ASA R2システムでのストレージ構成

Enterprise applications

NetApp
February 10, 2026

目次

ASA R2システムでのストレージ構成

概要

NetApp ASA R2は、ミッションクリティカルなワークロードを実行するSANのみのお客様向けの、シンプルで強力なソリューションです。ONTAPストレージソリューションを実行するASA R2プラットフォームとMicrosoft SQL Serverを組み合わせることで、今日の最も要求の厳しいアプリケーション要件を満たすエンタープライズレベルのデータベースストレージ設計が可能になります。

次のASAプラットフォームは、すべてのSANプロトコル（iSCSI、FC、NVMe/FC、NVMe/TCP）をサポートするASA R2システムに分類されます。iSCSI、FC、NVMe/FC、NVMe/TCPの各プロトコルでは、対称アクティブ/アクティブアーキテクチャのマルチパスがサポートされているため、ホストとストレージの間のすべてのパスがアクティブ/最適化されます。

- ASA A1K
- ASA A90
- ASA A70
- ASA A50
- ASA A30
- ASA A20

詳細については、を参照してください。 "NetApp ASA"

SQL Server on ONTAPソリューションを最適化するには、SQL ServerのI/Oパターンと特性を理解する必要があります。SQL Serverデータベース用のストレージレイアウトを適切に設計するには、SQL Serverのパフォーマンス要件を満たしながら、インフラ全体の管理性を最大限に高める必要があります。また、ストレージレイアウトを適切に配置すれば、初期導入を成功させ、ビジネスの成長に合わせて環境をスムーズに拡張できます。

データストレージ設計

Microsoftでは、データファイルとログファイルを別 々 のドライブに配置することを推奨しています。データを同時に更新して要求するアプリケーションでは、ログファイルに書き込み負荷がかかり、（アプリケーションによっては）データファイルの読み取り/書き込み負荷が高くなります。データを取得する場合、ログファイルは必要ありません。そのため、データの要求は、そのドライブに配置されたデータファイルから満たすことができます。

新しいデータベースを作成するときは、データとログ用に別 々 のドライブを指定することを推奨します。データベース作成後にファイルを移動するには、データベースをオフラインにする必要があります。Microsoftのその他の推奨事項については、 "データファイルとログファイルを別 々 のドライブに配置"。

ストレージユニットに関する考慮事項

ASAのストレージユニットとは、SCSI / FCホストの場合はLUN、NVMeホストの場合はNVMeネームスペースを指します。サポートされるプロトコルに基づいて、LUN、NVMeネームスペース、またはその両方を作成するように求められます。データベース導入用のストレージユニットを作成する前に、SQL ServerのI/Oパターンと特性がワークロードやバックアップとリカバリの要件に応じてどのように変わるかを理解しておくことが

重要です。ストレージユニットに関する次のNetAppの推奨事項を参照してください。

- 管理の複雑さを避けるため、同じホスト上で実行される複数のSQL Server間で同じストレージユニットを共有しないでください。同じホストで複数のSQL Serverインスタンスを実行する場合は、ノードのストレージユニット数の上限に近い場合を除き、共有は避け、データ管理を容易にするために、ホストごとにインスタンスごとにストレージユニットを個別に配置してください。
- ドライブレターではなくNTFSマウントポイントを使用して、Windowsのドライブレターの制限（26文字）を超えます。
- Snapshotスケジュールと保持ポリシーを無効にします。代わりに、SnapCenterを使用して、SQL Serverデータストレージユニットのスナップショットコピーを調整します。
- SQL Serverシステムデータベースは専用のストレージユニットに配置します。
- tempdbは、特にI/O負荷の高いDBCC CHECKDB処理のために、SQL Serverが一時的なワークスペースとして使用するシステムデータベースです。したがって、このデータベースは専用のストレージユニットに配置します。ストレージユニットの数が課題となる大規模な環境では、慎重に計画したあとに、tempdbとシステムデータベースを同じストレージユニットに統合できます。tempdbのデータ保護は、SQL Serverを再起動するたびにこのデータベースが再作成されるため、優先度の高いものではありません。
- (`.mdf`)ランダムな読み取り/書き込みワークロードであるため、ユーザデータファイルを別のストレージユニットに配置します。トランザクションログバックアップは、データベースバックアップよりも頻繁に作成するのが一般的です。このため、トランザクションログファイル(`.ldf`)をデータファイルとは別のストレージユニットまたはVMDKに配置し、それぞれに個別のバックアップスケジュールを作成できるようにします。また、この分離により、ログファイルのシーケンシャルライトI/Oとデータファイルのランダムリード/ライトI/Oが分離され、SQL Serverのパフォーマンスが大幅に向上します。
- ユーザデータベースファイルとログバックアップを格納するログディレクトリは、バックアップポリシーのSnapMirror機能でSnapshotが保持ポリシーによって上書きされないように、別々のストレージユニットに配置してください。
- データベースファイルとデータベース以外のファイル（フルテキスト検索関連ファイルなど）を同じストレージユニットに混在させないでください。
- データベースのセカンダリファイルを（ファイルグループの一部として）別のストレージユニットに配置すると、SQL Serverデータベースのパフォーマンスが向上します。この分離は、データベースのファイルがそのストレージユニットを他のファイルと共有していない`.mdf`場合にのみ有効`.mdf`です。
- Windowsサーバのディスクマネージャを使用してディスクをフォーマットするときは、パーティションの割り当て単位サイズが64Kに設定されていることを確認してください。
- マウントポイントをホストするストレージユニットにユーザデータベースまたはシステムデータベースを配置しないでください。
- を参照してください "最新SANに対するONTAPのベストプラクティスに基づくMicrosoft Windowsとネイティブ MPIO" WindowsのマルチパスサポートをMPIOプロパティのiSCSIデバイスに適用するには、次の手順を実行します。
- Always Onフェイルオーバークラスタインスタンスを使用している場合は、Windowsサーバフェイルオーバークラスタノード間で共有されるストレージユニットにユーザデータベースを配置する必要があります。また、物理ディスククラスタリソースは、SQL Serverインスタンスに関連付けられたクラスタグループに割り当てられます。

データベースファイルとファイルグループ

初期導入段階では、SQL Serverデータベースファイルを ONTAP に適切に配置することが重要です。これにより、パフォーマンス、スペース管理、バックアップとリストアの最

適な時間が確保され、ビジネス要件に合わせて設定できます。

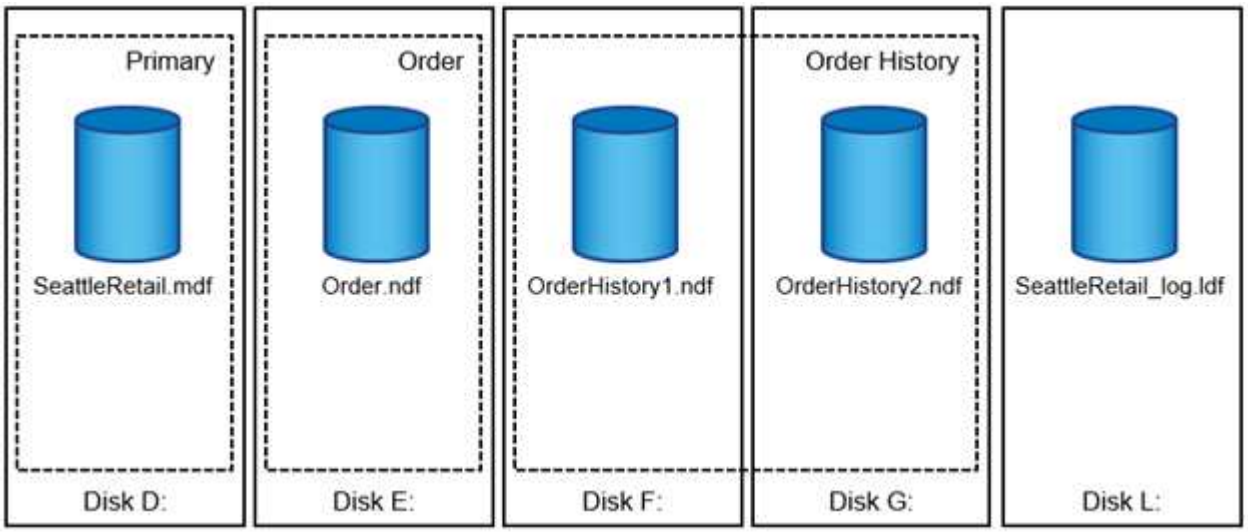
理論的には、SQL Server（64ビット）ではインスタンスあたり32、767個のデータベースと524、272TBのデータベースサイズがサポートされますが、通常のインストールでは複数のデータベースが使用されます。ただし、SQL Serverで処理できるデータベースの数は、負荷とハードウェアによって異なります。SQL Serverインスタンスでは、数十、数百、場合によっては数千の小規模データベースをホストしていることも珍しくありません。

データベースファイルとファイルグループ

各データベースは、1つ以上のデータファイルと1つ以上のトランザクションログファイルで構成されます。トランザクションログには、データベーストランザクションに関する情報と、各セッションで行われたすべてのデータ変更が格納されます。データが変更されるたびに、SQL Serverはトランザクションログに十分な情報を格納して、アクションを元に戻す（ロールバックする）か、やり直す（再生する）かを指定します。SQL Serverトランザクションログは、データの整合性と堅牢性に関するSQL Serverの評価に不可欠な要素です。トランザクションログは、SQL Serverの不可分性、整合性、分離、耐久性（ACID）機能に不可欠です。SQL Serverは、データページが変更されるとすぐにトランザクションログに書き込みます。すべてのData Manipulation Language（DML）ステートメント（SELECT、INSERT、UPDATE、DELETEなど）は完全なトランザクションであり、トランザクションログによってセットベースの操作全体が確実に実行され、トランザクションの不可分性が保証されます。

