」ルールは、「bi-monthly」というSnapMirrorラベルが割り当てられたSnapshotコピーだけをレプリケートする必要があることを指定します。SnapMirrorラベルは、ソースでのSnapshotポリシーの設定時に指定します。

各ポリシー タイプは、システム定義の1つ以上のルールに関連付けられています。これらのルールは、ポリシー タイプの指定時にポリシーに自動的に割り当てられます。次の表は、システム定義のルールを示しています。

表 7 :

システム定義のルール	ルールを使用するポリシー タイプ	結果
sm_created	async-mirror、mirror-vault、Sync、StrictSync	SnapMirrorで作成されたSnapshotコピーが初期化および更新の際に転送されます。
all_source_snapshots	async-mirror	ソース上の新しいSnapshotコピーが初期化および更新の際に転送されます。

システム定義のルール	ルールを使用するポリシー タイプ	結果
daily	vault、mirror-vault	ソース上の（SnapMirrorラベルが「daily」である）新しいSnapshotコピーが初期化および更新の際に転送されます。
weekly	vault、mirror-vault	ソース上の（SnapMirrorラベルが「weekly」である）新しいSnapshotコピーが初期化および更新の際に転送されます。
monthly	mirror-vault	ソース上の（SnapMirrorラベルが「monthly」である）新しいSnapshotコピーが初期化および更新の際に転送されます。
app_consistent	Sync、StrictSync	SnapMirrorラベルが「app_consistent」のSnapshotコピーがソースからデスティネーションに同期的にレプリケートされます。 ONTAP 9.7以降でサポート。

「async-mirror」ポリシー タイプの場合を除き、デフォルト ポリシーまたはカスタム ポリシーに対して追加のルールを必要に応じて指定できます。次に例を示します。

- デフォルトの`MirrorAndVault`ポリシーの場合は、「bi-monthly」というルールを作成し、ソース上のSnapMirrorラベルが「bi-monthly」のSnapshotコピーに一致させることができます。
- ポリシー タイプが「mirror-vault」のカスタム ポリシーの場合は、「bi-weekly」というルールを作成し、ソース上のSnapMirrorラベルが「bi-weekly」のSnapshotコピーに一致させることができます。

手順

ポリシーのルールを定義します。`snapmirror policy add-rule -vserver` *`SVM`* `-policy` *`policy_for_rule`* `-snapmirror-label` *`snapmirror-label`* `-keep` *`retention_count`*

コマンド構文全体については、マニュアル ページを参照してください。

次の例は、SnapMirrorラベルが`bi-monthly`のルールをデフォルトの`MirrorAndVault`ポリシーに追加します。

```
cluster_dst::> snapmirror policy add-rule -vserver svm1 -policy
 MirrorAndVault -snapmirror-label bi-monthly -keep 6
```

次の例は、SnapMirrorラベルが`bi-weekly`のルールをカスタムの`my_snapvault`ポリシーに追加します。

```
cluster_dst::> snapmirror policy add-rule -vserver svm1 -policy my_snapvault
 -snapmirror-label bi-weekly -keep 26
```

次の例は、SnapMirrorラベルが`app_consistent`のルールをカスタムのSyncポリシーに追加します。

```
cluster_dst::> snapmirror policy add-rule -vserver svm1 -policy Sync -
snapmirror-label app_consistent -keep 1
```

このSnapMirrorラベルに一致するSnapshotコピーをソース クラスタからレプリケートできます。

```
cluster_src::> snapshot create -vserver vs1 -volume vol1 -snapshot snapshot1
 -snapmirror-label app_consistent
```

デスティネーションでローカル コピーを作成するスケジュールの定義

SnapVault関係とユニファイド レプリケーション関係の場合は、最後に転送されたSnapshotコピーのコピーをデスティネーションで作成することによって、更新したSnapshotコピーが破損する可能性を防ぐことができます。この「ローカル コピー」はソース上の保持ルールに関係なく保持されるため、元はSnapMirrorによって転送されたSnapshotがソースで使用できなくなった場合でも、そのコピーをデスティネーションで使用できます。

このタスクについて

`snapmirror policy add-rule`コマンドの-scheduleオプションで、ローカル コピーを作成するスケジュールを指定します。

手順

デスティネーションでローカル コピーを作成するスケジュールを定義します。`snapmirror policy add-rule -vserver` *`SVM`* `-policy` *`policy_for_rule`* `-snapmirror-label` *`snapmirror-label`* `-schedule` *`schedule`*

コマンド構文全体については、マニュアル ページを参照してください。ジョブ スケジュールの作成方法の例については、レプリケーション ジョブ スケジュールの作成を参照してください。

次の例は、ローカル コピーを作成するスケジュールをデフォルトの`MirrorAndVault`ポリシーに追加します。

```
cluster_dst::> snapmirror policy add-rule -vserver svm1 -policy
 MirrorAndVault -snapmirror-label my_monthly -schedule my_monthly
```

次の例は、ローカル コピーを作成するスケジュールをカスタムの`my_unified`ポリシーに追加します。

```
cluster_dst::> snapmirror policy add-rule -vserver svm1 -policy my_unified -
snapmirror-label my_monthly -schedule my_monthly
```

レプリケーション関係の作成

プライマリ ストレージのソース ボリュームとセカンダリ ストレージのデスティネーション ボリュームの関係はデータ保護関係と呼ばれます。`snapmirror create`コマンドを使用すると、SnapMirror DR、SnapVault、またはユニファイド レプリケーションのデータ保護関係を作成できます。

始める前に

- ソース / デスティネーション クラスタとSVMのピア関係が確立されている必要があります。

  クラスタ / SVMピアリング エクスプレス ガイド
- デスティネーション ボリューム上の言語は、ソース ボリューム上の言語と同じである必要があります。

このタスクについて

- DPモードで呼び出されたSnapMirrorとXDPモードで呼び出されたSnapMirrorでは、バージョンに依存するさまざまなアプローチでさまざまなレプリケーション・エンジンが使用されていました。
- DPモードで起動するSnapMirrorでは、バージョンに依存するレプリケーション エンジンを使用していました。そのため、プライマリ ストレージとセカンダリ ストレージのONTAPバージョンが同じである必要がありました。

  ```
  cluster_dst::>  snapmirror create -type DP -source-path ... -
  destination-path ...
  ```

- XDPモードで起動するSnapMirrorでは、バージョンに依存しないレプリケーション エンジンを使用していました。そのため、プライマリ ストレージとセカンダリ ストレージのONTAPバージョンが異なっていてもかまいませんでした。

  ```
  cluster_dst::>  snapmirror create -type XDP -source-path ... -
  destination-path ...
  ```

パフォーマンスの改善に伴い、レプリケーション スループットではバージョンに依存するモードの方がわずかに優れてはいるものの、バージョンに依存しないSnapMirrorの方がはるかに大きなメリットが得られるようになりました。そのため、XDPモードが新しいデフォルト値となり、コマンドラインまたは新規 / 既存のスクリプトにおけるDPモードの起動は自動的にXDPモードに変換されます。

既存の関係には影響しません。DPタイプの既存の関係は引き続きDPタイプになります。次の表は、想定される動作を示しています。

表 8 :

指定するモード	タイプ	デフォルト ポリシー（ポリシーを指定しない場合）
DP	XDP	MirrorAllSnapshots（SnapMirror DR）
なし	XDP	MirrorAllSnapshots（SnapMirror DR）
XDP	XDP	XDPDefault（SnapVault）

以下の手順の例も参照してください。

以下は例外です。

- SVMデータ保護関係のデフォルトは引き続きDPモードです。

  デフォルトの`MirrorAllSnapshots`ポリシーを使用してXDPモードを使用するには、XDPを明示的に指定してください。
- 負荷共有データ保護関係のデフォルトは引き続きDPモードです。
- SnapLockデータ保護関係のデフォルトは引き続きDPモードです。
- 次のクラスタ全体のオプションを設定した場合、DPを明示的に指定した場合のデフォルトは引き続きDPモードです。

  ```
  options replication.create_data_protection_rels.enable on
  ```

  DPを明示的に指定しない場合、このオプションは無視されます。

ONTAP 9.7以降では、デスティネーションボリュームに最大1019個のスナップショットコピーを含めることができます。

SnapMirror Synchronous関係がサポートされます。

手順

デスティネーション クラスタから、レプリケーション関係を作成します。`snapmirror create -source-path` *`SVM:volume|cluster://SVM/volume`*`, ... -destination-path` *`SVM:volume|cluster://SVM/volume`*`, ... -type DP|XDP -schedule` *`schedule`* `-policy` *`policy`*

コマンド構文全体については、マニュアル ページを参照してください。

注 : scheduleパラメータは、SnapMirror Synchronous関係の作成時には指定しません。

次の例は、デフォルトの`MirrorLatest`ポリシーを使用して、SnapMirror DR関係を作成します。

```
cluster_dst::> snapmirror create -source-path svm1:volA -destination-path
 svm_backup:volA_dst -type XDP -schedule my_daily -policy MirrorLatest
```

次の例は、デフォルトの`XDPDefault`ポリシーを使用して、SnapVault関係を作成します。

```
cluster_dst::> snapmirror create -source-path svm1:volA -destination-path
 svm_backup:volA_dst -type XDP -schedule my_daily -policy XDPDefault
```

次の例は、デフォルトのMirrorAndVaultポリシーを使用して、ユニファイド レプリケーション関係を作成します。

```
cluster_dst:> snapmirror create -source-path svm1:volA -destination-path
 svm_backup:volA_dst -type XDP -schedule my_daily -policy MirrorAndVault
```

次の例は、カスタムのmy_unifiedポリシーを使用して、ユニファイド レプリケーション関係を作成します。

```
cluster_dst::> snapmirror create -source-path svm1:volA -destination-path
 svm_backup:volA_dst -type XDP -schedule my_daily -policy my_unified
```

次の例は、デフォルトのSyncポリシーを使用して、SnapMirror Synchronous関係を作成します。

```
cluster_dst::> snapmirror create -source-path svm1:volA -destination-path
 svm_backup:volA_dst -type XDP -policy Sync
```

次の例は、デフォルトのStrictSyncポリシーを使用して、SnapMirror Synchronous関係を作成します。

```
cluster_dst::> snapmirror create -source-path svm1:volA -destination-path
 svm_backup:volA_dst -type XDP -policy StrictSync
```

次の例は、SnapMirror DR関係を作成します。DPタイプは自動的にXDPに変換され、ポリシーは指定されません。デフォルト ポリシーはMirrorAllSnapshotsです。

```
cluster_dst::> snapmirror create -source-path svm1:volA -destination-path
 svm_backup:volA_dst -type DP -schedule my_daily
```

次の例は、SnapMirror DR関係を作成します。タイプまたはポリシーは指定されません。デフォルト ポリシーはMirrorAllSnapshotsです。

```
cluster_dst::> snapmirror create -source-path svm1:volA -destination-path
 svm_backup:volA_dst -schedule my_daily
```

次の例は、SnapMirror DR関係を作成します。ポリシーは指定されません。デフォルト ポリシーはXDPDefaultです。

```
cluster_dst::> snapmirror create -source-path svm1:volA -destination-path
 svm_backup:volA_dst -type XDP -schedule my_daily
```

次の例は、事前定義されたSnapCenterSyncポリシーを使用してSnapMirror Synchronous関係を作成します。

```
cluster_dst::> snapmirror create -source-path svm1:volA -destination-path
 svm_backup:volA_dst -type XDP -policy SnapCenterSync
```

注： 事前定義されたSnapCenterSyncポリシーのタイプはSyncです。このポリシーでは、snapmirror-labelが「app_consistent」のSnapshotコピーがすべてレプリケートされます。

次のタスク

snapmirror showコマンドを使用して、SnapMirror関係が作成されたことを確認します。コマンド構文全体については、マニュアル ページを参照してください。

関連タスク

ポリシーのルールの定義（31ページ）

レプリケーション関係の初期化

すべての関係タイプでは、初期化の際にベースライン転送が実行されます。これにより、ソース ボリュームのSnapshotコピーが作成され、そのコピーおよびコピーが参照するすべてのデータ ブロックがデスティネーション ボリュームに転送されます。それ以外の転送の内容はポリシーによって異なります。

始める前に

ソース / デスティネーション クラスタとSVMのピア関係が確立されている必要があります。

クラスタ / SVMピアリング エクスプレス ガイド

このタスクについて

初期化には時間がかかる場合があります。ベースライン転送はオフピークの時間帯に実行することを推奨します。

SnapMirror Synchronous関係がサポートされます。

手順

レプリケーション関係を初期化します。`snapmirror initialize -source-path SVM:volume|cluster://SVM/volume, ... -destination-path SVM:volume|cluster://SVM/volume, ...`

コマンド構文全体については、マニュアル ページを参照してください。

注： このコマンドはデスティネーションSVMまたはデスティネーション クラスタから実行する必要があります。

次の例は、ソース ボリューム`volA`（`svm1`上）とデスティネーション ボリューム`volA_dst`（`svm_backup`上）の間の関係を初期化します。

```
cluster_dst::> snapmirror initialize -source-path svm1:volA -destination-
path svm_backup:volA_dst
```

例：ヴォールト-ヴォールト カスケードの設定

レプリケーション関係を順を追って設定する方法の具体例を示します。この例で設定するヴォールト-ヴォールト カスケード構成を使用すると、「my-weekly」というラベルの付いた251個を超えるSnapshotコピーを保持できます。

始める前に

- ソース / デスティネーション クラスタとSVMのピア関係が確立されている必要があります。
- ONTAP 9.7以降が実行されている必要があります。

このタスクについて

この例では以下の事項を前提としています。

- SnapMirrorラベルが「my-daily」、「my-weekly」、「my-monthly」のSnapshotコピーをソース クラスタで設定済みである。
- セカンダリ デスティネーション クラスタと3番目のデスティネーション クラスタで「volA」というデスティネーション ボリュームを設定済みである。
- セカンダリ デスティネーション クラスタと3番目のデスティネーション クラスタで「my_snapvault」というレプリケーション ジョブ スケジュールを設定済みである。

この例では、次に示す2つのカスタム ポリシーに基づいてレプリケーション関係を作成する方法を示します。

- 「snapvault_secondary」ポリシーは、7個の日単位Snapshotコピー、52個の週単位Snapshotコピー、180個の月単位Snapshotコピーをセカンダリ デスティネーション クラスタに保持します。
- 「snapvault_tertiary」ポリシーは、250個の週単位Snapshotコピーを3番目のデスティネーション クラスタに保持します。

手順

1. セカンダリ デスティネーション クラスタで、「snapvault_secondary」ポリシーを作成します。`cluster_secondary::> snapmirror policy create -policy snapvault_secondary -type vault -comment “Policy on secondary for vault to vault cascade” -vserver svm_secondary`
2. セカンダリ デスティネーション クラスタで、ポリシーの「my-daily」ルールを定義します。`cluster_secondary::> snapmirror policy add-rule -policy snapvault_secondary -snapmirror-label my-daily -keep 7 -vserver svm_secondary`
3. セカンダリ デスティネーション クラスタで、ポリシーの「my-weekly」ルールを定義します。`cluster_secondary::> snapmirror policy add-rule -policy snapvault_secondary -snapmirror-label my-weekly -keep 52 -vserver svm_secondary`
4. セカンダリ デスティネーション クラスタで、ポリシーの「my-monthly」ルールを定義します。`cluster_secondary::> snapmirror policy add-rule -policy snapvault_secondary -snapmirror-label my-monthly -keep 180 -vserver svm_secondary`
5. セカンダリ デスティネーション クラスタで、ポリシーを検証します。`cluster_secondary::> snapmirror policy show snapvault_secondary -instance`

```
                    Vserver: svm_secondary
     SnapMirror Policy Name: snapvault_secondary
     SnapMirror Policy Type: vault
               Policy Owner: cluster-admin
                Tries Limit: 8
          Transfer Priority: normal
  Ignore accesstime Enabled: false
    Transfer Restartability: always
Network Compression Enabled: false
             Create Snapshot: false
                    Comment: Policy on secondary for vault to vault cascade
      Total Number of Rules: 3
                  Total Keep: 239
                       Rules: SnapMirror Label     Keep  Preserve Warn Schedule Prefix
                              ----------------     ----  -------- ---- -------- ------
                              my-daily                7  false       0 -        -
                              my-weekly              52  false       0 -        -
                              my-monthly            180  false       0 -        -
```

