Oracleのディザスタリカバリ

Enterprise applications

NetApp
May 09, 2024

目次

Oracleのディザスタリカバリ

ONTAPによるOracleデータベースのディザスタリカバリ

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アクティブ同期はSnapMirror Synchronous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。

SnapMirrorアクティブ同期とMetroClusterの内部的な動作は大きく異なりますが、ホストにとってはほぼ同じ結果になります。主な違いは粒度です。同期レプリケートするワークロードのみを選択する場合は、SnapMirrorアクティブ同期が適しています。環境全体やデータセンターをレプリケートする必要がある場合は、MetroClusterをお勧めします。また、SnapMirrorアクティブ同期は現在SAN専用ですが、MetroClusterはSAN、NFS、SMBなどのマルチプロトコルです。

MetroCluster

MetroCluster物理アーキテクチャとOracleデータベース

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セツゾクノ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

[注]「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。

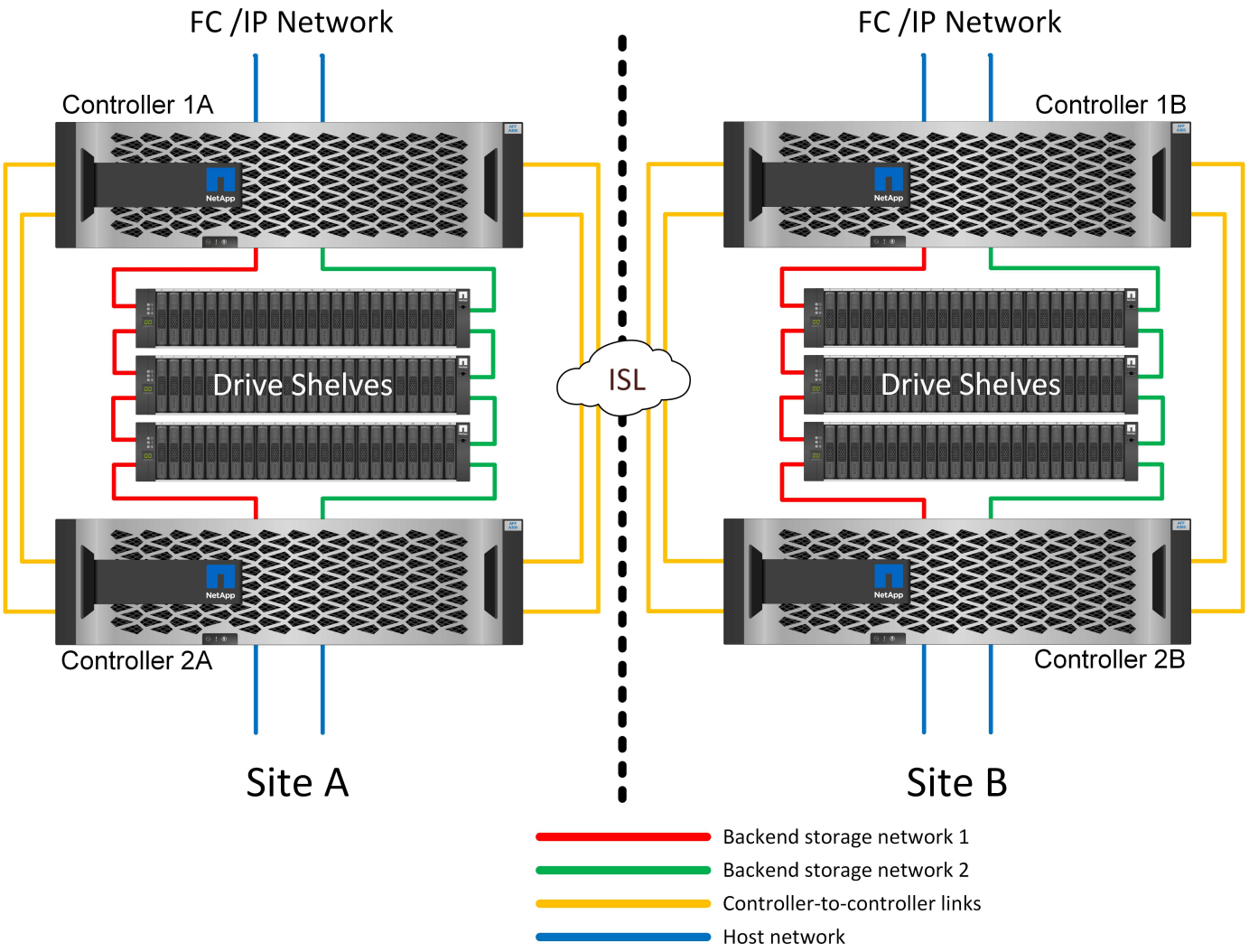
MetroCluster IP の略

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。

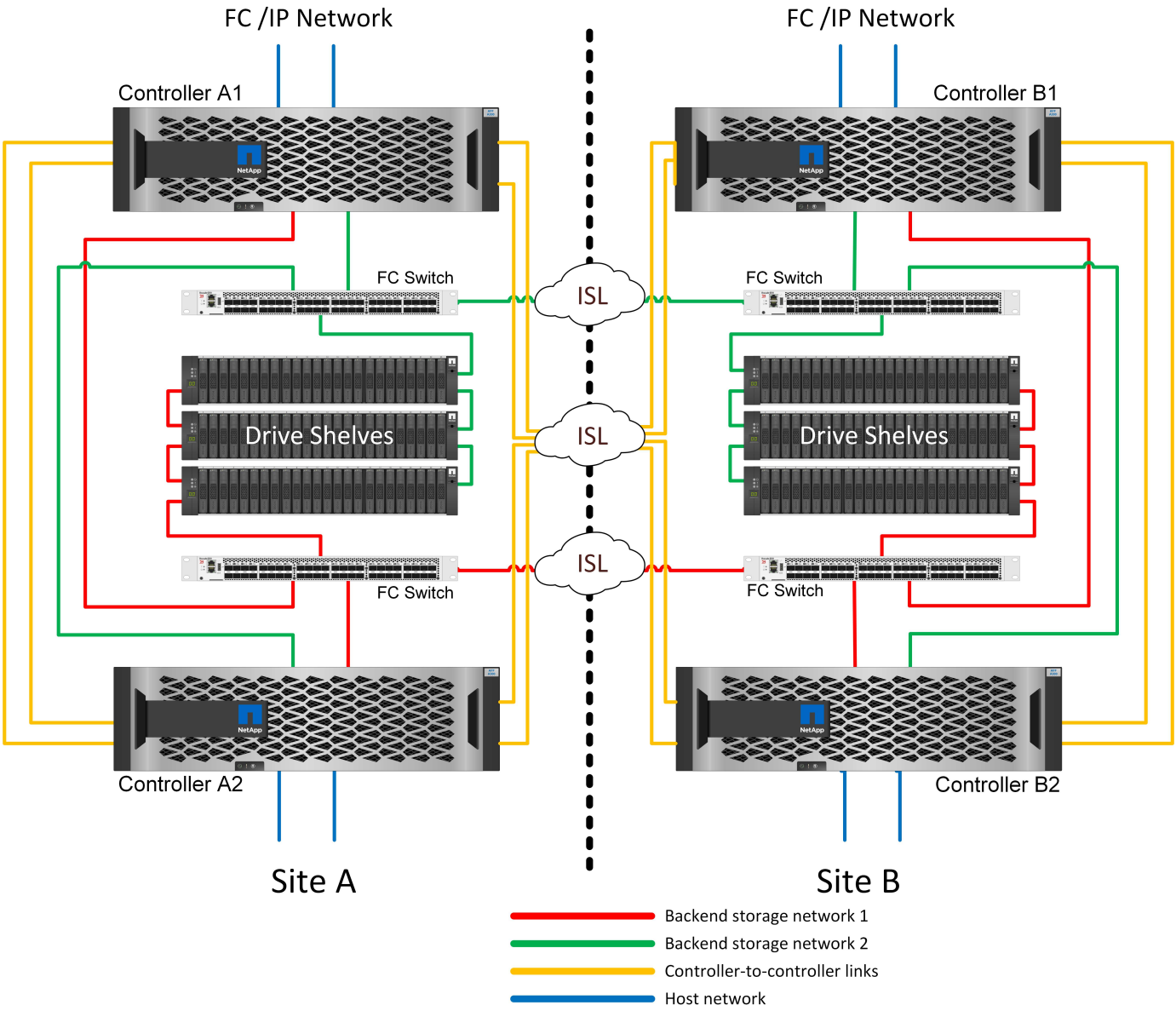
サイト間接続はコントローラのみであるため、アーキテクチャもシンプルです。FC SAN接続MetroClusterでは、コントローラが反対側サイトのドライブに直接書き込むため、追加のSAN接続、スイッチ、およびブリッジが必要になります。一方、IP構成のコントローラは、コントローラを介して反対側のドライブに書き込みます。

