# Oracleのディザスタリカバリ

Enterprise applications

NetApp

February 10, 2026

# 目次


Oracleのディザスタリカバリ . . . 1
概要 . . . 1
SM-ASとMCCの比較 . . . 1
MetroCluster . . . 2
MetroClusterによるディザスタリカバリ . . . 2
物理アーキテクチャ . . . 2
論理アーキテクチャ . . . 6
SyncMirror . . . 12
MetroClusterとNVFAIL . . . 13
Oracleシングルインスタンス . . . 15
Oracle拡張RAC . . . 16
SnapMirrorアクティブ同期 . . . 20
概要 . . . 20
ONTAPメディエーター . . . 21
SnapMirrorアクティブ同期優先サイト . . . 23
ネットワークトポロジ . . . 24
Oracleの構成 . . . 31
障害シナリオ . . . 43

# Oracleのディザスタリカバリ

## 概要

ディザスタリカバリとは、火災によってストレージシステムやサイト全体が破壊されるなど、重大な災害が発生した場合にデータサービスをリストアすることです。

このドキュメントは、以前に公開されたテクニカルレポート_TR-4591：『Oracle Data Protection_and_TR-4592：Oracle on MetroCluster』を差し替えます。_

ディザスタリカバリは、もちろんSnapMirrorを使用してデータを単純にレプリケーションすることで実現できます。多くのお客様は、ミラーされたレプリカを1時間に何度も更新します。

ほとんどのお客様にとって、DRに必要なのはデータのリモートコピーだけではなく、そのデータを迅速に利用できることです。NetAppは、このニーズに対応する2つのテクノロジを提供します。MetroClusterとSnapMirrorのアクティブ同期です。

MetroClusterとは、低レベルの同期ミラーリングストレージと多数の追加機能を含むハードウェア構成のONTAPのことです。MetroClusterなどの統合ソリューションは、今日の複雑なスケールアウトデータベース、アプリケーション、仮想化インフラストラクチャを簡素化します。複数の外部データ保護製品や戦略を、1つのシンプルな中央集中型ストレージアレイに置き換えます。また、単一のクラスタストレージシステム内に、バックアップ、リカバリ、ディザスタリカバリ、高可用性（HA）が統合されています。

SnapMirrorアクティブ同期（SM-AS）はSnapMirror同期に基づいています。MetroClusterでは、各ONTAPコントローラがドライブデータをリモートサイトにレプリケートします。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。

### SM-ASとMCCの比較

SM-ASとMetroClusterは全体的な機能が似ていますが、RPO=0レプリケーションの実装方法と管理方法には重要な違いがあります。SnapMirrorの非同期および同期はDR計画の一部としても使用できますが、HAレプリケーションテクノロジとしては設計されていません。

- MetroCluster構成は、複数のサイトにノードが分散された統合クラスタのようなものです。SM-ASは、同期的にレプリケートされるRPO=0のLUNにサービスを提供する独立した2つのクラスタのように動作します。
- MetroCluster構成のデータには、常に1つの特定のサイトからしかアクセスできません。データの2つ目のコピーは反対側のサイトに存在しますが、データはパッシブです。ストレージシステムのフェイルオーバーがないとアクセスできません。
- MetroClusterとSM-ASによるミラーリングは、さまざまなレベルで実行されます。MetroClusterミラーリングはRAIDレイヤで実行されます。下位レベルのデータは、SyncMirrorを使用してミラーリングされた形式で格納されます。ミラーリングは、LUN、ボリューム、プロトコルの各レイヤでは実質的に使用されません。
- 一方、SM-ASミラーリングはプロトコルレイヤで行われます。2つのクラスタは、全体的に独立したクラスタです。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。

- MetroClusterの主な役割は大規模なレプリケーションです。RPO=0でRTOがほぼゼロのアレイ全体をレプリケートできます。フェイルオーバーが1つしかなく、容量とIOPSの点で非常に適切に拡張できるため、フェイルオーバープロセスが簡易化されます。
- 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。アプリケーションを両方のサイトで実行しておくことで、フェイルオーバー処理中の全体的なRTOを短縮できます。

# MetroCluster

## MetroClusterによるディザスタリカバリ

MetroClusterは、サイト間のRPO=0の同期ミラーリングでOracleデータベースを保護するONTAPの機能です。また、単一のMetroClusterシステムで数百のデータベースをサポートするまでスケールアップできます。

使い方も簡単です。MetroClusterを使用しても、エンタープライズアプリケーションやデータベースの運用に最適な条件が追加されたり変更されたりするとは限りません。

通常のベストプラクティスも引き続き適用され、必要なデータ保護がRPO=0の場合はMetroClusterで対応します。しかし、ほとんどのお客様は、RPO=0のデータ保護だけでなく、災害時のRTOを向上させ、サイトメンテナンス作業の一環として透過的なフェイルオーバーを実現するためにMetroClusterを使用しています。

## 物理アーキテクチャ

MetroCluster環境でのOracleデータベースの動作を理解するには、MetroClusterシステムの物理設計についてある程度の説明が必要です。

このドキュメントは、以前に公開されていたテクニカルレポート（TR-4592：『Oracle on MetroCluster』）に代わるものです。_

**MetroCluster**は**3**種類の構成で使用できます。

- IPセツソクノHAヘア
- FCセツソクノHAヘア
- シングルコントローラ、FC接続

「接続」という用語は、サイト間レプリケーションに使用されるクラスタ接続を指します。ホストプロトコルを指しているわけではありません。MetroCluster構成では、クラスタ間通信に使用される接続の種類に関係なく、すべてのホスト側プロトコルが通常どおりサポートされます。

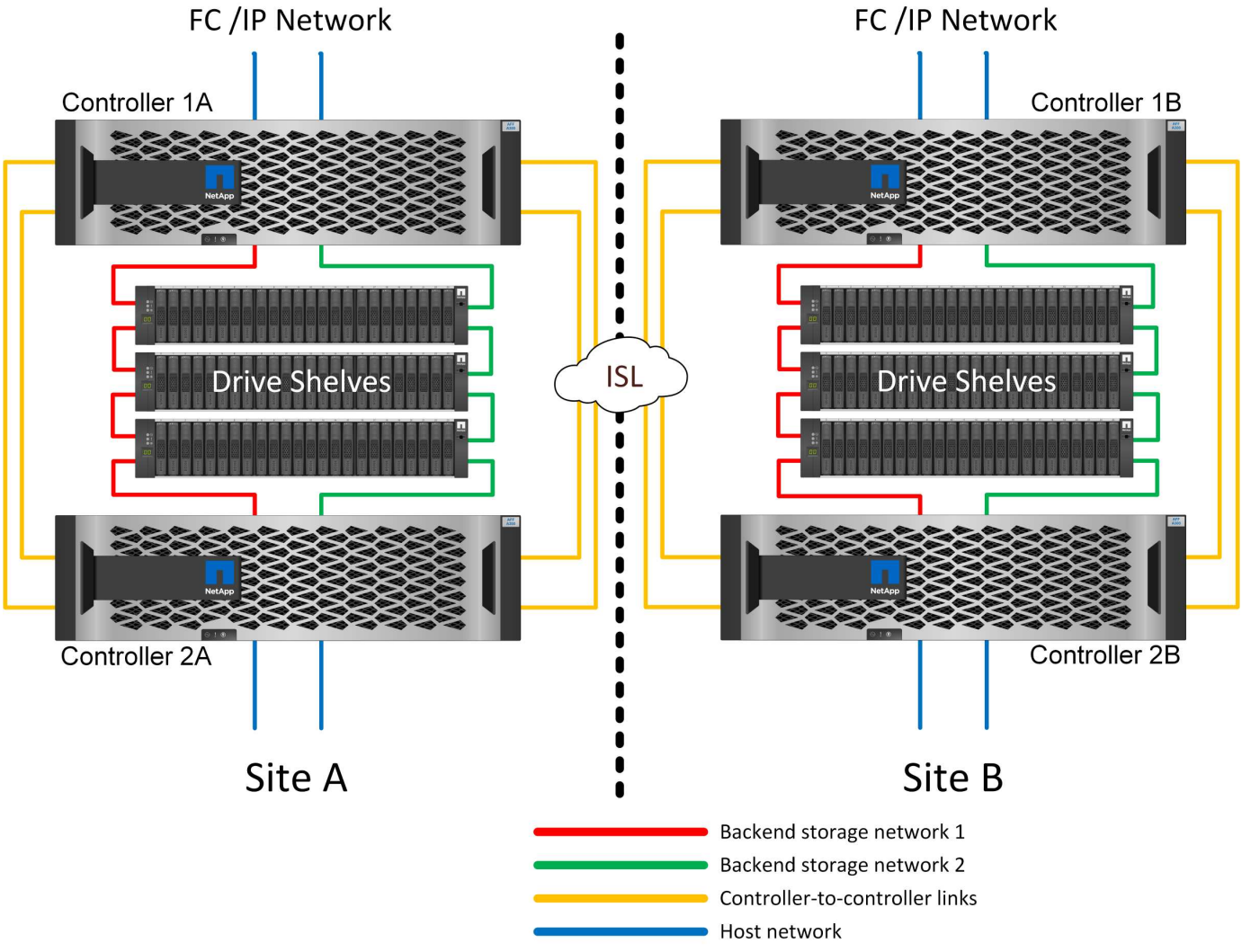
**MetroCluster IP** の略

HAペアMetroCluster IP構成では、サイトごとに2ノードまたは4ノードを使用します。この設定オプションを使用すると、2ノードオプションに比べて複雑さとコストが増加しますが、サイト内の冗長性という重要なメリットがあります。単純なコントローラ障害では、WAN経由のデータアクセスは必要ありません。データアクセスは、代替ローカルコントローラを介してローカルのままです。

ほとんどのお客様は、インフラストラクチャの要件がシンプルであるため、IP接続を選択しています。これまでは、ダークファイバやFCスイッチを使用した場合、サイト間での高速接続のプロビジョニングは一般的に容易でしたが、今日では、高速で低レイテンシのIP回線がより容易に利用可能になっています。

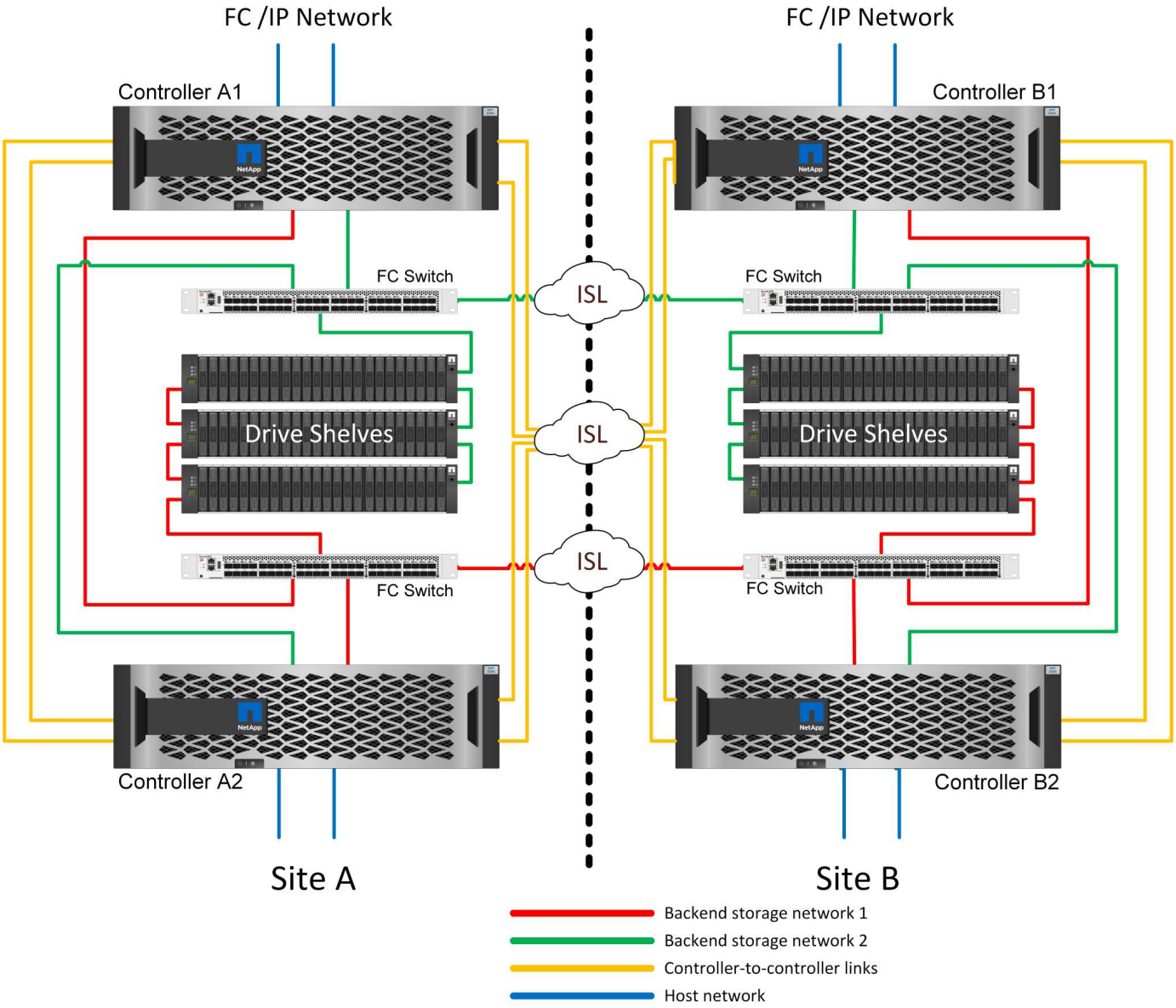
サイト間接続はコントローラのみであるため、アーキテクチャもシンプルです。FC SAN接続MetroClusterでは、コントローラが反対側サイトのドライブに直接書き込むため、追加のSAN接続、スイッチ、およびブリッジが必要になります。一方、IP構成のコントローラは、コントローラを介して反対側のドライブに書き込みます。

追加情報については、ONTAPの公式ドキュメントを参照してください。 "MetroCluster IP 解決策のアーキテクチャと設計"。



### HAペアFC SAN接続MetroCluster

HAペアMetroCluster FC構成では、サイトごとに2ノードまたは4ノードを使用します。この設定オプションを使用すると、2ノードオプションに比べて複雑さとコストが増加しますが、サイト内の冗長性という重要なメリットがあります。単純なコントローラ障害では、WAN経由のデータアクセスは必要ありません。データアクセスは、代替ローカルコントローラを介してローカルのままです。



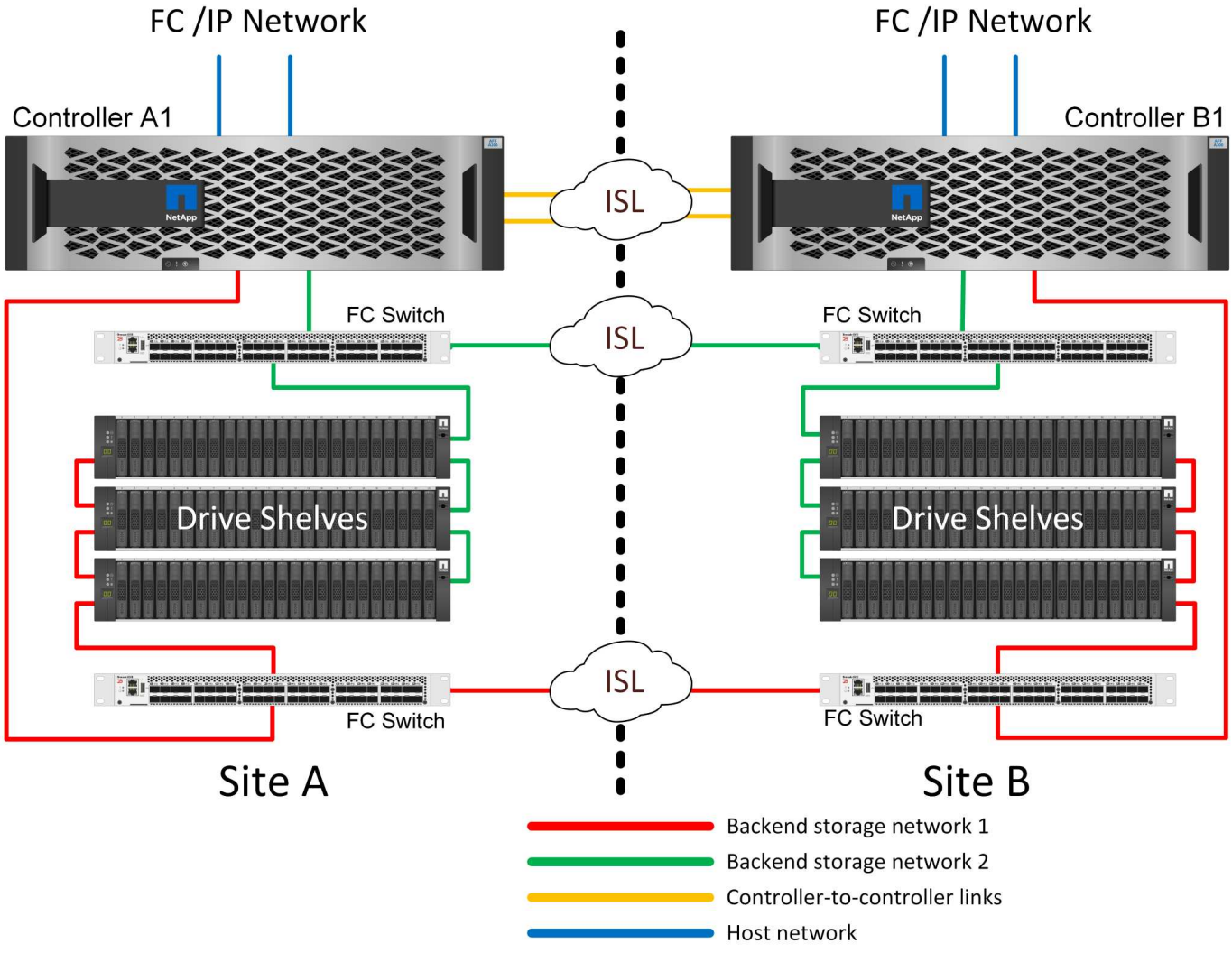

一部のマルチサイトインフラは、アクティブ/アクティブ運用向けに設計されたものではなく、プライマリサイトやディザスタリカバリサイトとして使用されます。この場合、一般にHAペアMetroClusterオプションが推奨される理由は次のとおりです。