各データベースには1つのプライマリデータファイルがあり、デフォルトでは.mdf拡張子が付いています。また、各データベースにセカンダリデータベースファイルを含めることもできます。これらのファイルには、デフォルトで.ndf拡張子が付いています。

すべてのデータベースファイルはファイルグループにグループ化されます。ファイルグループは論理ユニットであり、データベース管理を簡素化します。論理オブジェクトの配置と物理データベースファイルを分離できます。データベースオブジェクトテーブルを作成するときは、基になるデータファイルの設定を気にすることなく、配置するファイルグループを指定します。



ファイルグループ内に複数のデータファイルを配置できるため、複数のストレージデバイスに負荷を分散して、システムのI/Oパフォーマンスを向上させることができます。一方、SQL Serverはトランザクションログにシーケンシャルに書き込むため、トランザクションログには複数のファイルを使用するメリットはありません。

ファイルグループ内の論理オブジェクトの配置と物理データベースファイルの配置を分離することで、データベースファイルのレイアウトを微調整し、ストレージサブシステムを最大限に活用できます。与えられたワー

クロードをサポートするデータファイルの数は、アプリケーションに影響を与えることなく、I/O要件と想定容量をサポートするために必要に応じて変えることができます。データベースレイアウトのバリエーションは、データベースファイルではなくファイルグループにデータベースオブジェクトを配置するアプリケーション開発者には透過的です。

* NetAppでは、システムオブジェクト以外にプライマリファイルグループを使用しないことを推奨しています。ユーザオブジェクト用に別のファイルグループまたはファイルグループのセットを作成すると、特に大規模なデータベースの場合、データベースの管理とディザスタリカバリが容易になります。

データベースインスタンスファイルの初期化

データベースを作成するとき、または既存のデータベースに新しいファイルを追加するときに、初期ファイルサイズと自動拡張パラメータを指定できます。SQL Serverでは、Proportional Fill Algorithmを使用して、データを書き込むデータファイルを選択します。ファイルで使用可能な空きスペースに比例してデータ量が書き込まれます。ファイル内の空きスペースが多いほど、処理する書き込み数も多くなります。

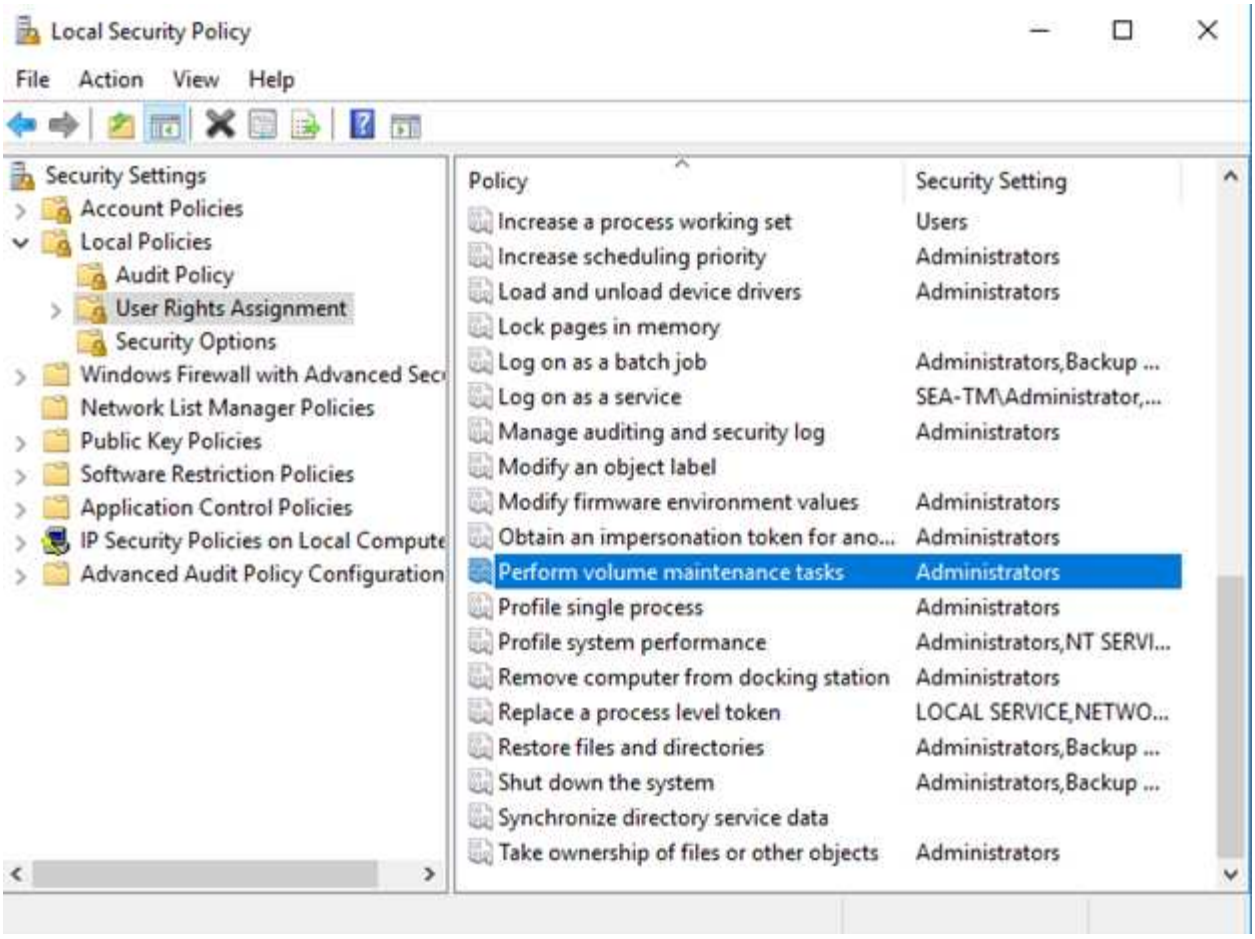
* NetAppでは、1つのファイルグループ内のすべてのファイルに同じ初期サイズと自動拡張パラメータを設定し、拡張サイズをパーセンテージではなくメガバイト単位で定義することを推奨しています*。これにより、Proportional Fill Algorithmは、データファイル間で書き込みアクティビティのバランスを均等に調整できます。

SQL Serverは、ファイルを拡張するたびに、新しく割り当てられたスペースをゼロでいっぱいにします。このプロセスは、対応するファイルへの書き込みが必要なすべてのセッションをブロックします。トランザクションログが増加した場合は、トランザクションログレコードを生成します。

SQL Serverは常にトランザクションログをゼロにし、その動作を変更することはできません。ただし、インスタントファイルの初期化を有効または無効にすることで、データファイルを初期化するかどうかを制御できます。インスタントファイルの初期化を有効にすると、データファイルの増加を高速化し、データベースの作成やリストアに必要な時間を短縮できます。

インスタントファイルの初期化には、わずかなセキュリティリスクが伴います。このオプションを有効にすると、データファイルの未割り当て部分に、以前に削除されたOSファイルの情報を含めることができます。データベース管理者はこのようなデータを調べることができます。

インスタントファイルの初期化を有効にするには、「ボリュームメンテナンスタスクの実行」とも呼ばれるSA_MANAGE_VOLUME_name権限をSQL Serverスタートアップアカウントに追加します。これは、次の図に示すように、ローカルセキュリティポリシー管理アプリケーション（secpol.msc）で実行できます。「Perform volume maintenance task」権限のプロパティを開き、SQL Serverスタートアップアカウントをユーザのリストに追加します。



