



ZFS STORAGE
APPLIANCE

Oracleテクニカル・ホワイト・ペーパー
2014年3月

Oracle ZFS Storage Applianceへの データ・マイグレーション

目次

目次.....	1
はじめに.....	2
データ・マイグレーション方法の比較.....	2
同期.....	2
同期によるデータ・マイグレーションの利点.....	5
不利点.....	5
制御 (Interposition).....	6
制御によるデータ・マイグレーションの利点.....	10
不利点.....	10
では、どちらが最善の方法でしょうか。.....	11
Oracle ZFS Storage Appliance Shadowのシャドウ・マイグレーション機能 を使用したデータ・マイグレーションの達成.....	11
シャドウ・マイグレーションを使用した、NFSソースからのデータ・マイ グレーション.....	12
性能に関する考慮事項.....	13
シャドウ・マイグレーションを使用した、ローカル・ソースからのデー タの移動.....	19
分析を利用したシャドウ・マイグレーションの監視.....	21
付録：参考資料.....	23

はじめに

Oracle ZFS Storage Applianceは、ストレージ性能と効率性を高める独自のOracle ソフトウェア統合を実現するために設計されたアプリケーション・エンジニアド・ストレージ・システムです。

Oracle ZFS Storage Appliance管理者が利用できるツールの1つに、シャドウ・マイグレーションがあります。シャドウ・マイグレーションを使用すると、管理者はOracle ZFS Storage Applianceに固有の特徴と機能を利用して、NFS (Network File System) プロトコルを使用するレガシー・システムからデータを移動したり、レガシー・システムで以前保持されていたファイル・システムを同じクライアント・ベースに対して提供したりすることができます。

このホワイト・ペーパーでは、シャドウ・マイグレーションをはじめとするデータ・マイグレーションを実行する方法の概要と、移行方法を選択する際に管理者が考慮することについて説明します。また、転送するファイル・システムに存在するファイル・サイズの種類ごとに、基本的な性能テストの結果についても示します。

このドキュメントの説明では、レガシー・ファイル・システムという用語を、データ・マイグレーション・プロセス開始時にレガシー・ストレージに存在する正式なデータソースと定義します。また、置換ファイル・システムという用語を、置換ストレージに通常存在する移行先と定義します。

マイグレーション・プロセスが正常に完了すると、置換ファイル・システムが正式なデータソースとなり、元のレガシー・ストレージは用途を変更したり、サービスを停止したりすることができます。

データ・マイグレーション方法の比較

従来から、データ・マイグレーション・プロジェクトでは、同期または制御という2つの方法が利用されています。次のセクションでは、異なるこれらの方法と、それぞれのおもな利点と不利点について説明します。

同期

同期は、一般的にはデータをレガシー・ソースから置換ソースに直接コピーすることで実行されますが、その間クライアントは主要なルートを経由して、引き続きそのデータにアクセスできます。