- 2ノードMetroClusterクラスタはHAシステムですが、コントローラに予期しない障害が発生した場合や計画的メンテナンスを行う場合は、反対側のサイトでデータサービスをオンラインにする必要があります。サイト間のネットワーク接続が必要な帯域幅をサポートできない場合は、パフォーマンスが低下します。唯一の選択肢は、さまざまなホストOSと関連サービスを代替サイトにフェイルオーバーすることです。HAペアMetroClusterクラスタでは、コントローラが停止すると同じサイト内で単純なフェイルオーバーが発生するため、この問題は解消されます。

- 一部のネットワークトポロジは、サイト間アクセス用に設計されていませんが、異なるサブネットまたは分離されたFC SANを使用します。この場合、代替コントローラが反対側のサイトのサーバにデータを提供できないため、2ノードMetroClusterクラスタはHAシステムとして機能しなくなります。完全な冗長性を実現するには、HAペアMetroClusterオプションが必要です。
- 2サイトインフラを単一の高可用性インフラとみなす場合は、2ノードMetroCluster構成が適しています。ただし、サイト障害後もシステムが長時間機能しなければならない場合は、HAペアが推奨されます。HAペアは、単一サイト内でHAを提供し続けるためです。

### 2ノードFC SAN接続MetroCluster

2ノードMetroCluster構成では、サイトごとに1つのノードのみが使用されます。設定とメンテナンスが必要なコンポーネントが少ないため、HAペアオプションよりもシンプルな設計になっています。また、ケーブル配線やFCスイッチの点でインフラストラクチャの必要性も軽減されています。最後に、コストを削減します。




この設計の明らかな影響は、1つのサイトでコントローラに障害が発生した場合、反対側のサイトからデータを利用できることです。この制限は必ずしも問題ではありません。多くの企業は、本質的に単一のインフラとして機能する、拡張された高速で低レイテンシのネットワークを使用したマルチサイトデータセンター運用を行っています。このような場合は、2ノードバージョンのMetroClusterが推奨されます。2ノードシステムは現在、複数のサービスプロバイダでペタバイト規模で使用されています。

#### **MetroCluster**の耐障害性機能

MetroCluster 解決策 には単一点障害（Single Point of Failure）はありません。

- 各コントローラに、ローカルサイトのドライブシェルフへの独立したパスが2つあります。
- 各コントローラに、リモートサイトのドライブシェルフへの独立したパスが2つあります。
- 各コントローラには、反対側のサイトのコントローラへの独立したパスが2つあります。
- HAペア構成では、各コントローラからローカルパートナーへのパスが2つあります。

つまり、構成内のコンポーネントを1つでも削除しても、MetroClusterのデータ提供機能を損なうことはありません。2つのオプションの耐障害性の違いは、サイト障害後もHAペアバージョンが全体的なHAストレージシステムになる点だけです。

## 論理アーキテクチャ

### MetroCluster環境でOracleデータベースがどのように動作するかを理解するAlsopでは、MetroClusterシステムの論理機能について説明する必要があります。

#### サイト障害からの保護：**NVRAM**と**MetroCluster**

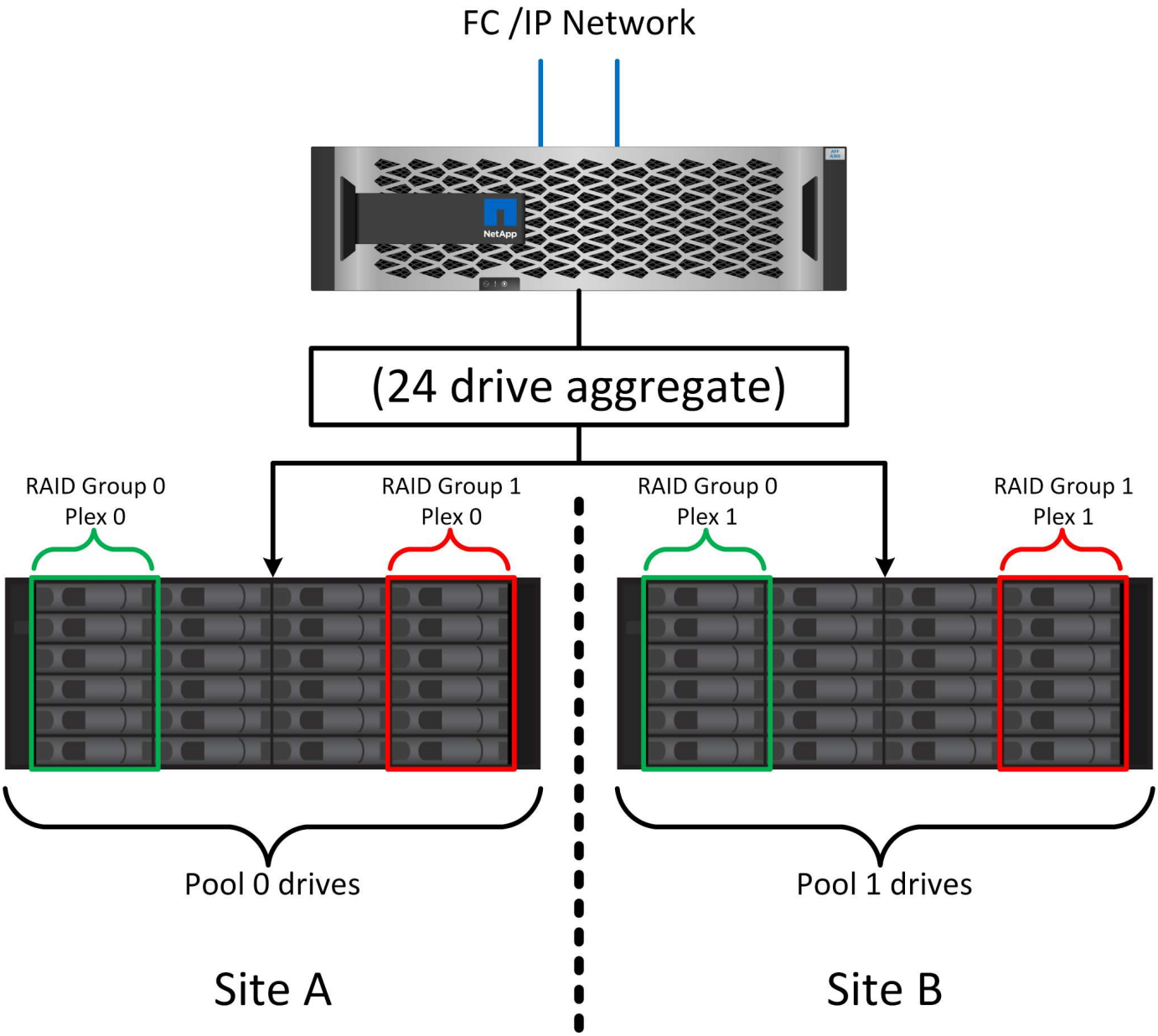
MetroClusterは、次の方法でNVRAMデータ保護を拡張します。

- 2ノード構成では、NVRAMデータがスイッチ間リンク（ISL）を使用してリモートパートナーにレプリケートされます。
- HAペア構成では、NVRAMデータがローカルパートナーとリモートパートナーの両方にレプリケートされます。
- 書き込みは、すべてのパートナーにレプリケートされるまで確認応答されません。このアーキテクチャは、NVRAMデータをリモートパートナーにレプリケートすることで、転送中のI/Oをサイト障害から保護します。このプロセスは、ドライブレベルのデータレプリケーションには関係ありません。アグリゲートを所有するコントローラは、アグリゲート内の両方のプレックスに書き込むことでデータレプリケーションを実行しますが、サイトが失われた場合でも転送中のI/Oの損失からデータを保護する必要があります。レプリケートされたNVRAMデータは、障害が発生したコントローラをパートナーコントローラがテイクオーバーする必要がある場合にのみ使用されます。

#### サイトおよびシェルフ障害からの保護：**SyncMirror**とプレックス

SyncMirrorは、RAID DPやRAID-TECを強化するミラーリングテクノロジですが、これに代わるものではありません。2つの独立したRAIDグループの内容をミラーリングします。論理構成は次のとおりです。

1. ドライブは、場所に基づいて2つのプールに構成されます。1つのプールはサイトAのすべてのドライブで構成され、2つ目のプールはサイトBのすべてのドライブで構成されます。
2. 次に、アグリゲートと呼ばれる共通のストレージプールが、RAIDグループのミラーセットに基づいて作成されます。各サイトから同じ数のドライブが引き出されます。たとえば、20ドライブのSyncMirrorアグリゲートは、サイトAの10本のドライブとサイトBの10本のドライブで構成されます。
3. サイト上の各ドライブセットは、ミラーリングを使用せずに、完全に冗長化された1つ以上のRAID DPグループまたはRAID-TECグループとして自動的に構成されます。ミラーリングの下でRAIDを使用することで、サイトが失われた場合でもデータを保護できます。




上の図は、SyncMirror構成の例を示しています。24ドライブのアグリゲートをコントローラに作成しました。このアグリゲートは、サイトAで割り当てられたシェルフの12本のドライブと、サイトBで割り当てられたシェルフの12本のドライブで構成されています。ドライブは2つのミラーRAIDグループにグループ化されました。RAIDグループ0には、サイトAの6ドライブのプレックスが含まれており、サイトBの6ドライブのプレックスにミラーリングされています。同様に、RAIDグループ1にはサイトAの6ドライブのプレックスが含まれており、サイトBの6ドライブのプレックスにミラーリングされています。

SyncMirrorは通常、MetroClusterシステムにリモートミラーリングを提供するために使用され、各サイトにデータのコピーが1つずつ配置されます。場合によっては、1つのシステムで追加レベルの冗長性を提供するために使用されます。特に、シェルフレベルの冗長性を提供します。ドライブシェルフにはすでにデュアル電源装置とコントローラが搭載されており、全体的には板金をほとんど使用していませんが、場合によっては追加の保護が保証されることがあります。たとえば、あるNetAppのお客様は、自動車テストで使用するモバイルリアルタイム分析プラットフォームにSyncMirrorを導入しています。システムは、独立した電源供給と独立したUPSシステムを備えた2つの物理ラックに分かれていました。

冗長性エラー：**NVFAIL**

前述したように、書き込みの確認応答は、少なくとも1台の他のコントローラでローカルのNVRAMとNVRAMに記録されるまで返されません。このアプローチにより、ハードウェア障害や停電が発生しても、転送中のI/Oが失われることはありません。ローカルのNVRAMに障害が発生したり、他のノードへの接続に障害が発生したりすると、データはミラーリングされなくなります。

ローカルNVRAMからエラーが報告されると、ノードはシャットダウンします。このシャットダウンにより、HAペアが使用されている場合はパートナーコントローラにフェイルオーバーされます。MetroClusterでは、動作は選択した全体的な設定によって異なりますが、リモートノートに自動的にフェイルオーバーされる場合があります。いずれの場合も、障害が発生したコントローラが書き込み処理を認識していないため、データは失われません。

リモートノードへのNVRAMレプリケーションがブロックされるサイト間接続障害は、より複雑な状況です。書き込みがリモートノードにレプリケートされなくなるため、コントローラで重大なエラーが発生した場合にデータが失われる可能性があります。さらに重要なことは、このような状況で別のノードにフェイルオーバーしようとするとデータが失われることです。

制御要素は、NVRAMが同期されているかどうかです。NVRAMが同期されていれば、ノード間のフェイルオーバーを安全に実行でき、データ損失のリスクはありません。MetroCluster構成では、NVRAMと基盤となるアグリゲートのプレックスが同期されていれば、データ損失のリスクなしにスイッチオーバーを実行できます。

データが同期されていない場合、ONTAPは、フェイルオーバーまたはスイッチオーバーを強制的に実行しないかぎり、フェイルオーバーまたはスイッチオーバーを許可しません。この方法で条件を変更すると、元のコントローラにデータが残っている可能性があり、データ損失が許容されることが確認されます。

データベースやその他のアプリケーションは、ディスク上のデータのより大きな内部キャッシュを保持するため、フェイルオーバーやスイッチオーバーを強制的に実行した場合に特に破損の影響を受けやすくなります。強制的なフェイルオーバーまたはスイッチオーバーが発生した場合、以前に確認済みの変更は事実上破棄されます。ストレージアレイの内容は実質的に時間を逆方向にジャンプし、キャッシュの状態はディスク上のデータの状態を反映しなくなります。

この状況を回避するために、ONTAPでは、NVRAMの障害に対する特別な保護をボリュームに設定できます。この保護メカニズムがトリガーされると、ボリュームがNVFAILという状態になります。この状態になると、原因アプリケーションがクラッシュするI/Oエラーが発生します。このクラッシュにより、古いデータを使用しないようにアプリケーションがシャットダウンされます。コミットされたトランザクションデータがログに含まれている必要があるため、データが失われないようにしてください。次の手順では、管理者がホストを完全にシャットダウンしてから、LUNとボリュームを手動で再度オンラインに戻します。これらの手順にはいくつかの作業が含まれる可能性がありますが、このアプローチはデータの整合性を確保するための最も安全な方法です。すべてのデータがこの保護を必要とするわけではありません。そのため、NVFAILの動作はボリューム単位で設定できます。

#### **HA**ペアと**MetroCluster**

MetroClusterには、2ノードとHAペアの2つの構成があります。2ノード構成の動作は、NVRAMに関してはHAペアと同じです。突然の障害が発生した場合、パートナーノードはNVRAMデータを再生してドライブの整合性を確保し、確認済みの書き込みが失われていないことを確認できます。

HAペア構成では、ローカルパートナーノードにもNVRAMがレプリケートされます。MetroClusterを使用しないスタンドアロンHAペアの場合と同様に、単純なコントローラ障害ではパートナーノードでNVRAMが再生されます。サイト全体が突然失われた場合、リモートサイトには、ドライブの整合性を確保してデータの提供を開始するために必要なNVRAMも用意されています。

MetroClusterの重要な側面の1つは、通常の運用状態ではリモートノードがパートナーデータにアクセスできないことです。各サイトは本質的に、反対のサイトのパーソナリティを想定できる独立したシステムとして機能します。このプロセスはスイッチオーバーと呼ばれ、計画的スイッチオーバーでは、サイトの処理が無停止で反対側のサイトに移行されます。また、サイトが失われ、ディザスタリカバリの一環として手動または自動のスイッチオーバーが必要になる計画外の状況も含まれます。

## スイッチオーバーとスイッチバック

スイッチオーバーとスイッチバックという用語は、MetroCluster構成のリモートコントローラ間でボリュームを移行するプロセスを指します。このプロセスでは、リモートノードのみが環境されます。4ボリューム構成でMetroClusterを使用する場合のローカルノードのフェイルオーバーは、前述したテイクオーバーとギブバックのプロセスと同じです。

### 計画的スイッチオーバーとスイッチバック

計画的スイッチオーバーまたはスイッチバックは、ノード間のテイクオーバーやギブバックと似ています。このプロセスには複数の手順があり、数分かかるように見える場合もありますが、実際には、ストレージリソースとネットワークリソースを複数のフェーズで正常に移行します。完全なコマンドの実行に必要な時間よりもはるかに短時間で制御転送が行われる瞬間。

テイクオーバー/ギブバックとスイッチオーバー/スイッチバックの主な違いは、FC SAN接続への影響です。ローカルのテイクオーバー/ギブバックでは、ローカルノードへのFCパスがすべて失われ、ホストのネイティブMPIOを使用して使用可能な代替パスに切り替えます。ポートは再配置されません。スイッチオーバーとスイッチバックでは、コントローラの仮想FCターゲットポートがもう一方のサイトに移行します。一時的にSAN上に存在しなくなり、代わりのコントローラに再表示されます。

#### **SyncMirror**タイムアウト

SyncMirrorは、シェルフ障害から保護するONTAPのミラーリングテクノロジです。シェルフが離れた場所に配置されている場合は、リモートデータ保護が実現します。

SyncMirrorは汎用同期ミラーリングを提供しません。その結果、可用性が向上します。一部のストレージシステムでは、一定のオールオアナッシングミラーリング（Dominoモードと呼ばれることもあります）を使用します。リモートサイトへの接続が失われるとすべての書き込みアクティビティが停止する必要があるため、この形式のミラーリングはアプリケーションで制限されます。そうしないと、書き込みは一方のサイトに存在し、もう一方のサイトには存在しません。通常、このような環境では、サイト間の接続が短時間（30秒など）以上切断された場合にLUNがオフラインになるように構成されます。

この動作は、一部の環境に適しています。ただし、ほとんどのアプリケーションには、通常の動作条件下で保証された同期レプリケーションを提供しながら、レプリケーションを一時停止できる解決策が必要です。サイト間の接続が完全に失われると、多くの場合、災害が近い状況とみなされます。通常、このような環境は、接続が修復されるか、データを保護するために環境をシャットダウンする正式な決定が下されるまで、オンラインのままでデータを提供します。リモートレプリケーションの障害のみが原因でアプリケーションを自動的にシャットダウンする必要があるのは珍しいことです。

SyncMirrorは、タイムアウトの柔軟性を備えた同期ミラーリングの要件に対応しています。リモートコントローラやプレックスへの接続が失われると、30秒のタイマーがカウントダウンを開始します。カウンタが0に達すると、ローカルデータを使用して書き込みI/O処理が再開されます。データのリモートコピーは使用可能ですが、接続が回復するまで時間内に凍結されます。再同期では、アグリゲートレベルのSnapshotを使用してシステムをできるだけ迅速に同期モードに戻します。

特に、多くの場合、この種の汎用的なオールオアナッシングDominoモードレプリケーションは、アプリケーションレイヤでより適切に実装されています。たとえば、Oracle DataGuardには最大保護モードが用意されており、どのような状況でも長時間のインスタンスレプリケーションが保証されます。設定可能なタイムアウ

トを超えてレプリケーションリンクに障害が発生すると、データベースはシャットダウンします。

#### ファブリック接続**MetroCluster**による自動無人スイッチオーバー

Automatic Unattended Switchover（AUSO；自動無人スイッチオーバー）は、クロスサイトHAの形式を提供するファブリック接続MetroClusterの機能です。前述したように、MetroClusterには2つのタイプ（各サイトに1台のコントローラを配置する場合と、各サイトに1台のHAペアを配置する場合）があります。HAオプションの主な利点は、コントローラの計画的シャットダウンと計画外シャットダウンのどちらでもすべてのI/Oをローカルで処理できることです。シングルノードオプションのメリットは、コスト、複雑さ、インフラの削減です。