6. セカンダリ デスティネーション クラスタで、ソース クラスタとの関係を作成します。`cluster_secondary::> snapmirror create -source-path svm_primary:volA -destination-path svm_secondary:volA -type XDP -schedule my_snapvault -policy snapvault_secondary`
7. セカンダリ デスティネーション クラスタで、ソース クラスタとの関係を初期化します。`cluster_secondary::> snapmirror initialize -source-path svm_primary:volA -destination-path svm_secondary:volA`
8. 3番目のデスティネーション クラスタで、「snapvault_tertiary」ポリシーを作成します。`cluster_tertiary::> snapmirror policy create -policy snapvault_tertiary -type vault -comment “Policy on tertiary for vault to vault cascade” -vserver svm_tertiary`
9. 3番目のデスティネーション クラスタで、ポリシーの「my-weekly」ルールを定義します。`cluster_tertiary::> snapmirror policy add-rule -policy snapvault_tertiary -snapmirror-label my-weekly -keep 250 -vserver svm_tertiary`
10. 3番目のデスティネーション クラスタで、ポリシーを検証します。`cluster_tertiary::> snapmirror policy show snapvault_tertiary -instance`

```
                    Vserver: svm_tertiary
     SnapMirror Policy Name: snapvault_tertiary
     SnapMirror Policy Type: vault
```

```
                Policy Owner: cluster-admin
                 Tries Limit: 8
           Transfer Priority: normal
   Ignore accesstime Enabled: false
     Transfer Restartability: always
 Network Compression Enabled: false
             Create Snapshot: false
                     Comment: Policy on tertiary for vault to vault cascade
      Total Number of Rules: 1
                  Total Keep: 250
                       Rules: SnapMirror Label     Keep  Preserve Warn Schedule Prefix
                              ----------------     ----  -------- ---- -------- ------
                              my-weekly             250  false       0 -        -
```

11. 3番目のデスティネーション クラスタで、セカンダリ クラスタとの関係を作成します。`cluster_tertiary::> snapmirror create -source-path svm_secondary:volA -destination-path svm_tertiary:volA -type XDP -schedule my_snapvault -policy snapvault_tertiary`
12. 3番目のデスティネーション クラスタで、セカンダリ クラスタとの関係を初期化します。`cluster_tertiary::> snapmirror initialize -source-path svm_secondary:volA -destination-path svm_tertiary:volA`

既存のDPタイプの関係からXDPへの変換

既存のDPタイプの関係を簡単にXDPに変換して、バージョンに依存しないSnapMirrorを活用できます。

このタスクについて

SnapMirrorでは、既存のDPタイプの関係をXDPに自動的に変換しません。関係を変換するには、既存の関係を解除して削除し、新しいXDP関係を作成して関係を再同期する必要があります。背景情報については、SnapMirrorのデフォルトの変更（DPからXDP）（19ページ）を参照してください。

注： SnapMirror 関係タイプを DP から XDP に変換すると、オートサイズやスペースギャランティなどのスペース関連の設定はデスティネーションにレプリケートされなくなります。

手順

1. 既存のDPタイプの関係を休止します。 `snapmirror quiesce -source-path SVM:volume|cluster://SVM/volume, ... -destination-path SVM:volume|cluster://SVM/volume, ...`

   コマンド構文全体については、マニュアル ページを参照してください。

   注： このコマンドはデスティネーションSVMまたはデスティネーション クラスタから実行する必要があります。

   次の例は、`svm1`上のソース ボリューム`volA`と`svm_backup`上のデスティネーション ボリューム`volA_dst`の間の関係を休止します。

   ```
   cluster_dst::> snapmirror quiesce -source-path svm1:volA -destination-path svm_backup:volA_dst
   ```

2. 既存のDPタイプの関係を解除します。 `snapmirror break -source-path SVM:volume|cluster://SVM/volume, ... -destination-path SVM:volume|cluster://SVM/volume, ...`

   コマンド構文全体については、マニュアル ページを参照してください。

   注： このコマンドはデスティネーションSVMまたはデスティネーション クラスタから実行する必要があります。

   次の例は、`svm1`上のソース ボリューム`volA`と`svm_backup`上のデスティネーション ボリューム`volA_dst`の間の関係を解除します。

   ```
   cluster_dst::> snapmirror break -source-path svm1:volA -destination-path
    svm_backup:volA_dst
   ```

3. デスティネーションボリューム上の Snapshot コピーの自動削除を無効にします。 `volume snapshot autodelete modify -vserver SVM -volume volume -enabled false`

次の例は、デスティネーション ボリュームvolA_dstでSnapshotコピーの自動削除を無効にします。

```
cluster_dst::> volume snapshot autodelete modify -vserver svm_backup -
volume volA_dst -enabled false
```

4. 既存のDPタイプの関係を削除します。`snapmirror delete -source-path SVM:volume|cluster://SVM/volume, ... -destination-path SVM:volume|cluster://SVM/volume, ...`

   コマンド構文全体については、マニュアル ページを参照してください。

   注： このコマンドはデスティネーションSVMまたはデスティネーション クラスタから実行する必要があります。

   次の例は、svm1上のソース ボリュームvolAとsvm_backup上のデスティネーション ボリュームvolA_dstの間の関係を削除します。

```
cluster_dst::> snapmirror delete -source-path svm1:volA -destination-path
 svm_backup:volA_dst
```

5. 新しいXDPタイプの関係を作成します。 `snapmirror create -source-path SVM:volume|cluster://SVM/volume, ... -destination-path SVM:volume|cluster://SVM/volume, ... -type XDP -schedule schedule -policy policy`

   新しい関係では、同じソース ボリュームとデスティネーション ボリュームを使用する必要があります。コマンド構文全体については、マニュアル ページを参照してください。

   注： このコマンドはデスティネーションSVMまたはデスティネーション クラスタから実行する必要があります。

   次の例は、デフォルトのMirrorAllSnapshotsポリシーを使用して、svm1上のソース ボリュームvolAとsvm_backup上のデスティネーション ボリュームvolA_dstの間のSnapMirror DR関係を作成します。

```
cluster_dst::> snapmirror create -source-path svm1:volA -destination-
path svm_backup:volA_dst -type XDP -schedule my_daily -policy
 MirrorAllSnapshots
```

6. ソース ボリュームとデスティネーション ボリュームを再同期します。`snapmirror resync -source-path SVM:volume|cluster://SVM/volume, ... -destination-path SVM:volume|cluster://SVM/volume, ...`

   コマンド構文全体については、マニュアル ページを参照してください。

   注： このコマンドはデスティネーションSVMまたはデスティネーション クラスタから実行する必要があります。再同期の際にベースライン転送は不要ですが、再同期には時間がかかる場合があります。再同期はオフピークの時間帯に実行することを推奨します。

   次の例は、svm1上のソース ボリュームvolAとsvm_backup上のデスティネーション ボリュームvolA_dstの間の関係を再同期します。

```
cluster_dst::> snapmirror resync -source-path svm1:volA -destination-path
 svm_backup:volA_dst
```

次のタスク

snapmirror showコマンドを使用して、SnapMirror関係が作成されたことを確認します。コマンド構文全体については、マニュアル ページを参照してください。

SnapMirror関係のタイプの変換

SnapMirror Synchronousがサポートされます。非同期SnapMirror関係とSnapMirror Synchronous関係は、ベースライン転送を実行しなくても相互に変換することができます。

このタスクについて

SnapMirrorポリシーを変更して非同期SnapMirror関係とSnapMirror Synchronous関係を相互に変換することはできません。

手順

- **非同期SnapMirror関係からSnapMirror Synchronous関係への変換**
  a) デスティネーション クラスタから、非同期SnapMirror関係を削除します。`snapmirror delete -destination-path SVM:volume`

  ```
  cluster2::>snapmirror delete -destination-path vs1_dr:vol1
  ```

  b) ソース クラスタから、共通のSnapshotコピーは削除せずにSnapMirror関係を解放します。`snapmirror release -relationship-info-only true -destination-path dest_SVM:dest_volume`

  ```
  cluster1::>snapmirror release -relationship-info-only true -
  destination-path vs1_dr:vol1
  ```

  c) デスティネーション クラスタから、SnapMirror Synchronous関係を作成します。`snapmirror create -source-path src_SVM:src_volume -destination-path dest_SVM:dest_volume -policy sync-mirror`

  ```
  cluster2::>snapmirror create -source-path vs1:vol1 -destination-path
   vs1_dr:vol1 -policy sync
  ```

  d) SnapMirror Synchronous関係を再同期します。`snapmirror resync -destination-path dest_SVM:dest_volume`

  ```
  cluster2::>snapmirror resync -destination-path vs1_dr:vol1
  ```

- **SnapMirror Synchronous関係から非同期SnapMirror関係への変換**
  a) デスティネーション クラスタから、既存のSnapMirror Synchronous関係を休止します。`snapmirror quiesce -destination-path dest_SVM:dest_volume`

  ```
  cluster2::> snapmirror quiesce -destination-path vs1_dr:vol1
  ```

  b) デスティネーション クラスタから、非同期SnapMirror関係を削除します。`snapmirror delete -destination-path SVM:volume`

  ```
  cluster2::>snapmirror delete -destination-path vs1_dr:vol1
  ```

  c) ソース クラスタから、共通のSnapshotコピーは削除せずにSnapMirror関係を解放します。`snapmirror release -relationship-info-only true -destination-path dest_SVM:dest_volume`

  ```
  cluster1::>snapmirror release -relationship-info-only true -
  destination-path vs1_dr:vol1
  ```

  d) デスティネーション クラスタから、非同期SnapMirror関係を作成します。`snapmirror create -source-path src_SVM:src_volume -destination-path dest_SVM:dest_volume -policy MirrorAllSnapshots`

  ```
  cluster2::>snapmirror create -source-path vs1:vol1 -destination-path
   vs1_dr:vol1 -policy sync
  ```

  e) SnapMirror Synchronous関係を再同期します。`snapmirror resync -destination-path dest_SVM:dest_volume`

  ```
  cluster2::>snapmirror resync -destination-path vs1_dr:vol1
  ```

SnapMirror Synchronous関係のモードの変換

SnapMirror Synchronous関係がサポートされます。SnapMirror Synchronous関係のモードはStrictSyncとSyncの間で相互に変換できます。

このタスクについて

Snapmirror Synchronous関係のポリシーを変更してモードを変換することはできません。

手順

1. デスティネーション クラスタから、既存のSnapMirror Synchronous関係を休止します。`snapmirror quiesce -destination-path dest_SVM:dest_volume`

   ```
   cluster2::> snapmirror quiesce -destination-path vs1_dr:vol1
   ```

2. デスティネーション クラスタから、既存のSnapMirror Synchronous関係を削除します。`snapmirror delete -destination-path dest_SVM:dest_volume`

   ```
   cluster2::> snapmirror delete -destination-path vs1_dr:vol1
   ```

3. ソース クラスタから、共通のSnapshotコピーは削除せずにSnapMirror関係を解放します。`snapmirror release -relationship-info-only true -destination-path dest_SVM:dest_volume`

   ```
   cluster1::> snapmirror release -relationship-info-only true -destination-path vs1_dr:vol1
   ```

4. デスティネーション クラスタから、変換後のモードを指定してSnapMirror Synchronous関係を作成します。`snapmirror create -source-path vs1:vol1 -destination-path dest_SVM:dest_volume -policy Sync|StrictSync`

   ```
   cluster2::> snapmirror create -source-path vs1:vol1 -destination-path
    vs1_dr:vol1 -policy Sync
   ```

5. デスティネーション クラスタから、SnapMirror関係を再同期します。`snapmirror resync -destination-path dest_SVM:dest_volume`

   ```
   cluster2::> snapmirror resync -destination-path vs1_dr:vol1
   ```

SnapMirror DRデスティネーション ボリュームからのデータの提供

災害によってSnapMirror DR関係のプライマリ サイトが機能しなくなった場合は、システム停止を最小限に抑えてデスティネーション ボリュームからデータを提供できます。プライマリ サイトでサービスが復旧したら、ソース ボリュームを再アクティブ化できます。

デスティネーション ボリュームを書き込み可能にする

デスティネーション ボリュームからクライアントにデータを提供する前に、そのボリュームを書き込み可能にする必要があります。`snapmirror quiesce`コマンドを使用すると、デスティネーションに対するスケジュールされた転送を停止できます。`snapmirror abort`コマンドを使用すると、実行中の転送を停止できます。`snapmirror break`コマンドを使用すると、デスティネーションを書き込み可能にすることができます。

このタスクについて

この手順はデスティネーションSVMまたはデスティネーション クラスタから実行する必要があります。

手順

1. デスティネーションへのスケジュールされた転送を停止します。`snapmirror quiesce -source-path SVM:volume|cluster://SVM/volume, ... -destination-path SVM:volume|cluster://SVM/volume, ...`

   コマンド構文全体については、マニュアル ページを参照してください。

   次の例は、ソース ボリューム`volA`（`svm1`上）とデスティネーション ボリューム`volA_dst`（`svm_backup`上）の間のスケジュールされた転送を停止します。

```
cluster_dst::> snapmirror quiesce -source-path svm1:volA -destination-
path svm_backup:volA_dst
```

2. デスティネーションへの実行中の転送を停止します。`snapmirror abort -source-path SVM:volume|cluster://SVM/volume, ... -destination-path SVM:volume|cluster://SVM/volume, ...`

   コマンド構文全体については、マニュアル ページを参照してください。

   注： SnapMirror Synchronous関係ではこの手順は必要ありません。

   次の例は、ソース ボリュームvolA（svm1上）とデスティネーション ボリュームvolA_dst（svm_backup上）の間の実行中の転送を停止します。

```
cluster_dst::> snapmirror abort -source-path svm1:volA -destination-path
 svm_backup:volA_dst
```

3. SnapMirror DR関係を解除します。`snapmirror break -source-path SVM:volume|cluster://SVM/volume, ... -destination-path SVM:volume|cluster://SVM/volume, ...`

   コマンド構文全体については、マニュアル ページを参照してください。

   次の例は、ソース ボリュームvolA（svm1上）とデスティネーション ボリュームvolA_dst（svm_backup上）の間の関係を解除します。

```
cluster_dst::> snapmirror break -source-path svm1:volA -destination-path
 svm_backup:volA_dst
```

データ アクセスのためのデスティネーション ボリュームの設定

デスティネーション ボリュームを書き込み可能にしたあとで、データにアクセスできるようにそのボリュームを設定する必要があります。NASクライアント、NVMeサブシステム、およびSANホストは、ソース ボリュームが再アクティブ化されるまでの間、デスティネーション ボリュームのデータにアクセスできます。

NAS環境の場合：

1. ソース ボリュームがソースSVMでマウントされていたのと同じジャンクション パスを使用して、NASボリュームをネームスペースにマウントします。
2. デスティネーション ボリュームのCIFS共有に適切なACLを適用します。
3. デスティネーション ボリュームにNFSエクスポート ポリシーを割り当てます。
4. デスティネーション ボリュームにクォータ ルールを適用します。
5. デスティネーション ボリュームにクライアントをリダイレクトします。
6. NFSおよびCIFS共有をクライアントに再マウントします。

SAN環境の場合：

1. ボリューム内のLUNを適切なイニシエータ グループにマッピングします。
2. iSCSIの場合、SANホスト イニシエータからSAN LIFへのiSCSIセッションを作成します。
3. SANクライアントで、ストレージの再スキャンを実行して接続されたLUNを検出します。

NVMe環境については、SANアドミニストレーション ガイドを参照してください。

元のソース ボリュームの再アクティブ化

デスティネーションからデータを提供する必要がなくなった場合は、ソース ボリュームとデスティネーション ボリュームの間で元のデータ保護関係を再確立できます。

このタスクについて

次の手順は、元のソース ボリュームにあるベースラインが損なわれていないことを前提としています。ベースラインが損なわれている場合は、この手順を実行する前に、データの提供元のボリュームと元のソース ボリュームの間の関係を作成して初期化する必要があります。

手順

1. 元のデータ保護関係を削除します。`snapmirror delete -source-path SVM:volume|cluster://SVM/volume, ... -destination-path SVM:volume|cluster://SVM/volume, ...`

   コマンド構文全体については、マニュアル ページを参照してください。

   このコマンドはデスティネーションSVMまたはデスティネーション クラスタから実行する必要があります。

   次の例は、元のソース ボリュームvolA（svm1上）とデータの提供元のボリュームvolA_dst（svm_backup上）の間の関係を削除します。

   ```
   cluster_dst::> snapmirror delete -source-path svm1:volA -destination-path
    svm_backup:volA_dst
   ```

2. 元のデータ保護関係を反転します。`snapmirror resync -source-path SVM:volume|cluster://SVM/volume, ... -destination-path SVM:volume|cluster://SVM/volume, ...`