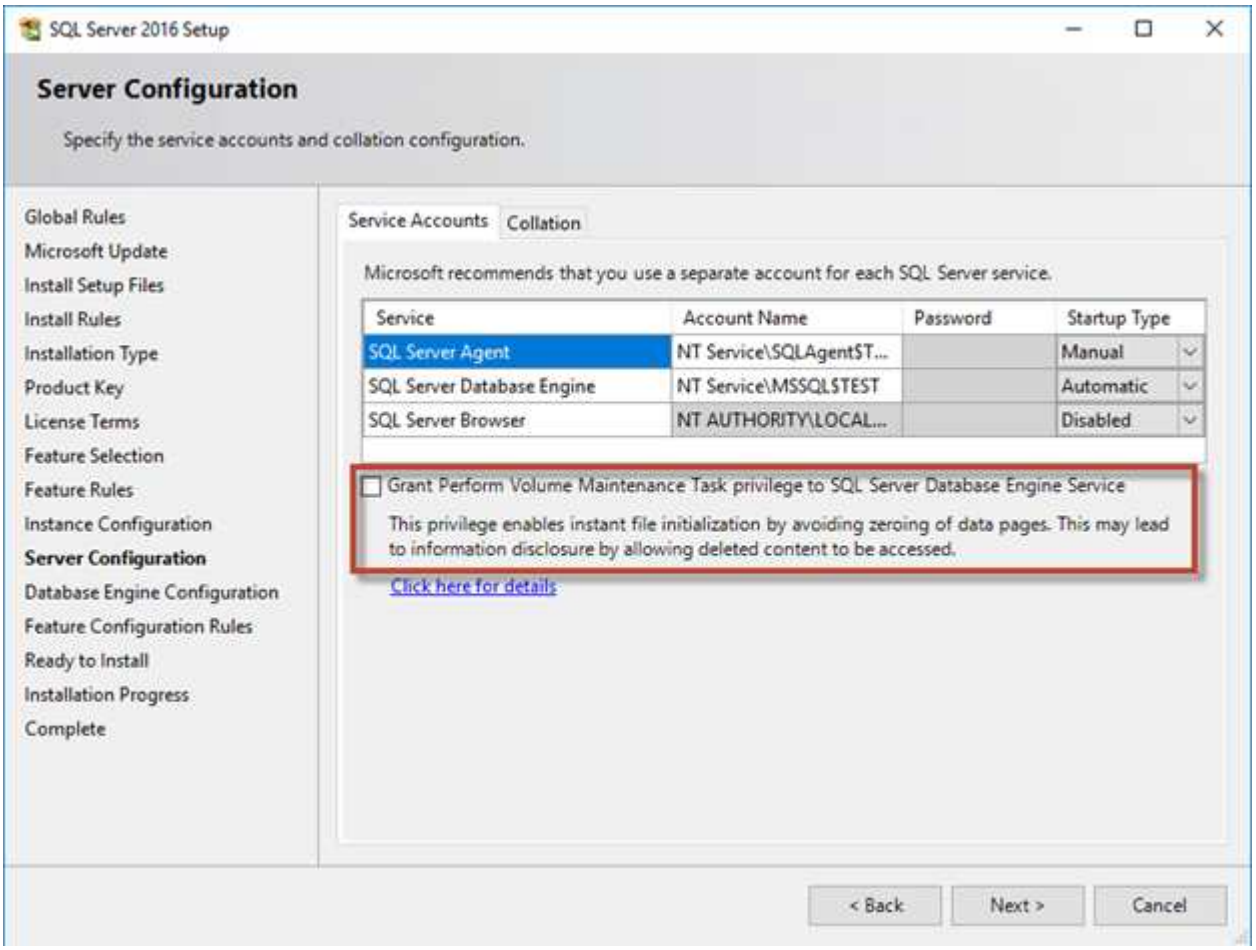
権限が有効になっているかどうかを確認するには、次の例のコードを使用します。このコードは、SQL Serverがエラーログに追加情報を書き込み、小さなデータベースを作成し、ログの内容を読み取るように強制する2つのトレースフラグを設定します。

```
DBCC TRACEON(3004,3605,-1)
GO
CREATE DATABASE DelMe
GO
EXECUTE sp_readerrorlog
GO
DROP DATABASE DelMe
GO
DBCC TRACEOFF(3004,3605,-1)
GO
```

インスタントファイルの初期化が有効になっていない場合、次の例に示すように、SQL Serverのエラーログには、LDFログファイルの初期化に加えてMDFデータファイルが初期化されていることが示されます。インスタントファイルの初期化を有効にすると、ログファイルの初期化のみが表示されます。

	LogDate	ProcessInfo	Text
365	2017-02-09 08:10:07.660	spid53	Ckpt dbid 3 flush delta counts.
366	2017-02-09 08:10:07.660	spid53	Ckpt dbid 3 logging active xact info.
367	2017-02-09 08:10:07.750	spid53	Ckpt dbid 3 phase 1 ended (8)
368	2017-02-09 08:10:07.750	spid53	About to log Checkpoint end.
369	2017-02-09 08:10:07.880	spid53	Ckpt dbid 3 complete
370	2017-02-09 08:10:08.130	spid53	Starting up database 'DelMe'.
371	2017-02-09 08:10:08.150	spid53	FixupLogTail(progress) zeroing C:\Program Files\Micros
372	2017-02-09 08:10:08.160	spid53	Zeroing C:\Program Files\Microsoft SQL Server\MSSQ
373	2017-02-09 08:10:08.170	spid53	Zeroing completed on C:\Program Files\Microsoft SQL
374	2017-02-09 08:10:08.710	spid53	Ckpt dbid 6 started
375	2017-02-09 08:10:08.710	spid53	About to log Checkpoint begin.

[Perform Volume Maintenance]タスクはSQL Server 2016では簡素化され、インストールプロセス中にオプションとして提供されます。この図は、SQL Serverデータベースエンジンサービスにボリュームメンテナンスタスクを実行する権限を付与するオプションを示しています。



データベースファイルのサイズを制御するもう1つの重要なデータベースオプションは、自動縮小です。このオプションを有効にすると、SQL Serverはデータベースファイルを定期的に縮小してサイズを縮小し、オペレーティングシステムにスペースを解放します。この処理はリソースを大量に消費するため、新しいデータがシステムに入ってくるとデータベースファイルが再び拡張されるため、あまり有用ではありません。データベースで自動縮小を有効にしないでください。

ログディレクトリ

ログディレクトリは、トランザクションログバックアップデータをホストレベルで格納するためにSQL Serverで指定します。SnapCenterを使用してログファイルをバックアップする場合は、SnapCenterで使用される各SQL Serverホストに、ログバックアップを実行するようにホストログディレクトリを設定する必要があります。

ログディレクトリを専用のストレージユニットに配置します。ホストログディレクトリのデータ量は、バックアップのサイズとバックアップを保持する日数によって異なります。SnapCenterでは、SQL Serverホストごとに1つのホストログディレクトリのみが許可されます。ホストログディレクトリは、SnapCenter -→ホスト -→プラグインの設定で設定できます。

- NetAppでは、ホストログディレクトリに次のことを推奨しています*。
- ホストログディレクトリが、バックアップSnapshotデータを破損する可能性のある他のタイプのデータと共有されていないことを確認してください。
- SnapCenterによるトランザクション・ログのコピー先となる専用のストレージ・ユニットに、ホスト・ログ・ディレクトリを作成します。
- Always Onフェイルオーバークラスタインスタンスを使用している場合、ホストログディレクトリに使用されるストレージユニットは、SnapCenterでバックアップされるSQL Serverインスタンスと同じクラスタグループ内のクラスタディスクリソースである必要があります。

データ保護

データベースのバックアップ戦略は、理論的な機能ではなく、特定されたビジネス要件に基づいて行う必要があります。ONTAPのSnapshotテクノロジとMicrosoft SQL Server APIを組み合わせることで、ユーザデータベースのサイズに関係なく、アプリケーションと整合性のあるバックアップを迅速に作成できます。より高度なデータ管理やスケールアウトデータ管理の要件に対応するために、NetAppはSnapCenterを提供しています。

SnapCenter

SnapCenterは、エンタープライズアプリケーション向けのNetAppデータ保護ソフトウェアです。SnapCenter Plug-in for SQL Serverや、SnapCenter Plug-in for Microsoft Windowsで管理されるOS処理を使用して、SQL Serverデータベースを迅速かつ簡単に保護できます。

SQL Serverインスタンスは、スタンドアロンセットアップ、フェイルオーバークラスタインスタンス、または常時稼働の可用性グループにすることができます。その結果、データベースの保護、クローニング、リストアをプライマリコピーまたはセカンダリコピーから単一コンソールで実行できます。SnapCenterでは、SQL Serverデータベースをオンプレミス、クラウド、ハイブリッド構成の両方で管理できます。また、開発やレポート作成のために、元のホストまたは代替ホストに数分でデータベースコピーを作成することもできます。

また、SQL Serverでは、作成時にSnapshotに正しいデータが存在するように、OSとストレージの間で調整を行う必要があります。ほとんどの場合、これを行う唯一の安全な方法は、SnapCenterまたはT-SQLを使用することです。この追加の調整なしで作成されたSnapshotは、確実にリカバリできない可能性があります。

SQL Server Plug-in for SnapCenterの詳細については、を参照してください。 "TR-4714：『Best Practice Guide for SQL Server using NetApp SnapCenter』"。

T-SQLスナップショットを使用したデータベースの保護

SQL Server 2022では、MicrosoftがT-SQLスナップショットを導入しました。これにより、バックアップ処理のスクリプト作成と自動化を行うことができます。フルサイズのコピーを実行する代わりに、Snapshot用にデータベースを準備できます。データベースのバックアップ準備が完了したら、ONTAP REST APIを使用してSnapshotを作成できます。

次に、バックアップワークフローの例を示します。

1. ALTERコマンドを使用してデータベースをフリーズします。これにより、基盤となるストレージ上で整合性のあるSnapshotを作成するためのデータベースが準備されます。フリーズ後、backupコマンドを使用してデータベースをフリーズ解除し、スナップショットを記録できます。
2. 新しいbackup groupコマンドとbackup serverコマンドを使用して、ストレージユニット上の複数のデータベースのスナップショットを同時に実行します。
3. データベースワークロードが複数のストレージユニットにまたがっている場合は、整合グループを作成して管理タスクを簡易化します。整合グループは、単一ユニットとして管理されるストレージユニットの集まりです。
4. フルバックアップまたはCOPY_ONLYフルバックアップを実行します。これらのバックアップもmsdbに記録されます。
5. スナップショットフルバックアップ後に通常のストリーミング方式で作成されたログバックアップを使用して、ポイントインタイムリカバリを実行します。必要に応じて、ストリーミング差分バックアップもサポートされます。

詳細については、を参照してください "T-SQLスナップショットについて知るためのMicrosoftのドキュメント"。

* NetAppでは* SnapCenterを使用してSnapshotコピーを作成することを推奨しています。前述のT-SQL方式も機能しますが、SnapCenterでは、バックアップ、リストア、クローニングのプロセスを完全に自動化できます。また、検出を実行して、正しいSnapshotが作成されていることを確認します。

ディザスタリカバリ

ディザスタリカバリ

エンタープライズデータベースやアプリケーションインフラでは、自然災害や予期しないビジネスの中断からダウンタイムを最小限に抑えて保護するために、レプリケーションが必要になることがよくあります。