AUSOの主な価値は、ファブリック接続MetroClusterシステムのHA機能を向上させることです。各サイトが反対側のサイトの健常性を監視し、データを提供するノードがなくなると、AUSOによって迅速なスイッチオーバーが実行されます。このアプローチは、可用性の点でHAペアに近い構成になるため、サイトごとにノードが1つだけのMetroCluster構成で特に役立ちます。

AUSOでは、HAペアレベルで包括的な監視を行うことはできません。HAペアには、ノード間の直接通信用の2本の冗長な物理ケーブルが含まれているため、きわめて高い可用性を実現できます。さらに、HAペアの両方のノードが冗長ループ上の同じディスクセットにアクセスできるため、1つのノードが別のノードの健常性を監視するための別のルートが提供されます。

MetroClusterクラスタは複数のサイトにまたがって存在し、ノード間の通信とディスクアクセスの両方がサイト間ネットワーク接続に依存します。クラスタの残りの部分のハートビートを監視する機能には制限があります。AUSOは、ネットワークの問題が原因で、もう一方のサイトが使用できない状況ではなく、実際にダウンしている状況を区別する必要があります。

その結果、HAペアのコントローラで、システムパニックなどの特定の理由で発生したコントローラ障害が検出された場合、テイクオーバーが要求されることがあります。また、接続が完全に失われた場合（ハートビートの損失とも呼ばれます）、テイクオーバーを促すこともあります。

MetroClusterシステムで自動スイッチオーバーを安全に実行できるのは、元のサイトで特定の障害が検出された場合のみです。また、ストレージシステムの所有権を取得するコントローラは、ディスクとNVRAMのデータが同期されていることを保証できる必要があります。コントローラは、ソースサイトとの通信が失われて稼働している可能性があるため、スイッチオーバーの安全性を保証できません。スイッチオーバーを自動化するためのその他のオプションについては、次のセクションのMetroCluster Tiebreaker（MCTB）解決策に関する情報を参照してください。

#### ファブリック接続**MetroCluster**を使用した**MetroCluster Tiebreaker**

この"NetApp MetroCluster Tiebreaker"ソフトウェアを第3のサイトで実行すると、MetroCluster環境の健全性を監視し、通知を送信できます。また、災害時にオプションでスイッチオーバーを強制的に実行することもできます。Tiebreakerの詳細についてはを参照して"NetApp Support Site"ください。MetroCluster Tiebreakerの主な目的はサイトの損失を検出することです。また、サイトの損失と接続の損失を区別する必要があります。たとえば、Tiebreakerがプライマリサイトに到達できなかったためにスイッチオーバーが発生しないようにします。そのため、Tiebreakerはリモートサイトがプライマリサイトに接続する能力も監視します。

AUSOによる自動スイッチオーバーもMCTBと互換性があります。AUSOは、特定の障害イベントを検出し、NVRAMとSyncMirrorのプレックスが同期されている場合にのみスイッチオーバーを実行するように設計されているため、非常に迅速に対応します。

一方、Tiebreakerはリモートに配置されているため、サイトの停止を宣言する前にタイマーが経過するのを待つ必要があります。Tiebreakerは最終的にAUSOの対象となるコントローラ障害を検出しますが、一般的にはAUSOがスイッチオーバーを開始しており、Tiebreakerが機能する前にスイッチオーバーを完了している可能性があります。Tiebreakerから送信される2つ目のswitchoverコマンドは拒否されます。

MCTBソフトウェアは、強制的なスイッチオーバー時に、NVRAM WASまたはプレックス（あるいはその両方）が同期されていることを検証しません。メンテナンス作業中に自動スイッチオーバーが設定されている場合は無効にして、NVRAMまたはSyncMirrorプレックスの同期が失われるようにしてください。

また、MCTBは、次の一連のイベントにつながるローリングディザスタに対応できない場合があります。

1. サイト間の接続が30秒以上中断されます。
2. SyncMirrorレプリケーションがタイムアウトし、プライマリサイトで処理が続行されるため、リモートレプリカは古くなります。
3. プライマリサイトが失われます。その結果、プライマリサイトにレプリケートされていない変更が存在します。その場合、次のようないくつかの理由でスイッチオーバーが望ましくない可能性があります。
   - 重要なデータはプライマリサイトに存在し、最終的にリカバリ可能になる可能性があります。スイッチオーバーによってアプリケーションの動作が継続されると、重要なデータは実質的に破棄されます。
   - サバイバーサイトのアプリケーションで、サイト障害時にプライマリサイトのストレージリソースを使用していた場合、データがキャッシュされている可能性があります。スイッチオーバーでは、キャッシュと一致しない古いバージョンのデータが生成されます。
   - サバイバーサイトのオペレーティングシステムで、サイト障害時にプライマリサイトのストレージリソースを使用していた場合、キャッシュデータがある可能性があります。スイッチオーバーでは、キャッシュと一致しない古いバージョンのデータが生成されます。最も安全な方法は、Tiebreakerがサイト障害を検出した場合にアラートを送信するように設定し、スイッチオーバーを強制的に実行するかどうかを決定することです。キャッシュされたデータを消去するには、アプリケーションやオペレーティングシステムのシャットダウンが必要になる場合があります。さらに、NVFAIL設定を使用して保護を強化し、フェイルオーバープロセスを合理化することもできます。

#### **MetroCluster IP**を使用した**ONTAP**メディエーター

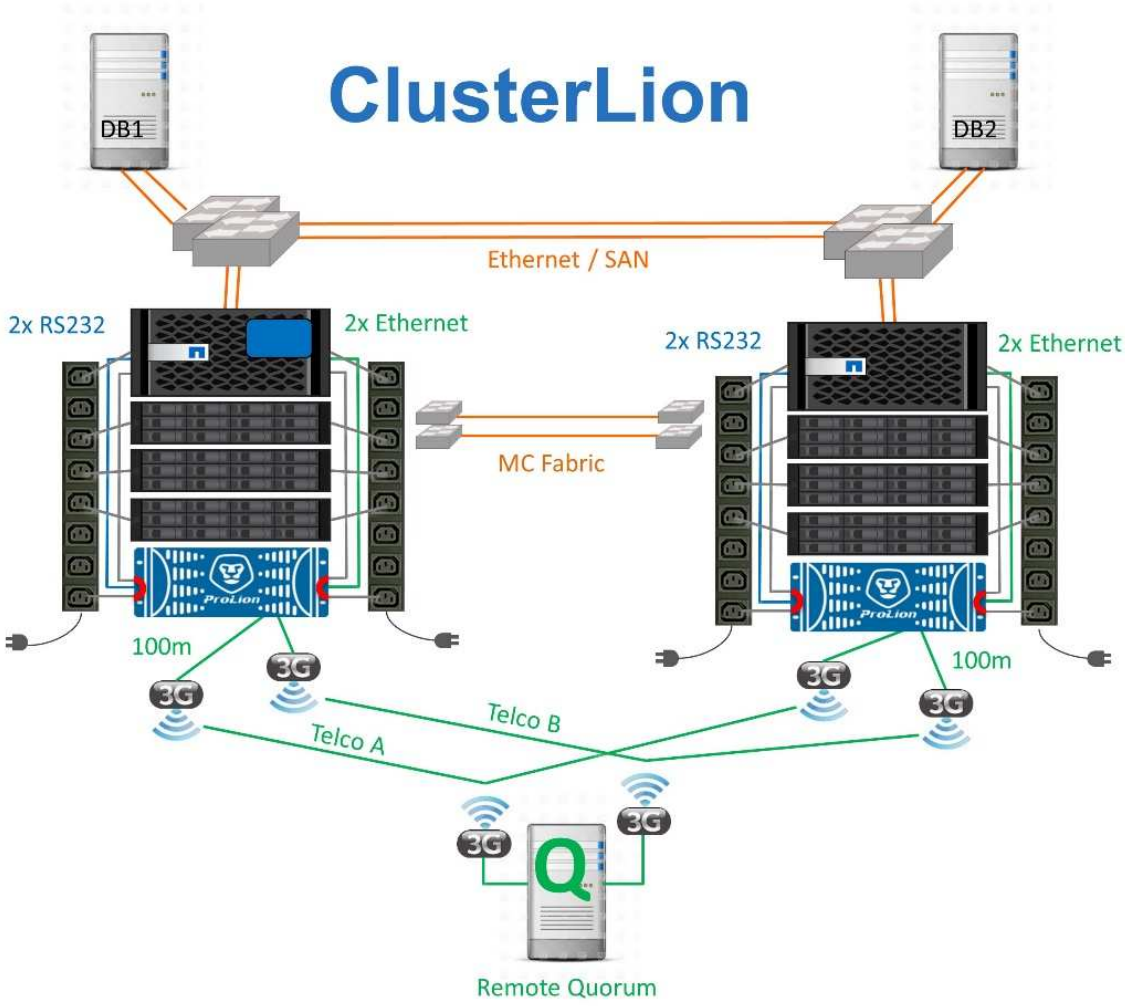
ONTAPメディエーターは、MetroCluster IPおよびその他の特定のONTAPソリューションで使用されます。これは、前述のMetroCluster Tiebreakerソフトウェアと同様に従来のTiebreakerサービスとして機能しますが、重要な機能を実行する自動無人スイッチオーバーも含まれています。

ファブリック接続MetroClusterは、反対側のサイトのストレージデバイスに直接アクセスできます。これにより、一方のMetroClusterコントローラがドライブからハートビートデータを読み取ることで、他のコントローラの健常性を監視できます。これにより、一方のコントローラがもう一方のコントローラの障害を認識し、スイッチオーバーを実行できるようになります。

一方、MetroCluster IPアーキテクチャでは、すべてのI/Oがコントローラとコントローラの接続を介して排他的にルーティングされるため、リモートサイトのストレージデバイスに直接アクセスすることはありません。これにより、コントローラで障害を検出してスイッチオーバーを実行する機能が制限されます。そのため、サイトの損失を検出して自動的にスイッチオーバーを実行するためには、ONTAPメディエーターがTiebreakerデバイスとして必要になります。

#### **ClusterLion**を使用した**3**番目の仮想サイト

ClusterLionは、仮想の第3サイトとして機能する高度なMetroCluster監視アプライアンスです。このアプローチにより、完全に自動化されたスイッチオーバー機能により、MetroClusterを2サイト構成で安全に導入できます。さらに、ClusterLionでは、追加のネットワークレベル監視を実行し、スイッチオーバー後の処理を実行できます。完全なドキュメントはProLionから入手できます。




- ClusterLionアプライアンスは、直接接続されたイーサネットケーブルとシリアルケーブルでコントローラの健常性を監視します。
- 2つのアプライアンスは、冗長3Gワイヤレス接続で相互に接続されています。
- ONTAPコントローラへの電源は、内部リレーを介して配線されます。サイト障害が発生すると、内部UPSシステムを搭載したClusterLionによって電源接続が切断されてからスイッチオーバーが実行されます。このプロセスにより、スプリットブレイン状態が発生しないようにします。
- ClusterLionは、30秒のSyncMirrorタイムアウト内にスイッチオーバーを実行するか、まったく実行しません。
- ClusterLionでは、NVRAMプレックスとSyncMirrorプレックスの状態が同期されていないかぎり、スイッチオーバーは実行されません。
- ClusterLionでは、MetroClusterが完全に同期されている場合にのみスイッチオーバーが実行されるため、NVFAILは必要ありません。この構成では、計画外スイッチオーバーが発生しても、拡張Oracle RACなどのサイトスパニング環境をオンラインのまま維持できます。
- ファブリック接続MetroClusterとMetroCluster IPの両方をサポート

## SyncMirror

MetroClusterシステムを使用したOracleデータ保護の基盤となるのは、最大パフォーマンスのスケールアウト同期ミラーリングテクノロジであるSyncMirrorです。

### SyncMirrorによるデータ保護

最も簡単な意味では、同期レプリケーションとは、変更がミラーされたストレージの両側に対して確認応答される前に行われなければならないことを意味します。たとえば、データベースがログを書き込んでいる場合やVMwareゲストにパッチを適用している場合は、書き込みが失われることはありません。プロトコルレベルでは、両方のサイトの不揮発性メディアにコミットされるまで、ストレージシステムは書き込みを確認応答しないでください。その場合にのみ、データ損失のリスクなしに作業を安全に進めることができます。

同期レプリケーションテクノロジの使用は、同期レプリケーション解決策を設計および管理するための最初のステップです。最も重要な考慮事項は、計画的および計画外のさまざまな障害シナリオで何が発生するかを理解することです。すべての同期レプリケーションソリューションが同じ機能を提供するわけではありません。Recovery Point Objective（RPO；目標復旧時点）がゼロ（つまりデータ損失ゼロ）の解決策が必要な場合は、すべての障害シナリオを考慮する必要があります。特に、サイト間の接続が失われてレプリケーションが不可能になった場合、どのような結果が予想されますか。

### SyncMirrorデータの可用性

MetroClusterレプリケーションは、同期モードに効率的に切り替えられるように設計されたNetApp SyncMirrorテクノロジに基づいています。この機能は、同期レプリケーションを必要とする一方で、データサービスに高可用性も必要とするお客様の要件を満たします。たとえば、リモートサイトへの接続が切断されている場合は、通常、ストレージシステムをレプリケートされていない状態で運用し続けることを推奨します。

多くの同期レプリケーションソリューションは、同期モードでしか動作できません。このタイプのall-or-nothingレプリケーションは、Dominoモードと呼ばれることがあります。このようなストレージシステムでは、データのローカルコピーとリモートコピーが非同期になるのではなく、データの提供が停止します。レプリケーションが強制的に解除された場合、再同期には非常に時間がかかり、ミラーリングの再確立中にデータが完全に失われる可能性があります。

リモートサイトに到達できない場合にSyncMirrorを同期モードからシームレスに切り替えることができるだけでなく、接続がリストアされたときにRPO=0状態に迅速に再同期することもできます。再同期中にリモートサイトにある古いデータコピーを使用可能な状態で保持することもできるため、データのローカルコピーとリモートコピーを常に維持できます。

Dominoモードが必要な場合、NetAppはSnapMirror Synchronous（SM-S）を提供します。Oracle DataGuardやSQL Server Always On可用性グループなど、アプリケーションレベルのオプションも用意されています。オプションとして、OSレベルのディスクミラーリングを使用できます。追加情報とオプションについては、担当のNetAppまたはパートナーアカウントチームにお問い合わせください。

## MetroClusterとNVFAIL

NVFailはONTAPの一般的なデータ整合性機能で、データベースを使用してデータ整合性を最大限に保護するように設計されています。

このセクションでは、基本的なONTAP NVFAILについて説明し、MetroCluster固有のトピックを扱います。

MetroClusterでは、少なくとも1台の他のコントローラのローカルNVRAMとNVRAMに書き込みが記録されるまで、書き込み確認は行われません。このアプローチにより、ハードウェア障害や停電が発生しても、転送中のI/Oが失われることはありません。ローカルのNVRAMに障害が発生したり、他のノードへの接続に障害が発生したりすると、データはミラーリングされなくなります。

ローカルNVRAMからエラーが報告されると、ノードはシャットダウンします。このシャットダウンによ

り、HAペアが使用されている場合はパートナーコントローラにフェイルオーバーされます。MetroClusterでは、動作は選択した全体的な設定によって異なりますが、リモートノートに自動的にフェイルオーバーされる場合があります。いずれの場合も、障害が発生したコントローラが書き込み処理を認識していないため、データは失われません。

リモートノードへのNVRAMレプリケーションがブロックされるサイト間接続障害は、より複雑な状況です。書き込みがリモートノードにレプリケートされなくなるため、コントローラで重大なエラーが発生した場合にデータが失われる可能性があります。さらに重要なことは、このような状況で別のノードにフェイルオーバーしようとするとデータが失われることです。

制御要素は、NVRAMが同期されているかどうかです。NVRAMが同期されていれば、ノード間のフェイルオーバーを安全に実行でき、データ損失のリスクはありません。MetroCluster構成では、NVRAMと基盤となるアグリゲートのプレックスが同期されていれば、データ損失のリスクなしにスイッチオーバーを安全に実行できます。

データが同期されていない場合、ONTAPは、フェイルオーバーまたはスイッチオーバーを強制的に実行しないかぎり、フェイルオーバーまたはスイッチオーバーを許可しません。この方法で条件を変更すると、元のコントローラにデータが残っている可能性があり、データ損失が許容されることが確認されます。

データベースは、ディスク上のデータのより大きな内部キャッシュを保持するため、フェイルオーバーやスイッチオーバーを強制的に実行した場合、データベースが破損する可能性が特に高くなります。強制的なフェイルオーバーまたはスイッチオーバーが発生した場合、以前に確認済みの変更は事実上破棄されます。ストレージアレイの内容は実質的に時間を逆方向に移動し、データベースキャッシュの状態はディスク上のデータの状態を反映しなくなります。

この状況からアプリケーションを保護するために、ONTAPでは、NVRAMの障害に対する特別な保護をボリュームに設定できます。この保護メカニズムがトリガーされると、ボリュームがNVFAILという状態になります。この状態になると、古いデータを使用しないように原因アプリケーションをシャットダウンするI/Oエラーが発生します。確認済みの書き込みはストレージシステムに残っているため、データが失われることはありません。データベースの場合は、コミットされたトランザクションデータがログに含まれている必要があります。

次の手順では、管理者がホストを完全にシャットダウンしてから、LUNとボリュームを手動で再度オンラインに戻します。これらの手順にはいくつかの作業が含まれる可能性がありますが、このアプローチはデータの整合性を確保するための最も安全な方法です。すべてのデータがこの保護を必要とするわけではありません。そのため、NVFAILの動作はボリューム単位で設定できます。

### 手動強制**NVFAIL**

サイト間に分散されているアプリケーションクラスタ（VMware、Oracle RACなど）でスイッチオーバーを強制的に実行する最も安全なオプションは、 `-force-nvfail-all` コマンドラインで。このオプションは、キャッシュされたすべてのデータが確実にフラッシュされるようにするための緊急措置として使用できます。災害が発生したサイトにもともと配置されていたストレージリソースをホストが使用している場合、I/Oエラーまたは古いファイルハンドルのいずれかを受信します。（`ESTALE`）エラー。Oracleデータベースがクラッシュし、ファイルシステムが完全にオフラインになるか、読み取り専用モードに切り替わります。