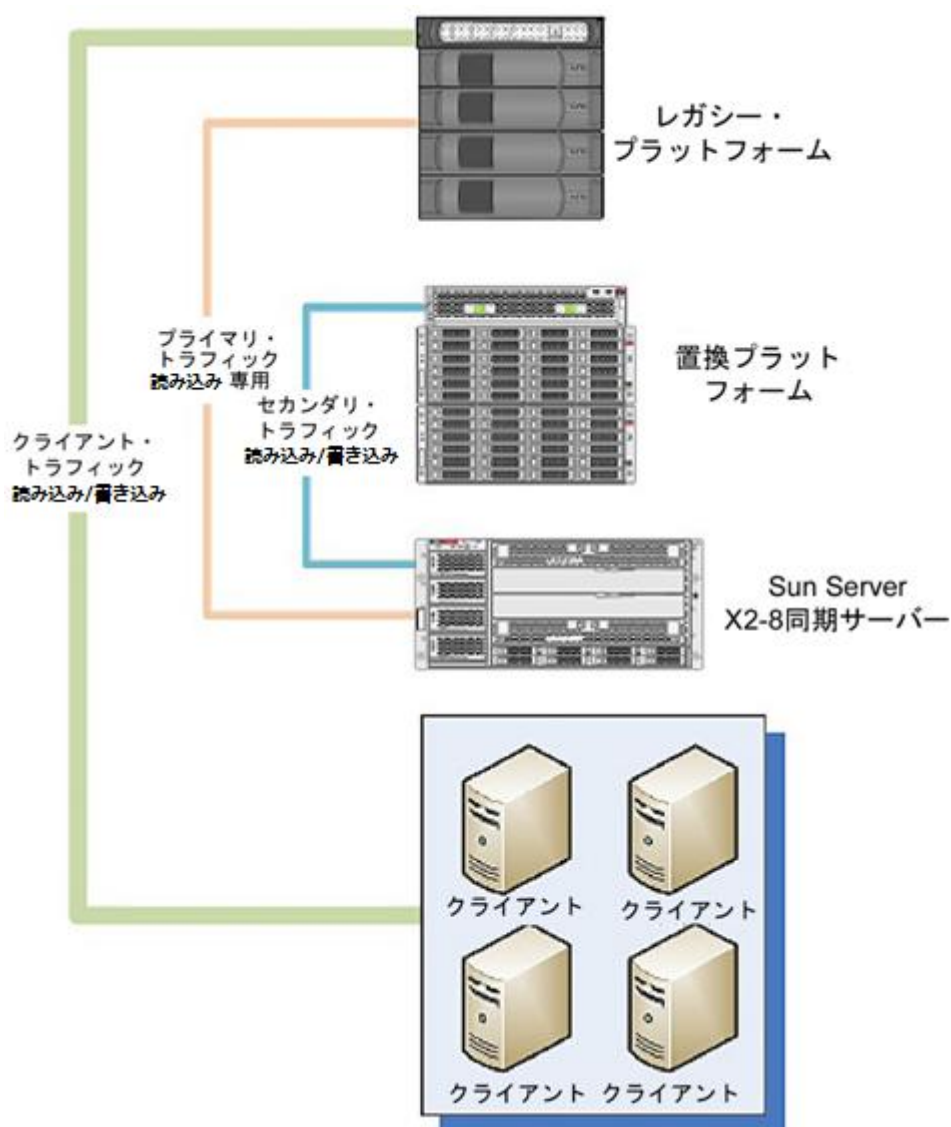


図1. 同期を使用したデータ・マイグレーションのアーキテクチャ

同期ユーティリティによって、通常レガシー・ソースと置換ソースの両方のデータの状態が検査され、コピーする最小のデータ量が決定されます。コピーが必要なのは、最後に同期ツールを実行した時点より後に行われた変更のみです。新しいファイルと更新されたファイルはコピーされ、削除されたファイルは除去されます。さらに新しいディレクトリが作成され、削除されたディレクトリは置換ソースから除去されます。

一部の同期ユーティリティでは、コピーが必要なファイルをチャンクに分割し、レガシー・ソースと置換ソースの両方でチャンクに対してチェックサムを行います。これらのチェックサムを比較して、コピーする最小のデータ量を決定します。チェックサムが一致した場合、そのチャンクをレガシー・ソースから置換ソースにコピーする必要はありません。またチェックサムが一致しない場合は、そのチャンクはネットワークを介してコピーされ、置換ファイルに適用されます。

同期の実行が開始されると、レガシー・ファイル・システムから置換ファイル・システムにコピーが必要なファイルのリストに対して、図2に例を示したようにアクション計画が作成されます。点線は、（現時点までは）空となっている置換ファイル・システムにコピーされる計画のファイルを示します。

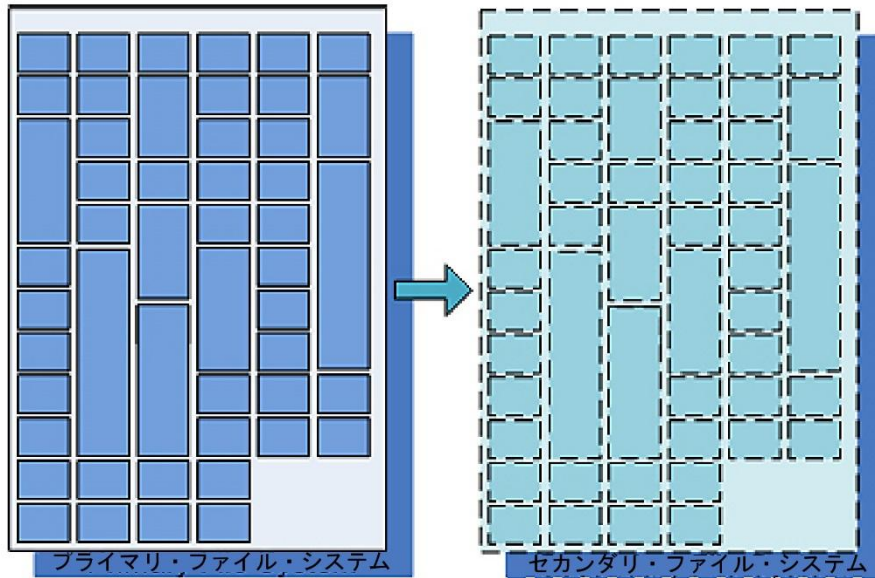


図2. ファイル・コピーの“アクション計画”

計画が作成された後、レガシー・ファイル・システムと置換ファイル・システムの間でファイルがコピーされます。レガシー・ファイル・システムに追加が発生しても、アクション計画には含まれないため、追加分はファイル・システム間でコピーされません。

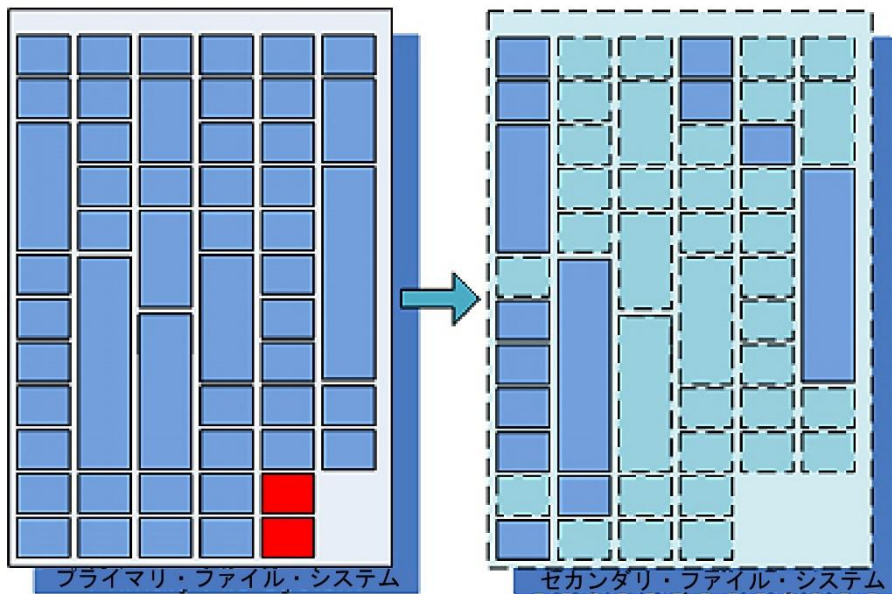


図3. 同期“コピー”フェーズ

図3に示すように、2つの新しいファイルがレガシー・ファイル・システムに作成されます（赤いボックスで示しています）。アクション計画には、これに対応するボックスはありません。