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。

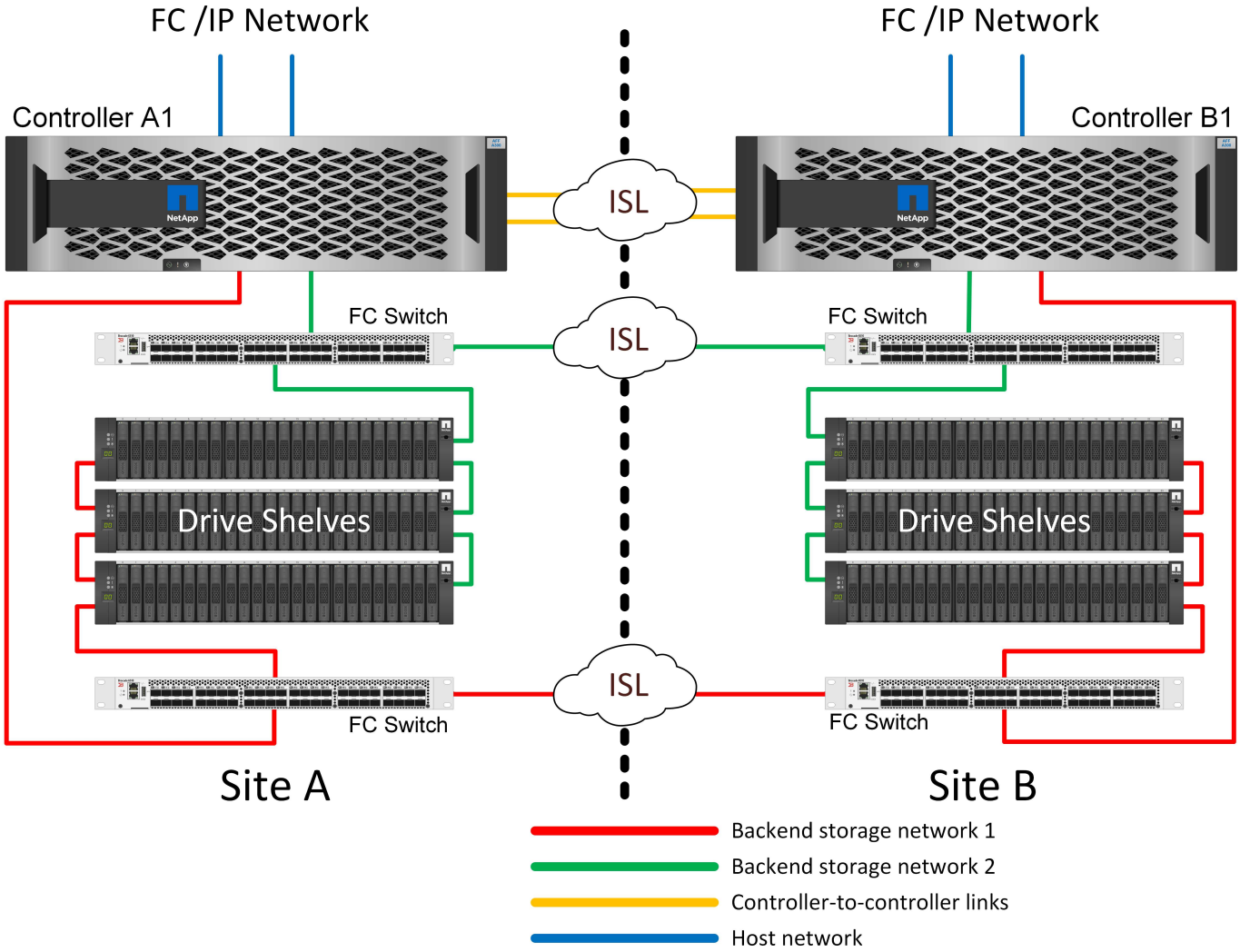


一部のマルチサイトインフラは、アクティブ/アクティブ運用向けに設計されたものではなく、プライマリサイトやディザスタリカバリサイトとして使用されます。この場合、一般にHAペアMetroClusterオプションが推奨される理由は次のとおりです。

- 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。

MetroCluster論理アーキテクチャと**Oracle**データベース

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**

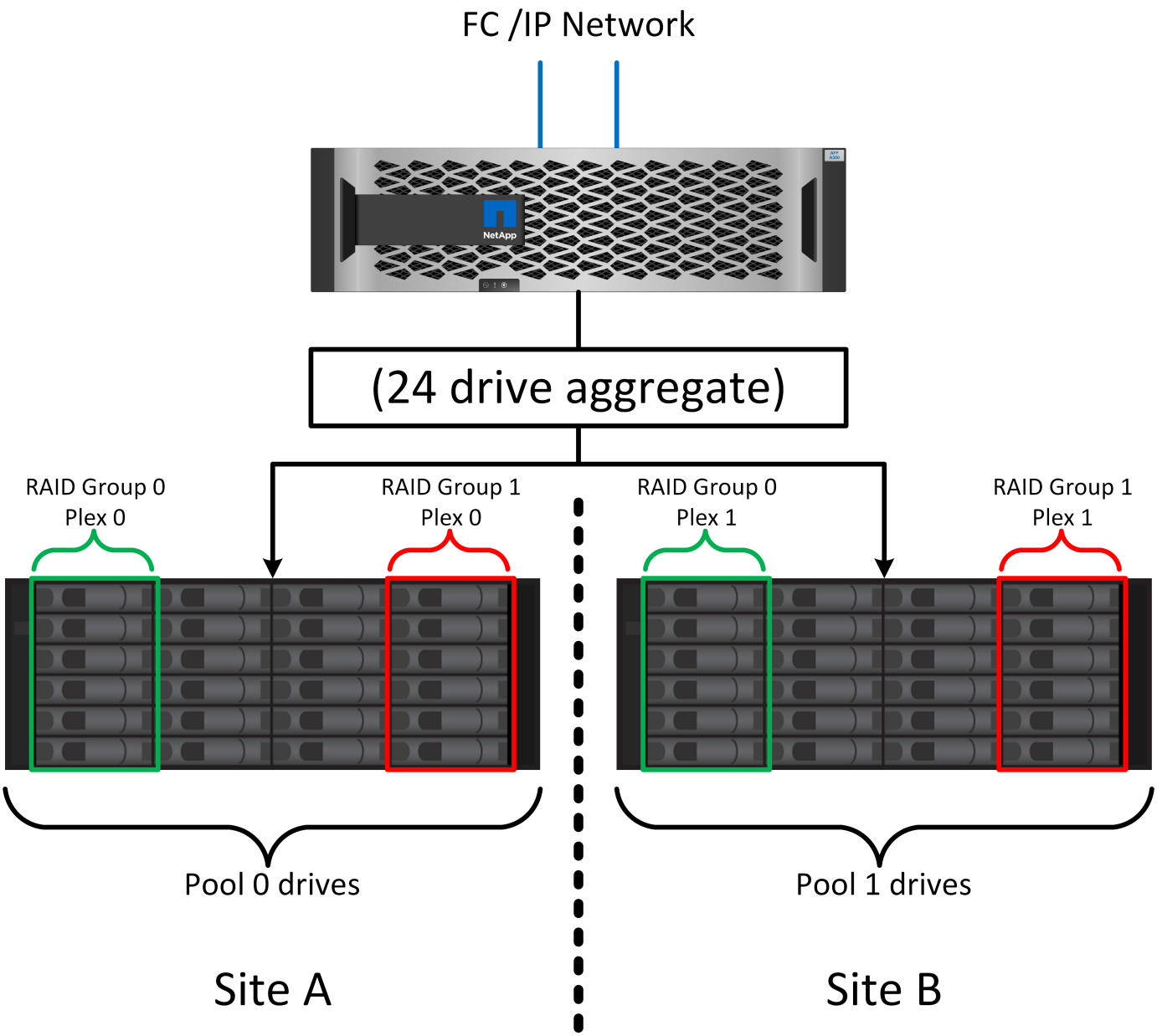
MetroClusterは、次の方法でNVRAMデータ保護を拡張します。

- 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**

。 "NetApp MetroCluster Tiebreaker" ソフトウェアを第3のサイトで実行して、MetroCluster環境の健全性を監視し、通知を送信し、必要に応じて災害時にスイッチオーバーを強制的に実行できます。Tiebreakerの完全な概要は、 "NetApp Support Site"ただし、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。

注意： MCTBソフトウェアは、強制的なスイッチオーバー時にNVRAMが同期されていること、またはプレックスが同期されていることを確認しません。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