SQL Server Always-On可用性グループレプリケーション機能は優れたオプションであり、NetAppには、データ保護とAlways-Onを統合するためのオプションが用意されています。ただし、場合によっては、次のオプションを使用してONTAPレプリケーションテクノロジを検討することもできます。

SnapMirror

SnapMirrorテクノロジーは、LANおよびWAN経由でデータを複製するための高速で柔軟なエンタープライズソリューションを提供します。最初のミラーリングの作成後は、変更されたデータブロックのみがデスティネーションに転送されるため、必要なネットワーク帯域幅が大幅に削減されますSnapMirror。同期モードまたは非同期モードのいずれかで設定できます。NetApp ASAでのSnapMirror同期レプリケーションは、SnapMirror

アクティブ同期を使用して設定します。

SnapMirrorアクティブ同期

多くのお客様にとって、ビジネス継続性には、単にデータのリモートコピーを所有するだけでなく、SnapMirrorのアクティブ同期を使用してNetApp ONTAPで可能なデータを迅速に利用できる機能が必要です。

SnapMirrorアクティブ同期を使用すると、基本的には2つの異なるONTAPシステムでLUNデータの独立したコピーを維持しながら、このLUNの単一インスタンスを提供できます。ホストの観点からは、単一のLUNエンティティです。SnapMirrorアクティブ同期は、iSCSI / FCベースのLUNでサポートされます。

SnapMirrorアクティブ同期はRPO=0のレプリケーションを提供し、2つの独立したクラスタ間で簡単に実装できます。データの2つのコピーが同期されると、2つのクラスタは書き込みをミラーリングするだけで済みます。一方のクラスタで書き込みが発生すると、もう一方のクラスタにレプリケートされます。書き込みの確認応答がホストに送信されるのは、両方のサイトで書き込みが完了した場合だけです。このプロトコルスプリット動作以外では、2つのクラスタは通常のONTAPクラスタです。

SM-ASの主なユースケースの1つに、きめ細かなレプリケーションがあります。すべてのデータを1つのユニットとしてレプリケートしたくない場合や、特定のワークロードを選択的にフェイルオーバーできる必要がある場合があります。

SM-ASのもう1つの主なユースケースは、アクティブ/アクティブ処理です。アクティブ/アクティブ処理では、データの完全に使用可能なコピーを、同じパフォーマンス特性を持つ2つの異なるクラスタに配置し、必要に応じてSANをサイト間で拡張する必要がありません。アプリケーションがサポートされていれば、両方のサイトでアプリケーションをすでに実行しておくことができます。これにより、フェイルオーバー処理中の全体的なRTOが削減されます。

SnapMirror

SnapMirror for SQL Serverの推奨事項は次のとおりです。

- 迅速なデータリカバリのニーズが高い場合は、SnapMirrorアクティブ同期を使用した同期レプリケーションを使用し、RPOの柔軟性を高める非同期ソリューションを使用します。
- SnapCenterを使用してデータベースをバックアップし、Snapshotをリモートクラスタにレプリケートする場合は、整合性を確保するためにコントローラからのSnapMirror更新のスケジュールを設定しないでください。代わりに、フルバックアップまたはログバックアップの完了後にSnapCenterからのSnapMirror更新を有効にしてSnapMirrorを更新します。
- SQL Serverデータを含むストレージユニットをクラスタ内の複数のノードに分散して、すべてのクラスタノードでSnapMirrorレプリケーションアクティビティを共有できるようにします。この分散により、ノードリソースの使用が最適化されます。

SnapMirrorの詳細については、を参照してください。 "TR-4015：『SnapMirrorの設定およびベストプラクティスガイド- ONTAP 9』"。

SnapMirrorアクティブ同期

概要

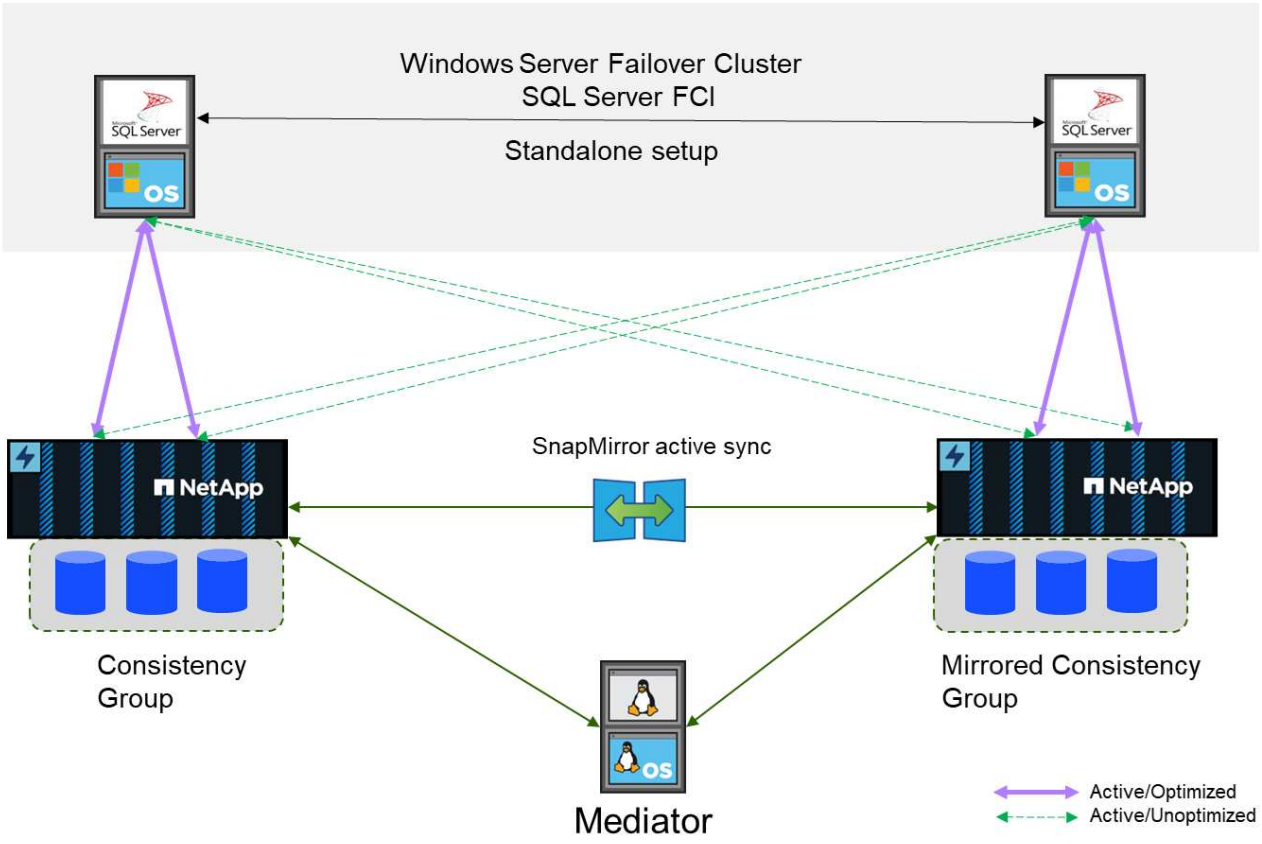
SnapMirror Active Syncを使用すると、ストレージやネットワークが停止しても、個 々のSQL Serverデータベースとアプリケーションの運用を継続できます。透過的なストレ

ージフェイルオーバーにより、手動操作は不要です。

SnapMirrorアクティブ同期は、同期双方向レプリケーションを提供する対称アクティブ/アクティブアーキテクチャをサポートし、ビジネス継続性とディザスタリカバリを実現します。複数の障害ドメインにわたるデータへの同時読み取り/書き込みアクセスにより、重要なSANワークロードのデータアクセスを保護し、運用を中断させず、災害やシステム障害時のダウンタイムを最小限に抑えることができます。

SQL Serverホストは、ファイバチャネル（FC）LUNまたはiSCSI LUNを使用してストレージにアクセスします。レプリケートされたデータのコピーをホストする各クラスタ間のレプリケーション。この機能はストレージレベルのレプリケーションであるため、スタンドアロンホストインスタンスまたはフェイルオーバークラスタインスタンス上で実行されているSQL Serverインスタンスは、どちらのクラスタでも読み取り/書き込み処理を実行できます。計画と設定の手順については、を参照してください"SnapMirror Active Syncに関するONTAPドキュメント"。

対称アクティブ/アクティブと**SnapMirror**アクティブのアーキテクチャ



同期レプリケーション

通常運用時には、1つの例外を除き、各コピーは常にRPO=0の同期レプリカになります。データをレプリケートできない場合、ONTAPでは、データのレプリケートという要件が解除され、一方のサイトのLUNがオフラインになる間に、一方のサイトでIOの提供が再開されます。

ストレージハードウェア

他のストレージディザスタリカバリソリューションとは異なり、SnapMirrorアクティブ同期は非対称プラットフォームの柔軟性を提供します。各サイトのハードウェアが同一である必要はありません。この機能を使用すると、SnapMirrorアクティブ同期をサポートするために使用するハードウェアのサイズを適正化できます。リ

モートストレージシステムは、本番環境のワークロードを完全にサポートする必要がある場合はプライマリサイトと同一にすることができますが、災害によってI/Oが減少した場合は、リモートサイトの小規模システムよりも対費用効果が高くなります。