スイッチオーバーの完了後、 `in-nvfailed-state` フラグをクリアし、LUNをオンラインにする必要があります。このアクティビティが完了したら、データベースを再起動できます。これらのタスクを自動化してRTOを短縮できます。

### dr-force-nvfail

一般的な安全対策として、 `dr-force-nvfail` 通常運用時にリモートサイトからアクセスされる可能性があるすべてのボリューム（フェイルオーバー前に使用されるアクティビティ）にフラグを付けます。この設定に

より、選択したリモートボリュームが `in-nvfailed-state` スイッチオーバー中。スイッチオーバーの完了後、 `in-nvfailed-state` フラグをクリアし、LUNをオンラインにする必要があります。これらのアクティビティが完了したら、アプリケーションを再起動できます。これらのタスクを自動化してRTOを短縮できます。

結果は、 `-force-nvfail-all` 手動スイッチオーバーのフラグ。ただし、影響を受けるボリュームの数は、古いキャッシュを使用するアプリケーションまたはオペレーティングシステムから保護する必要があるボリュームだけに制限される場合があります。

を使用しない環境には、次の2つの重要な要件があります。 `dr-force-nvfail` アプリケーションボリューム：

- 強制スイッチオーバーは、プライマリサイトの障害から30秒以内に実行する必要があります。
- メンテナンスタスクの実行中や、SyncMirrorプレックスやNVRAMレプリケーションが同期されていないその他の状況では、スイッチオーバーを実行しないでください。最初の要件を満たすには、Tiebreakerソフトウェアを使用します。Tiebreakerソフトウェアは、サイト障害から30秒以内にスイッチオーバーを実行するように設定されています。これは、サイト障害が検出されてから30秒以内にスイッチオーバーを実行する必要があるという意味ではありません。これは、サイトが動作していることが確認されてから30秒が経過した場合に強制的にスイッチオーバーを実行しても安全ではないことを意味します。

2つ目の要件は、MetroCluster構成が同期されていないことが判明した場合に、自動スイッチオーバー機能をすべて無効にすることで部分的に満たすことができます。NVRAMレプリケーションとSyncMirrorプレックスの健常性を監視できるTiebreaker解決策を使用することを推奨します。クラスタが完全に同期されていない場合、Tiebreakerはスイッチオーバーをトリガーしません。

NetApp MCTBソフトウェアは同期ステータスを監視できないため、何らかの理由でMetroClusterが同期されていない場合は無効にする必要があります。ClusterLionにはNVRAM監視機能とプレックス監視機能が搭載されており、MetroClusterシステムが完全に同期されていることが確認されないかぎり、スイッチオーバーをトリガーしないように設定できます。

## Oracleシングルインスタンス

前述したように、MetroClusterシステムが存在しても、データベースの運用に関するベストプラクティスが必ずしも追加されたり変更されたりするわけではありません。お客様のMetroClusterシステムで現在実行されているデータベースの大部分はシングルインスタンスであり、Oracle on ONTAPドキュメントに記載されている推奨事項に従っています。

### 事前設定されたOSを使用したフェイルオーバー

SyncMirrorはディザスタリカバリサイトにデータの同期コピーを提供しますが、そのデータを利用できるようにするには、オペレーティングシステムと関連するアプリケーションが必要です。基本的な自動化により、環境全体のフェイルオーバー時間を大幅に短縮できます。Veritas Cluster Server（VCS）などのClusterware製品は、サイト間でクラスタを作成するためによく使用されます。多くの場合、フェイルオーバープロセスは単純なスクリプトで実行できます。

プライマリノードが失われた場合、代替サイトでデータベースをオンラインにするようにクラスタウェア（またはスクリプト）が設定されます。1つは、データベースを構成するNFSリソースまたはSANリソース用に事前設定されたスタンバイサーバを作成する方法です。プライマリサイトに障害が発生すると、クラスタウェアまたはスクリプト化された代替サイトが次のような一連の処理を実行します。

1. MetroClusterスイッチオーバーの強制実行
2. FC LUNの検出の実行（SANのみ）
3. ファイルシステムのマウント、ASMディスクグループのマウント
4. データベースの起動

このアプローチの主な要件は、リモートサイトでOSを実行することです。Oracleバイナリを使用して事前に設定する必要があります。つまり、Oracleのパッチ適用などのタスクをプライマリサイトとスタンバイサイトで実行する必要があります。また、災害が発生した場合は、Oracleバイナリをリモートサイトにミラーリングしてマウントすることもできます。

実際のアクティベーション手順は簡単です。LUN検出などのコマンドでは、FCポートあたりのコマンド数が少なくて済みます。ファイル・システムのマウントは' `mount` コマンドを実行し、データベースとASMの両方をCLIで1つのコマンドで起動および停止できます。スイッチオーバーの前にディザスタリカバリサイトでボリュームとファイルシステムを使用していない場合は、 `dr-force- nvfail` ボリューム：

仮想**OS**によるフェイルオーバー

データベース環境のフェイルオーバーを拡張して、オペレーティングシステム自体を含めることができます。理論的には、このフェイルオーバーはブートLUNで実行できますが、ほとんどの場合、仮想OSで実行されます。手順の手順は次のようになります。

1. MetroClusterスイッチオーバーの強制実行
2. データベースサーバ仮想マシンをホストするデータストアのマウント
3. 仮想マシンの起動
4. データベースを手動で起動するか、データベースを自動的に起動するように仮想マシンを設定します。たとえば、ESXクラスタが複数のサイトにまたがっている場合があります。災害が発生した場合は、スイッチオーバー後にディザスタリカバリサイトで仮想マシンをオンラインにすることができます。災害発生時に仮想データベースサーバをホストするデータストアが使用されていないかぎり、 `dr-force- nvfail` 関連付けられているボリューム。

## Oracle拡張RAC

多くのお客様が、Oracle RACクラスタを複数のサイトにまたがって構成し、完全なアクティブ/アクティブ構成を実現することで、RTOを最適化しています。Oracle RACのクォーラム管理を含める必要があるため、設計全体が複雑になります。また、データは両方のサイトからアクセスされるため、強制的スイッチオーバーによって古いデータコピーが使用される可能性があります。

データのコピーは両方のサイトに存在しますが、データを提供できるのはアグリゲートを現在所有しているコントローラだけです。そのため、拡張RACクラスタでは、リモートのノードがサイト間接続でI/Oを実行する必要があります。その結果、I/Oレイテンシが増加しますが、このレイテンシは一般的には問題になりません。RACインターコネクトネットワークは複数のサイトにまたがって拡張する必要があるため、とにかく高速で低レイテンシのネットワークが必要です。レイテンシが増加して原因に問題が発生した場合は、クラスタをアクティブ/パッシブで運用できます。I/O負荷の高い処理は、アグリゲートを所有するコントローラに対してローカルなRACノードに対して実行する必要があります。リモートノードは、より軽いI/O処理を実行するか、純粋にウォームスタンバイサーバとして使用されます。

アクティブ/アクティブ拡張RACが必要な場合は、MetroClusterの代わりにSnapMirrorアクティブ同期を検討す

る必要があります。SM-ASレプリケーションでは、データの特定のレプリカを優先的に使用できます。したがって、すべての読み取りがローカルに行われる拡張RACクラスタを構築できます。読み取りI/Oがサイトを経由することはないため、レイテンシは最小限に抑えられます。すべての書き込みアクティビティは引き続きサイト間接続を転送する必要がありますが、このようなトラフィックは同期ミラーリング解決策では回避できません。

> 仮想ブートディスクを含むブートLUNをOracle RACで使用する場合は、 `misscount`パラメータの変更が必要になることがあります。RACタイムアウトパラメータの詳細については、を参照してください"ONTAPを使用したOracle RAC"。

**2**サイト構成

2サイトの拡張RAC構成では、すべてではないが多くの災害シナリオに無停止で対応できるアクティブ/アクティブデータベースサービスを提供できます。

**RAC**投票ファイル

MetroClusterに拡張RACを導入する場合は、クォーラム管理を最初に検討する必要があります。Oracle RACには、クォーラムを管理するための2つのメカニズム（ディスクハートビートとネットワークハートビート）があります。ディスクハートビートは、投票ファイルを使用してストレージアクセスを監視します。単一サイトのRAC構成では、基盤となるストレージシステムがHA機能を提供していれば、単一の投票リソースで十分です。

以前のバージョンのOracleでは、投票ファイルは物理ストレージデバイスに配置されていましたが、現在のバージョンのOracleでは、投票ファイルはASMディスクグループに格納されていました。

> Oracle RACはNFSでサポートされています。グリッドのインストールプロセスでは、一連のASMプロセスが作成され、グリッドファイルに使用されるNFSの場所がASMディスクグループとして提供されます。このプロセスはエンドユーザに対してほぼ透過的であり、インストール完了後にASMを継続的に管理する必要はありません。

2サイト構成の最初の要件は、無停止のディザスタリカバリプロセスを保証する方法で、各サイトが常に半数以上の投票ファイルにアクセスできるようにすることです。このタスクは、投票ファイルがASMディスクグループに格納される前は簡単でしたが、今日の管理者はASM冗長性の基本原則を理解する必要があります。

ASMディスクグループには3つの冗長性オプションがあります。 `external`、 `normal`および `high`。つまり、ミラーリングされていない、ミラーリングされている、3方向ミラーリングされているということです。という新しいオプションがあります。 `Flex` 利用可能ですが、めったに使用されません。冗長デバイスの冗長性レベルと配置によって、障害が発生した場合の動作が制御されます。例：

- 投票ファイルをに配置する `diskgroup` を使用 `external` 冗長性リソースを使用すると、サイト間接続が失われた場合に一方のサイトの削除が保証されます。
- 投票ファイルをに配置する `diskgroup` を使用 `normal` 各サイトにASMディスクが1つしかない冗長性を確保すると、どちらのサイトにもマジョリティクォーラムが存在しないためにサイト間接続が失われた場合に、両方のサイトでノードが削除されます。
- 投票ファイルをに配置する `diskgroup` を使用 `high` 一方のサイトに2本のディスクを配置し、もう一方のサイトに1本のディスクを配置する冗長性により、両方のサイトが動作していて相互にアクセスできる場合にアクティブ/アクティブ処理が可能になります。ただし、シングルディスクサイトがネットワークから分離されている場合、そのサイトは削除されます。

**RAC**ネットワークハートビート

Oracle RACネットワークハートビートは、クラスタインターコネクト経由でノードに到達できるかどうかを監視します。クラスタに残すには、あるノードが他のノードの半数以上にアクセスできる必要があります。この要件により、2サイトアーキテクチャのRACノード数は次のように選択されます。

- サイトごとに同じ数のノードを配置すると、ネットワーク接続が失われた場合に1つのサイトが削除されます。
- 一方のサイトにN個のノードを配置し、もう一方のサイトにN+1個のノードを配置すると、サイト間接続が失われてネットワーククォーラムに残っているノードの数が多くなり、削除するノードの数が少なくなります。

Oracle 12cR2より前のバージョンでは、サイト障害時にどの側で削除するかを制御することは不可能でした。各サイトのノード数が同じ場合、削除はマスターノード（通常は最初にブートするRACノード）によって制御されます。

Oracle 12cR2では、ノードの重み付け機能が導入されています。この機能により、管理者はOracleによるスプリットブレイン状態の解決方法をより細かく制御できます。簡単な例として、次のコマンドはRAC内の特定のノードの優先順位を設定します。

```
[root@host-a ~]# /grid/bin/crsctl set server css_critical yes
CRS-4416: Server attribute 'CSS_CRITICAL' successfully changed. Restart
Oracle High Availability Services for new value to take effect.
```

Oracle High-Availability Servicesを再起動すると、構成は次のようになります。

```
[root@host-a lib]# /grid/bin/crsctl status server -f | egrep
'^NAME|CSS_CRITICAL='
NAME=host-a
CSS_CRITICAL=yes
NAME=host-b
CSS_CRITICAL=no
```

ノード `host-a` が重要なサーバとして指定されました。2つのRACノードが分離されている場合は、 `host-a` 生き残って `host-b` 削除されます。

詳細については、Oracleのホワイトペーパー『Oracle Clusterware 12c Release 2 Technical Overview』を参照してください。」

12cR2より前のバージョンのOracle RACでは、CRSログを確認することでマスターノードを特定できます。

```
[root@host-a ~]# /grid/bin/crsctl status server -f | egrep
'^NAME|CSS_CRITICAL='
NAME=host-a
CSS_CRITICAL=yes
NAME=host-b
CSS_CRITICAL=no
 [root@host-a ~]# grep -i 'master node' /grid/diag/crs/host-
a/crs/trace/crsd.trc
2017-05-04 04:46:12.261525 :   CRSSE:2130671360: {1:16377:2} Master Change
Event; New Master Node ID:1 This Node's ID:1
2017-05-04 05:01:24.979716 :   CRSSE:2031576832: {1:13237:2} Master Change
Event; New Master Node ID:2 This Node's ID:1
2017-05-04 05:11:22.995707 :   CRSSE:2031576832: {1:13237:221} Master
Change Event; New Master Node ID:1 This Node's ID:1
2017-05-04 05:28:25.797860 :   CRSSE:3336529664: {1:8557:2} Master Change
Event; New Master Node ID:2 This Node's ID:1
```

このログは、マスターノードが 2 ノード host-a ID： 1。これはつまり host-a はマスターノードではありません。マスターノードのIDは、コマンドで確認できます。 olsnodes -n。

```
[root@host-a ~]# /grid/bin/olsnodes -n
host-a  1
host-b  2
```

IDがのノード 2 はです host-b`をクリックします。これはマスターノードです。各サイトに同じ数のノードがある構成では、 `host-b 2つのセットが何らかの理由でネットワーク接続を失った場合に存続するサイトです。

マスターノードを識別するログエントリがシステムから期限切れになる可能性があります。この場合、Oracle Cluster Registry（OCR）バックアップのタイムスタンプを使用できます。

```
[root@host-a ~]#  /grid/bin/ocrconfig -showbackup
host-b     2017/05/05 05:39:53     /grid/cdata/host-cluster/backup00.ocr
0
host-b     2017/05/05 01:39:53     /grid/cdata/host-cluster/backup01.ocr
0
host-b     2017/05/04 21:39:52     /grid/cdata/host-cluster/backup02.ocr
0
host-a     2017/05/04 02:05:36     /grid/cdata/host-cluster/day.ocr     0
host-a     2017/04/22 02:05:17     /grid/cdata/host-cluster/week.ocr     0
```

次の例では、マスターノードが host-b。また、マスターノードの変更も示します。 host-a 終了： host-b 5月4日の2時5分から21時39分までの間。マスターノードを識別するこの方法は、前回のOCRバックアップ以降にマスターノードが変更されている可能性があるため、CRSログもチェックされている場合にのみ使用で

きます。この変更が発生した場合は、OCRログに表示されます。

ほとんどのお客様は、環境全体と各サイトで同数のRACノードにサービスを提供する投票ディスクグループを1つ選択しています。ディスクグループは、データベースが格納されているサイトに配置する必要があります。接続が失われると、リモートサイトが削除されます。リモートサイトにはクォーラムがなくなり、データベースファイルにもアクセスできなくなりますが、ローカルサイトは通常どおり稼働し続けます。接続が回復したら、リモートインスタンスを再びオンラインにすることができます。

災害が発生した場合は、サバイバーサイトでデータベースファイルと投票ディスクグループをオンラインにするためにスイッチオーバーが必要です。災害によってAUSOでスイッチオーバーがトリガーされた場合、クラスタが同期されていてストレージリソースが正常にオンラインになるため、NVFAILはトリガーされません。AUSOは非常に高速な操作であり、 `disktimeout` 有効期限が切れます。

サイトが2つしかないため、自動化された外部タイブレークソフトウェアを使用することは不可能であり、強制スイッチオーバーは手動で行う必要があります。

**3**サイト構成

3つのサイトで拡張RACクラスタを構築する方がはるかに簡単です。MetroClusterシステムの各半分をホストする2つのサイトもデータベースワークロードをサポートし、3つ目のサイトはデータベースとMetroClusterシステムの両方のTiebreakerとして機能します。Oracle Tiebreakerの構成は、第3のサイトに投票に使用するASMディスクグループのメンバーを配置するだけで簡単に構成できます。また、RACクラスタに奇数のノードを配置するために、第3のサイトに運用インスタンスを配置することもできます。

> ⓘ 拡張RAC構成でNFSを使用する場合の重要な情報については、「クォーラム障害グループ」に関するOracleのドキュメントを参照してください。要するに、クォーラムリソースをホストする3番目のサイトへの接続が失われても、プライマリOracleサーバまたはOracle RACプロセスが停止しないように、NFSマウントオプションを変更してsoftオプションを含める必要がある場合があります。

# SnapMirrorアクティブ同期

## 概要

SnapMirror Active Syncを使用すると、非常に高可用性のOracleデータベース環境を構築できます。この環境では、2つの異なるストレージクラスタからLUNを使用できます。

SnapMirrorのアクティブな同期では、データの「プライマリ」コピーと「セカンダリ」コピーはありません。各クラスタはデータのローカルコピーから読み取りIOを提供でき、各クラスタはパートナーに書き込みをレプリケートします。その結果、対称IOビヘイビアが作成されます。

これにより、Oracle RACを両方のサイトで運用インスタンスを持つ拡張クラスタとして実行できます。または、RPO=0のアクティブ/パッシブデータベースクラスタを構築して、サイト停止中にシングルインスタンスデータベースをサイト間で移動できます。このプロセスは、PacemakerやVMware HAなどの製品を使用して自動化できます。これらすべてのオプションの基盤となるのは、SnapMirror Active Syncで管理される同期レプリケーションです。

同期レプリケーション

通常の運用では、1つの例外を除いて、SnapMirrorアクティブ同期は常にRPO=0同期レプリカを提供します。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。