   コマンド構文全体については、マニュアル ページを参照してください。

   このコマンドはデスティネーションSVMまたはデスティネーション クラスタから実行する必要があります。再同期の際にベースライン転送は不要ですが、再同期には時間がかかる場合があります。再同期はオフピークの時間帯に実行することを推奨します。

   次の例は、元のソース ボリュームvolA（svm1上）とデータの提供元のボリュームvolA_dst（svm_backup上）の間の関係を反転します。

   ```
   cluster_dst::> snapmirror resync -source-path svm_backup:volA_dst -
   destination-path svm1:volA
   ```

3. 反転した関係のソースSVMを停止します。`vserver stop -vserver SVM`

   コマンド構文全体については、マニュアル ページを参照してください。

   次の例は、反転した関係のソースSVMを停止します。

   ```
   cluster_src::> vserver stop svm_backup
   ```

4. 反転した関係を更新します。`snapmirror update -source-path SVM:volume|cluster://SVM/volume, ... -destination-path SVM:volume|cluster://SVM/volume, ...`

   コマンド構文全体については、マニュアル ページを参照してください。

   注： このコマンドはデスティネーションSVMまたはデスティネーション クラスタから実行する必要があります。ソースとデスティネーションに共通のSnapshotコピーが存在しない場合、このコマンドは失敗します。`snapmirror initialize`を使用して、関係を再初期化してください。

   次の例は、データの提供元のボリュームvolA_dst（svm_backup上）と元のソース ボリュームvolA（svm1上）の間の関係を更新します。

   ```
   cluster_dst::> snapmirror update -source-path svm_backup:volA_dst -
   destination-path svm1:volA
   ```

5. 反転した関係のスケジュールされた転送を停止します。 `snapmirror quiesce -source-path SVM:volume|cluster://SVM/volume, ... -destination-path SVM:volume|cluster://SVM/volume, ...`

   コマンド構文全体については、マニュアル ページを参照してください。

   注： このコマンドはデスティネーションSVMまたはデスティネーション クラスタから実行する必要があります。

   次の例は、データの提供元のボリュームvolA_dst（svm_backup上）と元のソース ボリュームvolA（svm1上）の間のスケジュールされた転送を停止します。

   ```
   cluster_dst::> snapmirror quiesce -source-path svm_backup:volA_dst -
   destination-path svm1:volA
   ```

6. 反転した関係の実行中の転送を停止します。 `snapmirror abort -source-path SVM:volume|cluster://SVM/volume, ... -destination-path SVM:volume|cluster://SVM/volume, ...`

   コマンド構文全体については、マニュアル ページを参照してください。

   注： このコマンドはデスティネーションSVMまたはデスティネーション クラスタから実行する必要があります。

   次の例は、データの提供元のボリュームvolA_dst（svm_backup上）と元のソース ボリュームvolA（svm1上）の間の実行中の転送を停止します。

   ```
   cluster_dst::> snapmirror abort -source-path svm_backup:volA_dst -
   destination-path svm1:volA
   ```

7. 反転した関係を解除します。 `snapmirror break -source-path SVM:volume|cluster://SVM/volume, ... -destination-path SVM:volume|cluster://SVM/volume, ...`

   コマンド構文全体については、マニュアル ページを参照してください。

   注： このコマンドはデスティネーションSVMまたはデスティネーション クラスタから実行する必要があります。

   次の例は、データの提供元のボリュームvolA_dst（svm_backup上）と元のソース ボリュームvolA（svm1上）の間の関係を解除します。

   ```
   cluster_dst::> snapmirror break -source-path svm_backup:volA_dst -
   destination-path svm1:volA
   ```

8. 元のソースSVMを起動します。`vserver start -vserver SVM`

   コマンド構文全体については、マニュアル ページを参照してください。

   次の例は、元のソースSVMを起動します。

   ```
   cluster_dst::> vserver start svm1
   ```

9. 反転したデータ保護関係を削除します。`snapmirror delete -source-path SVM:volume|cluster://SVM/volume, ... -destination-path SVM:volume|cluster://SVM/volume, ...`

   コマンド構文全体については、マニュアル ページを参照してください。

   このコマンドは反転した関係のソースSVMまたはソース クラスタから実行する必要があります。

   次の例は、元のソース ボリュームvolA（svm1上）とデータの提供元のボリュームvolA_dst（svm_backup上）の間の反転した関係を削除します。

   ```
   cluster_src::> snapmirror delete -source-path svm_backup:volA_dst -
   destination-path svm1:volA
   ```

10. 元のデータ保護関係を再確立します。`snapmirror resync -source-path SVM:volume|cluster://SVM/volume, ... -destination-path SVM:volume|cluster://SVM/volume, ...`

    コマンド構文全体については、マニュアル ページを参照してください。

    次の例は、元のソース ボリュームvolA（svm1上）と元のデスティネーション ボリュームvolA_dst（svm_backup上）の間の関係を再確立します。

    ```
    cluster_dst::> snapmirror resync -source-path svm1:volA -destination-path
     svm_backup:volA_dst
    ```

次のタスク

`snapmirror show`コマンドを使用して、SnapMirror関係が作成されたことを確認します。コマンド構文全体については、マニュアル ページを参照してください。

SnapMirrorデスティネーション ボリュームからのファイルのリストア

SnapMirrorデスティネーション ボリューム内のSnapshotコピーを使用して、個々のファイル、LUN、NVMeネームスペースをリカバリしたり、ボリュームの内容全体をリストアしたりできます。ファイルは元のソース ボリュームにリストアするか、別のボリュームにリストアできます。

SnapMirrorデスティネーションからの単一ファイル、LUN、NVMeネームスペースのリストア

単一ファイルまたはLUN、あるいは一連のファイルまたはLUNをSnapshotコピーからリストアしたり、NVMeネームスペースをSnapMirrorデスティネーション ボリュームからリストアしたりできます。ONTAP 9.7以降では、SnapMirror SynchronousデスティネーションからNVMeネームスペースをリストアすることもできます。ファイルは元のソース ボリュームにリストアするか、別のボリュームにリストアできます。

始める前に

ファイルまたはLUNをSnapMirror Synchronousデスティネーションからリストアするには、先に関係を削除して解放しておく必要があります。

このタスクについて

ファイルまたはLUNのリストア先のボリューム（デスティネーション ボリューム）は読み書き可能なボリュームにする必要があります。

- ソース ボリュームとデスティネーション ボリュームに共通のSnapshotコピーがある（通常、リストア先が元のソース ボリュームである）場合、SnapMirrorは増分リストアを実行します。
- それ以外の場合、SnapMirrorはベースライン リストアを実行します。これにより、指定されたSnapshotコピーおよびコピーが参照するすべてのデータ ブロックがデスティネーション ボリュームに転送されます。

手順

1. デスティネーション ボリューム内のSnapshotコピーの一覧を表示します。`volume snapshot show -vserver SVM -volume volume`

   コマンド構文全体については、マニュアル ページを参照してください。

   次の例は、`vserverB:secondary1`デスティネーション上のSnapshotコピーを示しています。

```
cluster_dst::> volume snapshot show -vserver vserverB -volume secondary1

Vserver     Volume      Snapshot                 State    Size  Total% Used%
-------     ------      ---------- ----------- ------   -----  ------ -----
vserverB    secondary1  hourly.2013-01-25_0005  valid   224KB     0%    0%
                        daily.2013-01-25_0010   valid   92KB      0%    0%
                        hourly.2013-01-25_0105  valid   228KB     0%    0%
                        hourly.2013-01-25_0205  valid   236KB     0%    0%
                        hourly.2013-01-25_0305  valid   244KB     0%    0%
                        hourly.2013-01-25_0405  valid   244KB     0%    0%
                        hourly.2013-01-25_0505  valid   244KB     0%    0%

7 entries were displayed.
```

2. 単一ファイルまたはLUN、あるいは一連のファイルまたはLUNを、SnapMirrorデスティネーション ボリューム内のSnapshotコピーからリストアします。`snapmirror restore -source-path SVM:volume|cluster://SVM/volume, ... -destination-path SVM:volume|cluster://SVM/volume, ... -source-snapshot snapshot -file-list source_file_path,@destination_file_path`

   コマンド構文全体については、マニュアル ページを参照してください。

   注： このコマンドはデスティネーションSVMまたはデスティネーション クラスタから実行する必要があります。

次のコマンドは、file1ファイルとfile2ファイルを元のデスティネーション ボリュームsecondary1内のSnapshotコピーdaily.2013-01-25_0010から、元のソース ボリュームprimary1のアクティブ ファイルシステム内の同じ場所にリストアします。

```
cluster_dst::> snapmirror restore -source-path vserverB:secondary1 -
destination-path vserverA:primary1 -source-snapshot daily.2013-01-25_0010
 -file-list /dir1/file1,/dir2/file2

[Job 3479] Job is queued: snapmirror restore for the relationship with
 destination vserverA:primary1
```

次のコマンドは、file1ファイルとfile2ファイルを元のデスティネーション ボリュームsecondary1内のSnapshotコピーdaily.2013-01-25_0010から、元のソース ボリュームprimary1のアクティブ ファイルシステム内の別の場所にリストアします。

@マークに続くパスがデスティネーション ファイルのパスで、元のソース ボリュームのルートからのパスを指定しています。この例では、file1がprimary1の/dir1/file1.newにリストアされ、file2が/dir2.new/file2にリストアされます。

```
cluster_dst::> snapmirror restore -source-path vserverB:secondary1 -
destination-path vserverA:primary1 -source-snapshot daily.2013-01-25_0010
 -file-list /dir/file1,@/dir1/file1.new,/dir2/file2,@/dir2.new/file2

[Job 3479] Job is queued: snapmirror restore for the relationship with
 destination vserverA:primary1
```

次のコマンドは、file1ファイルとfile3ファイルを元のデスティネーション ボリュームsecondary1内のSnapshotコピーdaily.2013-01-25_0010から、元のソース ボリュームprimary1のアクティブ ファイルシステム内の別の場所にリストアし、file2をsnap1からprimary1のアクティブ ファイルシステム内の同じ場所にリストアします。

この例では、file1が/dir1/file1.newにリストアされ、file3が/dir3.new/file3にリストアされます。

```
cluster_dst::> snapmirror restore -source-path vserverB:secondary1 -
destination-path vserverA:primary1 -source-snapshot daily.2013-01-25_0010
 -file-list /dir/file1,@/dir1/file1.new,/dir2/file2,/dir3/file3,@/
dir3.new/file3

[Job 3479] Job is queued: snapmirror restore for the relationship with
 destination vserverA:primary1
```

SnapMirrorデスティネーションからのボリューム内容のリストア

SnapMirrorデスティネーション ボリューム内のSnapshotコピーからボリューム全体の内容をリストアできます。ボリュームの内容は元のソース ボリュームにリストアするか、別のボリュームにリストアできます。

始める前に

ボリュームをSnapMirror Synchronousデスティネーションからリストアするには、先に関係を削除して解放しておく必要があります。

このタスクについて

リストア処理のデスティネーション ボリュームは次のいずれかにする必要があります。

- 読み書き可能なボリューム。このケースでは、ソース ボリュームとデスティネーション ボリュームに共通のSnapshotコピーがある（通常、リストア先が元のソース ボリュームである）場合、SnapMirrorは増分リストアを実行します。

注： 共通のSnapshotコピーがない場合、コマンドは失敗します。空の読み書き可能なボリュームにボリュームの内容をリストアすることはできません。

- 空のデータ保護ボリューム。このケースでは、SnapMirrorはベースライン リストアを実行します。これにより、指定されたSnapshotコピーおよびコピーが参照するすべてのデータ ブロックがソース ボリュームに転送されます。

ボリュームの内容のリストアはシステム停止を伴う処理です。リストア処理を実行しているときに、SnapVaultプライマリ ボリューム上でCIFSトラフィックを実行してはなりません。

リストア処理のデスティネーション ボリュームで圧縮が有効になっていて、ソース ボリュームで圧縮が有効になっていない場合は、デスティネーション ボリュームで圧縮を無効にします。リストア処理が完了したら、圧縮を再度有効にする必要があります。

デスティネーション ボリュームに対して定義されたクォータ ルールは、リストアの実行前に非アクティブ化されます。`volume quota modify`コマンドを使用すると、リストア処理の完了後にクォータ ルールを再アクティブ化できます。

手順

1. デスティネーション ボリューム内のSnapshotコピーの一覧を表示します。`volume snapshot show -vserver` *`SVM`* `-volume` *`volume`*

   コマンド構文全体については、マニュアル ページを参照してください。

   次の例は、`vserverB:secondary1`デスティネーション上のSnapshotコピーを示しています。

```
cluster_dst::> volume snapshot show -vserver vserverB -volume secondary1

Vserver     Volume      Snapshot                 State    Size  Total% Used%
-------     ------      ---------- ----------- ------   -----  ------ -----
vserverB    secondary1  hourly.2013-01-25_0005  valid   224KB     0%    0%
                        daily.2013-01-25_0010   valid   92KB      0%    0%
                        hourly.2013-01-25_0105  valid   228KB     0%    0%
                        hourly.2013-01-25_0205  valid   236KB     0%    0%
                        hourly.2013-01-25_0305  valid   244KB     0%    0%
                        hourly.2013-01-25_0405  valid   244KB     0%    0%
                        hourly.2013-01-25_0505  valid   244KB     0%    0%

7 entries were displayed.
```

2. SnapMirrorデスティネーション ボリューム内のSnapshotコピーからボリュームの内容をリストアします。`snapmirror restore -source-path` *`SVM`*`:`*`volume`*`|`*`cluster`*`://`*`SVM/volume`*`, ... -destination-path` *`SVM`*`:`*`volume`*`|`*`cluster`*`://`*`SVM/volume`*`, ... -source-snapshot` *`snapshot`*

   コマンド構文全体については、マニュアル ページを参照してください。

   注： このコマンドはデスティネーションSVMまたはデスティネーション クラスタから実行する必要があります。

   次のコマンドは、元のデスティネーション ボリューム`secondary1`内のSnapshotコピー`daily.2013-01-25_0010`から元のソース ボリューム`primary1`の内容をリストアします。

```
cluster_dst::> snapmirror restore -source-path vserverB:secondary1 -
destination-path vserverA:primary1 -source-snapshot daily.2013-01-25_0010

Warning: All data newer than Snapshot copy daily.2013-01-25_0010 on
 volume vserverA:primary1 will be deleted.

Do you want to continue? {y|n}: y

[Job 34] Job is queued: snapmirror restore from source
 vserverB:secondary1 for the snapshot daily.2013-01-25_0010.
```

3. リストアしたボリュームを再マウントし、ボリュームを使用するすべてのアプリケーションを再起動します。

手動によるレプリケーション関係の更新

ソース ボリュームが移動されたために更新に失敗した場合、レプリケーション関係を手動で更新しなければならないことがあります。

このタスクについて

レプリケーション関係を手動で更新するまで、SnapMirrorは移動されたソース ボリュームからの転送をすべて中止します。

SnapMirror Synchronous関係がサポートされます。これらの関係ではソース ボリュームとデスティネーション ボリュームは常に同期された状態ですが、セカンダリ クラスタの表示は1時間おきにしかプライマリと同期されません。デスティネーションで最新のデータを表示するには、`snapmirror update`コマンドを実行して手動更新を実行する必要があります。

手順

レプリケーション関係を手動で更新します。`snapmirror update -source-path *SVM*:*volume*|*cluster*:*//SVM/volume*, ... -destination-path *SVM*:*volume*|*cluster*:*//SVM/volume*, ...`

コマンド構文全体については、マニュアル ページを参照してください。

注： このコマンドはデスティネーションSVMまたはデスティネーション クラスタから実行する必要があります。ソースとデスティネーションに共通のSnapshotコピーが存在しない場合、このコマンドは失敗します。`snapmirror initialize`を使用して、関係を再初期化してください。

次の例は、ソース ボリューム`volA`（`svm1`上）とデスティネーション ボリューム`volA_dst`（`svm_backup`上）の間の関係を更新します。

```
cluster_src::> snapmirror update -source-path svm1:volA -destination-path
 svm_backup:volA_dst
```

レプリケーション関係の再同期

デスティネーション ボリュームを書き込み可能にしたあと、ソース ボリュームとデスティネーション ボリュームに共通のSnapshotコピーが存在しないために更新が失敗したあと、または関係のレプリケーション ポリシーを変更した場合には、レプリケーション関係の再同期が必要です。