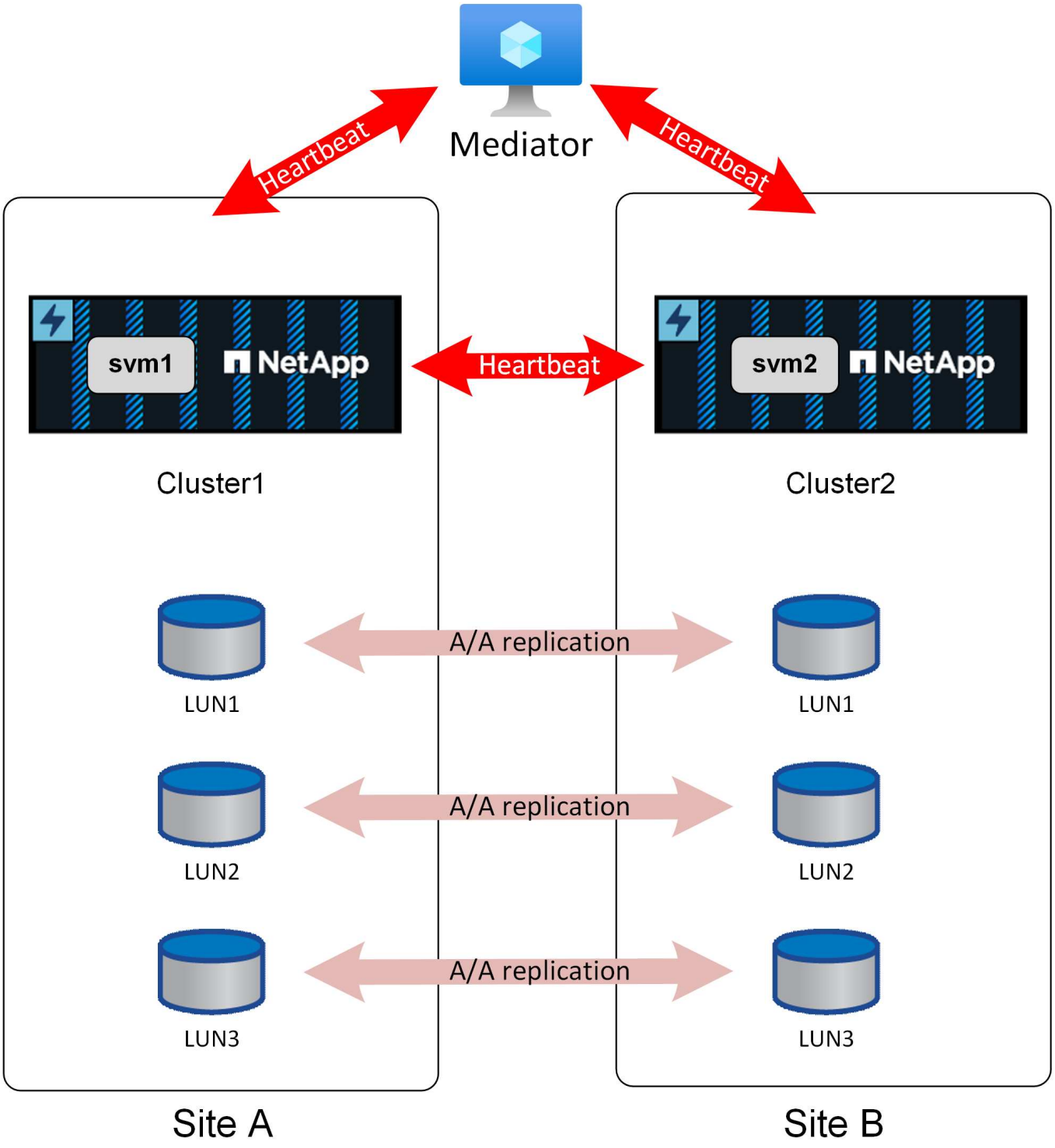
- ONTAPメディエーター**

ONTAPメディエーターは、NetAppサポートからダウンロードするソフトウェアアプリケーションで、通常は小規模な仮想マシンに導入されます。ONTAPメディエーターはTiebreakerではありません。これは、SnapMirrorのアクティブな同期レプリケーションに含まれる2つのクラスタの代替通信チャネルです。自動処理は、パートナーから直接接続またはメディエーター経由で受け取った応答に基づいてONTAPによって実行されます。

ONTAPメディエーター

フェイルオーバーを安全に自動化するにはメディエーターが必要です。理想的には、独立した3つ目のサイトに配置しますが、レプリケーションに参加しているクラスタの1つと同じ場所に配置すれば、ほとんどのニーズに対応できます。

メディエーターは実際にはTiebreakerではありませんが、それは事実上それが提供する機能です。処理は行われず、代わりにクラスタ間の通信用の代替通信チャネルを提供します。



自動フェイルオーバーの最大の課題はスプリットブレインの問題であり、この問題は2つのサイト間の接続が失われた場合に発生します。何が起こるべきでしょうか? 2つの異なるサイトがデータのサバイバーコピーとして自分自身を指定する必要はありませんが、1つのサイトでは、反対側のサイトが実際に失われたことと、反対側のサイトと通信できないことを区別するにはどうすればよいでしょうか。

ここでメディエーターが写真に入ります3番目のサイトに配置され、各サイトからそのサイトへの個別のネットワーク接続がある場合は、他のサイトの正常性を検証するための追加のパスが各サイトに用意されています。上の図をもう一度見て、次のシナリオを検討してください。

- 一方または両方のサイトからメディエーターに障害が発生した場合、またはメディエーターに到達できない場合はどうなりますか？
  - 2つのクラスタは、レプリケーションサービスに使用されるのと同じリンクを介して相互に通信できます。
  - データは引き続きRPO=0の保護で提供される
- サイトAに障害が発生した場合の動作
  - サイトBは、両方の通信チャネルがダウンしたことを確認します。
  - サイトBがデータサービスをテイクオーバーするが、RPO=0ミラーリングなし
- サイトBで障害が発生した場合の動作
  - サイトAでは、両方の通信チャネルがダウンしていることが確認されます。
  - サイトAがデータサービスをテイクオーバーするが、RPO=0ミラーリングなし

もう1つ考慮すべきシナリオがあります。データレプリケーションリンクの停止です。サイト間のレプリケーションリンクが失われた場合、RPO=0のミラーリングは明らかに不可能です。ではどうすればいいのでしょうか。

これは、優先サイトのステータスによって制御されます。SM-AS関係では、一方のサイトがもう一方のサイトのセカンダリになります。これは通常の運用には影響せず、すべてのデータアクセスは対称的ですが、レプリケーションが中断された場合は、運用を再開するためにこの関係を解除する必要があります。その結果、優先サイトはミラーリングなしで処理を継続し、レプリケーション通信がリストアされるまでセカンダリサイトはIO処理を停止します。

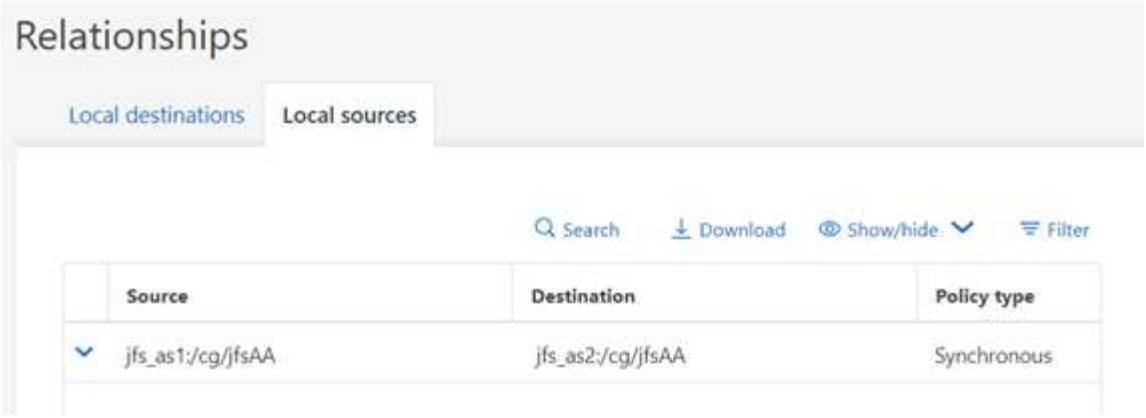
優先サイト

SnapMirrorのアクティブな同期の動作は対称ですが、重要な例外が1つあります（推奨サイト構成）。

SnapMirrorアクティブ同期では、一方のサイトが「ソース」で、もう一方が「デスティネーション」と見なされます。これは一方向のレプリケーション関係を意味しますが、IO動作には適用されません。レプリケーションは双方向であり、対称であり、IO応答時間はミラーの両側で同じです。

> `source`指定は、優先サイトを制御します。レプリケーションリンクが失われた場合、ソースコピー上のLUNパスは引き続きデータを提供しますが、デスティネーションコピー上のLUNパスは、レプリケーションが再確立されてSnapMirrorが同期状態に戻るまで使用できなくなります。その後、パスでデータの提供が再開されます。

ソース/デスティネーションの設定はSystemManagerで確認できます。



または、CLIで次の操作を行います。

```
Cluster2::> snapmirror show -destination-path jfs_as2:/cg/jfsAA

                           Source Path: jfs_as1:/cg/jfsAA
                      Destination Path: jfs_as2:/cg/jfsAA
                     Relationship Type: XDP
                Relationship Group Type: consistencygroup
                    SnapMirror Schedule: -
                 SnapMirror Policy Type: automated-failover-duplex
                      SnapMirror Policy: AutomatedFailOverDuplex
                            Tries Limit: -
                      Throttle (KB/sec): -
                           Mirror State: Snapmirrored
                    Relationship Status: InSync
```