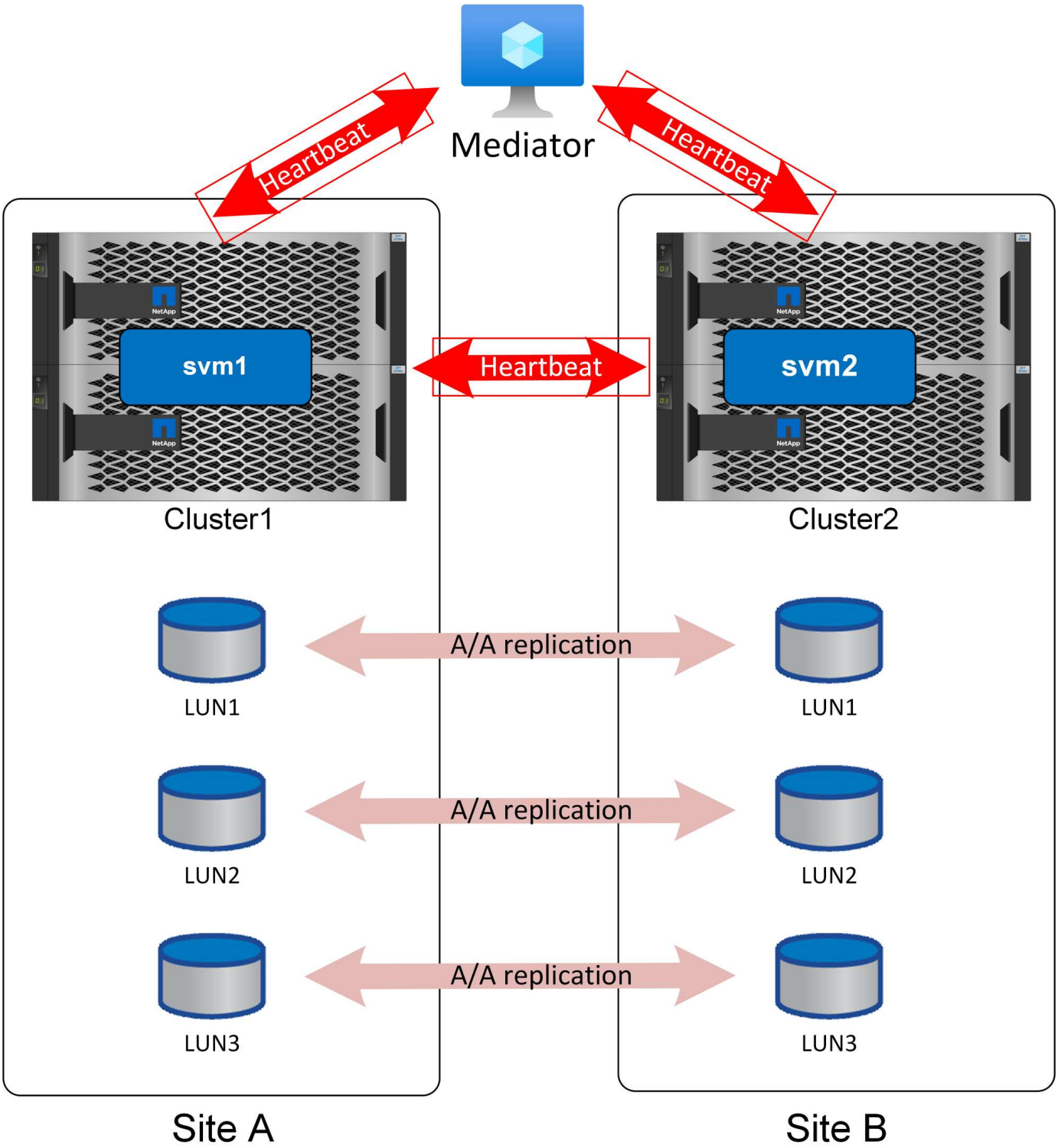
**ONTAP**メディエーター

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、SnapMirrorのアクティブな同期ではTiebreakerになりません。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。

調停者は実際にはタイブレーカーではありませんが、実質的にはそれがその機能を提供します。メディエーターは、クラスター ノードの状態を判断するのに役立ち、サイト障害が発生した場合の自動切り替えプロセスを支援します。Mediator はいかなる状況でもデータを転送しません。




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。

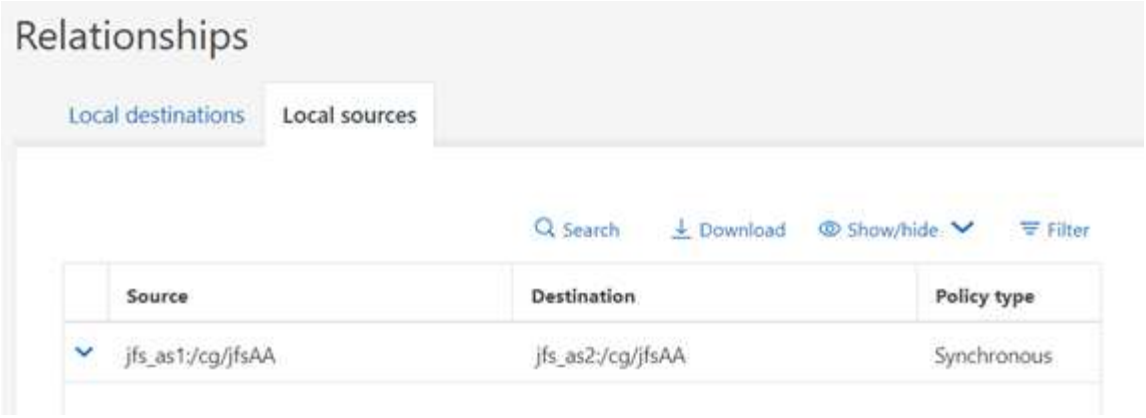
## SnapMirrorアクティブ同期優先サイト

### 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。

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。

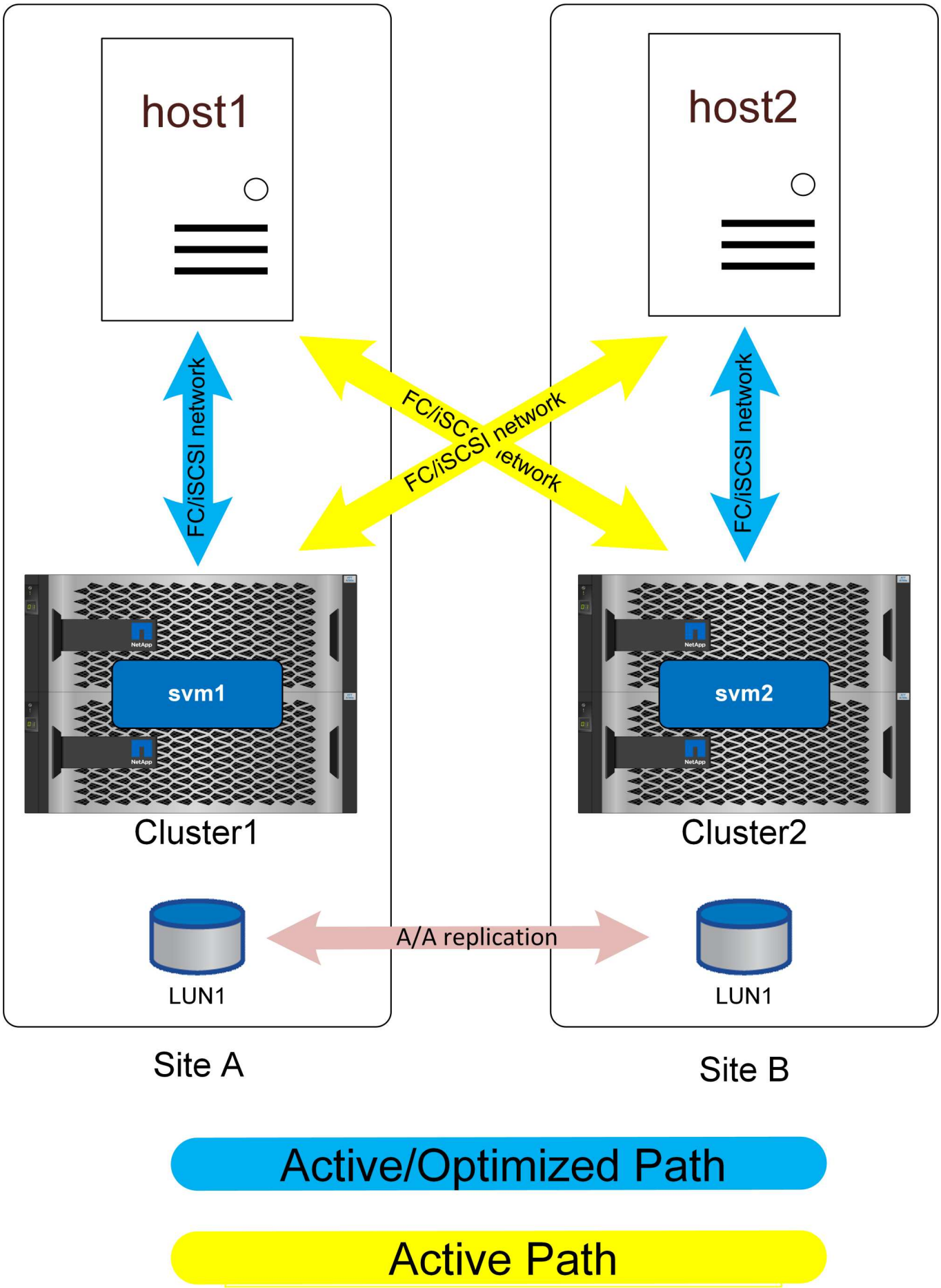
2番目のオプション手順は、ホストプロキシミティを設定することです。これはアクセスを制御するのではなく、_priority_を制御します。

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。

**AFF**

AFFシステムで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
svm2
Cluster1
Cluster2
A/A replication
LUN1
LUN1
Site A
Site B
Active/Optimized Path
Active Path

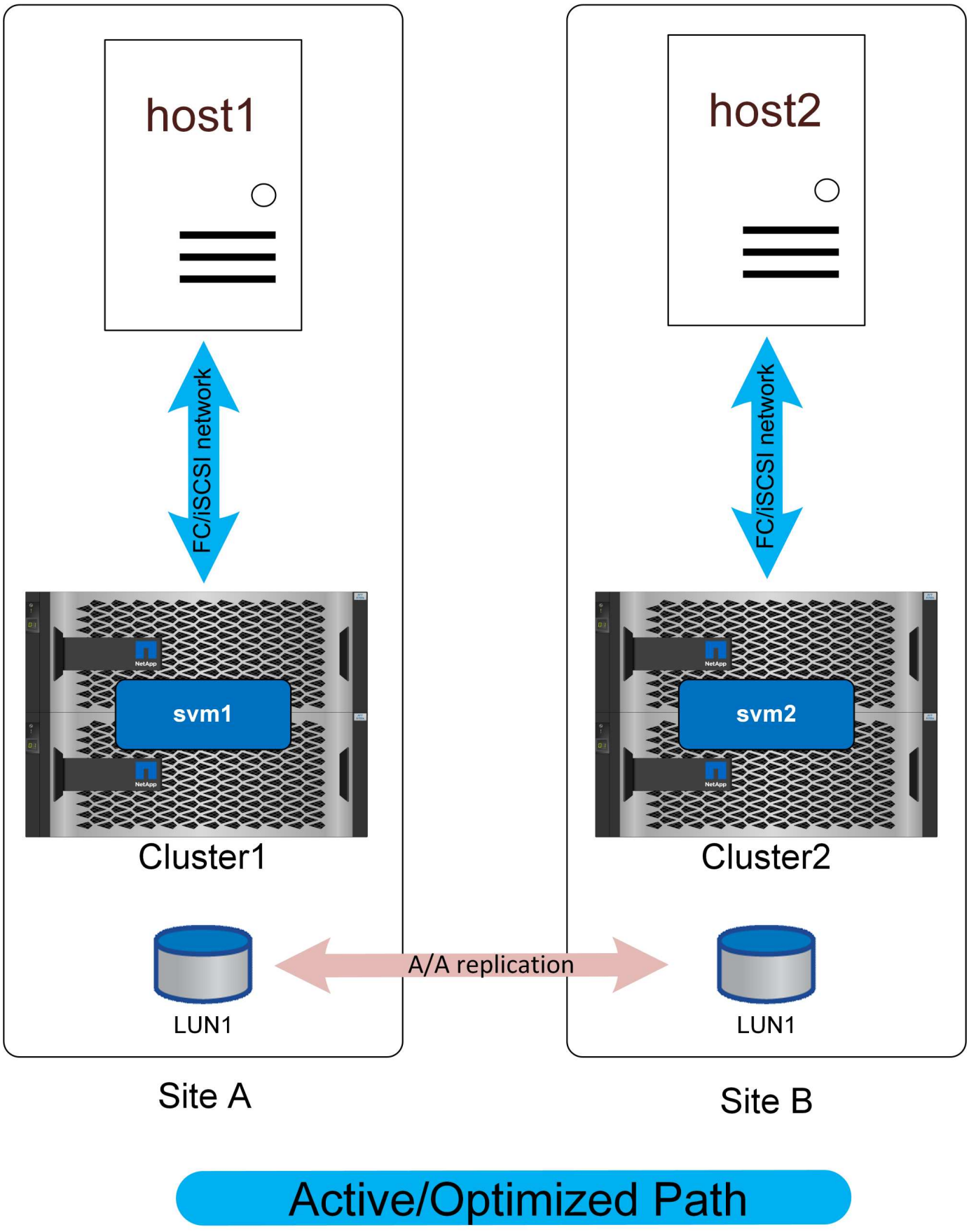
通常の運用では、すべてのIOがローカルIOになります。読み取りと書き込みはローカルストレージアレイから処理されます。もちろん、書き込みIOも確認応答の前にローカルコントローラでリモートシステムにレプリケートする必要がありますが、すべての読み取りIOはローカルで処理されるため、サイト間のSANリンクを経由して余分なレイテンシが発生することはありません。

非最適パスが使用されるのは、すべてのアクティブ/最適パスが失われた場合だけです。たとえば、サイトAのアレイ全体に電力が供給されなくなっても、サイトAのホストはサイトBのアレイへのパスに引き続きアクセスできるため、レイテンシは高くなりますが運用を継続できます。

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。

#### ASA

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。



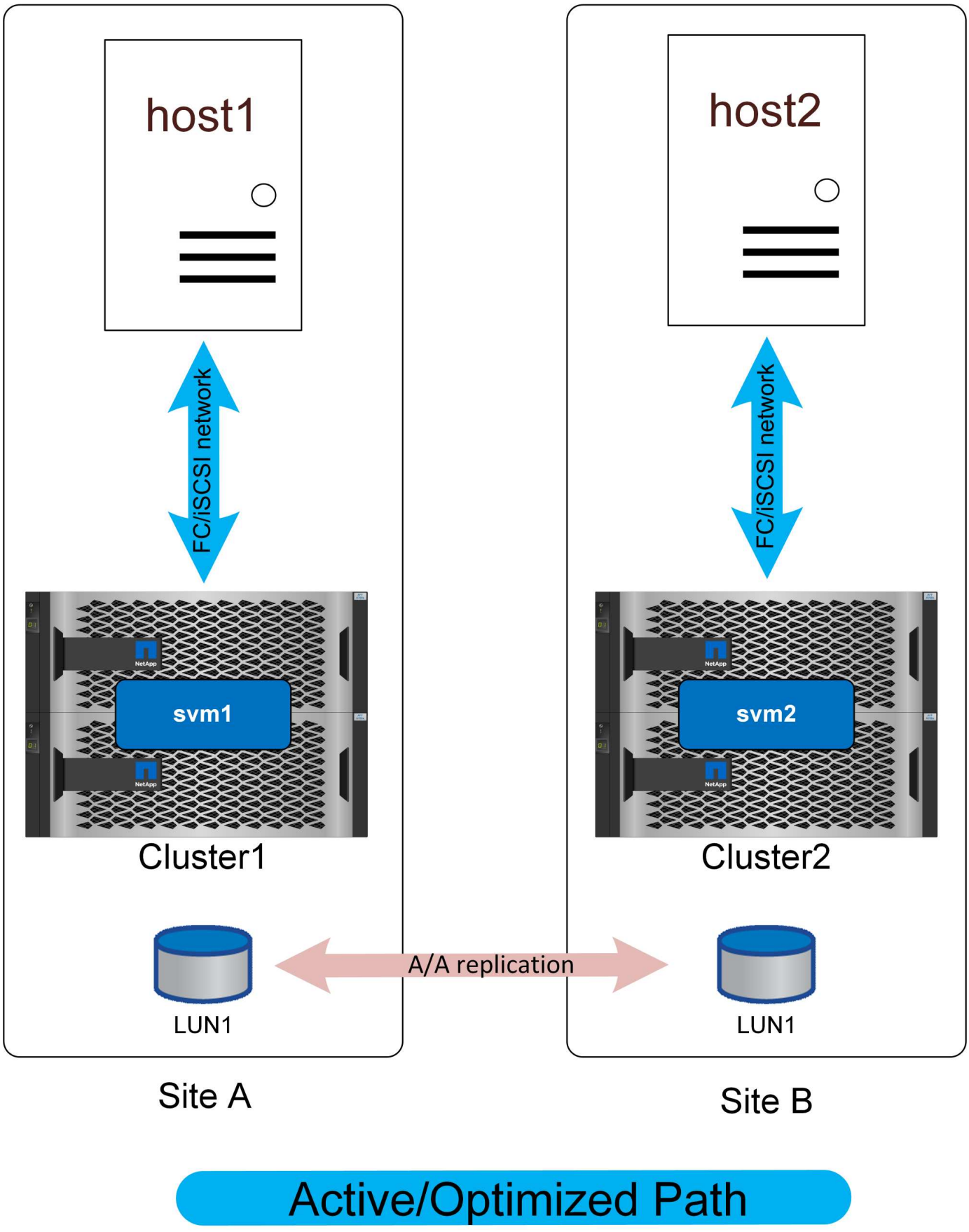

アクセスが不均一なASA構成は、AFFの場合とほとんど同じように機能します。アクセスが統一されている場合、IOはWANを通過します。これは望ましい場合とそうでない場合があります。

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。対照的に、AFFでは、これらのWAN交差パスは使用可能なローカルパスがない場合にのみ使用され、すべてのIOがローカルIOになるため、日 々 のパフォーマンスが向上します。不均一なアクセスネットワークを使用するASAは、サイト間の遅延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得るためのオプションです。