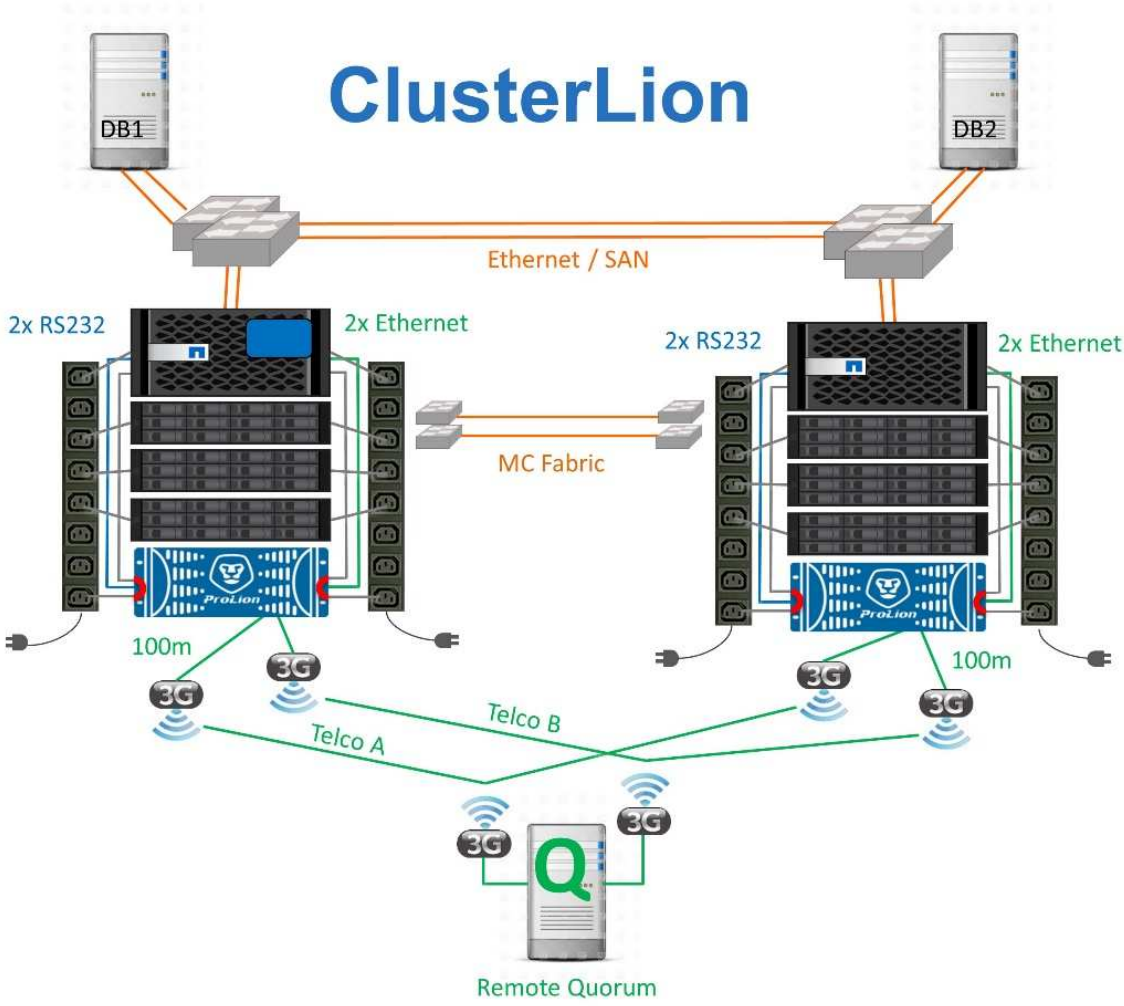
ONTAPメディエーターは、MetroCluster IPおよびその他の特定のONTAPソリューションで使用されます。これは、前述のMetroCluster Tiebreakerソフトウェアと同様に従来のTiebreakerサービスとして機能しますが、自動無人スイッチオーバーの実行という重要な機能も備えています。

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

Oracleデータベースと**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、ホスト側のディスクミラーリングのタイムアウト延長など、アプリケーションレベルのオプションもあります。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によるOracleデータベースのフェイルオーバー

```
 Metrocluster is an ONTAP feature that can protect your Oracle databases
with RPO=0 synchronous mirroring across sites, and it scales up to support
hundreds of databases on a single MetroCluster system. It's also simple to
use. The use of MetroCluster does not necessarily add to or change any
best practices for operating a enterprise applications and databases.
```

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。

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。Oracle RAC、Veritas Cluster Server（VCS）、VMware HA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、代替サイトでアプリケーション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、アプリケーションを構成する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
4. アプリケーションの起動

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。アプリケーション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、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、アプリケーション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、または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する

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。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データベース、**MetroCluster**、**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。

MetroCluster上のOracle単一インスタンス

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。

MetroCluster上の拡張Oracle 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/アクティブの拡張RACが必要な場合は、MetroClusterではなくASMミラーリングを検討する必要があります。ASMミラーリングでは、データの特定のレプリカを優先することができます。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がOracle RACで使用されている場合は、 `misscount` パラメータの変更が必要になる場合があります。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

SnapMirrorアクティブ同期ありのOracleデータベース

SnapMirrorアクティブ同期により、個 々 のOracleデータベースおよびアプリケーション環境に対して、選択的なRPO=0同期ミラーリングが可能になります。

SnapMirrorアクティブ同期は、SAN向けに強化されたSnapMirror機能です。これにより、ホストは、LUNをホストしているシステムとレプリカをホストしているシステムの両方からLUNにアクセスできます。

SnapMirror Active SyncとSnapMirror Syncはレプリケーションエンジンを共有しますが、SnapMirror Active Syncには、エンタープライズアプリケーションに対する透過的なアプリケーションフェイルオーバーやフェイルバックなどの追加機能が含まれています。