重要なのは、ソースがcluster1のSVMであることです。前述のように、「ソース」と「デスティネーション」という用語は、レプリケートされたデータのフローを表していません。両方のサイトが書き込みを処理し、反対側のサイトにレプリケートできます。実際には、両方のクラスタがソースとデスティネーションです。1つのクラスタをソースとして指定すると、レプリケーションリンクが失われた場合に、どのクラスタが読み取り/書き込みストレージシステムとして残っているかが制御されます。

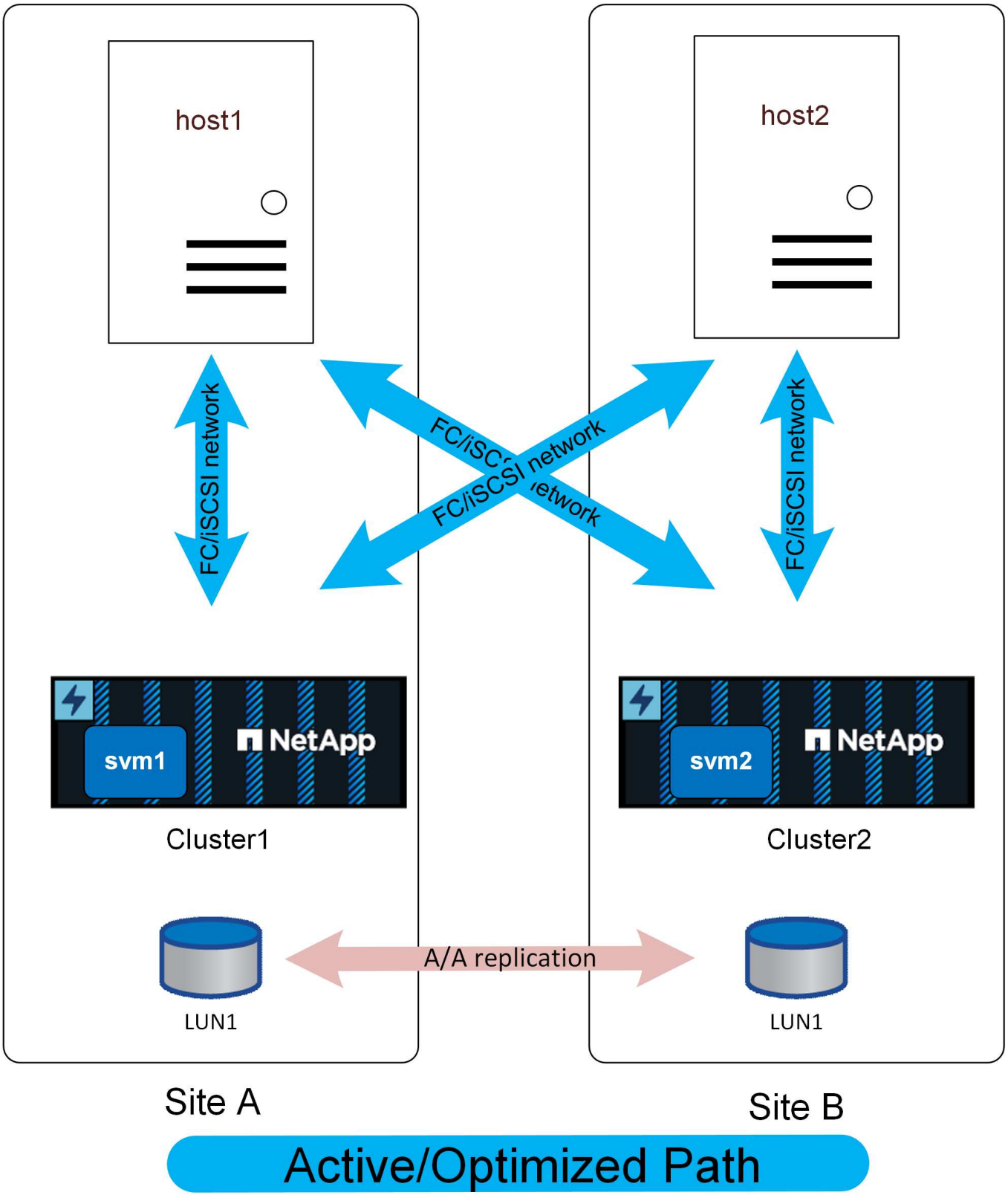
ネットワークトポロジ

均一なアクセス

統一されたアクセスネットワークとは、ホストが両方のサイト（または同じサイト内の障害ドメイン）のパスにアクセスできることを意味します。

SM-ASの重要な機能の1つは、ホストがどこにあるかを認識するようにストレージシステムを設定できることです。LUNを特定のホストにマッピングするときに、LUNが特定のストレージシステムに近接しているかどうかを指定できます。

NetApp ASAシステムは、クラスタ上のすべてのパスでアクティブ/アクティブマルチパスを提供します。これはSM-AS設定にも適用されます。



アクセスが統一されている場合、IOはWANを通過します。これはフルメッシュネットワーククラスタであり、すべてのユースケースに適している場合とそうでない場合があります。

2つのサイトがファイバ接続で100m離れている場合、WANを経由する追加のレイテンシは検出されませんが、サイト間の距離が離れていると、両方のサイトで読み取りパフォーマンスが低下します。不均一なアクセ

スネットワークを使用するASAは、サイト間のレイテンシアクセスペナルティを発生させることなく、ASAのコストと機能のメリットを享受したり、ホストプロキシミティ機能を使用して両方のサイトでサイトローカルの読み取り/書き込みアクセスを許可したりすることができます。

低レイテンシ構成でSM-ASを使用するASAには、2つの興味深い利点があります。まず、I/Oは2倍のパスを使用して2倍のコントローラで処理できるため、1台のホストのパフォーマンスが実質的に2倍になります。2つ目は、単一サイト環境では、ホストへのアクセスを中断することなくストレージシステム全体が失われる可能性があるため、非常に高い可用性を提供することです。

近接設定

プロキシミティとは、特定のホストWWNまたはiSCSIイニシエータIDがローカルホストに属していることを示すクラスタ単位の構成を指します。これは、LUNアクセスを設定するための2番目のオプションの手順です。

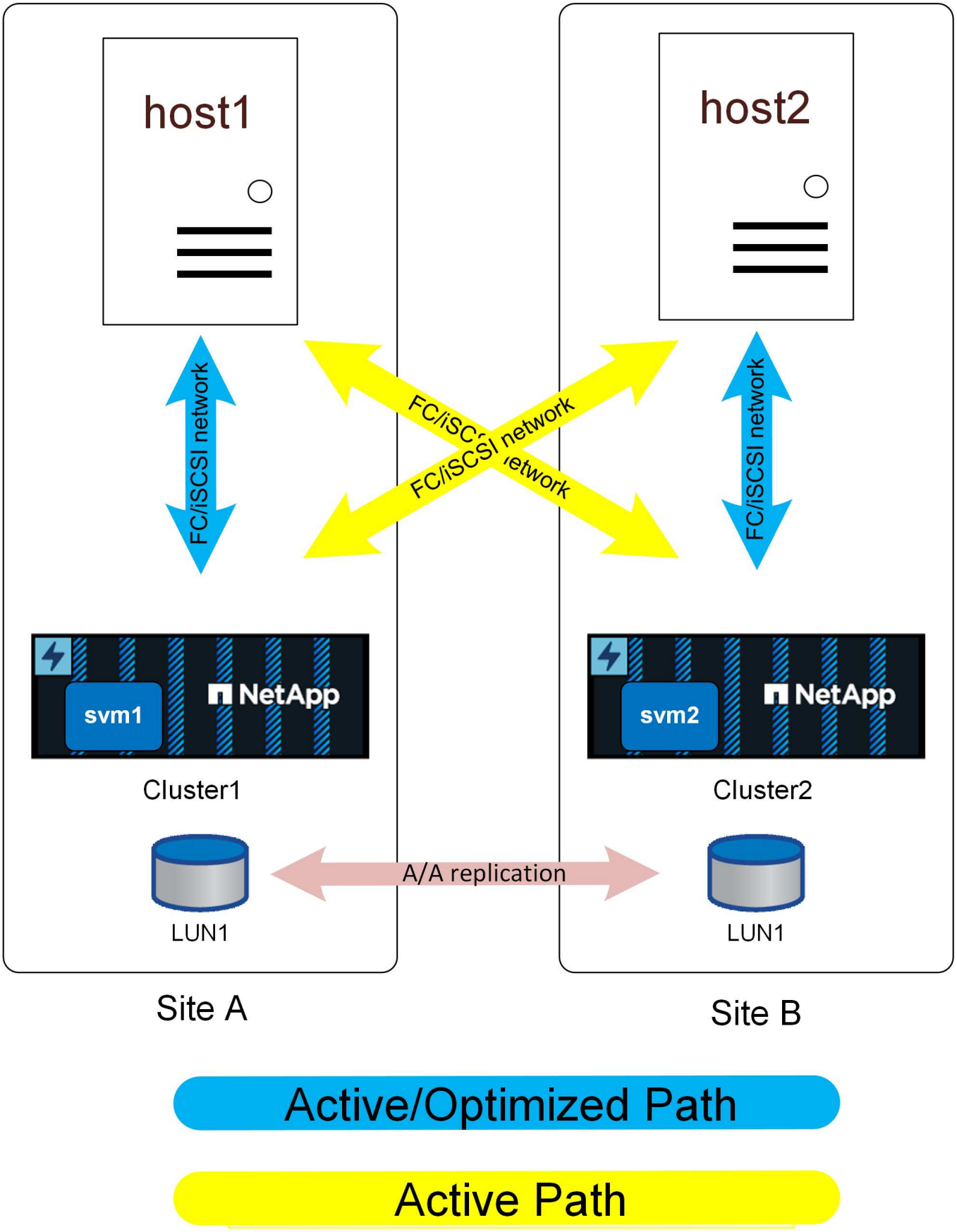
最初の手順では、通常のigroup設定を行います。各LUNは、そのLUNにアクセスする必要があるホストのWWN/iSCSI IDを含むigroupにマッピングする必要があります。これは、どのホストがLUNに_access_toを持つかを制御します。