ファイル選択プロセスの間または後で、レガシー・データソースが変更された場合、その変更は同期が実行されている間に完全に失われます。

アクション計画に含まれるレガシー・ファイル・システムのファイルが変更された場合は、その変更は、コピー・フェーズがそのファイルに達する前に行われないとコピーされません。同期プロセスでそのファイルが処理された後に変更が行われると、その変更は失われます。そのため、場合によっては、同期プロセスを複数回実行して、完了していない変更を取得して、処理する必要があります。そのため、同期プロセスを反復プロセスとし、ファイル・ソースの変動のしやすさに応じて反復回数を設定します。

ただし、最終的には、レガシー・ソースへのアクセスを停止して、すべての変更を置換ソースに確実に適用する必要があります。このプロセスを正常に終了させるには、レガシー・ソースに対する更新アクションは、一切許可すべきではありません。監査やコンプライアンス要件のために、レガシー・ソースのファイルに対して更新アクセスの時間が必要な場合は、読み込み専用アクセスも禁止する必要があります。

この停止時間に最後の同期を行い、レガシー・ソースと置換ソース間での変更が不要になるまで、同期プロセスを適用する必要があります。

同期の実行が“クリーン”の状態になったら、レガシー・ソースを無効にして、クライアントは置換ソースを正式なソースとして参照する必要があります。

両方のソースのIPアドレスを交換すると、管理スタッフの負担を軽減しながら切り替えることができます。管理スタッフは、すべてのクライアントに、新しい正式なソースを参照させるようにしなくて済みます。

同期によるデータ・マイグレーションの利点

この方法のもっとも重要な利点は、初回の同期プロセス後に、レガシー・ソースと置換ソースに2つのコピーが保持できることです。置換ソースは、最後に同期を実行した時の状態であるため、レガシー・ソースよりも古い可能性があります。ディザスタ・リカバリ・ポリシーに対しては十分であると考えられます。

同期プロセスは、最後の同期の停止時間にコピーする必要があるデータ量を削減するために、複数回実行することが可能です。スイッチオーバーを‘許可’するかどうかは、最後の同期を実行する直前に判断できます。

不利点

初回の実行を除き、すべての同期プロセスの実行時間は、ファイル・システムの変動のしやすさにより大きく異なります。最後の同期に必要な停止時間は、明確な方法で定量化するのは困難（または不可能）な場合があります。最悪の場合には、ファイル・システムのコンテンツすべてが、新しく作成されたファイルで変更されてしまうこともあります。このような場合、最後の停止時間は、初回の同期の実行に必要な停止時間よりも長くなる可能性があります。

この最後のコピーの間には、クライアント・アクセスは一切許可できず、また、許可された停止時間枠に対しても許容できません。

同期を使用する際の別の欠点として、最後の停止時間の同期を実行するまで、クライアントが古いストレージや古いサーバーを使用し続けてしまい、その間、追加の機能とサービスを備えた置換ストレージや置換サーバーが完全にアイドル状態となってしまう点があげられます。つまり、最後の同期プロセスの実行が終了するまで、投資収益が得られません。

レガシー・プラットフォームで同期自体が実行できない場合は、同期サーバーがネットワーク上でソース・データを読み込み、そのデータを置換プラットフォームに再送信するため、ネットワークのホップが増加します。

制御（Interposition）

制御を使用する場合は、レガシー・ソースとレガシー・ソースのクライアント・ユーザーとの間にデバイスを配置します。このインタポーザ・デバイスは、クライアントに対して透過的に、バックグラウンドでレガシー・ソースから置換ソースへの移行を実行します。

インタポーザが配置されている間、レガシー・ファイル・システムを変更してはなりません。行われた変更は無視されるか、レガシー・ファイル・システムの変更箇所によっては、対象のファイルが破損する場合があります。レガシー・ファイル・システムは、インタポーザ以外のデバイスと共有しないのが理想的です。