実際には、個 々 のワークロードに対して選択的かつきめ細かなRPO=0の同期レプリケーションを有効にすることで、MetroClusterのきめ細かなバージョンと同様に機能します。下位レベルのパスの動作はMetroClusterとは大きく異なりますが、ホスト側から見た結果はほぼ同じです。

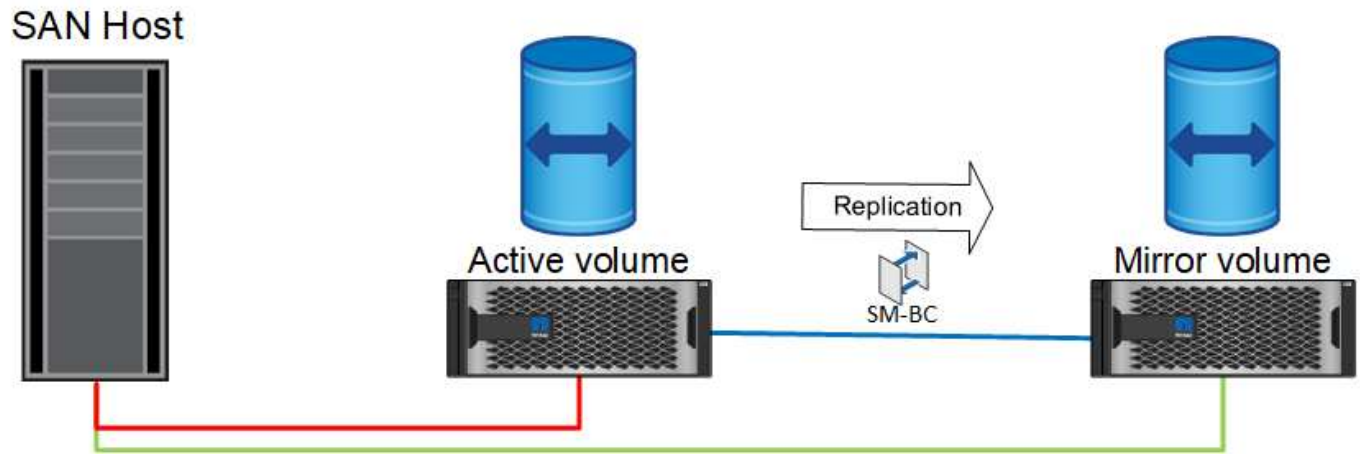
パスアクセス

SnapMirrorをアクティブに同期すると、プライマリとリモートの両方のストレージアレイから、ホストオペレ

ーティングシステムからストレージデバイスを認識できるようになります。パスは、ストレージシステムとホストの間の最適なパスを特定するための業界標準プロトコルであるAsymmetric Logical Unit Access（ALUA；非対称論理ユニットアクセス）を通じて管理されます。

I/Oへのアクセスに最も短いデバイスパスはアクティブ/最適パスとみなされ、残りのパスはアクティブ/非最適パスとみなされます。

SnapMirrorアクティブ同期関係は、異なるクラスタにあるSVMのペア間で確立されます。どちらのSVMもデータを提供できますが、ALUAは、LUNが配置されているドライブの所有権を現在持っているSVMを優先的に使用します。リモートSVMへのIOは、SnapMirrorのアクティブな同期インターコネクトを使用して経由でプロキシされます。



同期レプリケーション

通常の運用では、1つの例外を除き、リモートコピーは常にRPO=0の同期レプリカです。データをレプリケートできない場合、SnapMirrorのアクティブな同期により、データをレプリケートしてIOの提供を再開する必要がなくなります。このオプションは、レプリケーションリンクの損失がほぼ災害になると考えているお客様や、データをレプリケートできないときに業務の停止を望まないお客様に適しています。

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す。Mediatorは、プライマリサイトとリモートサイトの両方のストレージクラスタのフェイルオーバー処理を自動化します。オンプレミスまたはクラウドでホストされた小規模な仮想マシン（VM）に導入できます。設定後は、両方のサイトのフェイルオーバーシナリオを監視するための第3のサイトとして機能します。

SnapMirror Active Syncを使用した**Oracle**データベースのフェイルオーバー

SnapMirrorのアクティブな同期でOracleデータベースをホストする主な理由は、計画的

ストレージイベントと計画外ストレージイベントの発生時に透過的なフェイルオーバーを実現するためです。

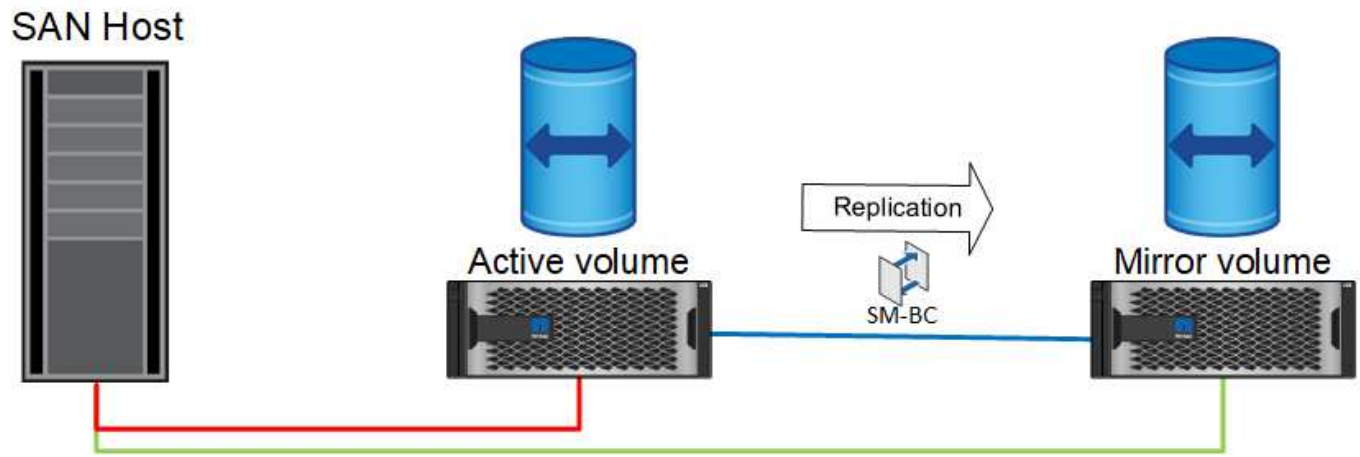
SnapMirror Active Syncでは、計画的フェイルオーバーと計画外フェイルオーバーの2種類のストレージフェイルオーバー処理がサポートされます。どちらの処理も多少異なります。計画的フェイルオーバーは、リモートサイトへの迅速なスイッチオーバーのために管理者が手動で開始し、計画外フェイルオーバーは3番目のサイトのメディエーターによって自動的に開始されます。計画的フェイルオーバーの主な目的は、パッチ適用とアップグレードの差分実行、ディザスタリカバリテストの実行、または完全なアクティブ同期機能を実証するためのサイト間の運用の切り替えという正式なポリシーの採用です。