2番目のオプション手順は、ホストプロキシミティを設定することです。これはアクセスを制御するのではなく、_priority_を制御します。

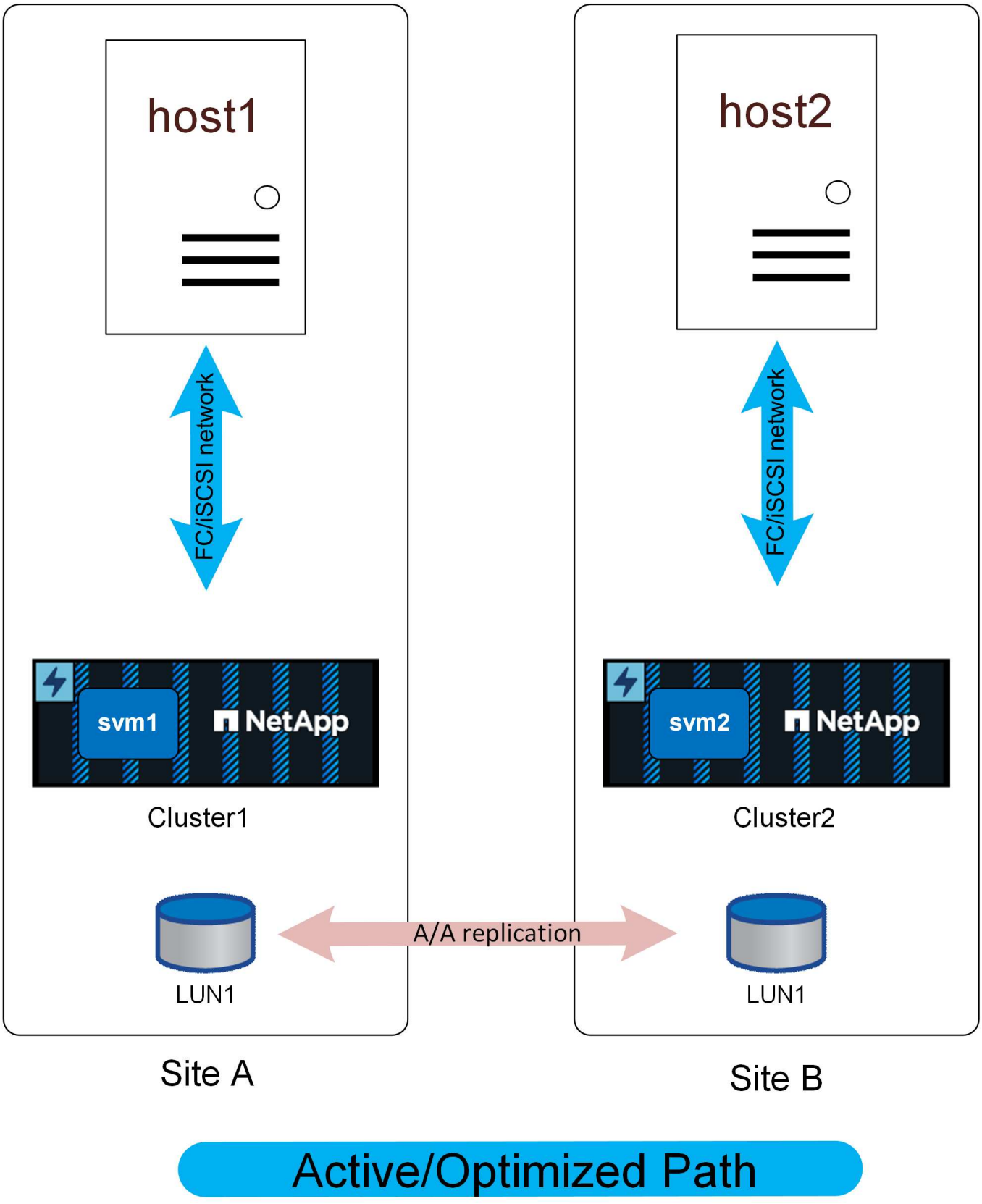
たとえば、サイトAのホストがSnapMirror Active Syncで保護されているLUNにアクセスするように設定されている場合、SANがサイト間で拡張されるため、サイトAのストレージまたはサイトBのストレージを使用してそのLUNへのパスを使用できます。

近接設定を使用しない場合、両方のストレージシステムがアクティブな最適パスをアドバタイズするため、そのホストは両方のストレージシステムを均等に使用します。SANのレイテンシやサイト間の帯域幅に制限がある場合は、この設定を解除できない可能性があります。また、通常動作中に各ホストがローカルストレージシステムへのパスを優先的に使用するように設定することもできます。これは、ホストWWN/iSCSI IDをローカルクラスタに近接ホストとして追加することで設定します。これは、CLIまたはSystemManagerで実行できます。

ホストプロキシミティが設定されている場合、パスは次のように表示されます。


host1
host2
FC/iSCSI network
FC/iSCSI network
FC/iSCSI network
FC/iSCSI network
svm1
NetApp
svm2
NetApp
Cluster1
Cluster2
A/A replication
LUN1
LUN1
Site A
Site B
Active/Optimized Path
Active Path

非ユニフォームアクセスネットワークとは、各ホストがローカルストレージシステム上のポートにしかアクセスできないことを意味します。SANを複数のサイト（または同じサイト内の障害ドメイン）に拡張することはできません。



このアプローチの主なメリットはSANのシンプルさです。SANをネットワーク経由で拡張する必要がなくなります。お客様によっては、サイト間の接続遅延が十分でない場合や、サイト間ネットワーク経由でFC SANトラフィックをトンネリングするためのインフラストラクチャが不足している場合があります。

不均一なアクセスの欠点は、レプリケーションリンクの喪失などの特定の障害シナリオで、一部のホストがストレージにアクセスできなくなることです。ローカルストレージの接続が失われると、単一のホストでのみ実行されている非クラスタデータベースなど、単一インスタンスとして実行されるアプリケーションは失敗します。データは保護されますが'データベース・サーバはアクセスできなくなりますリモートサイトで、できれば自動化されたプロセスを使用して再起動する必要があります。たとえば、VMware HAは、あるサーバでオールパスダウン状態を検出し、パスが使用可能な別のサーバでVMを再起動できます。

一方、Oracle RACなどのクラスタ化されたアプリケーションは、2つの異なるサイトで同時に利用可能なサービスを提供できます。サイトが失われても、アプリケーションサービス全体が失われるわけではありません。サバイバーサイトでは、引き続きインスタンスを使用して実行できます。

多くの場合、サイト間リンク経由でストレージにアクセスするアプリケーションによるレイテンシのオーバーヘッドは許容できません。つまり、サイトのストレージが失われると、障害が発生したサイトのサービスをシャットダウンする必要が生じるため、統一されたネットワークの可用性の向上は最小限で済みます。

この図では、わかりやすいように、ローカルクラスタを経由する冗長パスを示していません。ONTAPストレージシステム自体はHAであるため、コントローラ障害が発生してもサイト障害は発生しません。影響を受けるサイトで使用されるローカルパスが変更されるだけです。