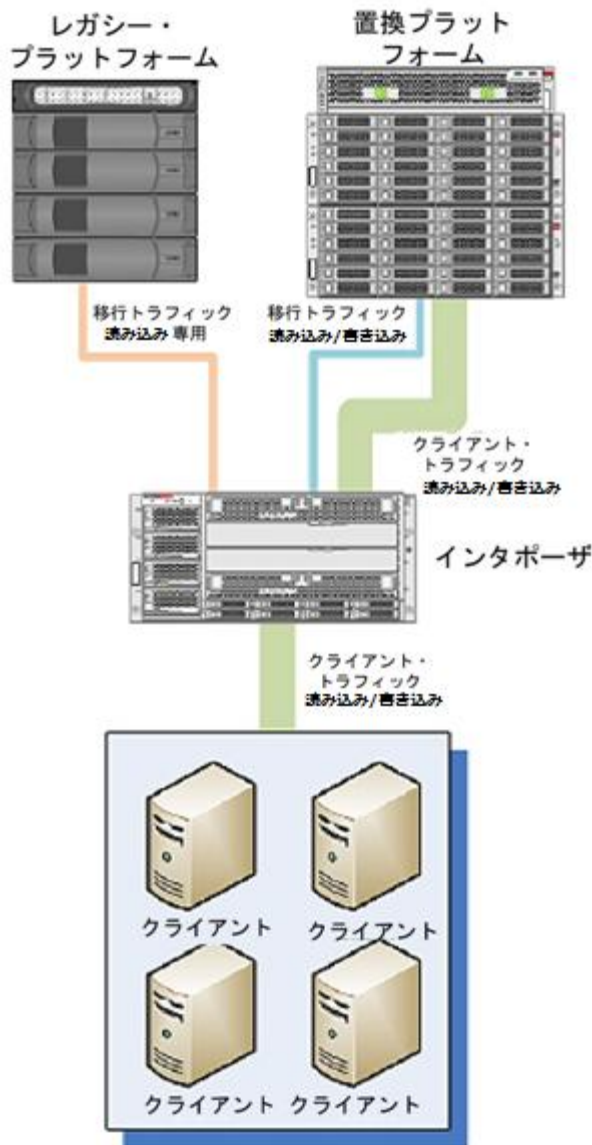


図4. 制御を使用したデータ・マイグレーションのアーキテクチャ

マイグレーション・プロセスの開始時に、クライアントがインタポーザ・デバイスをポイントするための停止時間が必要です。

マイグレーション・プロセスの間、インタポーザ・デバイスは、対象のデータがすでにレガシー・ソースから置換ソースに移行しているかどうか、および要求されているアクセスの種類（読み込み専用または読み込み/書き込み）に従って、データソースを選択する必要があります。

ファイルが新しいプラットフォームにコピー済みであれば、新しいプラットフォームによってI/Oリクエストはすべて受け付けられます。ファイルが新しいプラットフォームにコピー済みでなければ、マイグレーション・プロセスは最初にファイルを新しいプラットフォームに移行してから、I/Oリクエストの進行を許

可します。I/Oリクエストは、新しいプラットフォームによって受け付けられます。

したがって、この方法では、I/Oリクエストのレイテンシが長くなる可能性があります。ただし、このことが当てはまるのは、ファイルが自動プロセスによりまだ移行されていないか、または移行開始後、I/Oリクエストの対象になっている場合のみです。対象のファイルが大きい場合は、制御は許容できない手法となる可能性があります。これは、I/Oリクエストの進行を許可するまでに、ファイルの移行で発生するレイテンシが長くなるためです。

ファイルが小さい場合は、このことは問題にならない可能性があります。小さなファイルは比較的高速にコピーされるためです。アプリケーションで小さなファイルを大量に必要とする場合は、アプリケーションが初めてそれらのファイルにアクセスするときに、レイテンシが非常に長くなる可能性があります。

注：すべてのファイルを移行するには、インタポーザがそれらのファイルを読み込み可能である必要があります。よって、インタポーザ・デバイスには、レガシー・ファイル・システムへの有効な'root'アクセス権が必要です。また、これらのファイルおよびフォルダに対するアクセス権とアクセス制御リストも確実に置換ファイル・システムにコピーするために、レガシー・プラットフォーム、置換プラットフォーム、およびインタポーザを同じネーム・サービス構成にする必要があります。

移行が完了すると、インタポーザ・デバイスを除去できます。完全に移行された置換ソースをクライアントに参照させるためには、停止時間が必要です。この置換ソースが、データの正式なコピーとなります。

制御マイグレーションが開始されると、クライアントはレガシー・ファイル・システムに置換ファイル・システムを“介して”アクセスします。置換ファイル・システムでは、レガシー・ファイル・システムに存在するすべてのファイルを参照できます。これらのファイルは置換ファイル・システムに存在しているように見えますが、実際にはデータは転送されていません。このことを、図5に点線で表しています。

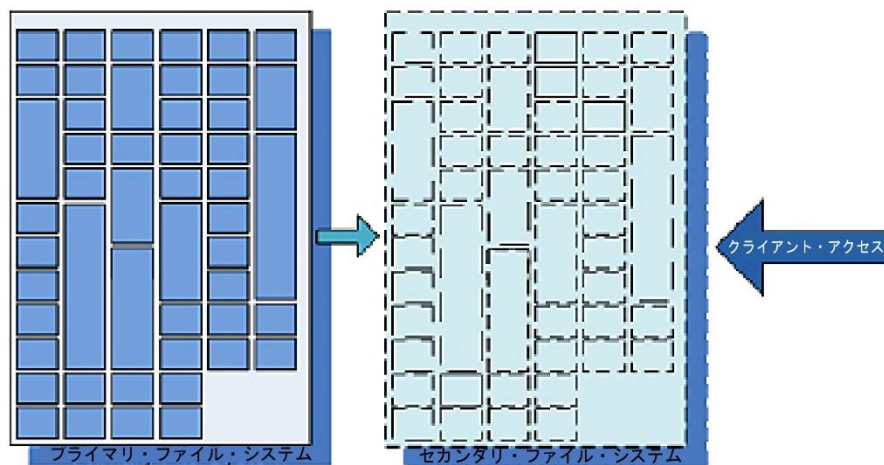


図5. 制御の開始

移行の進行に伴って、ファイルがレガシー・ファイル・システムから置換ファイル・システムにコピーされます。ファイルの読み込みリクエストにより、2つのシーケンスのうちのいずれかが行われます。どちらが行われるかは、ファイルがレガシー・ファイル・システムから置換ファイル・システムにすでに移行されているかどうかによって決まります。