次の図は、通常、フェイルオーバー、フェイルバックの各処理中に何が発生するかを示しています。わかりやすいように、レプリケートされたLUNを表しています。実際のSnapMirrorアクティブ同期構成では、レプリケーションはボリュームに基づいて行われます。各ボリュームには1つ以上のLUNが含まれていますが、わかりやすくするためにボリュームレイヤは削除されています。

通常運用時

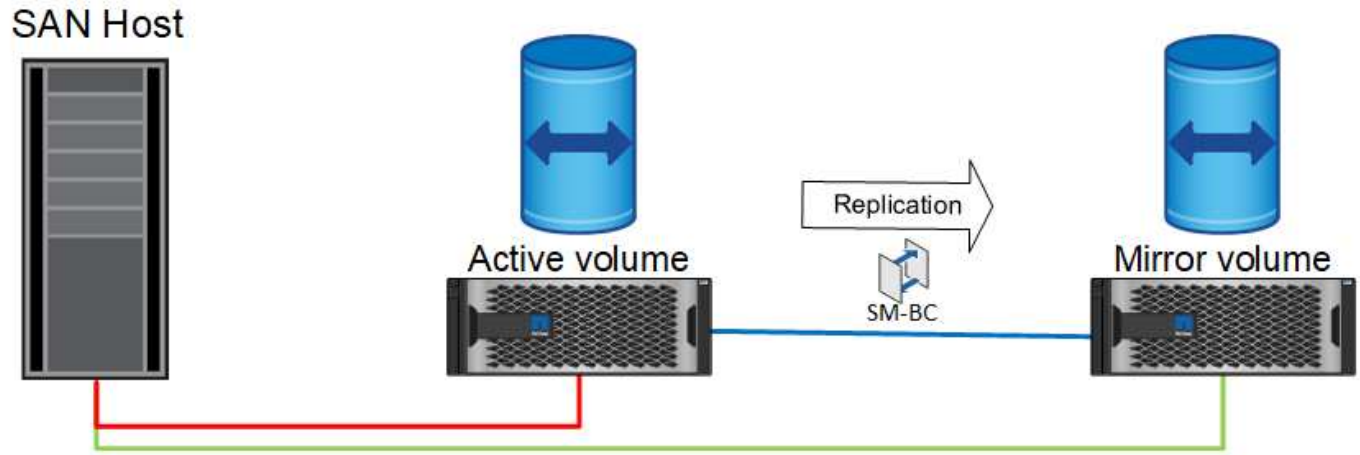
通常の操作では、ローカルレプリカまたはリモートレプリカからLUNにアクセスできます。赤い線はALUAによってアドバタイズされる最適パスを示し、そのパスにIOが優先的に送信されることになります。

緑の線はアクティブなパスですが、そのパスのIOをSnapMirrorのアクティブな同期パス経由で渡す必要があるため、レイテンシが高くなります。追加のレイテンシは、SnapMirrorアクティブ同期に使用されるサイト間のインターコネクトの速度によって異なります。



失敗

計画的フェイルオーバーまたは計画外フェイルオーバーのためにアクティブなミラーコピーを使用できなくなった場合は、明らかに使用できなくなります。ただし、リモートシステムには同期レプリカがあり、リモートサイトへのSANパスはすでに存在します。リモートシステムは、そのLUNのIOを処理できます。