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。

不均一なアクセス

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。

## Oracleの構成

概要

### SnapMirrorアクティブ同期を使用しても、データベースの運用に関するベストプラクティスが追加または変更されるとは限りません。

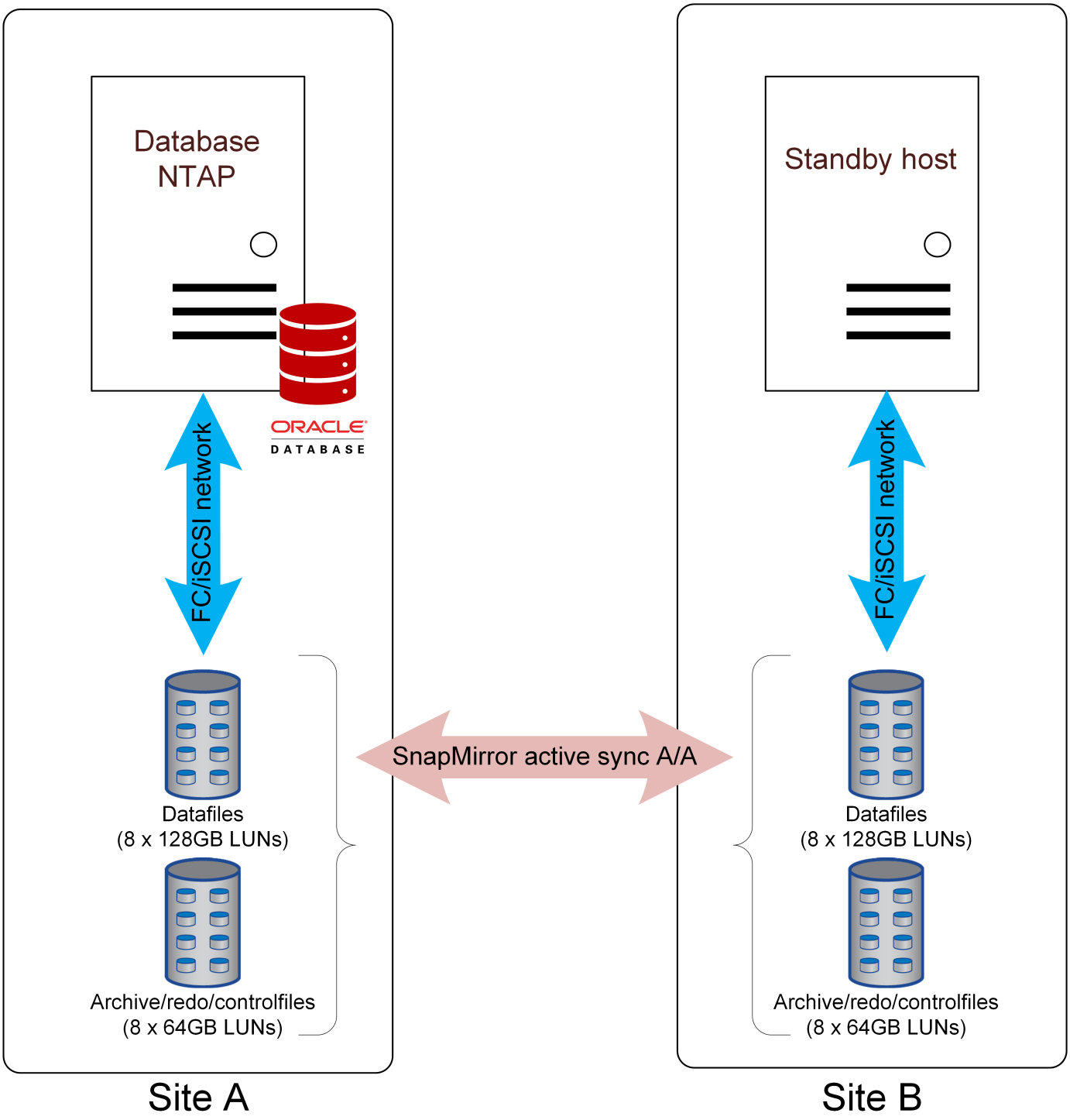
最適なアーキテクチャは、ビジネス要件によって異なります。たとえば、データ損失に対するRPO=0の保護が目標であるにもかかわらず、RTOが緩和されている場合は、Oracleのシングルインスタンスデータベースを使用し、SM-ASでLUNをレプリケートすれば十分であり、Oracleのライセンスの問題からより安価になる可能性があります。リモートサイトに障害が発生しても運用は中断されず、プライマリサイトが停止すると、サバイバーサイトのLUNはオンラインで使用可能な状態になります。

RTOの方が厳しい場合は、スクリプトやPacemakerやAnsibleなどのクラスタウェアを使用した基本的なアクティブ/パッシブ自動化を使用すると、フェイルオーバー時間が短縮されます。たとえば、プライマリサイトでVMの障害を検出し、リモートサイトでVMをアクティブにするようにVMware HAを設定できます。

最後に、きわめて高速なフェイルオーバーを実現するために、Oracle RACを複数のサイトに導入できます。データベースは常にオンラインで両方のサイトで利用できるため、RTOは基本的にゼロになります。

#### Oracleシングルインスタンス

### 以下に説明する例は、SnapMirrorアクティブ同期レプリケーションを使用してOracleシングルインスタンスデータベースを導入するための多数のオプションの一部を示しています。




事前設定された**OS**を使用したフェイルオーバー

SnapMirror Active Syncはディザスタリカバリサイトにデータの同期コピーを作成しますが、そのデータを利用できるようにするには、オペレーティングシステムと関連するアプリケーションが必要です。基本的な自動化により、環境全体のフェイルオーバー時間を大幅に短縮できます。PacemakerなどのClusterware製品は、サイト間でクラスタを作成するためによく使用されます。多くの場合、フェイルオーバープロセスは単純なスクリプトで実行できます。

プライマリノードが失われると、クラスタウェア（またはスクリプト）によって代替サイトでデータベースがオンラインになります。1つは、データベースを構成するSANリソース用に事前設定されたスタンバイサーバを作成する方法です。プライマリサイトに障害が発生すると、クラスタウェアまたはスクリプト化された代替

サイトが次のような一連の処理を実行します。

1. プライマリサイトの障害を検出
2. FCまたはiSCSI LUNの検出の実行
3. ファイルシステムのマウント、ASMディスクグループのマウント
4. データベースの起動

このアプローチの主な要件は、リモートサイトでOSを実行することです。Oracleバイナリを使用して事前に設定する必要があります。つまり、Oracleのパッチ適用などのタスクをプライマリサイトとスタンバイサイトで実行する必要があります。また、災害が発生した場合は、Oracleバイナリをリモートサイトにミラーリングしてマウントすることもできます。

実際のアクティベーション手順は簡単です。LUN検出などのコマンドでは、FCポートあたりのコマンド数が少なくて済みます。ファイルシステムのマウントはコマンドにすぎませ `mount`ん。データベースとASMの両方を、1つのコマンドでCLIから開始および停止できます。

#### 仮想OSによるフェイルオーバー

データベース環境のフェイルオーバーを拡張して、オペレーティングシステム自体を含めることができます。理論的には、このフェイルオーバーはブートLUNで実行できますが、ほとんどの場合、仮想OSで実行されます。手順の手順は次のようになります。

1. プライマリサイトの障害を検出
2. データベースサーバ仮想マシンをホストするデータストアのマウント
3. 仮想マシンの起動
4. データベースを手動で起動するか、仮想マシンでデータベースが自動的に起動するように設定します。

たとえば、ESXクラスタが複数のサイトにまたがっているとします。災害が発生した場合は、スイッチオーバー後にディザスタリカバリサイトで仮想マシンをオンラインにすることができます。

#### ストレージ障害からの保護

上の図は"不均一なアクセス"、の使用方法を示しています。SANが複数のサイトにまたがっているわけではありません。これは設定が簡単で、現在のSAN機能では唯一の選択肢となる場合もありますが、プライマリストレージシステムに障害が発生すると、アプリケーションがフェイルオーバーされるまでデータベースが停止します。

耐障害性を高めるために、このソリューションをとともに導入することもでき"均一なアクセス"ます。これにより、アプリケーションは、反対側のサイトからアドバタイズされたパスを使用して動作を継続できます。

### Oracle拡張RAC

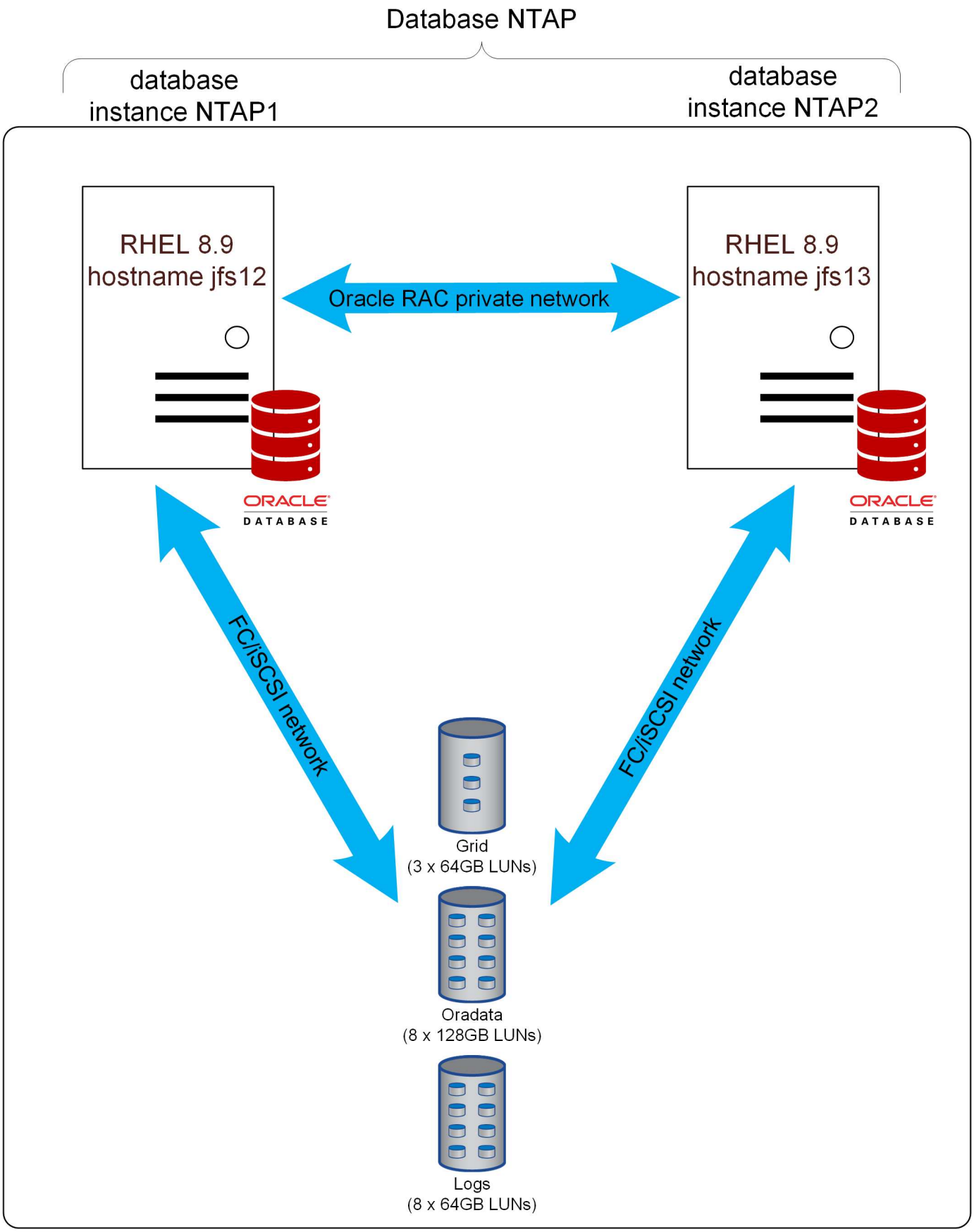
多くのお客様が、Oracle RACクラスタを複数のサイトにまたがって構成し、完全なアクティブ/アクティブ構成を実現することで、RTOを最適化しています。Oracle RACのクォーラム管理を含める必要があるため、設計全体が複雑になります。

従来の拡張RACクラスタでは、ASMミラーリングを使用してデータを保護していました。このアプローチは機能しますが、多くの手動設定手順が必要になり、ネットワークインフラストラクチャにオーバーヘッドが発生します。一方、SnapMirrorのアクティブな同期機能でデータレプリケーションを実行できるようにすること

で、ソリューションが大幅に簡易化されます。同期、中断後の再同期、フェイルオーバー、クォーラム管理などの操作が容易になります。また、SANの設計と管理を簡素化するために、SANをサイト間に分散させる必要もありません。

### レプリケーション

SnapMirrorアクティブ同期のRAC機能を理解するには、ストレージをミラーリングされたストレージでホストされている単一のLUNセットとして表示することが重要です。例：




プライマリコピーまたはミラーコピーはありません。論理的には、各LUNのコピーは1つだけで、そのLUNは2つの異なるストレージシステム上にあるSANパスで使用できます。ホストから見ると、ストレージフェイルオーバーは発生せず、代わりにパスが変更されます。さまざまな障害イベントが発生すると、LUNへの特定

のパスが失われても、他のパスはオンラインのままになる可能性があります。SnapMirrorのアクティブな同期により、すべての運用パスで同じデータを利用できるようになります。

### ストレージ構成

この構成例では、ASMディスクは、エンタープライズストレージの単一サイトRAC構成と同じように設定されています。ストレージシステムはデータ保護を提供するため、ASM外部冗長性が使用されます。

### ユニフォームアクセスと非インフォームアクセス

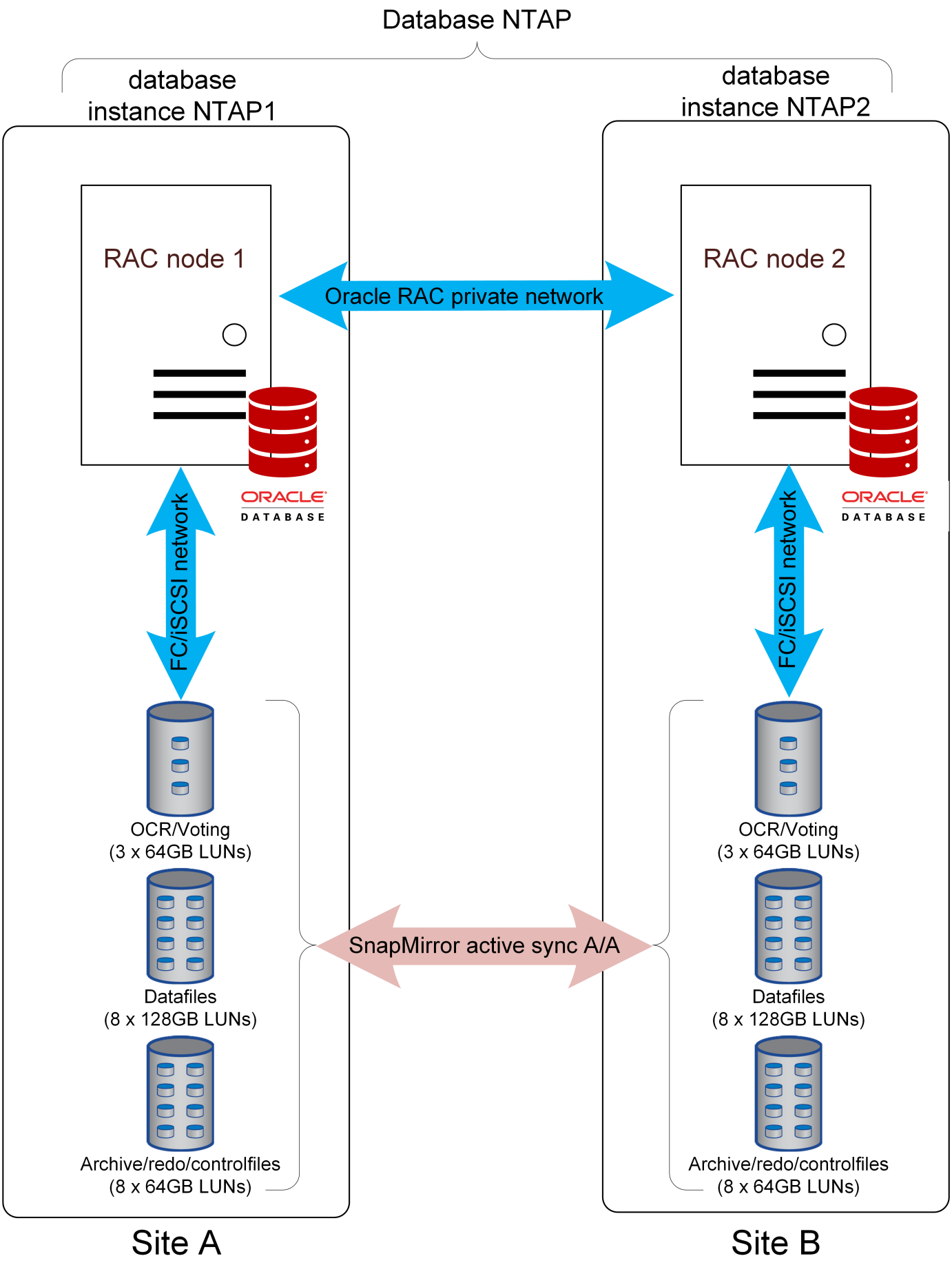
SnapMirrorアクティブ同期上のOracle RACで最も重要な考慮事項は、均一アクセスと非均一アクセスのどちらを使用するかです。

アクセスが統一されているため、各ホストは両方のクラスタのパスを認識できます。一様でないアクセスとは、ホストがローカルクラスタへのパスのみを認識できることを意味します。

どちらのオプションも特に推奨または推奨されないものではありません。ダークファイバを使用してサイトを接続しているお客様もいれば、そのような接続を利用していないお客様や、SANインフラで長距離ISLをサポートしていないお客様もいます。

### 不均一なアクセス

アクセスが一様でない場合、SANの観点からはより簡単に設定できます。


Database NTAP
database instance NTAP1
database instance NTAP2
RAC node 1
RAC node 2
Oracle RAC private network
FC/iSCSI network
FC/iSCSI network
ORACLE DATABASE
ORACLE DATABASE
OCR/Voting (3 x 64GB LUNs)
Datafiles (8 x 128GB LUNs)
Archive/redo/controlfiles (8 x 64GB LUNs)
SnapMirror active sync A/A
OCR/Voting (3 x 64GB LUNs)
Datafiles (8 x 128GB LUNs)
Archive/redo/controlfiles (8 x 64GB LUNs)
Site A
Site B