- ファイルが置換ファイル・システムにすでに移行されている場合は、I/Oリクエストは遅延なく処理

され、置換ファイル・システムにより受け付けられます。この例を、図6に示します。

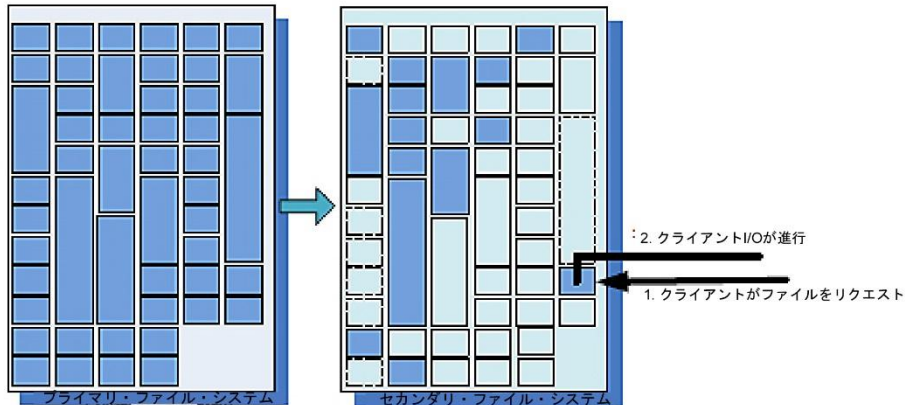


図6. インタポーザによる読み込み

- ケース1: ファイルが置換ファイル・システムにすでに移行されている場合

- ファイルがまだ移行されていない場合は、インタポーザがファイルの移行を開始している間、I/Oリクエストは一時停止されます。ファイルの移行が完了すると、I/Oリクエストは進行を許可され、置換ファイル・システムにより受け付けられます。この例を、図7に示します。

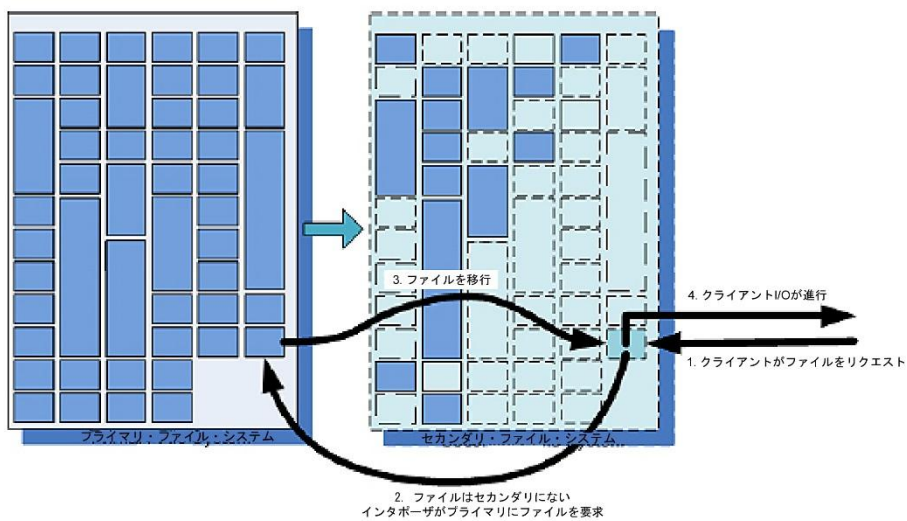


図7. インタポーザによる読み込み

- ケース2: ファイルがプライマリ/レガシー・ファイル・システムからまだ移行されていない場合

ファイルを変更すると、(ファイルがまだコピーされていないと仮定して)マイグレーション・プロセスによってレガシー・ファイル・システムから置換ファイル・システムにコピーされ、その変更が置換ファイル・システムで実行されます。同様に、新しいファイルが作成されると、レガシー・ファイル・システムを変更することなく、そのファイルは置換ファイル・システムに作成されます。図8に、このシナリオを示します。新しいファイルは赤で示しています。

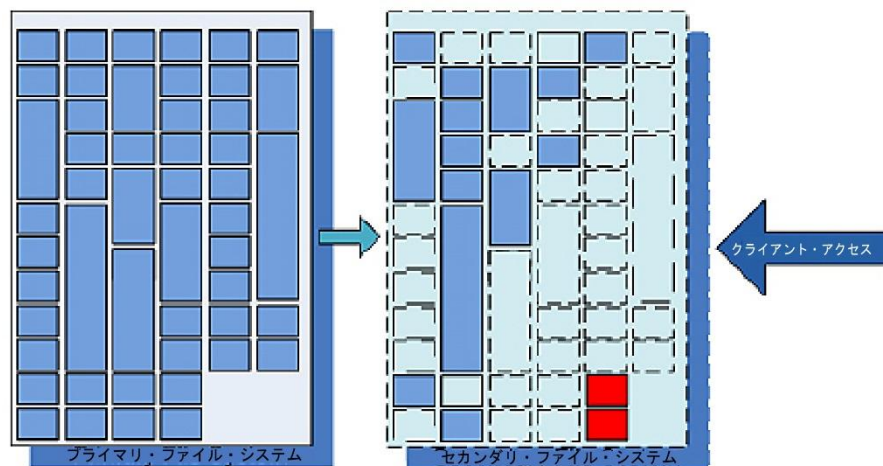


図8. 制御進行中。セカンダリ/置換ファイル・システム上に新しいファイルを直接作成

この間、クライアント・アクセスは、置換ファイル・システムによって受け付けられます。最終的には、すべてのファイルが置換ファイル・システムにコピーされると、制御マイグレーションは自動的に停止し、レガシー・ファイル・システムと置換ファイル・システム間の関係を維持する必要はなくなります。

制御によるデータ・マイグレーションの利点

いったんデータ・マイグレーションが開始されれば、クライアントは移行先のプラットフォームの機能とサービスを利用可能となります。

停止時間が必要なのは、クライアント・アクセス・パスとしてインタポーザ・デバイスを参照するためにクライアント構成を変更するときのみです。また、この停止時間に、インタポーザ・デバイスを削除したり、新しい置換ファイル・システムを参照するようクライアント構成をさらに変更したりすることもできます。