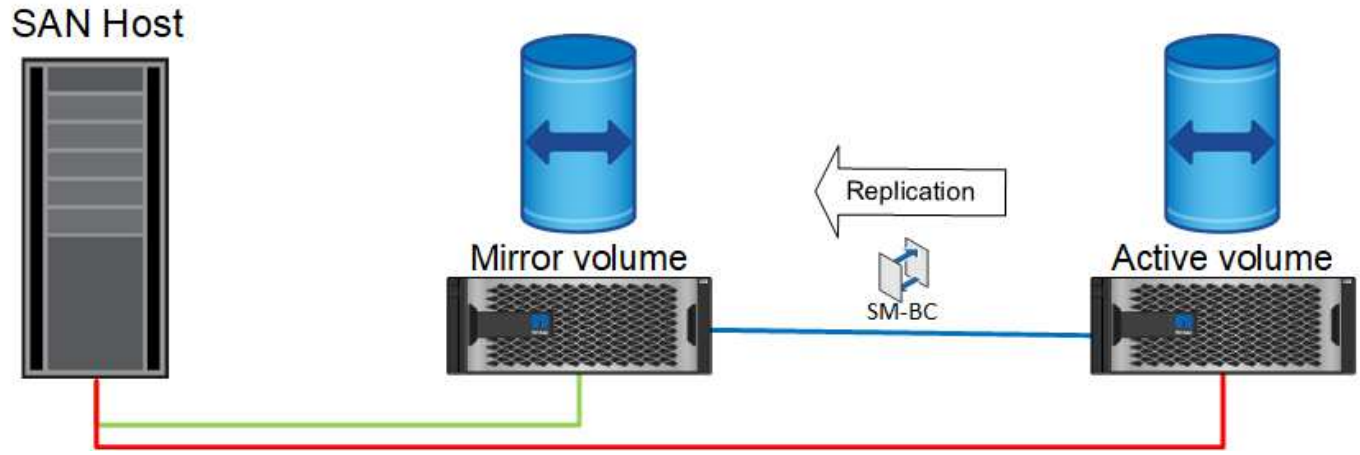


フェイルオーバー

フェイルオーバーを実行すると、リモートコピーがアクティブコピーになります。パスが[Active]から[Active]/[Optimized]に変更され、IOは引き続きデータ損失なしで処理されます。

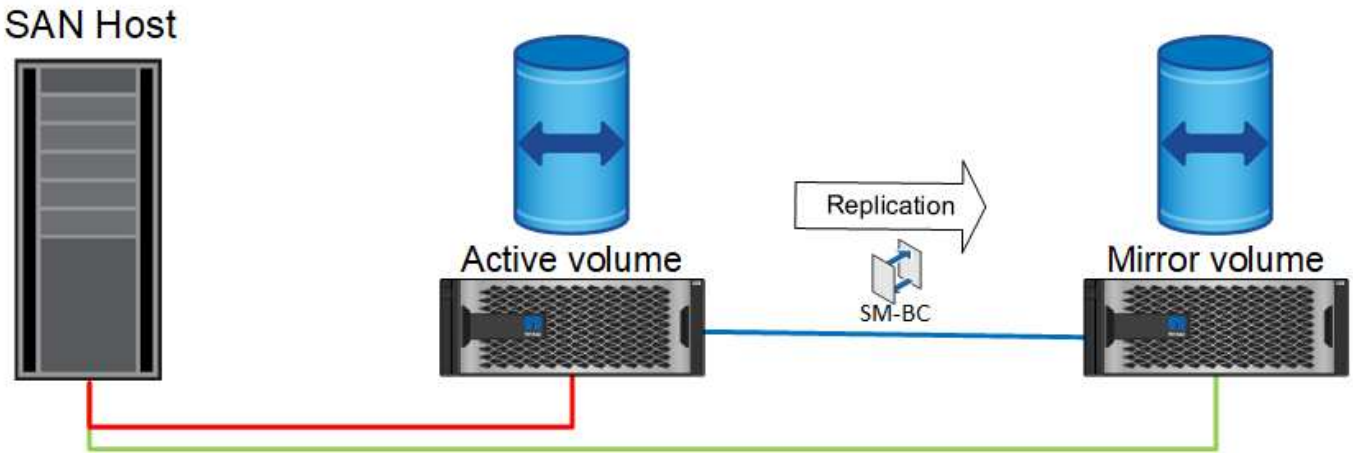
修復する

ソースシステムが稼働状態に戻ると、SnapMirrorアクティブ同期はレプリケーションを再同期できますが、逆方向に実行されます。現在の構成は、アクティブミラーサイトが反転されている点を除き、基本的には開始点と同じです。



フェイルバック

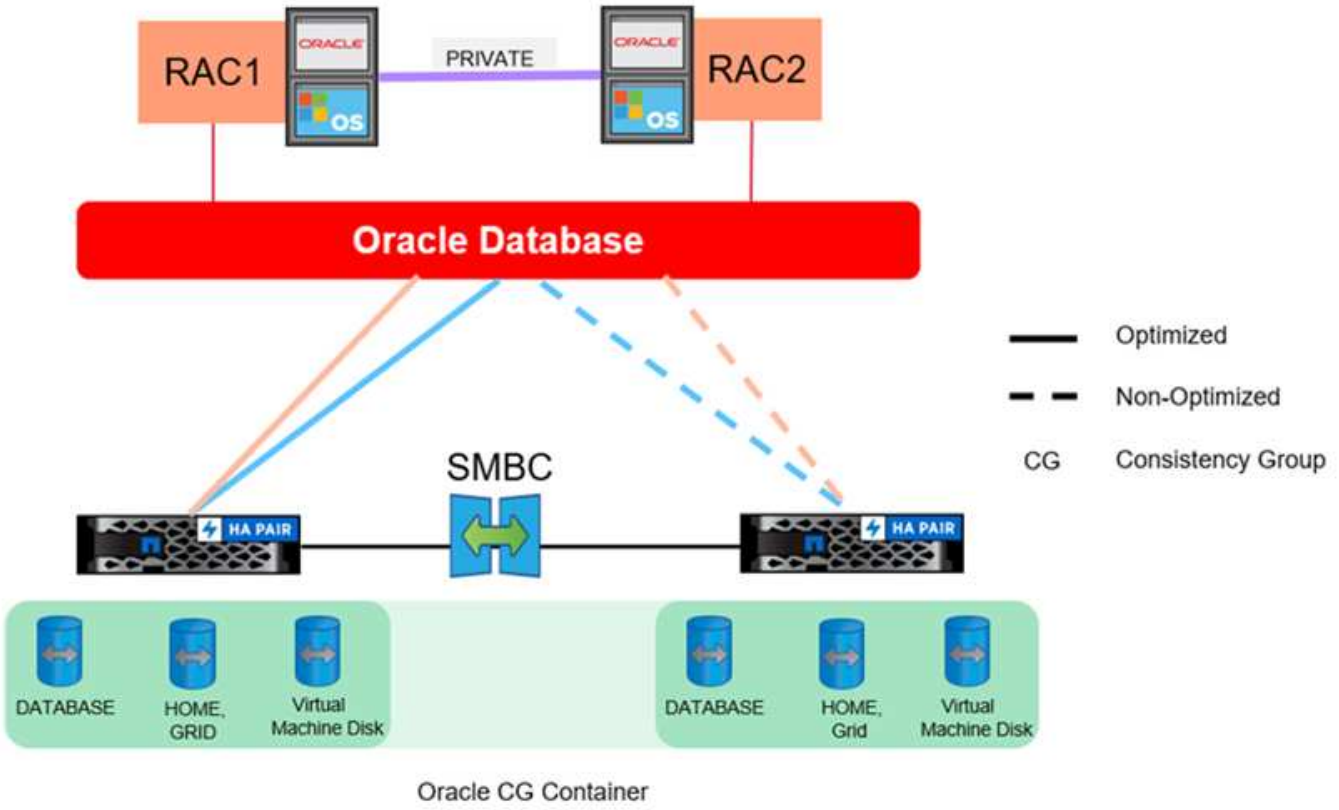
必要に応じて、管理者はフェイルバックを実行し、LUNのアクティブコピーを元のコントローラに戻すことができます。