このアプローチの主な欠点"不均一なアクセス"は、サイト間のONTAP接続が失われたり、ストレージシステムが失われたりすると、一方のサイトのデータベースインスタンスが失われることです。これは明らかに望ましくありませんが、シンプルなSAN構成と引き換えに許容可能なリスクになる可能性があります。

均一なアクセス

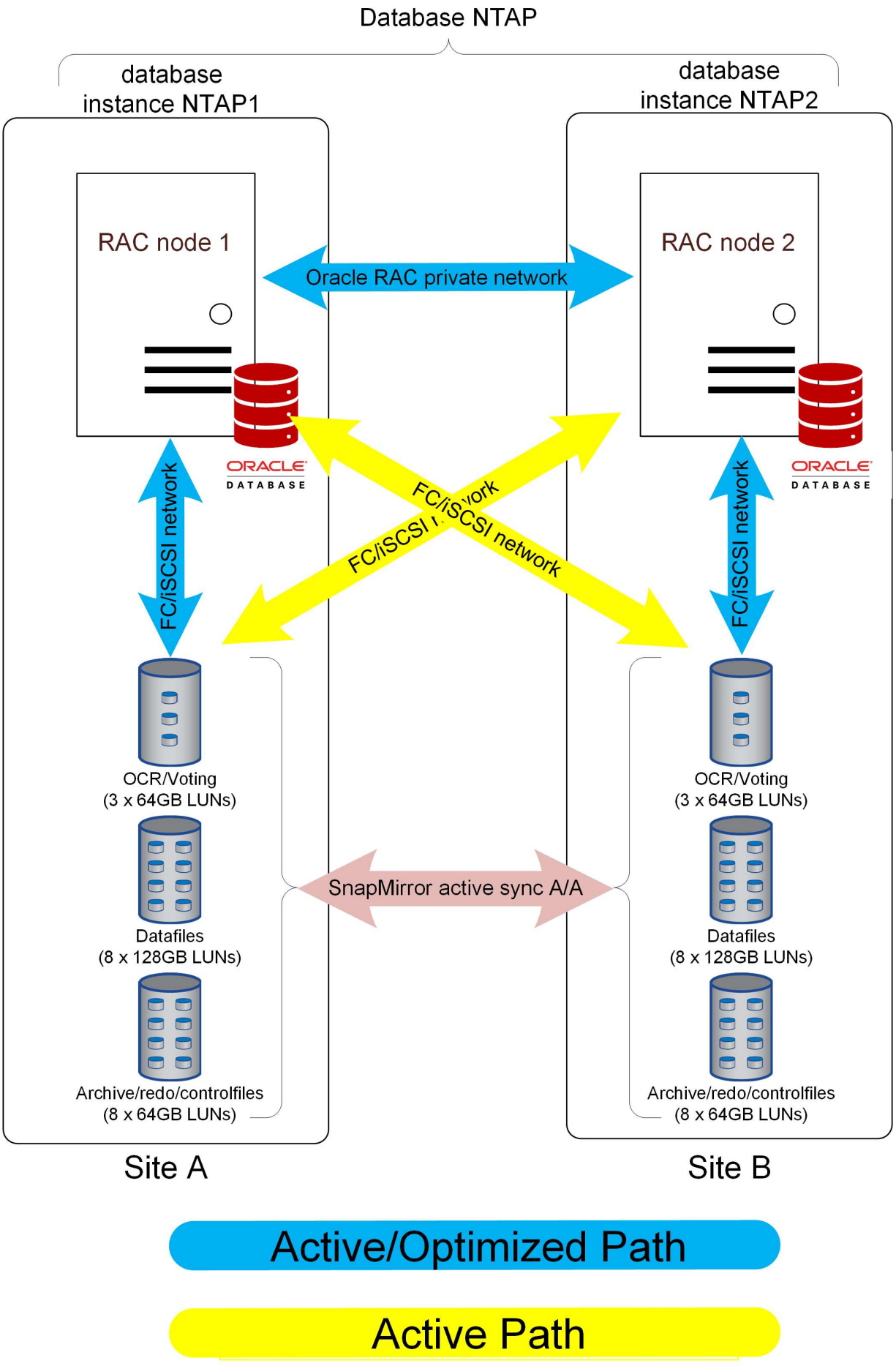
アクセスを統一するには、SANをサイト間に拡張する必要があります。主なメリットは、ストレージシステムが停止してもデータベースインスタンスが失われないことです。その結果、現在使用されているパスがマルチパスに変更されます。

不均一アクセスを設定するには、いくつかの方法があります。

次の図には、単純なコントローラ障害時に使用されるアクティブだが最適化されていないパスもありますが、この図では省略しています。

近接設定を使用した**AFF**

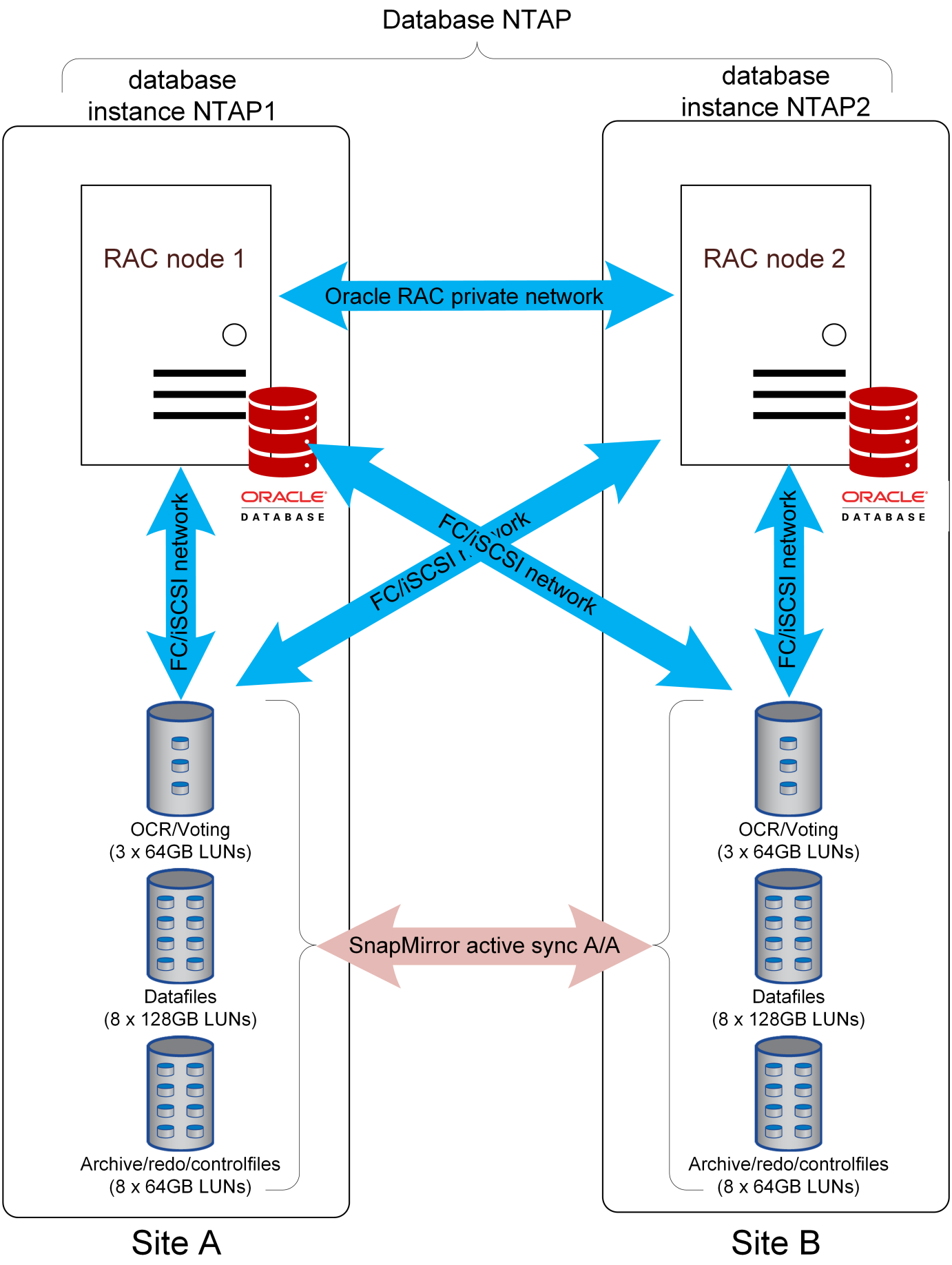
サイト間のレイテンシが大きい場合は、ホストとの近接設定を使用してAFFシステムを設定できます。これにより、各ストレージシステムはどのホストがローカルでどのホストがリモートであるかを認識し、パスの優先順位を適切に割り当てることができます。


Database NTAP
database instance NTAP1
database instance NTAP2
RAC node 1
RAC node 2
Oracle RAC private network
FC/iSCSI network
FC/iSCSI network
FC/iSCSI network
FC/iSCSI network
ORACLE DATABASE
ORACLE DATABASE
OCR/Voting (3 x 64GB LUNs)
OCR/Voting (3 x 64GB LUNs)
SnapMirror active sync A/A
Datafiles (8 x 128GB LUNs)
Datafiles (8 x 128GB LUNs)
Archive/redo/controlfiles (8 x 64GB LUNs)
Archive/redo/controlfiles (8 x 64GB LUNs)
Site A
Site B
Active/Optimized Path
Active Path

通常運用時は、各Oracleインスタンスがローカルのアクティブ/最適パスを優先的に使用します。その結果、すべての読み取りはブロックのローカルコピーによって処理されます。これにより、レイテンシが最小限に抑えられます。書き込みIOも同様に、ローカルコントローラへのパスに送信されます。IOは確認される前にレプリケートする必要がありますが、その場合もサイト間ネットワークを通過するための追加のレイテンシが発生しますが、同期レプリケーションソリューションではこれを回避することはできません。

### 近接設定なしの**ASA / AFF**

サイト間のレイテンシがそれほど高くない場合は、ホストとの近接を設定せずにAFFシステムを構成するか、ASAを使用できます。


Database NTAP
database instance NTAP1
database instance NTAP2
RAC node 1
RAC node 2
Oracle RAC private network
FC/iSCSI network
FC/iSCSI network
FC/iSCSI network
FC/iSCSI network
ORACLE DATABASE
ORACLE DATABASE
OCR/Voting (3 x 64GB LUNs)
OCR/Voting (3 x 64GB LUNs)
SnapMirror active sync A/A
Datafiles (8 x 128GB LUNs)
Datafiles (8 x 128GB LUNs)
Archive/redo/controlfiles (8 x 64GB LUNs)
Archive/redo/controlfiles (8 x 64GB LUNs)
Site A
Site B

各ホストが両方のストレージシステムのすべての動作パスを使用できるようになります。これにより、各ホストが1つだけでなく2つのクラスタの潜在的なパフォーマンスを利用できるようになるため、パフォーマンスが大幅に向上します。

ASAでは、両方のクラスタへのすべてのパスがアクティブで最適化されているとみなされるだけでなく、パートナーコントローラのパスもアクティブになります。その結果、常にクラスタ全体でオールアクティブなSANパスが作成されます。

ASAシステムは、不均一なアクセス設定でも使用できます。サイト間パスは存在しないため、IOがISLを経由してもパフォーマンスに影響はありません。

**RAC Tiebreaker**

## SnapMirrorアクティブ同期を使用した拡張RACはIOに関して対称アーキテクチャですが、スプリットブレイン管理に接続される例外が1つあります。

レプリケーションリンクが失われ、どちらのサイトにもクォーラムがない場合はどうなりますか。何が起こるべきでしょうか? この質問は、Oracle RACとONTAPの両方の動作に当てはまります。サイト間で変更をレプリケートできない場合に運用を再開するには、一方のサイトを停止し、もう一方のサイトを使用できなくなる必要があります。

は、"ONTAPメディエーター"ONTAPレイヤでこの要件に対応します。RACのタイブレークには複数のオプションがあります。

#### Oracleタイブレーカー

Oracle RACのスプリットブレインリスクを管理する最善の方法は、奇数のRACノードを使用すること（できれば3つ目のサイトのTiebreakerを使用すること）です。3つ目のサイトが使用できない場合は、Tiebreakerインスタンスを2つのサイトの一方のサイトに配置して、優先サバイバーサイトに指定できます。

#### OracleおよびCSS_CRITICAL

ノード数が偶数の場合、Oracle RACのデフォルトの動作では、クラスタ内のいずれかのノードの重要度が他のノードよりも高くなります。優先度の高いノードを含むサイトはサイト分離を継続し、もう一方のサイトのノードは削除されます。優先順位付けは複数の要因に基づいて行われますが、設定を使用してこの動作を制御することもできます `css_critical`。

"例"アーキテクチャでは、RACノードのホスト名はjfs12およびjfs13です。の現在の設定は `css_critical`次のとおりです。

```
[root@jfs12 ~]# /grid/bin/crsctl get server css_critical
CRS-5092: Current value of the server attribute CSS_CRITICAL is no.

[root@jfs13 trace]# /grid/bin/crsctl get server css_critical
CRS-5092: Current value of the server attribute CSS_CRITICAL is no.
```

jfs12が設定されたサイトを優先サイトにする場合は、サイトAのノードでこの値をyesに変更し、サービスを再起動します。

```
[root@jfs12 ~]# /grid/bin/crsctl set server css_critical yes
CRS-4416: Server attribute 'CSS_CRITICAL' successfully changed. Restart
Oracle High Availability Services for new value to take effect.

[root@jfs12 ~]# /grid/bin/crsctl stop crs
CRS-2791: Starting shutdown of Oracle High Availability Services-managed
resources on 'jfs12'
CRS-2673: Attempting to stop 'ora.crsd' on 'jfs12'
CRS-2790: Starting shutdown of Cluster Ready Services-managed resources on
server 'jfs12'
CRS-2673: Attempting to stop 'ora.ntap.ntappdb1.pdb' on 'jfs12'
…
CRS-2673: Attempting to stop 'ora.gipcd' on 'jfs12'
CRS-2677: Stop of 'ora.gipcd' on 'jfs12' succeeded
CRS-2793: Shutdown of Oracle High Availability Services-managed resources
on 'jfs12' has completed
CRS-4133: Oracle High Availability Services has been stopped.

[root@jfs12 ~]# /grid/bin/crsctl start crs
CRS-4123: Oracle High Availability Services has been started.
```

## 障害シナリオ

### 概要

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

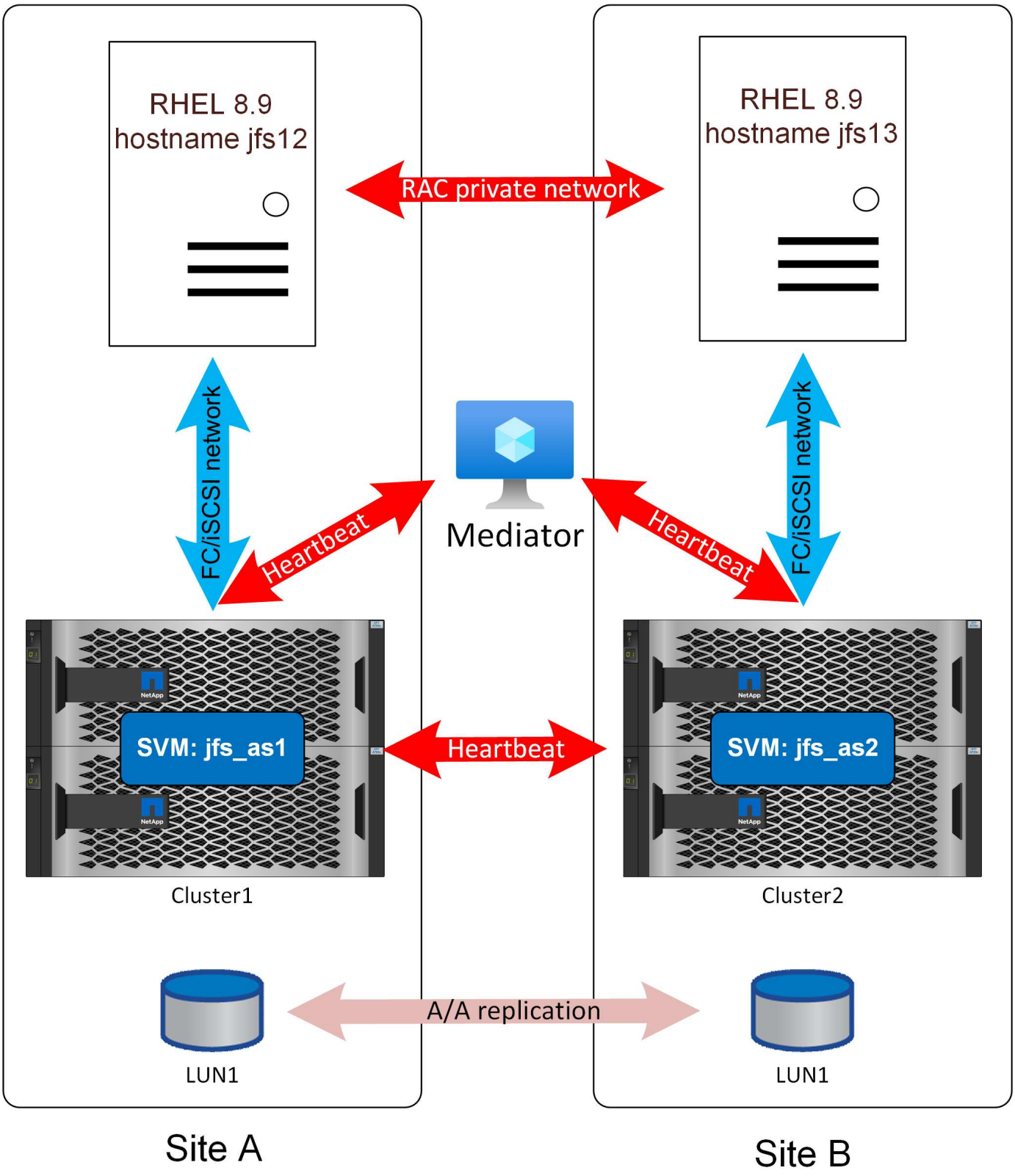
優先サイトを変更するには、簡単な操作が必要です。クラスタ間でレプリケーション動作の権限が切り替わるため、IOは1~2秒間停止しますが、それ以外の場合はIOには影響しません。