不利点

インタポーザ・デバイスは、マイグレーション・プロジェクトの終了後、必要もなく長期間放置されることが多くなります。新しいストレージ・プラットフォームにクライアントを参照させるために必要な比較的短い停止時間が、長引いてしまうこともあります。インタポーザを介する間接的な処理が、各リクエストに対して引き起こす追加のレイテンシはわずかですが、プロジェクト全体の期間では、このレイテンシが長時間となることがあります。インタポーザ・デバイスによって、置換プラットフォームでは利用できないその他のデータ・サービスが提供されるのであれば、移行終了後はこのような追加のレイテンシは完全に無駄な時間となります。

インタポーザ・デバイスがクライアントと、レガシー・プラットフォームおよび置換プラットフォームの間のデータ・パスに配置された時点で、マイグレーション・プロジェクトに専心する必要があります。デバイスが配置されると、すべての作業を元に戻すことは難しくなります。また置換プラットフォームだけに存在する新しく追加したファイルおよび変更したファイルは、すべてレガシー・プラットフォームにコピーして戻す必要があります。

では、どちらが最善の方法でしょうか。

この質問には、“状況による”と答えるしかありません。対象のファイル・システムに対するアクセスのパターンと、そのファイル・システムのファイル・サイズの分布によって、選択肢が異なります。データ・マイグレーション・プロジェクトには、両方の方法を導入できます。ただし、同じファイル・システムに両方の方法は導入できません。

非常に大きなファイルが存在する場合は、同期のほうが適している可能性があります。これは、特に読み込み専用モードでファイルにアクセスする場合に当てはまります。比較的変動の大きいファイルがマイグレーション・ソースに多数存在している場合は、停止時間の要件が定量化できる制御モードのほうが、同期よりもはるかに許容しやすいことがあります。

ただし、あらゆる状況に対応できる完全な方法はありません。ローカル・ポリシーとアクセス・パターンを検討して、移行する各ファイル・システムへの影響が最小になるのはどちらの方法であるかを判断する必要があります。

Oracle ZFS Storage Appliance Shadowのシャドウ・マイグレーション機能を使用したデータ・マイグレーションの達成

Oracle ZFS Storage Applianceは、インタポーザ・デバイスとして機能できます。そのおもな利点の1つに、マイグレーション・プロセスの終了時に削除する必要がないことが挙げられます。Oracle ZFS Storage Applianceでは、シャドウ・マイグレーションと呼ばれるサービスにより、このような制御による移行機能を提供します。

シャドウ・マイグレーションを使用すると、ソースがOracle ZFS Storage Applianceに対してローカルとして定義されている場合に、NFSを使用するレガシー・プラットフォームからデータ・マイグレーションしたり、プール間でデータ・マイグレーションしたりすることができます。

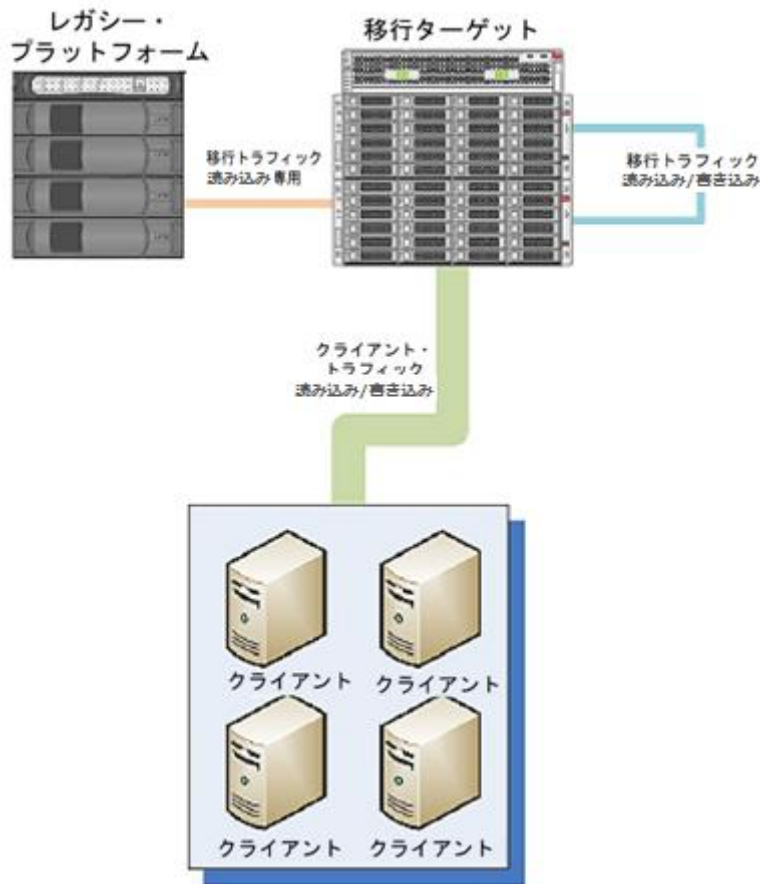


図9. Oracle ZFS Storage Applianceシャドウ・マイグレーションのアーキテクチャ

シャドウ・マイグレーションを使用した、NFSソースからのデータ・マイグレーション

Oracle ZFS Storage Applianceを使用してレガシー・ソースからデータ・マイグレーションする場合は、レガシー・プラットフォームによって提供される性能が、可能な限り高い帯域幅をOracle ZFS Storage Applianceに対して提供できるよう構成することを強くお勧めします。このようにすると、すべてのクライアント・アクセスの参照先が、レガシー・プラットフォームからOracle ZFS Storage Applianceへと切り替えられるだけでなく、レガシー・プラットフォームとOracle ZFS Storage Appliance間のネットワーク・リンクを競合から解放し、可能な限り最大の帯域幅を提供できるよう構成できます。具体的にこれを実現するには、専用のEthernetスイッチをこれらのプラットフォームの間に配置して、これらのプラットフォームのみサービスを提供するようにします。