このタスクについて

再同期の際にベースライン転送は不要ですが、再同期には時間がかかる場合があります。再同期はオフピークの時間帯に実行することを推奨します。

手順

ソース ボリュームとデスティネーション ボリュームを再同期します。`snapmirror resync -source-path *SVM*:*volume*|*cluster*:*//SVM/volume*, ... -destination-path *SVM*:*volume*|*cluster*:*//SVM/volume*, ... -type DP|XDP -schedule *schedule* -policy *policy*`

コマンド構文全体については、マニュアル ページを参照してください。

注： このコマンドはデスティネーションSVMまたはデスティネーション クラスタから実行する必要があります。

次の例は、ソース ボリューム`volA`（`svm1`上）とデスティネーション ボリューム`volA_dst`（`svm_backup`上）の間の関係を再同期します。

```
cluster_dst::> snapmirror resync -source-path svm1:volA -destination-path
 svm_backup:volA_dst
```

ボリューム レプリケーション関係の削除

`snapmirror delete`コマンドと`snapmirror release`コマンドを使用すると、ボリューム レプリケーション関係を削除できます。続いて、不要なデスティネーション ボリュームを手動で削除することができます。

このタスクについて

`snapmirror release`コマンドは、SnapMirrorによって作成されたSnapshotコピーをソースから削除します。-relationship-info-onlyオプションを使用するとSnapshotコピーを保持できます。

手順

1. SnapMirror Synchronous関係がある場合は、レプリケーション関係を休止します。 `snapmirror quiesce -destination-path SVM:volume|cluster://SVM/volume`

   ```
   cluster_src::> snapmirror quiesce -destination-path svm_backup:volA_dst
   ```

2. レプリケーション関係を削除します。`snapmirror delete -source-path SVM:volume|cluster://SVM/volume, ... -destination-path SVM:volume|cluster://SVM/volume, ...`

   コマンド構文全体については、マニュアル ページを参照してください。

   注： このコマンドはデスティネーション クラスタまたはデスティネーションSVMから実行する必要があります。

   次の例は、ソース ボリューム`volA`（`svm1`上）とデスティネーション ボリューム`volA_dst`（`svm_backup`上）の間の関係を削除します。

   ```
   cluster_dst::> snapmirror delete -source-path svm1:volA -destination-path
    svm_backup:volA_dst
   ```

3. ソースSVMからレプリケーション関係情報をリリースします。`snapmirror release -source-path SVM:volume|cluster://SVM/volume, ... -destination-path SVM:volume|cluster://SVM/volume, ...`

   コマンド構文全体については、マニュアル ページを参照してください。

   注： このコマンドはソース クラスタまたはソースSVMから実行する必要があります。

   次の例は、ソースSVM `svm1`から指定されたレプリケーション関係の情報をリリースします。

   ```
   cluster_src::> snapmirror release -source-path svm1:volA -destination-
   path svm_backup:volA_dst
   ```

ストレージ効率の管理

SnapMirrorは、ソース ボリュームとデスティネーション ボリュームでストレージ効率を維持します。ただし、例外が1つあり、デスティネーションでポストプロセス データ圧縮が有効になっている場合、ストレージ効率は維持されません。その場合、デスティネーションではすべてのストレージ効率が失われます。この問題を修正するには、デスティネーションでポストプロセス圧縮を無効にして、関係を手動で更新し、Storage Efficiencyを再度有効にする必要があります。

始める前に

- ソース / デスティネーション クラスタとSVMのピア関係が確立されている必要があります。

  クラスタ / SVMピアリング エクスプレス ガイド
- デスティネーションでポストプロセス圧縮を無効にする必要があります。

このタスクについて

volume efficiency showコマンドを使用して、ボリュームで効率性が有効になっているかどうかを確認できます。詳細については、マニュアル ページを参照してください。

SnapMirror監査ログを表示して転送の説明を見つけることで、SnapMirrorがストレージ効率を維持しているかどうかを確認できます。転送の説明に「transfer_desc = logical transfer」と表示されている場合、SnapMirrorはストレージ効率を維持していません。転送の説明に「Transfer _desc = Logical Transfer with Storage Efficiency」と表示されている場合、SnapMirrorはストレージの効率性を維持しています。次に例を示します。

```
Fri May 22 02:13:02 CDT 2020 ScheduledUpdate[May 22 02:12:00]:cc0fbc29-
b665-11e5-a626-00a09860c273 Operation-Uuid=39fbcf48-550a-4282-a906-
df35632c73a1 Group=none Operation-Cookie=0 action=End source=<sourcepath>
 destination=<destpath> status=Success bytes_transferred=117080571
 network_compression_ratio=1.0:1 transfer_desc=Logical Transfer - Optimized
 Directory Mode
```

ストレージを使用した論理転送

Storage Efficiencyを再度有効にするための手動更新が不要になりました。SnapMirrorでは、ポストプロセス圧縮が無効になったことを検出すると、スケジュールによる次回の更新の際にStorage Efficiencyを自動的に再度有効にします。

デスティネーション ボリュームが書き込み可能になったあとで、ETERNUS AXシリーズがStorage Efficiencyの設定をETERNUS HXシリーズとは異なる方法で管理します。

- snapmirror breakコマンドを使用してデスティネーション ボリュームを書き込み可能にすると、ボリューム上のキャッシング ポリシーが自動的に「auto」（デフォルト）に設定されます。

  注： この動作はFlexVolにのみ該当し、FlexGroupボリュームには該当しません。
- 再同期の際、元の設定に関係なく、キャッシング ポリシーは自動的に「none」に設定され、重複排除とインライン圧縮は自動的に無効になります。必要に応じて、設定を手動で変更する必要があります。

注： Storage Efficiencyが有効な状態での手動更新には時間がかかる場合があります。この処理はオフピークの時間帯に実行することを推奨します。

手順

レプリケーション関係を更新して、Storage Efficiencyを再度有効にします。snapmirror update -source-path *SVM*:*volume*|*cluster*:*//SVM/volume*, ... -destination-path *SVM*:*volume*|*cluster*:*//SVM/volume*, ... -enable-storage-efficiency true

コマンド構文全体については、マニュアル ページを参照してください。

注： このコマンドはデスティネーションSVMまたはデスティネーション クラスタから実行する必要があります。ソースとデスティネーションに共通のSnapshotコピーが存在しない場合、このコマンドは失敗します。snapmirror initializeを使用して、関係を再初期化してください。

次の例は、svm1上のソース ボリュームvolAとsvm_backup上のデスティネーション ボリュームvolA_dstの間の関係を更新し、Storage Efficiencyを再度有効にします。

```
cluster_dst::> snapmirror update -source-path svm1:volA -destination-path
 svm_backup:volA_dst -enable-storage-efficiency true
```

SnapMirrorグローバル スロットルの使用

ノード単位のSnapMirror転送とSnapVault転送にグローバル ネットワーク スロットルを使用できます。

このタスクについて

SnapMirrorグローバル スロットルは、送受信されるSnapMirror転送およびSnapVault転送で使用する帯域幅を制限します。この制限は、クラスタ内のすべてのノードで適用されます。

たとえば、送信スロットルを100Mbpsに設定した場合は、クラスタ内の各ノードで送信帯域幅が100Mbpsに設定されます。グローバル スロットルを無効にすると、すべてのノードで無効になります。

注： このスロットルは、`volume move`転送や負荷共有ミラー転送には影響しません。データ転送速度は多くの場合ビット/秒（bps）で表されますが、スロットル値はキロバイト/秒（KBps）で入力する必要があります。

グローバル スロットルは、SnapMirror転送およびSnapVault転送の関係ごとのスロットル機能と連動します。関係ごとのスロットルは、関係ごとの転送の帯域幅の合計がグローバル スロットルの値を超えるまで有効で、超えたあとはグローバル スロットルが有効になります。スロットル値`0`を指定すると、グローバル スロットルが無効になります。

重要： グローバル スロットルは、SnapMirror Synchronous関係が含まれるクラスタでは有効にしないでください。

手順

1. グローバル スロットルを有効にします。`options -option-name replication.throttle.enable on|off`
   次の例は、`cluster_dst`でSnapMirrorグローバル スロットルを有効にする方法を示しています。

   ```
   cluster_dst::> options -option-name replication.throttle.enable on
   ```

2. 受信転送で使用される総帯域幅について最大値を指定します。 `options -option-name replication.throttle.incoming.max_kbs` *KBps*

   推奨される最小スロットル帯域幅は4kbps で、最大は2Tbpsです。 このオプションのデフォルト値は`unlimited`です。これは、使用できる総帯域幅に制限がないことを意味します。

   次の例は、受信転送で使用される総帯域幅について最大値を100Mbpsに設定する方法を示しています。

   ```
   cluster_dst::> options -option-name replication.throttle.incoming.max_kbs
    12500
   ```

   注： 100Mbps = 12500KBps

3. 送信転送で使用される総帯域幅について最大値を指定します。 `options -option-name replication.throttle.outgoing.max_kbs` *KBps*

   KBpsは最大転送速度（KB/秒）です。有効な転送速度の値は1～125000です。このオプションのデフォルト値は`unlimited`です。これは、使用できる総帯域幅に制限がないことを意味します。

   次の例は、送信転送で使用される総帯域幅について最大値を100Mbpsに設定する方法を示しています。

   ```
   cluster_dst::> options -option-name replication.throttle.outgoing.max_kbs
    12500
   ```

SnapMirror SVMレプリケーションの概要

SnapMirrorを使用すると、SVM間のデータ保護関係を作成できます。このタイプのデータ保護関係では、SVMのすべてまたは一部の設定がNFSエクスポートおよびSMB共有からRBACにレプリケートされます。また、SVMが所有するボリューム内のデータもレプリケートされます。

サポートされている関係タイプ

レプリケート可能なのはデータ提供用SVMのみです。サポートされているデータ保護関係タイプは次のとおりです。

- *SnapMirror DR*：通常、デスティネーションにはソースSVM上に現在あるSnapshotコピーだけが含まれます。
- *SnapMirror*ユニファイド レプリケーション：デスティネーションはDRと長期保持の両用途に設定されます。

これらの関係タイプの詳細については、SnapMirrorボリューム レプリケーションの概要（12ページ）を参照してください。

レプリケーション ポリシーのポリシー タイプによって、そのポリシーがサポートする関係のタイプが決まります。次の表は、使用可能なポリシー タイプを示しています。

表 9 :

ポリシー タイプ	関係タイプ
async-mirror	SnapMirror DR
mirror-vault	ユニファイド レプリケーション

デフォルトのSVMレプリケーションの変更（DP→XDP）

SVMデータ保護関係のデフォルトがXDPモードに変更されました。

新しいデフォルトは既存の関係には影響しません。DPタイプの既存の関係は引き続きDPタイプになります。次の表は、想定される動作を示しています。

表 10 :

指定するモード	タイプ	デフォルト ポリシー（ポリシーを指定しない場合）
DP	XDP	MirrorAllSnapshots（SnapMirror DR）
なし	XDP	MirrorAllSnapshots（SnapMirror DR）
XDP	XDP	MirrorAndVault（ユニファイド レプリケーション）

デフォルトの変更の詳細については、SnapMirrorのデフォルトの変更（DPからXDP）（19ページ）を参照してください。

注 : バージョンに依存しないレプリケーションは、SVMレプリケーションではサポートされません。

SnapMirror関係に対応したONTAPバージョン（79ページ）

SVM設定のレプリケート方法

SVMレプリケーション関係の内容は、以下のフィールドによって決定します。

- `snapmirror create`コマンドの-identity-preserve trueオプションは、SVMの設定全体をレプリケートします。

-identity-preserve falseオプションは、SVMのボリュームと認証および許可の設定と、SVM DR関係でレプリケートされる設定（54ページ）に指定されたプロトコルとネーム サービスの設定のみをレプリケートします。

- `snapmirror policy create`コマンドの-discard-configs networkオプションは、ソースとデスティネーションのSVMが異なるサブネットにある場合を想定し、LIFと関連ネットワークの設定をSVMレプリケーションの対象から除外します。
- `volume modify`コマンドの-vserver-dr-protection unprotectedオプションは、指定されたボリュームをSVMレプリケーションの対象から除外します。

上記の点を除き、SVMレプリケーションはボリューム レプリケーションとほぼ同じです。ボリューム レプリケーションとほぼ同じワークフローをSVMレプリケーションにも使用できます。

サポートの詳細

次の表は、SnapMirror SVMレプリケーションのサポートの詳細を示しています。

表 11 :

リソースまたは機能	サポートの詳細
関係タイプ	• SnapMirror DR • SnapMirrorユニファイド レプリケーション
レプリケーションの範囲	クラスタ間のみ。同じクラスタ内のSVMをレプリケートすることはできません。
バージョンへの非依存性	サポートされていません。
構成タイプ	• 1つのソースから1つのデスティネーション • ONTAP 9.7以降はファンアウト（ただしファンアウトのデスティネーションは2つのみ） デフォルトでは、-identity-preserve trueの関係はソースSVMごとに1つだけ許可されます。
ボリューム暗号化	• ソースで暗号化されたボリュームがデスティネーションで暗号化されます。 • オンボード キー マネージャまたはKMIPサーバをデスティネーションで設定する必要があります。 • 新しい暗号化キーはデスティネーションで生成されます。 • ボリューム暗号化をサポートするノードがデスティネーションに含まれていない場合、レプリケーションは成功しますが、デスティネーション ボリュームは暗号化されません。
FabricPool	ONTAP 9.7以降のFabricPoolでは、SnapMirror SVMレプリケーションがサポートされます。

リソースまたは機能	サポートの詳細
MetroCluster	ONTAP 9.7以降のMetroCluster構成では、SnapMirror SVMレプリケーションがサポートされます。 • MetroCluster構成をSVM DR関係のデスティネーションにすることはできません。 • SVM DR関係のソースとして使用できるのは、MetroCluster構成のアクティブなSVMだけです。 スイッチオーバー前の同期元のSVMとスイッチオーバー後の同期先のSVMのどちらもソースに使用できます。 • MetroCluster構成が安定した状態のときはMetroClusterの同期先のSVMがオンラインでないため、同期先のボリュームをSVM DR関係のソースにすることはできません。 • SVM DR関係のソースが同期元のSVMの場合、ソースのSVM DR関係情報がMetroClusterパートナーにレプリケートされます。 • スイッチオーバーおよびスイッチバックの実行中に、SVM DRのデスティネーションへのレプリケーションが失敗することがあります。 ただし、スイッチオーバーまたはスイッチバック処理の完了後、SVM DRの次回のスケジュールされている更新は成功します。
SnapMirror Synchronous	SVM DRではサポートされません。

SVM DR関係でレプリケートされる設定

次の表は、`snapmirror create`-identity-preserve オプションと`snapmirror policy create`-discard-configs networkオプションでレプリケートされる設定を示しています。

表 12 :

レプリケートされる設定		-identity-preserve true		-identity-preserve false
		-discard-configs networkが設定されていないポリシー	-discard-configs networkが設定されたポリシー	
ネットワーク	NAS LIF	○	×	×
	LIFのKerberos設定	○	×	×
	SAN LIF	×	×	×
	ファイアウォールポリシー	○	○	×
	Routes	○	×	×
	ブロードキャストドメイン	×	×	×
	サブネット	×	×	×
	IPspace	×	×	×
CIFS	CIFSサーバ	○	○	×
	ローカル グループおよびローカルユーザ	○	○	○

レプリケートされる設定		**-identity-preserve true**		**-identity-preserve false**
		-discard-configs networkが設定されていないポリシー	**-discard-configs networkが設定されたポリシー**	
	権限	○	○	○
	シャドウ コピー	○	○	○
	BranchCache	○	○	○
	サーバ オプション	○	○	○
	サーバ セキュリティ	○	○	×
	ホーム ディレクトリ、共有	○	○	○
	シンボリックリンク	○	○	○
	Fpolicyポリシー、Fsecurityポリシー、およびFsecurity NTFS	○	○	○
	ネーム マッピングおよびグループ マッピング	○	○	○
	監査情報	○	○	○
NFS	エクスポート ポリシー	○	○	×
	エクスポート ポリシー ルール	○	○	×
	NFSサーバ	○	○	×
RBAC	セキュリティ証明書	○	○	×
	ログイン ユーザ、公開鍵、ロール、およびロール設定	○	○	○
	SSL	○	○	×
ネーム サービス	DNSおよびDNSホスト	○	○	×
	UNIXユーザおよびUNIXグループ	○	○	○
	Kerberos RealmおよびKerberosキーブロック	○	○	×
	LDAPおよびLDAPクライアント	○	○	×