### サンプルアーキテクチャ

## このセクションで示す障害の詳細な例は、次のアーキテクチャに基づいています。

> これは、SnapMirrorアクティブ同期でOracleデータベースを使用する場合のオプションの1つにすぎません。この設計は、いくつかのより複雑なシナリオを説明するために選択されました。

この設計では、サイトAがに設定されていると仮定し"優先サイト"ます。




**RAC**インターコネクト障害

Oracle RACレプリケーションリンクが失われると、SnapMirror接続が切断されますが、デフォルトでタイムアウトが短くなる点が異なります。デフォルト設定では、Oracle RACノードはストレージ接続が失われてから200秒待機してから削除されますが、RAC

ネットワークハートビートが失われてからは30秒しか待機しません。

CRSメッセージは次のようになります。30秒のタイムアウトの経過が表示されます。CSS_CRITICALはサイトAにあるjfs12に設定されているため、これが存続するサイトとなり、サイトBのjfs13が削除されます。

```
2024-09-12 10:56:44.047 [ONMD(3528)]CRS-1611: Network communication with
node jfs13 (2) has been missing for 75% of the timeout interval.  If this
persists, removal of this node from cluster will occur in 6.980 seconds
2024-09-12 10:56:48.048 [ONMD(3528)]CRS-1610: Network communication with
node jfs13 (2) has been missing for 90% of the timeout interval.  If this
persists, removal of this node from cluster will occur in 2.980 seconds
2024-09-12 10:56:51.031 [ONMD(3528)]CRS-1607: Node jfs13 is being evicted
in cluster incarnation 621599354; details at (:CSSNM00007:) in
/gridbase/diag/crs/jfs12/crs/trace/onmd.trc.
2024-09-12 10:56:52.390 [CRSD(6668)]CRS-7503: The Oracle Grid
Infrastructure process 'crsd' observed communication issues between node
'jfs12' and node 'jfs13', interface list of local node 'jfs12' is
'192.168.30.1:33194;', interface list of remote node 'jfs13' is
'192.168.30.2:33621;'.
2024-09-12 10:56:55.683 [ONMD(3528)]CRS-1601: CSSD Reconfiguration
complete. Active nodes are jfs12 .
2024-09-12 10:56:55.722 [CRSD(6668)]CRS-5504: Node down event reported for
node 'jfs13'.
2024-09-12 10:56:57.222 [CRSD(6668)]CRS-2773: Server 'jfs13' has been
removed from pool 'Generic'.
2024-09-12 10:56:57.224 [CRSD(6668)]CRS-2773: Server 'jfs13' has been
removed from pool 'ora.NTAP'.
```

**SnapMirror**通信障害

SnapMirrorのアクティブな同期レプリケーションリンクの場合、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いため、書き込みIOを完了できません。

サイト**A**

レプリケーションリンク障害が発生したサイトAでは、レプリケーションリンクが本当に動作不能であると判断される前に、ONTAPが書き込みをレプリケートしようとするため、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ONTAPクラスタ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。

レプリケーションリンクはタイムスタンプ15：19：44でカットされました。Oracle RACからの最初の警告は、200秒のタイムアウト（Oracle RACパラメータdisktimeoutで制御）が近づくと、100秒後に通知されます。

```
2024-09-10 15:21:24.702 [ONMD(2792)]CRS-1615: No I/O has completed after
50% of the maximum interval. If this persists, voting file
/dev/mapper/grid2 will be considered not functional in 99340 milliseconds.
2024-09-10 15:22:14.706 [ONMD(2792)]CRS-1614: No I/O has completed after
75% of the maximum interval. If this persists, voting file
/dev/mapper/grid2 will be considered not functional in 49330 milliseconds.
2024-09-10 15:22:44.708 [ONMD(2792)]CRS-1613: No I/O has completed after
90% of the maximum interval. If this persists, voting file
/dev/mapper/grid2 will be considered not functional in 19330 milliseconds.
2024-09-10 15:23:04.710 [ONMD(2792)]CRS-1604: CSSD voting file is offline:
/dev/mapper/grid2; details at (:CSSNM00058:) in
/gridbase/diag/crs/jfs13/crs/trace/onmd.trc.
2024-09-10 15:23:04.710 [ONMD(2792)]CRS-1606: The number of voting files
available, 0, is less than the minimum number of voting files required, 1,
resulting in CSSD termination to ensure data integrity; details at
(:CSSNM00018:) in /gridbase/diag/crs/jfs13/crs/trace/onmd.trc
2024-09-10 15:23:04.716 [ONMD(2792)]CRS-1699: The CSS daemon is
terminating due to a fatal error from thread:
clssnmvDiskPingMonitorThread; Details at (:CSSSC00012:) in
/gridbase/diag/crs/jfs13/crs/trace/onmd.trc
2024-09-10 15:23:04.731 [OCSSD(2794)]CRS-1652: Starting clean up of CRSD
resources.
```

200秒の投票ディスクタイムアウトに達すると、このOracle RACノードはクラスタから削除され、リブートされます。

ネットワーク相互接続の全体的な障害

## サイト間のレプリケーションリンクが完全に失われると、SnapMirrorアクティブ同期とOracle RAC接続の両方が中断されます。

Oracle RACのスプリットブレイン検出は、Oracle RACストレージのハートビートに依存します。サイト間の接続が失われてRACネットワークハートビートとストレージレプリケーションサービスの両方が同時に失われると、RACサイトはRACインターコネクトまたはRAC投票ディスクを介してサイト間通信できなくなります。その結果、ノード数が偶数になると、両方のサイトがデフォルト設定で削除される可能性があります。正確な動作は、イベントのシーケンス、RACネットワークおよびディスクハートビートポーリングのタイミングによって異なります。

2サイト停止のリスクには、2つの方法で対処できます。まず、"Tiebreaker"構成を使用できます。

3つ目のサイトが利用できない場合は、RACクラスタでmiscountパラメータを調整することでこのリスクに対処できます。デフォルトでは、RACネットワークハートビートタイムアウトは30秒です。通常、RACは障害が発生したRACノードを特定してクラスタから削除するために使用します。また、投票ディスクハートビートにも接続されています。

たとえば、Oracle RACとストレージレプリケーションサービスの両方でサイト間トラフィックを伝送するコンジットがバックホーでカットされると、30秒間のミスカウントのカウントダウンが開始されます。RAC優先サイトノードが30秒以内に反対サイトとの接続を再確立できない場合、および同じ30秒以内に反対サイト

が停止していることを投票ディスクを使用して確認できない場合、優先サイトノードも削除されます。その結果、データベースが完全に停止します。

ミスマウントポーリングが発生したタイミングによっては、30秒でSnapMirrorアクティブ同期がタイムアウトし、優先サイトのストレージでサービスが再開されるまでに30秒では不十分な場合があります。この30秒のウィンドウは増やすことができます。

```
[root@jfs12 ~]# /grid/bin/crsctl set css misscount 100
CRS-4684: Successful set of parameter misscount to 100 for Cluster
Synchronization Services.
```

この値を指定すると、優先サイト上のストレージシステムは、ミスカウントのタイムアウトが切れる前に処理を再開できます。その結果、LUNパスを削除したサイトのノードのみが削除されます。以下の例：

```
2024-09-12 09:50:59.352 [ONMD(681360)]CRS-1612: Network communication with
node jfs13 (2) has been missing for 50% of the timeout interval.  If this
persists, removal of this node from cluster will occur in 49.570 seconds
2024-09-12 09:51:10.082 [CRSD(682669)]CRS-7503: The Oracle Grid
Infrastructure process 'crsd' observed communication issues between node
'jfs12' and node 'jfs13', interface list of local node 'jfs12' is
'192.168.30.1:46039;', interface list of remote node 'jfs13' is
'192.168.30.2:42037;'.
2024-09-12 09:51:24.356 [ONMD(681360)]CRS-1611: Network communication with
node jfs13 (2) has been missing for 75% of the timeout interval.  If this
persists, removal of this node from cluster will occur in 24.560 seconds
2024-09-12 09:51:39.359 [ONMD(681360)]CRS-1610: Network communication with
node jfs13 (2) has been missing for 90% of the timeout interval.  If this
persists, removal of this node from cluster will occur in 9.560 seconds
2024-09-12 09:51:47.527 [OHASD(680884)]CRS-8011: reboot advisory message
from host: jfs13, component: cssagent, with time stamp: L-2024-09-12-
09:51:47.451
2024-09-12 09:51:47.527 [OHASD(680884)]CRS-8013: reboot advisory message
text: oracssdagent is about to reboot this node due to unknown reason as
it did not receive local heartbeats for 10470 ms amount of time
2024-09-12 09:51:48.925 [ONMD(681360)]CRS-1632: Node jfs13 is being
removed from the cluster in cluster incarnation 621596607
```

Oracleサポートでは、設定の問題を解決するために、miscountパラメータやdisktimeoutパラメータを変更することを強く推奨していません。ただし、SANブート、仮想化、ストレージレプリケーションの構成など、多くの場合、これらのパラメータの変更は保証され、やむを得ない場合があります。たとえば、SANまたはIPネットワークの安定性に問題があり、その結果RACが削除された場合は、原因となっている問題を修正し、ミスカウントやdisktimeoutの値を加算しないでください。構成エラーに対処するためにタイムアウトを変更すると、問題がマスキングされ、問題が解決されません。基盤となるインフラの設計要素に基づいてRAC環境を適切に設定するためにこれらのパラメータを変更することは異なり、Oracleのサポートステートメントと一致しています。SANブートでは、disktimeoutに合わせて最大200までミスカウントを調整するのが一般的です。詳細については、を参照してください"リンクをクリックしてください"。

## ストレージシステムまたはサイト障害の結果は、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、書き込み時のIOポーズが約15秒になります。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。

ストレージシステムのみが影響を受けた場合、障害が発生したサイトのOracle RACノードはストレージサービスを失い、削除とその後のリブートの前に同じ200秒のディスクタイムアウトカウントダウンを入力します。

```
2024-09-11 13:44:38.613 [ONMD(3629)]CRS-1615: No I/O has completed after
50% of the maximum interval. If this persists, voting file
/dev/mapper/grid2 will be considered not functional in 99750 milliseconds.
2024-09-11 13:44:51.202 [ORAAGENT(5437)]CRS-5011: Check of resource "NTAP"
failed: details at "(:CLSN00007:)" in
"/gridbase/diag/crs/jfs13/crs/trace/crsd_oraagent_oracle.trc"
2024-09-11 13:44:51.798 [ORAAGENT(75914)]CRS-8500: Oracle Clusterware
ORAAGENT process is starting with operating system process ID 75914
2024-09-11 13:45:28.626 [ONMD(3629)]CRS-1614: No I/O has completed after
75% of the maximum interval. If this persists, voting file
/dev/mapper/grid2 will be considered not functional in 49730 milliseconds.
2024-09-11 13:45:33.339 [ORAAGENT(76328)]CRS-8500: Oracle Clusterware
ORAAGENT process is starting with operating system process ID 76328
2024-09-11 13:45:58.629 [ONMD(3629)]CRS-1613: No I/O has completed after
90% of the maximum interval. If this persists, voting file
/dev/mapper/grid2 will be considered not functional in 19730 milliseconds.
2024-09-11 13:46:18.630 [ONMD(3629)]CRS-1604: CSSD voting file is offline:
/dev/mapper/grid2; details at (:CSSNM00058:) in
/gridbase/diag/crs/jfs13/crs/trace/onmd.trc.
2024-09-11 13:46:18.631 [ONMD(3629)]CRS-1606: The number of voting files
available, 0, is less than the minimum number of voting files required, 1,
resulting in CSSD termination to ensure data integrity; details at
(:CSSNM00018:) in /gridbase/diag/crs/jfs13/crs/trace/onmd.trc
2024-09-11 13:46:18.638 [ONMD(3629)]CRS-1699: The CSS daemon is
terminating due to a fatal error from thread:
clssnmvDiskPingMonitorThread; Details at (:CSSSC00012:) in
/gridbase/diag/crs/jfs13/crs/trace/onmd.trc
2024-09-11 13:46:18.651 [OCSSD(3631)]CRS-1652: Starting clean up of CRSD
resources.
```

ストレージサービスが失われたRACノードのSANパスの状態は次のようになります。

```
oradata7 (3600a0980383041334a3f55676c697347) dm-20 NETAPP,LUN C-Mode
size=128G features='3 queue_if_no_path pg_init_retries 50' hwhandler='1
alua' wp=rw
|-+- policy='service-time 0' prio=0 status=enabled
| `- 34:0:0:18 sdam 66:96  failed faulty running
`-+- policy='service-time 0' prio=0 status=enabled
  `- 33:0:0:18 sdaj 66:48  failed faulty running
```

Linuxホストはパスの損失を200秒よりもはるかに早く検出しましたが、データベースに関しては、障害が発生したサイトのホストへのクライアント接続は、デフォルトのOracle RAC設定で200秒間フリーズします。フルデータベース処理は削除が完了するまで再開されません。

一方、反対側のサイトのOracle RACノードでは、もう一方のRACノードの損失が記録されます。それ以外の場合は、通常どおり動作し続けます。

```
2024-09-11 13:46:34.152 [ONMD(3547)]CRS-1612: Network communication with
node jfs13 (2) has been missing for 50% of the timeout interval.  If this
persists, removal of this node from cluster will occur in 14.020 seconds
2024-09-11 13:46:41.154 [ONMD(3547)]CRS-1611: Network communication with
node jfs13 (2) has been missing for 75% of the timeout interval.  If this
persists, removal of this node from cluster will occur in 7.010 seconds
2024-09-11 13:46:46.155 [ONMD(3547)]CRS-1610: Network communication with
node jfs13 (2) has been missing for 90% of the timeout interval.  If this
persists, removal of this node from cluster will occur in 2.010 seconds
2024-09-11 13:46:46.470 [OHASD(1705)]CRS-8011: reboot advisory message
from host: jfs13, component: cssmonit, with time stamp: L-2024-09-11-
13:46:46.404
2024-09-11 13:46:46.471 [OHASD(1705)]CRS-8013: reboot advisory message
text: At this point node has lost voting file majority access and
oracssdmonitor is rebooting the node due to unknown reason as it did not
receive local hearbeats for 28180 ms amount of time
2024-09-11 13:46:48.173 [ONMD(3547)]CRS-1632: Node jfs13 is being removed
from the cluster in cluster incarnation 621516934
```

メディエーター障害

## 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。

安全な自動運用を実現するためにメディエーターが必要になるのは、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：

- 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、読み取りと書き込みの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。

サービスの復旧

SnapMirrorは自己回復型です。SnapMirrorのアクティブな同期では、レプリケーション関係に問題があることを自動的に検出し、RPO=0の状態に戻します。同期レプリケーションが再確立されると、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。

アプリケーションとその構成方法によって異なり、一般的にこのようなタスクは簡単に自動化できます。SnapMirrorのアクティブな同期自体は自己修復機能であり、電源と接続が復旧した時点で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必要ありません。

手動フェイルオーバー

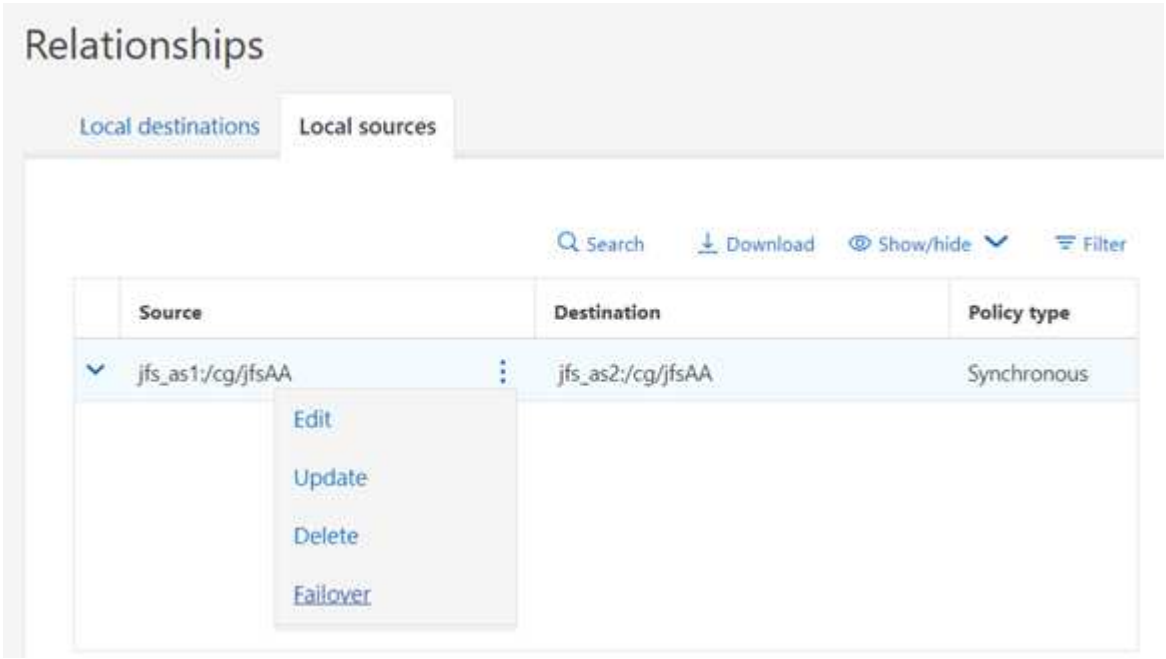
「フェイルオーバー」という用語は、双方向のレプリケーションテクノロジであるため、SnapMirror Active Syncを使用したレプリケーションの方向を指していません。代わりに、「failover」とは、障害発生時にどのストレージシステムが優先サイトになるかを意味します。

たとえば、メンテナンスのためにサイトをシャットダウンする前やDRテストを実行する前に、フェイルオーバーを実行して優先サイトを変更できます。

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。

GUI の例：




CLIを使用して元に戻す例：

```
Cluster2::> snapmirror failover start -destination-path jfs_as2:/cg/jfsAA
[Job 9575] Job is queued: SnapMirror failover for destination
"jfs_as2:/cg/jfsAA                      ".

Cluster2::> snapmirror failover show

Source    Destination                                           Error
Path      Path        Type     Status    start-time end-time   Reason
--------- ----------  -------- --------- ---------- ---------- ----------
jfs_as1:/cg/jfsAA
          jfs_as2:/cg/jfsAA
                      planned  completed 9/11/2024  9/11/2024
                                         09:29:22   09:29:32

The new destination path can be verified as follows:


Cluster1::> snapmirror show -destination-path jfs_as1:/cg/jfsAA

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