可能であれば、マイグレーション・プロセスのために、プラットフォームに複数のインターフェースを割り当てます。ただし、どの構成でも、Oracle ZFS Storage Applianceへのクライアント・アクセスを許可する必要があります。これは、Oracle ZFS Storage Applianceは、レガシー・プラットフォームを置換すると、クライアント側のサービスになるためです。移行するボリュームも、すでに説明したように読み込み専用として構成す

る必要があります。

図10は、移行中に可能な物理ネットワーク構成を図で示しています。

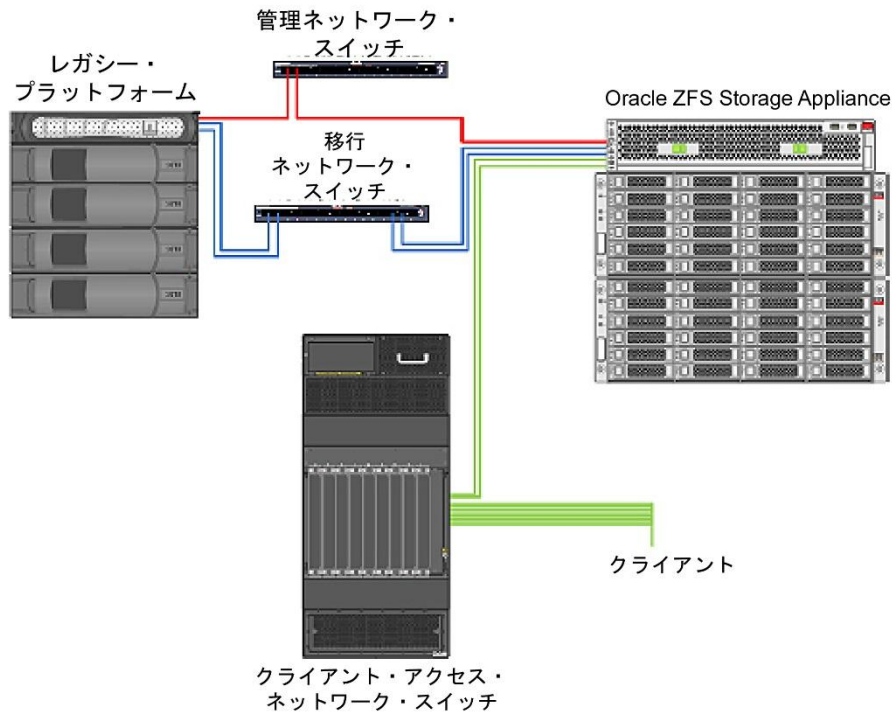


図10. 可能な物理ネットワーク構成

クライアント・アクセスを無効にするには、クライアントがアクセスできるスイッチまたはルーターからレガシー・プラットフォームを削除します。この操作の結果、場合によっては、両方のプラットフォームのIPアドレスを再構成して、マイグレーション・プロセスをサポートするために作成された一時的なネットワークを反映する必要があります。

クライアントは、置換Oracle ZFS Storage Applianceのサービスを使用するよう再構成する必要があります。または、Oracle ZFS Storage Applianceを、レガシー・デバイスのネットワーク・アドレスを取得するよう構成することもできます。

性能に関する考慮事項

シャドウ・マイグレーションの性能は、次のさまざまな要因によって異なります。

- ファイル・サイズのプロファイル
- ファイルの数
- 移行ネットワーク上でのネットワークの競合
- 移行ネットワークのネットワーク構成

- 移行先プールのデータ保護の構成
- ログとキャッシュの構成
- 移行先プールのデバイスの数
- レガシー・デバイスの性能
- 移行中のファイル・システムの変動
- 移行中の既存データへのアクセスのパターン

最適なマイグレーション性能を実現する最初の手順は、ソース・デバイスと移行先デバイス間のネットワーク・パスの競合をなくすことと、ネットワーク・インタフェースとネットワーク・デバイスによって、可能な限り最高の帯域幅を提供することです。そのためには、図10の構成で示すように、一時的にネットワーク・スイッチを使用して、マイグレーション・トラフィックのみを伝送することもあります。

管理者アクセス権をレガシー・プラットフォームと置換プラットフォームの両方で維持し続けて、管理者が性能とハードウェア障害の発生を監視できるようにします。

また、複数のインタフェースを1つの仮想インタフェースとして構成することもできます。Oracle ZFS Storage Applianceファミリでは、複数のインタフェースをIPマルチパス（IPMP）として構成するか、LACPアグリゲーションでより高度な構成にします。これらの構成のメリットを得るには、レガシー・プラットフォームでも類似の構成をサポートする必要があります。利用可能な方法と、用語はメーカーによって異なります。

マイグレーション・プロセス中のクライアント・アクセス要件を慎重に検討してから、すべてのネットワーク・デバイスをマイグレーション・プロセスに割り当てます。

ファイル・サイズ・プロファイルは、マイグレーション性能に著しく影響する可能性があります。非常に小さなファイルが大量にあるほうが、非常に大きなファイルが全体的に同じサイズである場合よりも、移行に時間がかかります。この原因となるのは、正しい属性でファイルを作成するオーバーヘッドや個々のファイルの開閉という単純な操作です。

どの程度移行時間が変わる可能性があるかを例として示すために、次のファイル・プロファイルに基づいてテストを実施し、64GBのデータを使用して移行時間を測定しました。