レプリケートされる設定		**-identity-preserve true**		**-identity-preserve false**
		-discard-configs networkが設定されていないポリシー	**-discard-configs networkが設定されたポリシー**	
	Netgroup	○	○	×
	NIS	○	○	×
	WebおよびWebアクセス	○	○	×
ボリューム	オブジェクト	○	○	○
	Snapshotコピー、Snapshotポリシー、および自動削除ポリシー	○	○	○
	効率化ポリシー	○	○	○
	クォータ ポリシーおよびクォータ ポリシー ルール	○	○	○
	リカバリ キュー	○	○	○
ルート ボリューム	ネームスペース	○	○	○
	ユーザ データ	×	×	×
	qtree	×	×	×
	クォータ	×	×	×
	ファイルレベルのQoS	×	×	×
	属性：ルート ボリュームの状態、スペース ギャランティ、サイズ、オートサイズ、およびファイル総数	×	×	×
ストレージQoS	QoSポリシー グループ	○	○	○
Fibre Channel（FC）		×	×	×
iSCSI		×	×	×
LUN	オブジェクト	○	○	○
	igroup数	×	×	×
	ポートセット	×	×	×
	シリアル番号	×	×	×
SNMP	v3ユーザ	○	○	×

SnapMirror SVMレプリケーションの管理

SnapMirrorを使用すると、SVMのすべてまたは一部の設定、およびSVMが所有するボリュームをレプリケートできます。つまり、サービスが復旧したらソースSVMを再アクティブ化できるのと同様に、プライマリ サイトで災害が発生したときにデスティネーションSVMを簡単にアクティブ化できます。

SVMの設定のレプリケート

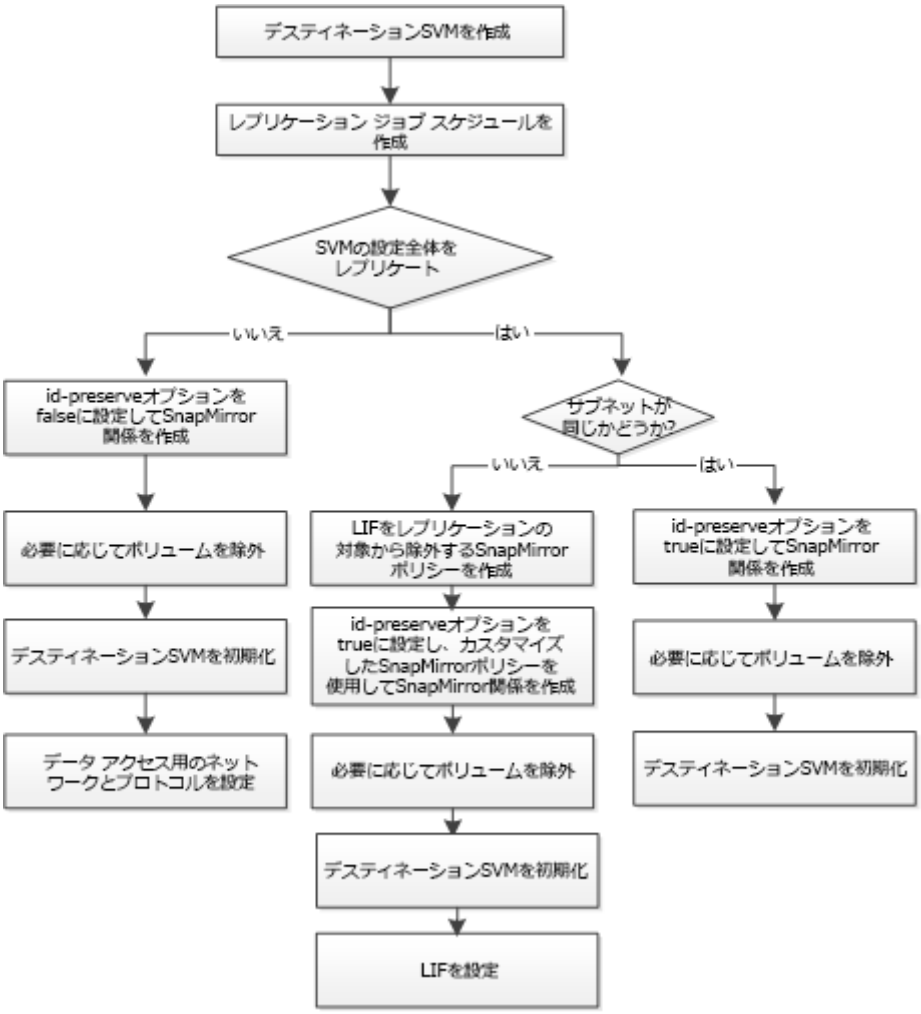
SVMレプリケーションはボリューム レプリケーションとほぼ同じです。レプリケート対象となるSVMの設定の量を指定する作業を除き、ボリュームをレプリケートする場合とほぼ同じワークフローを使用してSVMをレプリケートできます。

SnapMirrorレプリケーションのワークフロー（27ページ）

SnapMirror SVMレプリケーションのワークフロー

SnapMirror SVMレプリケーションでは、デスティネーションSVMを作成し、レプリケーション ジョブ スケジュールを作成し、SnapMirror関係を作成して初期化します。

注： このワークフローでは、デフォルト ポリシーまたはカスタム レプリケーション ポリシーをすでに使用していることを前提としています。



SVMの設定全体のレプリケート

`snapmirror create`コマンドの-identity-preserve trueオプションを使用すると、SVMの設定全体をレプリケートできます。

始める前に

ソース / デスティネーション クラスタとSVMのピア関係が確立されている必要があります。

コマンド構文全体については、マニュアル ページを参照してください。

このタスクについて

このワークフローでは、デフォルト ポリシーまたはカスタム レプリケーション ポリシーをすでに使用していることを前提としています。

手順

1. デスティネーションSVMを作成します。 `vserver create -vserver *SVM_name* -subtype dp-destination`

   SVM名はソース クラスタとデスティネーション クラスタの間で一意である必要があります。

   次の例は、`svm_backup`という名前のデスティネーションSVMを作成します。

   ```
   cluster_dst:> vserver create -vserver svm_backup -subtype dp-destination
   ```

2. レプリケーション ジョブ スケジュールを作成します。 `job schedule cron create -name *job_name* -month *month* -dayofweek *day_of_week* -day *day_of_month* -hour *hour* -minute *minute*`

   -month、-dayofweek、-hourに`all`を指定すると、ジョブを毎月、毎日、1時間ごとにそれぞれ実行できます。

   次の例は、土曜日の午前3時に実行する`my_weekly`というジョブ スケジュールを作成します。

   ```
   cluster_dst::> job schedule cron create -name my_weekly -dayofweek
    "Saturday" -hour 3 -minute 0
   ```

3. デスティネーションSVMまたはデスティネーション クラスタから、レプリケーション関係を作成します。 `snapmirror create -source-path *SVM_name*: -destination-path *SVM_name*: -type DP|XDP -schedule *schedule* -policy *policy* -identity-preserve true`

   注： -source-pathオプションと-destination-pathオプションでは、SVM名のあとにコロン（:）を入力する必要があります。

   次の例は、デフォルトの`MirrorAllSnapshots`ポリシーを使用して、SnapMirror DR関係を作成します。

   ```
   cluster_dst::> snapmirror create -source-path svm1: -destination-path
    svm_backup: -type XDP -schedule my_daily -policy MirrorAllSnapshots -
   identity-preserve true
   ```

   次の例は、デフォルトの`MirrorAndVault`ポリシーを使用して、ユニファイド レプリケーション関係を作成します。

   ```
   cluster_dst:> snapmirror create -source-path svm1: -destination-path
    svm_backup: -type XDP -schedule my_daily -policy MirrorAndVault -
   identity-preserve true
   ```

   ポリシー タイプが`async-mirror`のカスタム ポリシーを作成した場合、次の例はSnapMirror DR関係を作成します。

   ```
   cluster_dst::> snapmirror create -source-path svm1: -destination-path
    svm_backup: -type XDP -schedule my_daily -policy my_mirrored -identity-
   preserve true
   ```

ポリシー タイプがmirror-vaultのカスタム ポリシーを作成した場合、次の例はユニファイド レプリケーション関係を作成します。

```
cluster_dst::> snapmirror create -source-path svm1: -destination-path
 svm_backup: -type XDP -schedule my_daily -policy my_unified -identity-
preserve true
```

4. デスティネーションSVMを停止します。vserver stop*SVM name*
   次の例は、dvs1という名前のデスティネーションSVMを停止します。

```
cluster_dst::> vserver stop -vserver dvs1
```

5. デスティネーションSVMまたはデスティネーション クラスタから、SVMレプリケーション関係を初期化します。 snapmirror initialize -source-path *SVM_name*: -destination-path *SVM_name*:
   次の例は、ソースSVM svm1とデスティネーションSVM svm_backupの間の関係を初期化します。

```
cluster_dst::> snapmirror initialize -source-path svm1: -destination-path
 svm_backup:
```

SVMレプリケーション対象からのLIFと関連ネットワークの設定の除外

ソースSVMとデスティネーションSVMが異なるネットワーク サブネットにある場合は、snapmirror policy createコマンドの-discard-configs networkオプションを使用して、LIFと関連ネットワークの設定をSVMレプリケーション対象から除外できます。

始める前に

ソース / デスティネーション クラスタとSVMのピア関係が確立されている必要があります。

このタスクについて

SVMレプリケーション関係を作成する場合は、snapmirror createコマンドの-identity-preserveオプションをtrueに設定する必要があります。

コマンド構文全体については、マニュアル ページを参照してください。

手順

1. デスティネーションSVMを作成します。vserver create -vserver *SVM* -subtype dp-destination
   SVM名はソース クラスタとデスティネーション クラスタの間で一意である必要があります。
   次の例は、svm_backupという名前のデスティネーションSVMを作成します。

```
cluster_dst:> vserver create -vserver svm_backup -subtype dp-destination
```

2. ジョブ スケジュールを作成します。 job schedule cron create -name *job_name* -month *month* -dayofweek *day_of_week* -day *day_of_month* -hour *hour* -minute *minute*
   -month、-dayofweek、-hourにallを指定すると、ジョブを毎月、毎日、1時間ごとにそれぞれ実行できます。
   次の例は、土曜日の午前3時に実行するmy_weeklyというジョブ スケジュールを作成します。

```
cluster_dst::> job schedule cron create -name my_weekly -dayofweek
 "Saturday" -hour 3 -minute 0
```

3. カスタム レプリケーション ポリシーを作成します。 snapmirror policy create -vserver *SVM* -policy *policy* -type async-mirror|vault|mirror-vault -comment *comment* -tries *transfer_tries* -transfer-priority low|normal -is-network-compression-enabled true|false -discard-configs network
   コマンド構文全体については、マニュアル ページを参照してください。
   次の例は、LIFを除外するSnapMirror DR用のカスタム レプリケーション ポリシーを作成します。

```
cluster_dst::> snapmirror policy create -vserver svm1 -policy
 DR_exclude_LIFs -type async-mirror -discard-configs network
```

次の例は、LIFを除外するユニファイド レプリケーション用のカスタム レプリケーション ポリシーを作成します。

```
cluster_dst::> snapmirror policy create -vserver svm1 -policy
 unified_exclude_LIFs -type mirror-vault -discard-configs network
```

4. デスティネーションSVMまたはデスティネーション クラスタから次のコマンドを実行して、レプリケーション関係を作成します。`snapmirror create -source-path SVM: -destination-path SVM: -type DP|XDP -schedule schedule -policy policy -identity-preserve true|false`

   注： -source-pathオプションと-destination-pathオプションでは、SVM名のあとにコロン（:）を入力する必要があります。以下に例を紹介します。

   次の例は、LIFを除外するSnapMirror DR関係を作成します。

```
cluster_dst::> snapmirror create -source-path svm1: -destination-path
 svm_backup: -type XDP -schedule my_daily -policy DR_exclude_LIFs -
identity-preserve true
```

   次の例は、LIFを除外するSnapMirrorユニファイド レプリケーション関係を作成します。

```
cluster_dst::> snapmirror create -source-path svm1: -destination-path
 svm_backup: -type XDP -schedule my_daily -policy unified_exclude_LIFs -
identity-preserve true
```

5. デスティネーションSVMを停止します。`vserver stop`*SVM name*
   次の例は、dvs1という名前のデスティネーションSVMを停止します。

```
cluster_dst::> vserver stop -vserver dvs1
```

6. デスティネーションSVMまたはデスティネーション クラスタから、レプリケーション関係を初期化します。`snapmirror initialize -source-path SVM: -destination-path SVM:`
   コマンド構文全体については、マニュアル ページを参照してください。
   次の例は、ソースのsvm1とデスティネーションのsvm_backupの間の関係を初期化します。

```
cluster_dst::> snapmirror initialize -source-path svm1: -destination-path
 svm_backup:
```

次のタスク

災害発生時のデータ アクセス用に、デスティネーションSVMでネットワークとプロトコルを設定する必要があります。

SVMレプリケーション対象からのネットワーク、ネーム サービス、およびその他の設定の除外

`snapmirror create`コマンドの-identity-preserve falseオプションを使用すると、SVMのボリュームとセキュリティ設定だけをレプリケートできます。一部のプロトコルとネーム サービスの設定も保持されます。

始める前に

ソース / デスティネーション クラスタとSVMのピア関係が確立されている必要があります。

このタスクについて

保持されているプロトコルおよびネーム サービスの設定のリストについては、SVM DR関係でレプリケートされる設定（54ページ）を参照してください。

コマンド構文全体については、マニュアル ページを参照してください。

手順

1. デスティネーションSVMを作成します。`vserver create -vserver SVM -subtype dp-destination`

   SVM名はソース クラスタとデスティネーション クラスタの間で一意である必要があります。

   次の例は、`svm_backup`という名前のデスティネーションSVMを作成します。

   ```
   cluster_dst:> vserver create -vserver svm_backup -subtype dp-destination
   ```

2. レプリケーション ジョブ スケジュールを作成します。`job schedule cron create -name job_name -month month -dayofweek day_of_week -day day_of_month -hour hour -minute minute`

   -month、-dayofweek、-hourに`all`を指定すると、ジョブを毎月、毎日、1時間ごとにそれぞれ実行できます。

   次の例は、土曜日の午前3時に実行する`my_weekly`というジョブ スケジュールを作成します。

   ```
   cluster_dst::> job schedule cron create -name my_weekly -dayofweek
    "Saturday" -hour 3 -minute 0
   ```

3. ネットワーク、ネーム サービス、およびその他の設定を除外するレプリケーション関係を作成します。`snapmirror create -source-path SVM: -destination-path SVM: -type DP|XDP -schedule schedule -policy policy -identity-preserve false`

   注： -source-pathオプションと-destination-pathオプションでは、SVM名のあとにコロン（:）を入力する必要があります。以下に例を紹介します。

   このコマンドはデスティネーションSVMまたはデスティネーション クラスタから実行する必要があります。

   次の例は、デフォルトの`MirrorAllSnapshots`ポリシーを使用して、SnapMirror DR関係を作成します。この関係では、ネットワーク、ネーム サービス、およびその他の設定がSVMレプリケーションから除外されます。

   ```
   cluster_dst::> snapmirror create -source-path svm1: -destination-path
    svm_backup: -type XDP -schedule my_daily -policy MirrorAllSnapshots -
   identity-preserve false
   ```

   次の例は、デフォルトの`MirrorAndVault`ポリシーを使用して、ユニファイド レプリケーション関係を作成します。この関係では、ネットワーク、ネーム サービス、およびその他の設定が除外されます。

   ```
   cluster_dst:> snapmirror create svm1: -destination-path svm_backup: -type
    XDP -schedule my_daily -policy MirrorAndVault -identity-preserve false
   ```

   ポリシー タイプが`async-mirror`のカスタム ポリシーを作成した場合、次の例はSnapMirror DR関係を作成します。この関係では、ネットワーク、ネーム サービス、およびその他の設定がSVMレプリケーションから除外されます。

   ```
   cluster_dst::> snapmirror create -source-path svm1: -destination-path
    svm_backup: -type XDP -schedule my_daily -policy my_mirrored -identity-
   preserve false
   ```

   ポリシー タイプが`mirror-vault`のカスタム ポリシーを作成した場合、次の例はユニファイド レプリケーション関係を作成します。この関係では、ネットワーク、ネーム サービス、およびその他の設定がSVMレプリケーションから除外されます。

   ```
   cluster_dst::> snapmirror create -source-path svm1: -destination-path
    svm_backup: -type XDP -schedule my_daily -policy my_unified -identity-
   preserve false
   ```

4. デスティネーションSVMを停止します。`vserver stop`*SVM name*

次の例は、dvs1という名前のデスティネーションSVMを停止します。

```
destination_cluster::> vserver stop -vserver dvs1
```