シングルインスタンスの**Oracle**データベースと**SnapMirror**アクティブ同期

次の図は、Oracleデータベースのプライマリストレージクラスタとリモートストレージクラスタの両方からストレージデバイスをゾーニングまたは接続するシンプルな導入モデルを示しています。

Oracleはプライマリにのみ設定されます。このモデルは、ストレージ側で災害が発生した場合にシームレスなストレージフェイルオーバーに対応し、アプリケーションのダウンタイムを発生させることなくデータの損失をゼロにします。ただし、このモデルでは、サイト障害時のデータベース環境の高可用性を実現できません。このタイプのアーキテクチャは、データ損失ゼロの解決策でストレージサービスの高可用性を実現したいが、データベースクラスタが完全に失われた場合には手動での作業が必要である場合に便利です。



このアプローチでは、Oracleのライセンスコストも削減できます。リモートサイトでOracleデータベースノードを事前に設定するには、ほとんどのOracleライセンス契約に基づいてすべてのコアがライセンスされている必要があります。Oracleデータベースサーバのインストールと、稼働しているデータコピーのマウントにかかる時間が原因で遅延が発生しても問題ない場合は、コスト効率に優れた設計にすることができます。

Oracle RACとSnapMirrorのアクティブな同期

SnapMirror Active Syncを使用すると、ロードバランシングや個 々 のアプリケーションのフェイルオーバーなど、データセットのレプリケーションをきめ細かく制御できます。アーキテクチャ全体は拡張RACクラスタのように見えますが、一部のデータベースは特定のサイト専用で、全体の負荷は分散されます。

たとえば、6つのデータベースを個別にホストするOracle RACクラスタを構築できます。3つのデータベースのストレージは主にサイトAでホストされ、残りの3つのデータベースのストレージはサイトBでホストされます。この構成により、クロスサイトトラフィックが最小限に抑えられ、可能な限り最高のパフォーマンスが保証されます。また、ストレージシステムに対してローカルなデータベースインスタンスをアクティブパスで使用するようにアプリケーションを設定します。これにより、RACインターコネクトトラフィックが最小限に抑えられます。最後に、この全体的な設計により、すべてのコンピューティングリソースが均等に使用されます。ワークロードの変化に応じて、データベースをサイト間で選択的にフェイルバックして、ロードが均一になるようにすることができます。

きめ細かさを除けば、SnapMirror Active Syncareを使用するOracle RACの基本原則とオプションは "MetroCluster上のOracle RAC"

Oracleデータベースと SnapMirrorのアクティブな同期が失敗した場合

SnapMirror Active Sync（SM-AS）で障害が発生した場合は、それぞれ結果が異なりま

す。

シナリオ（ **Scenario** ）	結果
レプリケーションリンク障害	Mediatorはこのスプリットブレインシナリオを認識し、マスターコピーを保持するノードでI/Oを再開します。サイト間の接続がオンラインに戻ると、代替サイトで自動再同期が実行されます。
プライマリサイトストレージの障害	自動計画外フェイルオーバーはMediatorによって開始されます。 I/Oの中断はありません。
リモートサイトのストレージ障害	I/Oの中断はありません。ネットワークが原因で一時的に停止し、同期レプリケーションが中断され、マスターがI/O処理を継続する正当な所有者であることが確認されます（コンセンサス）。そのため、数秒間I/Oが一時停止してから、I/Oが再開されます。 サイトがオンラインの場合は自動再同期が実行されます。
MediatorまたはMediatorとストレージアレイの間のリンクの停止	I/Oは継続してリモートクラスタとの同期が維持されますが、Mediatorがないと、計画外フェイルオーバーや自動フェイルバックは実行できません。
HAクラスタ内の一方のストレージコントローラの停止	HAクラスタのパートナーノードでテイクオーバー（NDO）が試行されます。テイクオーバーに失敗すると、Mediatorはストレージの両方のノードが停止していることを認識し、リモートクラスタへの自動計画外フェイルオーバーを実行します。
ディスクノテイシ	IOは、連続して3つのディスク障害が発生しても継続されます。これはRAID-TECの一部です。
一般的な環境でサイト全体が停止する	障害が発生したサイトのサーバは、明らかに使用できなくなります。クラスタリングをサポートするアプリケーションは、両方のサイトで実行し、代替サイトで処理を継続するように設定できます。ただし、ほとんどのアプリケーションでは、SM-ASでメディエーターが必要な場合と同様の第3のサイトTiebreakerが必要です。 アプリケーションレベルのクラスタがない場合は、サバイバーサイトでアプリケーションを起動する必要があります。これは可用性に影響しますが、RPO=0は維持されます。データが失われることはありません。

著作権に関する情報

商標に関する情報