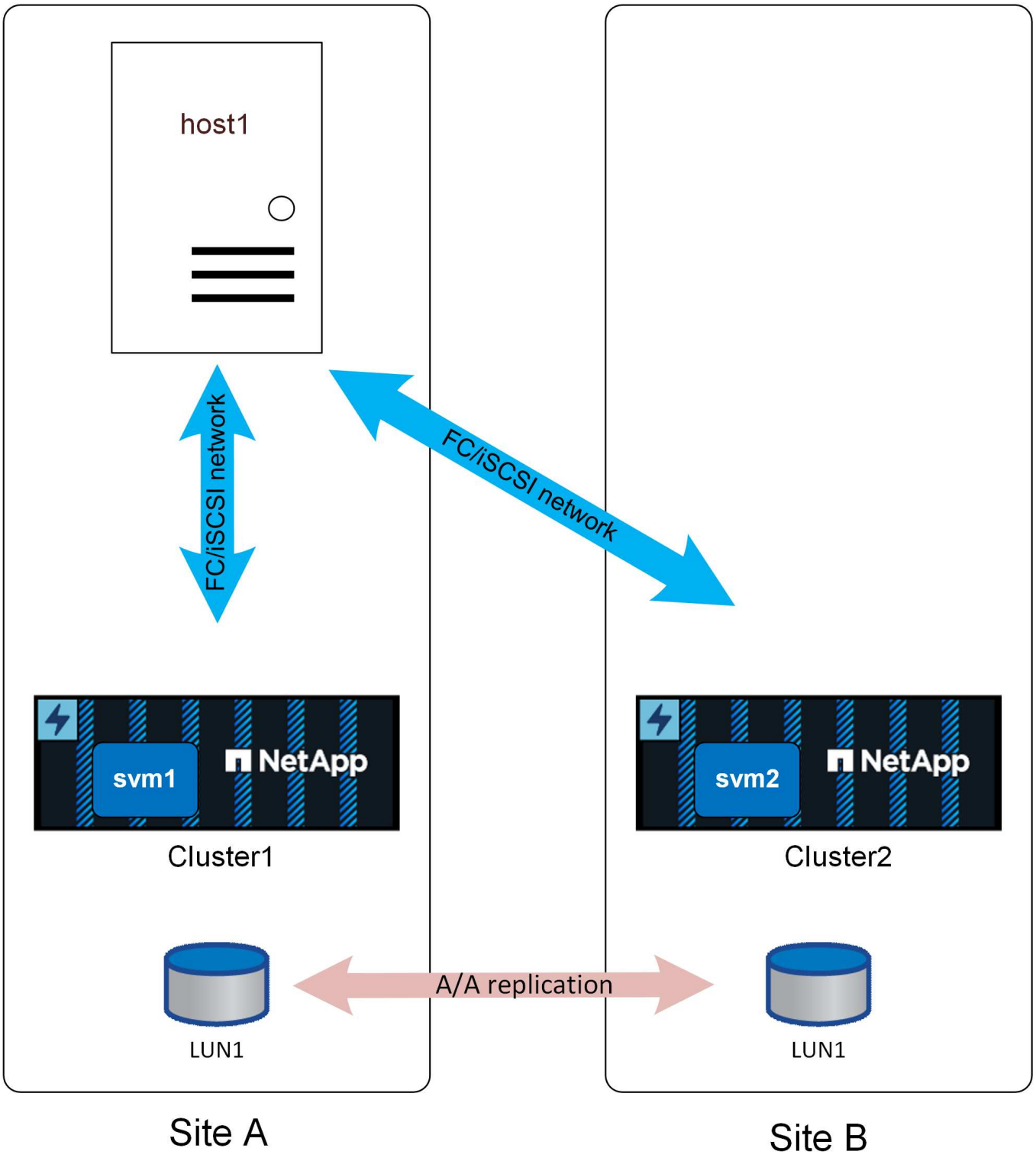
概要

SQL Serverは、いくつかの方法でSnapMirrorアクティブ同期と連携するように設定できます。適切な答えは、使用可能なネットワーク接続、RPOの要件、可用性の要件によって異なります。

SQL Serverのスタンドアロンインスタンス

ファイルレイアウトとサーバ設定のベストプラクティスは、ドキュメントで推奨されているものと同じ"ONTAP上のSQL Server"です。

スタンドアロン構成では、SQL Serverを1つのサイトでのみ実行できます。おそらく"均一（ Uniform ）"アクセスが使用されます。



アクセス方法が統一されていれば、どちらかのサイトでストレージ障害が発生してもデータベースの処理は中断されません。データベース・サーバを含むサイトで完全なサイト障害が発生すると'当然'システム停止が発生します

一部のお客様は、リモートサイトで実行されているOSを、構成済みのSQL Serverセットアップで構成し、本番インスタンスと同等のビルドバージョンで更新することもできます。フェイルオーバーを実行するには、代替サイトでSQL Serverのスタンドアロンインスタンスをアクティブ化し、LUNを検出して、データベースを起動する必要があります。ストレージ側からの操作が不要なため、Windows PowerShellコマンドレットを使

用して完全なプロセスを自動化できます。

"不均一"アクセスも使用できますが、データベースにストレージへの使用可能なパスがないために、データベースサーバが配置されていたストレージシステムで障害が発生すると、データベースが停止します。これは、場合によっては許容される可能性があります。SnapMirrorのアクティブな同期では引き続きRPO=0のデータ保護が提供され、サイト障害が発生した場合でも、稼働しているコピーがアクティブになり、前述の統一されたアクセスと同じ手順で運用を再開できます。

シンプルで自動化されたフェイルオーバープロセスは、仮想化ホストを使用してより簡単に設定できます。たとえば、SQL ServerデータファイルをブートVMDKとともにセカンダリストレージに同期的にレプリケートする場合は、災害が発生したときに代替サイトで環境全体をアクティブ化できます。管理者は、サバイバーサイトでホストを手動でアクティブ化することも、VMware HAなどのサービスを使用してプロセスを自動化することもできます。

SQL Serverフェイルオーバークラスタインスタンス

SQL Serverフェイルオーバーインスタンスは、物理サーバまたは仮想サーバ上でゲストオペレーティングシステムとして実行されているWindowsフェイルオーバークラスタでホストすることもできます。このマルチホストアーキテクチャは、SQL Serverインスタンスとストレージの耐障害性を提供します。このような導入は、強化されたパフォーマンスを維持しながら堅牢なフェイルオーバープロセスを必要とする、負荷の高い環境に役立ちます。フェイルオーバークラスタのセットアップでは、ホストまたはプライマリストレージが影響を受けると、SQLサービスがセカンダリホストにフェイルオーバーされ、同時にセカンダリストレージがIOを処理できるようになります。自動化スクリプトや管理者の介入は必要ありません。

障害シナリオ

完全なSnapMirrorアクティブ同期アプリケーションアーキテクチャを計画するには、さまざまな計画的フェイルオーバーシナリオと計画外フェイルオーバーシナリオでSM-ASがどのように対応するかを理解する必要があります。

次の例では、サイトAが優先サイトとして設定されているとします。

レプリケーション接続の切断

SM-ASレプリケーションが中断されると、クラスタが反対側のサイトに変更をレプリケートできなくなるため、書き込みIOを完了できません。

サイト**A**（優先サイト）

優先サイトでのレプリケーションリンク障害の結果、レプリケーションリンクが本当に到達不能であると判断される前に、ONTAPがレプリケートされた書き込み処理を再試行するため、書き込みIO処理が約15秒間中断されます。15秒が経過すると、サイトAのシステムが読み取りと書き込みのIO処理を再開します。SANパスは変更されず、LUNはオンラインのままです。

サイト**B**

サイトBはSnapMirrorアクティブ同期優先サイトではないため、約15秒後にLUNパスが使用できなくなります。

ストレージシステムの障害

ストレージシステム障害の結果は、レプリケーションリンクが失われた場合とほぼ同じです。サバイバーサイトでは、IOが約15秒間停止します。その15秒が経過すると、IOは通常どおりそのサイトで再開されます。

メディエーターの停止

メディエーターサービスはストレージの処理を直接制御しません。クラスタ間の代替制御パスとして機能します。これは主に、スプリットブレインのリスクを伴わずにフェイルオーバーを自動化することを目的としています。通常運用時は、各クラスタがパートナーに変更内容をレプリケートするため、各クラスタはパートナークラスタがオンラインでデータを提供していることを確認できます。レプリケーションリンクに障害が発生すると、レプリケーションは停止します。

安全な自動フェイルオーバーを実現するためにメディエーターが必要になるのは、そうしないと、双方向通信の切断がネットワークの停止によるものか実際のストレージ障害によるものかをストレージクラスタが判断できないためです。

メディエーターは、パートナーの健常性を確認するための代替パスを各クラスタに提供します。シナリオは次のとおりです。

- クラスタがパートナーに直接接続できる場合は、レプリケーションサービスが動作しています。対処は不要です。
- 優先サイトがパートナーに直接またはメディエーターを介してアクセスできない場合、パートナーが実際に使用できないか分離されてLUNパスがオフラインになっているとみなされます。その後、優先サイトでRPO=0の状態が解除され、読み取りI/Oと書き込みI/Oの両方の処理が続行されます。
- 非優先サイトがパートナーに直接接続できず、メディエーター経由で接続できる場合、そのサイトのパスはオフラインになり、レプリケーション接続が戻るまで待機します。
- 優先されないサイトがパートナーに直接、または動作中のメディエーターを介してアクセスできない場合、パートナーが実際に使用できないか分離され、LUNパスがオフラインになったとみなされます。優先されないサイトは、RPO=0状態の解放に進み、読み取りI/Oと書き込みI/Oの両方の処理を続行します。レプリケーションソースの役割を引き継ぎ、新しい優先サイトになります。

メディエーターが完全に使用できない場合：

- 非優先サイトまたはストレージシステムの障害など、何らかの理由でレプリケーションサービスに障害が発生すると、優先サイトでRPO=0状態が解放され、読み取りおよび書き込みIO処理が再開されます。非優先サイトのパスがオフラインになります。
- 優先サイトに障害が発生すると、非優先サイトでは、反対側のサイトが本当にオフラインであることを確認できず、そのため非優先サイトがサービスを再開しても安全ではないため、システムが停止します。

サービスのリストア

サイト間の接続のリストアや障害が発生したシステムの電源投入などの障害が解決されると、SnapMirrorのアクティブな同期エンドポイントは、障害のあるレプリケーション関係の存在を自動的に検出してRPO=0状態に戻します。同期レプリケーションが再確立されると、障害が発生したパスは再びオンラインになります。

多くの場合、クラスタ化されたアプリケーションは障害が発生したパスの復帰を自動的に検出し、それらのアプリケーションもオンラインに戻ります。また、ホストレベルのSANスキャンが必要な場合や、アプリケーションを手動でオンラインに戻す必要がある場合もあります。それはアプリケーションとそれがどのように構成されているかによって異なり、一般的にそのようなタスクは簡単に自動化することができます。ONTAP自体は自己回復型であり、RPO=0のストレージ処理を再開するためにユーザの介入は不要です。

手動フェイルオーバー

優先サイトを変更するには、簡単な操作が必要です。クラスタ間でレプリケーション動作の権限が切り替わるため、IOは1~2秒間停止しますが、それ以外の場合はIOには影響しません。

著作権に関する情報

商標に関する情報