5. SMBを使用する場合は、CIFSサーバも設定してください。

   CIFSのみ：CIFSサーバの作成（62ページ）を参照してください。

6. デスティネーションSVMまたはデスティネーション クラスタから、SVMレプリケーション関係を初期化します。 `snapmirror initialize -source-path SVM_name: -destination-path SVM_name:`

次のタスク

災害発生時のデータ アクセス用に、デスティネーションSVMでネットワークとプロトコルを設定する必要があります。

CIFSのみ：CIFSサーバの作成

ソースSVMでCIFS構成を使用しており、identity-preserveを`false`に設定した場合は、デスティネーションSVM用のCIFSサーバを作成する必要があります。SnapMirror関係の初期化の際は、共有などの一部のCIFS構成でCIFSサーバが必要です。

手順

1. `vserver start`コマンドを使用して、デスティネーションSVMを起動します。

```
destination_cluster::> vserver start -vserver dvs1
[Job 30] Job succeeded: DONE
```

2. `vserver show`コマンドを使用して、デスティネーションSVMの状態が`running`であること、およびサブタイプが`dp-destination`であることを確認します。

```
destination_cluster::> vserver show
                                   Admin      Operational Root
Vserver  Type    Subtype           State      State       Volume     Aggregate
-------- ------- ----------       ---------- ----------- ---------- ----------
dvs1     data    dp-destination    running    running       -          -
```

3. `network interface create`コマンドを使用して、LIFを作成します。

```
destination_cluster::>network interface create -vserver dvs1 -lif NAS1 -role data -data-protocol cifs -
home-node destination_cluster-01 -home-port a0a-101  -address 192.0.2.128 -netmask 255.255.255.128
```

4. `network route create`コマンドを使用して、ルートを作成します。

```
destination_cluster::>network route create -vserver dvs1 -destination 0.0.0.0/0
-gateway 192.0.2.1
```

   ネットワーク管理ガイド

5. `vserver services dns create`コマンドを使用して、DNSを設定します。

```
destination_cluster::>vserver services dns create -domains mydomain.example.com -vserver
dvs1 -name-servers 192.0.2.128 -state enabled
```

6. `vserver cifs domain preferred-dc add`コマンドを使用して、優先ドメイン コントローラを追加します。

```
destination_cluster::>vserver cifs domain preferred-dc add -vserver dvs1 -preferred-dc
192.0.2.128 -domain mydomain.example.com
```

7. `vserver cifs create`コマンドを使用して、CIFSサーバを作成します。

```
destination_cluster::>vserver cifs create -vserver dvs1 -domain mydomain.example.com
-cifs-server CIFS1
```

   SMB / CIFSリファレンス

8. `vserver stop`コマンドを使用して、デスティネーションSVMを停止します。

```
destination_cluster::> vserver stop -vserver dvs1
[Job 46] Job succeeded: DONE
```

SVMレプリケーション対象からのボリュームの除外

デフォルトでは、ソースSVMのすべてのRWデータ ボリュームがレプリケートされます。保護する必要がないボリュームがソースSVMに含まれている場合は、`volume modify`コマンドの-vserver-dr-

protection unprotectedオプションを使用して、SVMレプリケーション対象からボリュームを除外できます。

手順

1. SVMレプリケーション対象からボリュームを除外します。`volume modify -vserver *SVM* -volume *volume* -vserver-dr-protection unprotected`

   コマンド構文全体については、マニュアル ページを参照してください。

   次の例は、SVMレプリケーション対象からボリューム`volA_src`を除外します。

   ```
   cluster_src::> volume modify -vserver SVM1 -volume volA_src -vserver-dr-protection unprotected
   ```

   除外したボリュームをあとでSVMレプリケーション対象に含めるには、次のコマンドを実行します。`volume modify -vserver *SVM* -volume *volume* -vserver-dr-protection protected`

   次の例は、ボリューム`volA_src`をSVMレプリケーション対象に追加します。

   ```
   cluster_src::> volume modify -vserver SVM1 -volume volA_src -vserver-dr-protection protected
   ```

2. SVMの設定全体のレプリケート（58ページ）の説明に従って、SVMレプリケーション関係を作成して初期化します。

初期化後に設定されたジャンクション パスの更新

`snapmirror update`コマンドは、SVMレプリケーション関係を作成して初期化したあとに追加されたジャンクション パス情報をレプリケートしません。ジャンクション パス情報を更新するには、`snapmirror resync`を実行する必要があります。

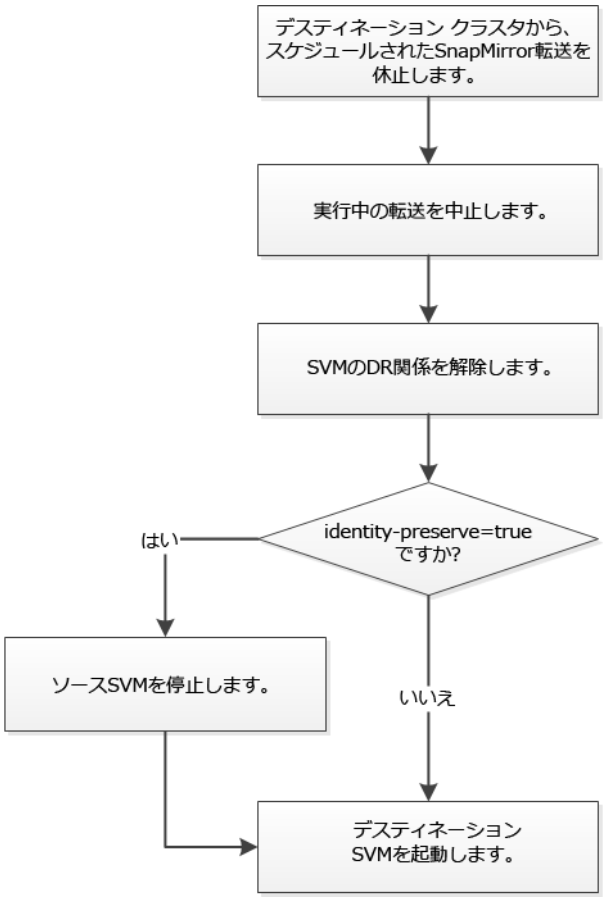
レプリケーション関係の再同期（48ページ）

SVM DRデスティネーションからのデータの提供

災害によってSVM DR関係のプライマリ サイトが機能しなくなった場合は、システム停止を最小限に抑えてデスティネーションSVMからデータを提供できます。プライマリ サイトでサービスが復旧したら、ソースSVMを再アクティブ化できます。

SVMのディザスタ リカバリのワークフロー

災害からリカバリしてデスティネーションSVMからデータを提供するには、デスティネーションSVMをアクティブ化する必要があります。デスティネーションSVMのアクティブ化では、スケジュールされたSnapMirror転送の停止、実行中のSnapMirror転送の中止、レプリケーション関係の解除、ソースSVMの停止、デスティネーションSVMの起動が実行されます。



SVMデスティネーション ボリュームを書き込み可能にする

クライアントにデータを提供する前に、SVMデスティネーション ボリュームを書き込み可能にする必要があります。この手順は、1つの例外を除き、ボリューム レプリケーションの手順とほぼ同じです。SVMレプリケーション関係の作成時に-identity-preserve trueを設定した場合は、デスティネーションSVMをアクティブ化する前にソースSVMを停止する必要があります。

このタスクについて

コマンド構文全体については、マニュアル ページを参照してください。

注： ディザスタ リカバリ シナリオでは、ソースSVMとそのデータにアクセスできないため、ソースSVMからディザスタ リカバリ デスティネーションSVMへのSnapMirror更新を実行できません。また、最後の再同期以降の更新が不正であるか破損している可能性があるためです。

手順

1. デスティネーションSVMデスティネーションクラスタから、デスティネーションへのスケジュールされた転送を停止します。 `snapmirror quiesce -source-path SVM: -destination-path SVM:`

   注： -source-pathオプションと-destination-pathオプションでは、SVM名のあとにコロン（:）を入力する必要があります。以下の例を参照してください。

   次の例は、ソースSVM `svm1`とデスティネーションSVM `svm_backup`の間のスケジュールされた転送を停止します。

   ```
   cluster_dst::> snapmirror quiesce -source-path svm1: -destination-path
    svm_backup:
   ```

2. デスティネーションSVMデスティネーションクラスタから、デスティネーションへの進行中の転送を停止します。 `snapmirror abort -source-path SVM: -destination-path SVM:`

注： -source-pathオプションと-destination-pathオプションでは、SVM名のあとにコロン（:）を入力する必要があります。以下の例を参照してください。

次の例は、ソースSVM svm1とデスティネーションSVM svm_backupの間の実行中の転送を停止します。

```
cluster_dst::> snapmirror abort -source-path svm1: -destination-path
 svm_backup:
```

3. デスティネーションSVMデスティネーションクラスタから、レプリケーション関係を解除します。
`snapmirror break -source-path SVM: -destination-path SVM:`

   注： -source-pathオプションと-destination-pathオプションでは、SVM名のあとにコロン（:）を入力する必要があります。以下の例を参照してください。

   次の例は、ソースSVM svm1とデスティネーションSVM svm_backupの間の関係を解除します。

```
cluster_dst::> snapmirror break -source-path svm1: -destination-path
 svm_backup:
```

4. SVMレプリケーション関係の作成時に-identity-preserve trueを設定した場合は、ソースSVMを停止します。`vserver stop -vserver SVM`
次の例は、ソースSVM svm1を停止します。

```
cluster_src::> vserver stop svm1
```

5. デスティネーションSVMを起動します。`vserver start -vserver SVM`
次の例は、デスティネーションSVM svm_backupを起動します。

```
cluster_dst::> vserver start svm_backup
```

次のタスク

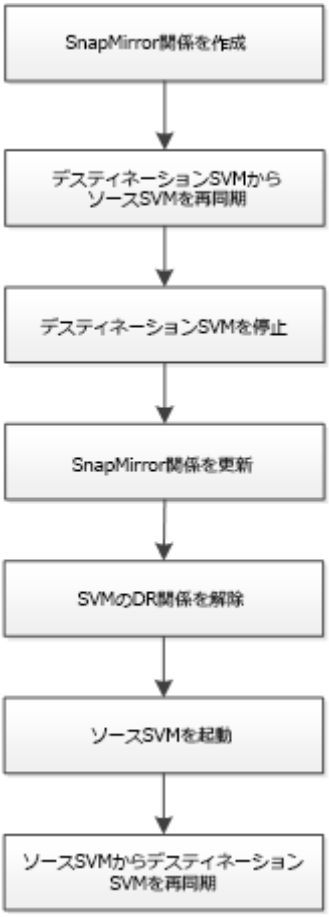
SVMデスティネーションボリュームをデータアクセス用に設定します（データ アクセスのためのデスティネーション ボリュームの設定（42ページ）を参照）。

ソースSVMの再アクティブ化

災害発生後もソースSVMが残っている場合は、プライマリ サイトでサービスが復旧したときにソースSVMを再アクティブ化できます。

ソースSVMの再アクティブ化ワークフロー

災害発生後もソースSVMが残っている場合は、そのソースSVMを再アクティブ化し、SVMディザスタ リカバリ関係を再作成して保護できます。



元のソースSVMの再アクティブ化

デスティネーションからデータを提供する必要がなくなった場合は、ソースSVMとデスティネーションSVMの間で元のデータ保護関係を再確立できます。この手順は、1つの例外を除き、ボリューム レプリケーションの手順とほぼ同じです。ソースSVMを再アクティブ化する前に、デスティネーションSVMを停止する必要があります。

始める前に

デスティネーション ボリュームからデータを提供している間にそのサイズを拡張した場合は、ソース ボリュームを再アクティブ化する前に、元のソース ボリュームを十分拡張できるようにmax-autosizeを手動で増やす必要があります。

デスティネーション ボリュームが自動的に拡張される状況（20ページ）

このタスクについて

この手順は、元のソース ボリュームにあるベースラインが損なわれていないことを前提としています。ベースラインが損なわれている場合は、この手順を実行する前に、データの提供元のボリュームと元のソース ボリュームの間の関係を作成して初期化する必要があります。

これらのコマンドのコマンド構文全体については、マニュアル ページを参照してください。

手順

1. 元のソースSVMまたはソース クラスタから、元のSVM DR関係と同じ設定、ポリシー、およびidentity-preserve設定を使用して、反転したSVM DR関係を作成します。`snapmirror create -source-path SVM: -destination-path SVM:`

   注： -source-pathオプションと-destination-pathオプションでは、SVM名のあとにコロン（:）を入力する必要があります。以下の例を参照してください。

次の例は、データの提供元であるSVM svm_backupと元のソースSVM svm1の間に関係を作成します。

```
cluster_src::> snapmirror create -source-path svm_backup: -destination-
path svm1:
```

2. 元のソースSVMまたはソース クラスタから次のコマンドを実行して、データ保護関係を反転します。snapmirror resync -source-path *SVM*: -destination-path *SVM*:

   注： -source-pathオプションと-destination-pathオプションでは、SVM名のあとにコロン（:）を入力する必要があります。以下の例を参照してください。

   再同期の際にベースライン転送は不要ですが、再同期には時間がかかる場合があります。再同期はオフピークの時間帯に実行することを推奨します。

   注： ソースとデスティネーションに共通のSnapshotコピーが存在しない場合、このコマンドは失敗します。snapmirror initializeを使用して関係を再初期化してください。

   次の例は、元のソースSVM svm1とデータの提供元であるSVM svm_backupの間の関係を反転します。

```
cluster_src::> snapmirror resync -source-path svm_backup: -destination-
path svm1:
```

3. 元のソースSVMへのデータ アクセスを再確立する準備ができたら、元のデスティネーションSVMを停止してSVMに接続されているクライアントをすべて切断します。 vserver stop -vserver *SVM*
   次の例は、現在データを提供している元のデスティネーションSVMを停止します。

```
cluster_dst::> vserver stop svm_backup
```

4. vserver showコマンドを使用して、元のデスティネーションSVMの状態がstoppedであることを確認します。

```
cluster_dst::> vserver show
                               Admin      Operational Root
Vserver        Type    Subtype    State      State       Volume     Aggregate
--------       ------- ---------- ---------- ----------- ---------- ----------
svm_backup     data    default    stopped    stopped     rv         aggr1
```

5. 元のソースSVMまたはソース クラスタから次のコマンドを実行して反転した関係を最後にもう一度更新し、元のデスティネーションSVMから元のソースSVMにすべての変更を転送します。 snapmirror update -source-path *SVM*: -destination-path *SVM*:

   注： -source-pathオプションと-destination-pathオプションでは、SVM名のあとにコロン（:）を入力する必要があります。以下の例を参照してください。

   次の例は、データの提供元である元のデスティネーションSVM svm_backupと元のソースSVM svm1の間の関係を更新します。

```
cluster_src::> snapmirror update -source-path svm_backup: -destination-
path svm1:
```

6. 元のソースSVMまたはソース クラスタから次のコマンドを実行して、反転した関係のスケジュールされた転送を停止します。 snapmirror quiesce -source-path *SVM*: -destination-path *SVM*:

   注： -source-pathオプションと-destination-pathオプションでは、SVM名のあとにコロン（:）を入力する必要があります。以下の例を参照してください。

   次の例は、データの提供元のSVM svm_backupと元のSVM svm1の間のスケジュールされた転送を停止します。

```
cluster_src::> snapmirror quiesce -source-path svm_backup: -destination-
path svm1:
```

7. 最後の更新が完了して関係のステータスが「Quiesced」になったら、元のソースSVMまたは元のソース クラスタから次のコマンドを実行して反転した関係を解除します。 snapmirror break -source-path *SVM*: -destination-path *SVM*:

注： -source-pathオプションと-destination-pathオプションでは、SVM名のあとにコロン（:）を入力する必要があります。以下の例を参照してください。

次の例は、データの提供元である元のデスティネーションSVM `svm_backup`と元のソースSVM `svm1`の間の関係を解除します。

```
cluster_src::> snapmirror break -source-path svm_backup: -destination-
path svm1:
```

8. 元のデスティネーションSVMまたはデスティネーション クラスタから、元のデータ保護関係を再確立します。`snapmirror resync -source-path SVM: -destination-path SVM:`

   注： -source-pathオプションと-destination-pathオプションでは、SVM名のあとにコロン（:）を入力する必要があります。以下の例を参照してください。

   次の例は、元のソースSVM `svm1`と元のデスティネーションSVM `svm_backup`の間の関係を再確立します。

```
cluster_dst::> snapmirror resync -source-path svm1: -destination-path
 svm_backup:
```

9. 元のソースSVMを停止していた場合は、元のソース クラスタから元のソースSVMを起動します。`vserver start -vserver SVM`
   次の例は、元のソースSVMを起動します。

```
cluster_src::> vserver start svm1
```

10. 元のソースSVMまたはソース クラスタから次のコマンドを実行して、反転したデータ保護関係を削除します。`snapmirror delete -source-path SVM: -destination-path SVM:`

    注： -source-pathオプションと-destination-pathオプションでは、SVM名のあとにコロン（:）を入力する必要があります。以下の例を参照してください。

    次の例は、元のデスティネーションSVM `svm_backup`と元のソースSVM `svm1`の間の反転した関係を削除します。

```
cluster_src::> snapmirror delete -source-path svm_backup: -destination-
path svm1:
```

11. 元のデスティネーションSVMまたはデスティネーション クラスタから、反転したデータ保護関係をリリースします。`snapmirror release -source-path SVM: -destination-path SVM:`

    注： -source-pathオプションと-destination-pathオプションでは、SVM名のあとにコロン（:）を入力する必要があります。以下の例を参照してください。

    次の例は、元のデスティネーションSVM svm_backupと元のソースSVM `svm1`の間の反転した関係をリリースします。

```
cluster_dst::> snapmirror release -source-path svm_backup: -destination-
path svm1:
```

次のタスク

`snapmirror show`コマンドを使用して、SnapMirror関係が作成されたことを確認します。コマンド構文全体については、マニュアル ページを参照してください。

元のソースSVMの再アクティブ化（FlexGroupボリュームのみ）

デスティネーションからデータを提供する必要がなくなった場合は、ソースSVMとデスティネーションSVMの間で元のデータ保護関係を再確立できます。FlexGroupボリュームを使用している場合に元のソースSVMを再アクティブ化するには、いくつか追加手順を実行する必要があります。まず、元のSVM DR関係を削除し、元の関係をリリースしてから関係を反転する必要があります。また、スケジュールされた転送を停止する前に、反転した関係をリリースし、元の関係を再作成する必要もあります。

手順

1. 元のデスティネーションSVMまたはデスティネーション クラスタから、元のSVM DR関係を削除します。`snapmirror delete -source-path SVM: -destination-path SVM:`

   注： -source-pathオプションと-destination-pathオプションでは、SVM名のあとにコロン（:）を入力する必要があります。以下の例を参照してください。

   次の例は、元のソースSVM svm1とデスティネーションSVM svm_backupの間の元の関係を削除します。

   ```
   cluster_dst::> snapmirror delete -source-path svm1: -destination-path
    svm_backup:
   ```

2. 元のソースSVMまたはソース クラスタから、Snapshotコピーは残したまま、元の関係をリリースします。`snapmirror release -source-path SVM: -destination-path SVM: -relationship-info-only true`

   注： -source-pathオプションと-destination-pathオプションでは、SVM名のあとにコロン（:）を入力する必要があります。以下の例を参照してください。

   次の例は、元のソースSVM svm1と元のデスティネーションSVM svm_backupの間の元の関係をリリースします。

   ```
   cluster_src::> snapmirror release -source-path svm1: -destination-path
    svm_backup: -relationship-info-only true
   ```

3. 元のソースSVMまたはソース クラスタから、元のSVM DR関係と同じ設定、ポリシー、およびidentity-preserve設定を使用して、反転したSVM DR関係を作成します。`snapmirror create -source-path SVM: -destination-path SVM:`

   注： -source-pathオプションと-destination-pathオプションでは、SVM名のあとにコロン（:）を入力する必要があります。以下の例を参照してください。

   次の例は、データの提供元であるSVM svm_backupと元のソースSVM svm1の間に関係を作成します。

   ```
   cluster_src::> snapmirror create -source-path svm_backup: -destination-
   path svm1:
   ```

4. 元のソースSVMまたはソース クラスタから次のコマンドを実行して、データ保護関係を反転します。`snapmirror resync -source-path *SVM*: -destination-path *SVM*:`

   注： -source-pathオプションと-destination-pathオプションでは、SVM名のあとにコロン（:）を入力する必要があります。以下の例を参照してください。

   再同期の際にベースライン転送は不要ですが、再同期には時間がかかる場合があります。再同期はオフピークの時間帯に実行することを推奨します。

   注： ソースとデスティネーションに共通のSnapshotコピーが存在しない場合、このコマンドは失敗します。`snapmirror initialize`を使用して関係を再初期化してください。

   次の例は、元のソースSVM svm1とデータの提供元であるSVM svm_backupの間の関係を反転します。

   ```
   cluster_src::> snapmirror resync -source-path svm_backup: -destination-
   path svm1:
   ```

5. 元のソースSVMへのデータ アクセスを再確立する準備ができたら、元のデスティネーションSVMを停止してSVMに接続されているクライアントをすべて切断します。 `vserver stop -vserver *SVM*`
   次の例は、現在データを提供している元のデスティネーションSVMを停止します。

   ```
   cluster_dst::> vserver stop svm_backup
   ```

6. `vserver show`コマンドを使用して、元のデスティネーションSVMの状態がstoppedであることを確認します。

```
cluster_dst::> vserver show
                                   Admin      Operational Root
Vserver        Type    Subtype    State      State       Volume     Aggregate
--------       ------- ---------- ---------- ----------- ---------- ----------
svm_backup     data    default    stopped    stopped     rv         aggr1
```

7. 元のソースSVMまたはソース クラスタから次のコマンドを実行して反転した関係を最後にもう一度更新し、元のデスティネーションSVMから元のソースSVMにすべての変更を転送します。 `snapmirror update -source-path SVM: -destination-path SVM:`

   注： -source-pathオプションと-destination-pathオプションでは、SVM名のあとにコロン（:）を入力する必要があります。以下の例を参照してください。

   次の例は、データの提供元である元のデスティネーションSVM svm_backupと元のソースSVM svm1の間の関係を更新します。

   ```
   cluster_src::> snapmirror update -source-path svm_backup: -destination-
   path svm1:
   ```

8. 元のソースSVMまたはソース クラスタから次のコマンドを実行して、反転した関係のスケジュールされた転送を停止します。 `snapmirror quiesce -source-path SVM: -destination-path SVM:`

   注： -source-pathオプションと-destination-pathオプションでは、SVM名のあとにコロン（:）を入力する必要があります。以下の例を参照してください。

   次の例は、データの提供元のSVM svm_backupと元のSVM svm1の間のスケジュールされた転送を停止します。

   ```
   cluster_src::> snapmirror quiesce -source-path svm_backup: -destination-
   path svm1:
   ```

9. 最後の更新が完了して関係のステータスが「Quiesced」になったら、元のソースSVMまたは元のソース クラスタから次のコマンドを実行して反転した関係を解除します。 `snapmirror break -source-path SVM: -destination-path SVM:`

   注： -source-pathオプションと-destination-pathオプションでは、SVM名のあとにコロン（:）を入力する必要があります。以下の例を参照してください。

   次の例は、データの提供元である元のデスティネーションSVM svm_backupと元のソースSVM svm1の間の関係を解除します。

   ```
   cluster_src::> snapmirror break -source-path svm_backup: -destination-
   path svm1:
   ```

10. 元のソースSVMまたはソース クラスタから、反転したSVM DR関係を削除します。`snapmirror delete -source-path SVM: -destination-path SVM:`

    注： -source-pathオプションと-destination-pathオプションでは、SVM名のあとにコロン（:）を入力する必要があります。以下の例を参照してください。

    次の例は、元のデスティネーションSVM svm_backupと元のソースSVM svm1の間の反転した関係を削除します。

    ```
    cluster_src::> snapmirror delete -source-path svm_backup: -destination-
    path svm1:
    ```

11. 元のデスティネーションSVMまたはデスティネーション クラスタから、Snapshotコピーは残したまま、反転した関係をリリースします。`snapmirror release -source-path SVM: -destination-path SVM: -relationship-info-only true`

    注： -source-pathオプションと-destination-pathオプションでは、SVM名のあとにコロン（:）を入力する必要があります。以下の例を参照してください。

    次の例は、元のデスティネーションSVM svm_backupと元のソースSVM svm1の間の反転した関係をリリースします。

    ```
    cluster_dst::> snapmirror release -source-path svm_backup: -destination-
    path svm1: -relationship-info-only true
    ```

12. 元のデスティネーションSVMまたはデスティネーション クラスタから、元の関係を再作成します。元のSVM DR関係と同じ設定、ポリシー、およびidentity-preserve設定を使用します。`snapmirror create -source-path SVM: -destination-path SVM:`

    注： -source-pathオプションと-destination-pathオプションでは、SVM名のあとにコロン（:）を入力する必要があります。以下の例を参照してください。

    次の例は、元のソースSVM `svm1`と元のデスティネーションSVM `svm_backup`の間に関係を作成します。

    ```
    cluster_dst::> snapmirror create -source-path svm1: -destination-path
     svm_backup:
    ```

13. 元のデスティネーションSVMまたはデスティネーション クラスタから、元のデータ保護関係を再確立します。`snapmirror resync -source-path SVM: -destination-path SVM:`

    注： -source-pathオプションと-destination-pathオプションでは、SVM名のあとにコロン（:）を入力する必要があります。以下の例を参照してください。

    次の例は、元のソースSVM `svm1`と元のデスティネーションSVM `svm_backup`の間の関係を再確立します。

    ```
    cluster_dst::> snapmirror resync -source-path svm1: -destination-path
     svm_backup:
    ```

14. 元のソースSVMを停止していた場合は、元のソース クラスタから元のソースSVMを起動します。`vserver start -vserver SVM`
    次の例は、元のソースSVMを起動します。

    ```
    cluster_src::> vserver start svm1
    ```

ボリューム レプリケーション関係からSVMレプリケーション関係への変換

ソース上の各ボリューム（ルート ボリュームを除く）がレプリケート対象であり、ソース上の各ボリューム（ルート ボリュームを含む）の名前がデスティネーション上のボリュームと同じである場合は、ボリューム間のレプリケーション関係を、そのボリュームを所有するStorage Virtual Machine（SVM）間のレプリケーション関係に変換できます。

このタスクについて

必要に応じて、`volume rename`コマンドを使用してデスティネーション ボリュームの名前を変更します。

手順

1. デスティネーションSVMまたはデスティネーション クラスタから次のコマンドを実行して、ソースとデスティネーションのボリュームを再同期します。`snapmirror resync -source-path SVM:volume -destination-path SVM:volume -type DP|XDP -schedule schedule -policy policy`

   コマンド構文全体については、マニュアル ページを参照してください。

   注： 再同期の際にベースライン転送は不要ですが、再同期には時間がかかる場合があります。再同期はオフピークの時間帯に実行することを推奨します。

   次の例は、`svm1`上のソース ボリューム`volA`と`svm_backup`上のデスティネーション ボリューム`volA`の間の関係を再同期します。

   ```
   cluster_dst::> snapmirror resync -source-path svm1:volA -destination-path
    svm_backup:volA
   ```

2. SVMの設定のレプリケート（57ページ）の説明に従って、ソースとデスティネーションのSVM間にSVMレプリケーション関係を作成します。

レプリケーション関係を作成するときは、`snapmirror create`コマンドで-identity-preserve trueオプションを使用する必要があります。

3. デスティネーションSVMを停止します。`vserver stop -vserver *SVM*`

   コマンド構文全体については、マニュアル ページを参照してください。

   次の例は、デスティネーションSVM`svm_backup`を停止します。

```
cluster_dst::> vserver stop svm_backup
```

4. デスティネーションSVMまたはデスティネーション クラスタから次のコマンドを実行して、ソースとデスティネーションのSVMを再同期します。`snapmirror resync -source-path *SVM*: -destination-path *SVM*: -type DP|XDP -schedule *schedule* -policy *policy*`

   コマンド構文全体については、マニュアル ページを参照してください。

   注： -source-pathオプションと-destination-pathオプションでは、SVM名のあとにコロン（:）を入力する必要があります。以下の例を参照してください。

   再同期の際にベースライン転送は不要ですが、再同期には時間がかかる場合があります。再同期はオフピークの時間帯に実行することを推奨します。

   次の例は、ソースSVM`svm1`とデスティネーションSVM`svm_backup`の間の関係を再同期します。

```
cluster_dst::> snapmirror resync -source-path svm1: -destination-path
 svm_backup:
```

SVMレプリケーション関係の削除

`snapmirror delete`コマンドと`snapmirror release`コマンドを使用すると、SVMレプリケーション関係を削除できます。続いて、不要なデスティネーション ボリュームを手動で削除することができます。

このタスクについて

`snapmirror release`コマンドは、SnapMirrorによって作成されたSnapshotコピーをソースから削除します。-relationship-info-onlyオプションを使用するとSnapshotコピーを保持できます。

これらのコマンドのコマンド構文全体については、マニュアル ページを参照してください。

手順

1. デスティネーションSVMまたはデスティネーション クラスタから次のコマンドを実行して、レプリケーション関係を解除します。`snapmirror break -source-path *SVM*: -destination-path *SVM*:`

   注： -source-pathオプションと-destination-pathオプションでは、SVM名のあとにコロン（:）を入力する必要があります。以下の例を参照してください。

   次の例は、ソースSVM`svm1`とデスティネーションSVM`svm_backup`の間の関係を解除します。

```
cluster_dst::> snapmirror break -source-path svm1: -destination-path
 svm_backup:
```

2. デスティネーションSVMまたはデスティネーション クラスタから次のコマンドを実行して、レプリケーション関係を削除します。`snapmirror delete -source-path *SVM*: -destination-path *SVM*:`

   注： -source-pathオプションと-destination-pathオプションでは、SVM名のあとにコロン（:）を入力する必要があります。以下の例を参照してください。

   次の例は、ソースSVM`svm1`とデスティネーションSVM`svm_backup`の間の関係を削除します。

```
cluster_dst::> snapmirror delete -source-path svm1: -destination-path
 svm_backup:
```

3. ソース クラスタまたはソースSVMから次のコマンドを実行して、ソースSVMからレプリケーション関係情報をリリースします。`snapmirror release -source-path SVM: -destination-path SVM:`

   注： -source-pathオプションと-destination-pathオプションでは、SVM名のあとにコロン（:）を入力する必要があります。以下の例を参照してください。

   次の例は、ソースSVM`svm1`から指定されたレプリケーション関係の情報をリリースします。

```
cluster_src::> snapmirror release -source-path svm1: -destination-path
 svm_backup:
```

SnapMirrorルート ボリューム レプリケーションの管理

NAS環境内のすべてのSVMには一意のネームスペースがあります。オペレーティング システムと関連情報を含むSVMルート ボリュームは、ネームスペース階層のエントリ ポイントです。ノードに障害やフェイルオーバーが発生したときに引き続きクライアントからデータにアクセスできるようにするには、SVMルート ボリュームの負荷共有ミラー コピーを作成する必要があります。

SVM ルートボリュームの負荷共有ミラーの主な目的は、負荷共有ではなく、ディザスタリカバリを目的としたものです。

- ルート ボリュームが一時的に使用できなくなった場合は、負荷共有ミラーによって、ルート ボリュームのデータへの読み取り専用アクセスが自動的に提供されます。
- ルート ボリュームが完全に使用できなくなった場合は、いずれかの負荷共有ボリュームを昇格して、ルート ボリュームのデータへの書き込みアクセスを提供できます。

負荷共有ミラー関係の作成および初期化

クラスタ内のそれぞれのルート ボリュームについて、負荷共有ミラー（LSM）を作成する必要があります。LSMは、ルート ボリュームを含まない任意のノード（HAペアのパートナー ノードなど）または別のHAペアに作成できます。2ノード クラスタの場合は、ルート ボリュームがあるノードのパートナーにLSMを作成する必要があります。

このタスクについて

同じノードにLSMを作成した場合、HAペアのディスクで障害が発生すると単一点障害となり、データのリカバリ元となる2つ目のコピーは存在しません。一方、ルート ボリュームを含まないノードや別のHAペアにLSMを作成すると、障害が発生してもデータに引き続きアクセスできます。

たとえば、4ノード クラスタの3つのノードにルート ボリュームがある場合、次のようになります。

- HA 1のノード1のルート ボリュームについて、HA 2のノード1またはHA 2のノード2にLSMを作成します。
- HA 1のノード2のルート ボリュームについて、HA 2のノード1またはHA 2のノード2にLSMを作成します。
- HA 2のノード1のルート ボリュームについて、HA 1のノード1またはHA 1のノード2にLSMを作成します。

手順

1. LSMのデスティネーション ボリュームを作成します。`volume create -vserver SVM -volume volume -aggregate aggregate -type DP -size size`

   デスティネーション ボリュームのサイズは、ルート ボリュームと同じかそれ以上である必要があります。

   ルート ボリュームとデスティネーション ボリュームの名前にサフィックス（`_root`、`_m1`など）を追加することを推奨します。

   コマンド構文全体については、マニュアル ページを参照してください。

   次の例は、`cluster_src`にルート ボリューム`svm1_root`の負荷共有ミラー ボリュームを作成します。

   ```
   cluster_src:> volume create -vserver svm1 -volume svm1_m1 -aggregate
    aggr_1 -size 1gb -state online -type DP
   ```

2. レプリケーション ジョブ スケジュールの作成（29ページ）の説明に従って、レプリケーション ジョブ スケジュールを作成します。

3. SVMルート ボリュームとLSMのデスティネーション ボリュームの間に負荷共有ミラー関係を作成します。 `snapmirror create -source-path SVM:volume|cluster://SVM/volume -destination-path SVM:volume|cluster://SVM/volume -type LS -schedule schedule`

   コマンド構文全体については、マニュアル ページを参照してください。

   次の例は、ルート ボリュームsvm1_rootと負荷共有ミラー ボリュームsvm1_m1の間に負荷共有ミラー関係を作成します。

   ```
   cluster_src::> snapmirror create -source-path svm1:svm1_root -
   destination-path svm1:svm1_m1 -type LS -schedule hourly
   ```

   負荷共有ミラー ボリュームのtype属性がDPからLSに変わります。
4. 負荷共有ミラーを初期化します。`snapmirror initialize-ls-set -source-path SVM:volume|cluster://SVM/volume`

   初期化には時間がかかる場合があります。ベースライン転送はオフピークの時間帯に実行することを推奨します。

   コマンド構文全体については、マニュアル ページを参照してください。

   次の例は、ルート ボリュームsvm1_rootの負荷共有ミラーを初期化します。

   ```
   cluster_src::> snapmirror initialize-ls-set -source-path svm1:svm1_root
   ```

負荷共有ミラー関係の更新

ルート ボリュームに加えた変更を、スケジュールされた次回の更新前に参照できるようにする場合は、負荷共有ミラー（LSM）関係を手動で更新する必要があります。たとえば、SVMルート ボリュームに新しいボリュームをマウントした場合はLSM関係を更新する必要があります。

手順

負荷共有ミラー関係を手動で更新します。`snapmirror update-ls-set -source-path SVM:volume|cluster://SVM/volume`
次の例は、ルート ボリュームsvm1_rootの負荷共有ミラー関係を更新します。

```
cluster_src::> snapmirror update-ls-set -source-path svm1:svm1_root
```

負荷共有ミラーの昇格

ルート ボリュームが完全に使用できなくなった場合は、負荷共有ミラー（LSM）ボリュームを昇格して、ルート ボリュームのデータへの書き込みアクセスを提供できます。

始める前に

このタスクを実行するには、advanced権限レベルのコマンドを使用する必要があります。

手順

1. advanced権限レベルに切り替えます。`set -privilege advanced`
2. LSMボリュームを昇格します。`snapmirror promote -destination-path SVM:volume|cluster://SVM/volume`

   コマンド構文全体については、マニュアル ページを参照してください。

   次の例は、ボリュームsvm1_m2を新しいSVMルート ボリュームとして昇格します。

   ```
   cluster_src::*> snapmirror promote -destination-path svm1:svm1_m2

   Warning: Promote will delete the offline read-write volume
   ```

```
          cluster_src://svm1/svm1_root and replace it with
          cluster_src://svm1/svm1_m2. Because the volume is offline,
          it is not possible to determine whether this promote will
          affect other relationships associated with this source.
Do you want to continue? {y|n}: y
```

「y」と入力します。LSMボリュームが読み取り / 書き込みボリュームとなり、元のルート ボリューム（アクセス可能である場合）が削除されます。

重要： 最後の更新が最近行われていない場合は、昇格されたルート ボリュームに元のルート ボリュームのデータがすべて含まれているとは限りません。

3. admin権限レベルに戻ります。`set -privilege admin`
4. ルート ボリュームに使用した命名規則に従って、昇格されたボリュームの名前を変更します。`volume rename -vserver` *`SVM`* `-volume` *`volume`* `-newname` *`new_name`*
   次の例は、昇格されたボリューム`svm1_m2`の名前を`svm1_root`に変更します。

   ```
   cluster_src::> volume rename -vserver svm11 -volume svm1_m2 -newname
    svm1_root
   ```

5. 負荷共有ミラー関係の作成および初期化（74ページ）の手順3（75ページ）～手順4（75ページ）に従って、名前を変更したルート ボリュームを保護します。

SnapMirrorの技術的な詳細

SnapMirrorデータ保護関係を作成する前に理解しておく必要のある便利な機能がいくつかあります。また、SnapMirror関係に対応したONTAPのバージョンとSnapMirrorの基本的な制限事項を把握しておく必要があります。

パス名のパターン マッチングの使用

パターン マッチングを使用すると、`snapmirror`コマンドでソース パスとデスティネーション パスを指定できます。

`snapmirror`コマンドでは、*`vserver`*`:`*`volume`*という形式の完全修飾パス名を使用します。SVM名を入力せずにパス名を省略できます。省略した場合、`snapmirror`コマンドではユーザのローカルのSVMコンテキストが使用されます。

SVMの名前が「vserver1」、ボリュームの名前が「vol1」とすると、完全修飾パス名は`vserver1:vol1`になります。

パス名にアスタリスク（*）をワイルドカードとして使用すると、一致する完全修飾パス名を選択できます。次の表に、ワイルドカードを使用して特定範囲のボリュームを選択する例を示します。

`*`	すべてのパスに一致します。
`vs*`	SVM名が`vs`で始まるすべてのSVMとボリュームが一致します。
`*:*src*`	ボリューム名に`src`文字列が含まれているすべてのSVMが一致します。
`*:vol*`	ボリューム名が`vol`で始まるすべてのSVMが一致します。

```
vs1::> snapmirror show -destination-path *:*dest*
                                                                        Progress
Source              Destination  Mirror        Relationship  Total      Last
Path          Type  Path         State         Status        Progress   Healthy Updated
------------- ---- ------------ ------------- -------------- ---------- ------- --------
vs1:sm_src2
              DP   vs2:sm_dest1
                                Snapmirrored  Idle           -          true    -
```

拡張クエリを使用した多数のSnapMirror関係の操作

拡張クエリを使用すると、複数のSnapMirror関係に対してSnapMirror処理を一度に実行できます。たとえば、初期化されていないSnapMirror関係が複数ある場合に、それらの関係を1つのコマンドで初期化できます。

このタスクについて

拡張クエリは、次のSnapMirror処理に適用できます。

- 初期化されていない関係の初期化
- 休止された関係の再開
- 解除した関係の再同期
- アイドル状態の関係の更新
- 関係のデータ転送の中止

手順

多数の関係に対してSnapMirror処理を実行します。 *`snapmirror command`* `{-state`*`state`*` } *`

次のコマンドを実行すると、Uninitialized状態のSnapMirror関係が初期化されます。

```
vs1::> snapmirror initialize {-state Uninitialized} *
```

ミラー-ヴォールト構成における共通のSnapshotコピーの保持

snapmirror snapshot-owner createコマンドを使用すると、ミラー-ヴォールト構成のセカンダリにラベル付きのSnapshotコピーを保持することができます。これにより、ヴォールト関係の更新用の共通のSnapshotコピーが確保されます。

このタスクについて

ミラー-ヴォールト ファンアウト構成またはカスケード構成を組み合わせて使用する場合は、ソース ボリュームとデスティネーション ボリュームに共通のSnapshotコピーが存在しなければ更新が失敗するという点を考慮する必要があります。

ミラー-ヴォールト ファンアウト構成またはカスケード構成におけるミラー関係では、このことは問題になりません。これは、更新の実行前にソース ボリュームのSnapshotコピーがSnapMirrorによって必ず作成されるためです。

ただし、ヴォールト関係の更新時にはソース ボリュームのSnapshotコピーがSnapMirrorによって作成されないため、ヴォールト関係ではこれが問題になる場合があります。ヴォールト関係のソースとデスティネーションの両方に共通のSnapshotコピーを少なくとも1つ確保するには、snapmirror snapshot-owner createを使用する必要があります。

手順

1. ソース ボリュームで、保持するラベル付きのSnapshotコピーに所有者を割り当てます。
   snapmirror snapshot-owner create -vserver *SVM* -volume *volume* -snapshot *snapshot* -owner *owner*
   次の例は、ApplicationAをSnapshotコピーsnap1の所有者として割り当てます。

   ```
   clust1::> snapmirror snapshot-owner create -vserver vs1 -volume vol1
   -snapshot snap1 -owner ApplicationA
   ```

2. 手動によるレプリケーション関係の更新（48ページ）の説明に従って、ミラー関係を更新します。
   または、ミラー関係のスケジュールされた更新が行われるまで待つこともできます。
3. ラベル付きのSnapshotコピーをヴォールト デスティネーションに転送します。snapmirror update -source-path *SVM*:*volume*|*cluster*://*SVM*/*volume*, ... -destination-path *SVM*:*volume*|*cluster*://*SVM*/*volume*, ... -source-snapshot *snapshot*
   コマンド構文全体については、マニュアル ページを参照してください。
   次の例は、Snapshotコピーsnap1を転送します。

   ```
   clust1::> snapmirror update -vserver vs1 -volume vol1
   -source-snapshot snap1
   ```

   ヴォールト関係の更新時にラベル付きのSnapshotコピーが保持されます。
4. ソース ボリュームで、ラベル付きのSnapshotコピーから所有者を削除します。snapmirror snapshot-owner delete -vserver *SVM* -volume *volume* -snapshot *snapshot* -owner *owner*
   次の例は、Snapshotコピーsnap1の所有者であるApplicationAを削除します。

   ```
   clust1::> snapmirror snapshot-owner delete -vserver vs1 -volume vol1
   -snapshot snap1 -owner ApplicationA
   ```

SnapMirror関係に対応したONTAPバージョン

SnapMirrorデータ保護関係を作成する前に、ソース ボリュームとデスティネーション ボリュームで互換性のあるONTAPバージョンが実行されていることを確認する必要があります。

注 : バージョンに依存しないレプリケーションは、SVMレプリケーションではサポートされません。

SnapMirror DR関係

「DP」タイプおよび「async-mirror」ポリシー タイプのSnapMirror関係の互換性は次のとおりです。

注 : 次の表で、左側の列がソース ボリュームのONTAPバージョン、上段がデスティネーション ボリュームで使用可能なONTAPバージョンです。

表 13 :

ソース	デスティネーション		
	9.7	9.8	9.9.1
9.7	○	○	○
9.8	×	○	○
9.9.1	×	×	○

注 : 双方向の互換性はありません。

ユニファイド レプリケーション関係

「XDP」タイプのSnapMirror関係（オンプレミスのONTAPまたはCloud Volumes ONTAPのリリースを使用）の互換性は次のとおりです。

注 : アスタリスク（*）が付いているバージョンは、Cloud Volumes ONTAPのみのリリースです。

表 14 :

ONTAPバージョン	以前のONTAPバージョンとの互換性			
	9.7	9.8	9.9.0*	9.9.1
9.7	○	-	-	-
9.8	○	○	-	-
9.9.0*	○	○	○	-
9.9.1	○	○	○	○

注 : 左側の列から上位（最新）のONTAPバージョンを、右側の列から下位のONTAPバージョンを選択し、互換性を確認してください。双方向の互換性があります。

SnapMirrorの制限事項

データ保護関係を作成する前に、SnapMirrorの基本的な制限事項を把握しておく必要があります。

- 1個のデスティネーション ボリュームにはソース ボリュームを1個だけ設定できます。

  注 : 1個のソース ボリュームには複数のデスティネーション ボリュームを設定できます。デスティネーション ボリュームを任意のタイプのSnapMirrorレプリケーション関係のソース ボリュームにすることができます。
- 1個のソース ボリュームから最大8個のデスティネーション ボリュームをファンアウトできます。

- SnapMirror DR関係のデスティネーションにファイルをリストアすることはできません。
- ソースまたはデスティネーションのSnapVaultボリュームを32ビット ボリュームにすることはできません。
- SnapVault関係のソース ボリュームをFlexCloneボリュームにはしないでください。

  注： 関係は機能しますが、FlexCloneボリュームによる効率化の効果が得られなくなります。

詳細情報の入手方法

SnapshotコピーとSnapMirrorレプリケーションに関連するタスクの詳細を富士通のドキュメント ライブラリで確認できます。

- コンセプト ガイド

  ONTAPデータ管理ソフトウェアの概念（データ保護や転送など）について説明します。
- クラスタ / SVMピアリング エクスプレス ガイド

  ソース クラスタとデスティネーション クラスタ間およびソースSVMとデスティネーションSVM間のピア関係を作成する方法について説明します。
- ONTAP System Managerを使用したクラスタ管理

  ONTAP System Managerを使用してSnapMirrorレプリケーションを実行する方法について説明します。
- SnapLock®テクノロジを使用したアーカイブとコンプライアンス パワー ガイド

  SnapLockボリューム内のWORMファイルをレプリケートする方法について説明します。
- Fujitsu ElementソフトウェアとETERNUS AX/HXシリーズのレプリケーション

  ElementとONTAPの間でデータをレプリケートする方法について説明します。ElementのSnapshotコピーをETERNUS AX/HXシリーズにバックアップする方法、ONTAPのLUNをElementシステムに移行する方法などを説明しています。
- NDMP構成エクスプレス ガイド

  NDMPを使用して他社製バックアップ アプリケーションでデータをテープに直接バックアップする方法について説明します。
- データ保護：テープ バックアップおよび リカバリ ガイド

  テープ バックアップおよびリカバリ機能を使用してデータをバックアップおよびリカバリする方法について説明します。
- SANアドミニストレーション ガイド

  NVMe、iSCSI、FCoE、およびFCの各プロトコルを設定および管理する方法（LUN、igroup、ターゲットの設定など）について説明します。

著作権に関する情報

Copyright 2021 FUJITSU LIMITED. All rights reserved.

このドキュメントは著作権によって保護されています。著作権所有者の書面による事前承諾がある場合を除き、画像媒体、電子媒体、および写真複写、記録媒体、テープ媒体、電子検索システムへの組み込みを含む機械媒体など、いかなる形式および方法による複製も禁止します。

富士通の著作物から派生したソフトウェアは、次に示す使用許諾条項および免責条項の対象となります。

このソフトウェアは、富士通によって「現状のまま」提供されています。富士通は明示的な保証、または商品性および特定目的に対する適合性の暗示的保証を含み、かつこれに限定されないいかなる暗示的な保証も行いません。富士通は、代替品または代替サービスの調達、使用不能、データ損失、利益損失、業務中断を含み、かつこれに限定されない、このソフトウェアの使用により生じたすべての直接的損害、間接的損害、偶発的損害、特別損害、懲罰的損害、必然的損害の発生に対して、損失の発生の可能性が通知されていたとしても、その発生理由、根拠とする責任論、契約の有無、厳格責任、不法行為（過失またはそうでない場合を含む）にかかわらず、一切の責任を負いません。

富士通は、ここに記載されているすべての製品に対する変更を随時、予告なく行う権利を保有します。富士通による明示的な書面による合意がある場合を除き、ここに記載されている製品の使用により生じる責任および義務に対して、富士通は責任を負いません。この製品の使用または購入は、富士通の特許権、商標権、または他の知的所有権に基づくライセンスの供与とはみなされません。

登録商標

富士通、富士通ロゴ、ETERNUSは富士通の登録商標です。会社名、製品名等の固有名詞は、各社の商号、商標または登録商標です。

https://www.fujitsu.com/jp/products/computing/storage/trademark/

マニュアルの更新について

本書の最新版や本装置に関連する最新の情報は、以下のサイトで公開されています。

https://www.fujitsu.com/jp/products/computing/storage/manual/

必要に応じてご使用モデルのマニュアルを参照してください。

FUJITSU Storage ETERNUS AX/HX Series

データ保護パワー ガイド

A3CA08733-A616-03

発行日: 2021 年 6 月

発行責任: 富士通株式会社

• 本書の内容は、改善のため事前連絡なしに変更することがあります。
• 本書の内容は、細心の注意を払って制作致しましたが、本書中の誤字、情報の抜け、本書情報の使用に起因する運用結果に関しましては、責任を負いかねますので予めご了承願います。
• 本書に記載されたデータの使用に起因する第三者の特許権およびその他の権利の侵害については、当社はその責を負いません。
• 無断転載を禁じます。