表1: マイグレーション・テストのファイル・プロファイル

ファイル・プロファイル	ファイルの数	説明
GB	64	同一ファイル。すべて1,073,741,824バイト
MB	65,536	同一ファイル。すべて1,048,576バイト
Mx (混在)	48,683	"実際の環境"のアプリケーション・ファイルを混在。平均サイズは1,410,187バイトで、標準偏差は19,018バイト 範囲は0バイトから2,146,435,072バイト

移行するファイルのプロファイルを多様化するだけでなく、シャドウ・マイグレーション・サービス・プロファイルに64、32、16および8スレッドを設定してそれぞれのテストを実行しました。

重複排除と圧縮は無効にして、ボリュームを同じセットのディスクから同じプール構成に移行し、その他の変動要因が加わらないようにしました。

ここに示した各プロファイルでは転送するデータ量は同じにしました。ただし、結果として作成されるシェアのサイズはわずかに異なりました。これは、ファイル数の違いによりディレクトリのサイズに差が生じたためです。

ソースのNFSのバージョンも複数テストして、差異が生じる可能性を示しました（以下のグラフでは、バージョン番号nをvnと簡略化して示しています）。NFSクライアントのマウント・パラメータは、Oracle ZFS Storage Applianceでは変更できません。よって、デフォルトまたはネゴシエーションされた値を使用する必要があります。

最後に、Maximum Transmission Unit (MTU) サイズは、デフォルトの1500バイトから9000バイトに変更されました。9000バイトは、Oracle ZFS Storage Applianceで利用可能な最大値です。

注：MTUのリセットは、簡単には決定できません。競合や衝突が頻繁に発生する環境や、ケーブルに影響する干渉がデータ・ストリームを破損させてしまうことがある環境では、失われたパケットの再送信が必要となるために、MTUのリセットは性能に悪影響を及ぼす可能性があります。この例では、導入したスイッチはマイグレーション・トラフィックのみを伝送し、ケーブルの長さを極めて短くしました。

マイグレーション・トラフィックをホストするために使用したスイッチは、実行したすべてのテストで、10GBの全二重に設定しました。

図11に示すグラフでは、64GBのデータの移行にかかった全体的な時間、図12では各構成で得られた平均スループットを示します。これらの図のポイントは、絶対的な時間と帯域幅ではなく、いくつかの変数の変更によって発生する相対的な差を示していることです。

選択した最大のファイル・サイズは、NFSバージョン2がサポートするファイル・サイズの制約により、ちょうど2GB未満でした。

9000MTUでの転送時間

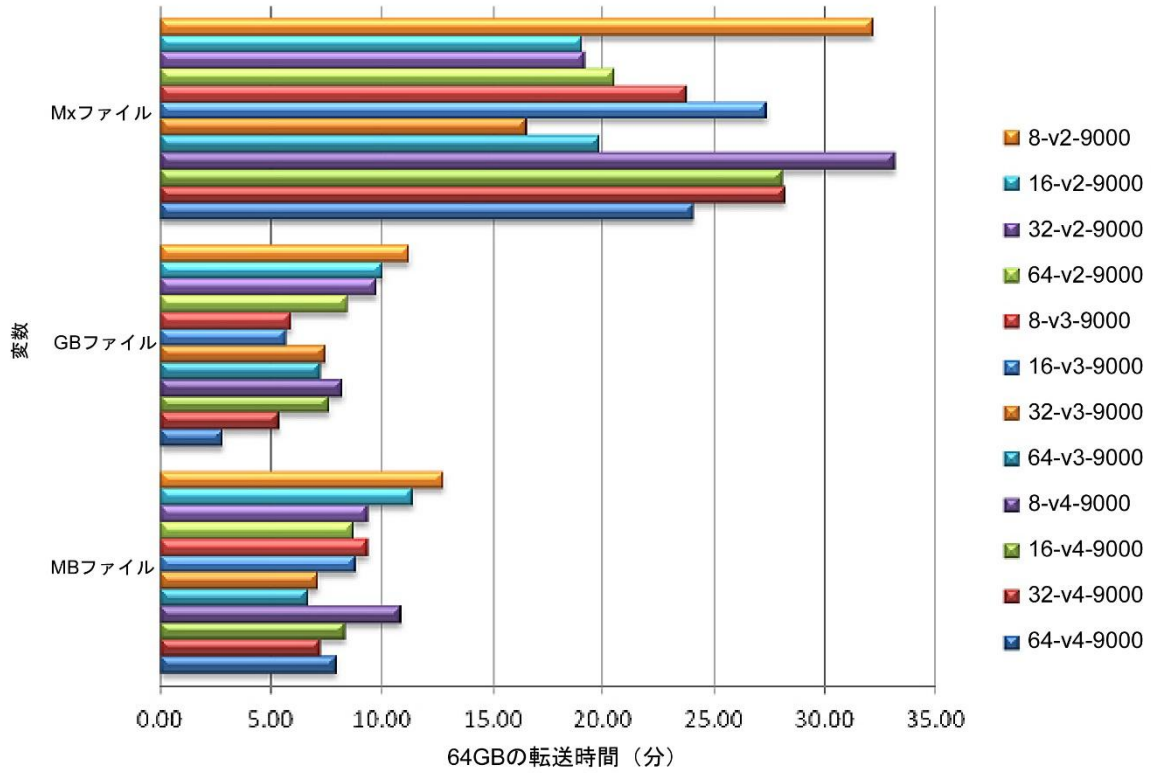


図11. MTUが9000の場合に、シャドウ・マイグレーションを使用した64GBのデータ・マイグレーションにかかった時間

データ系列の凡例の形式：

<スレッドの数> - v<NFSのバージョン> - <MTU>

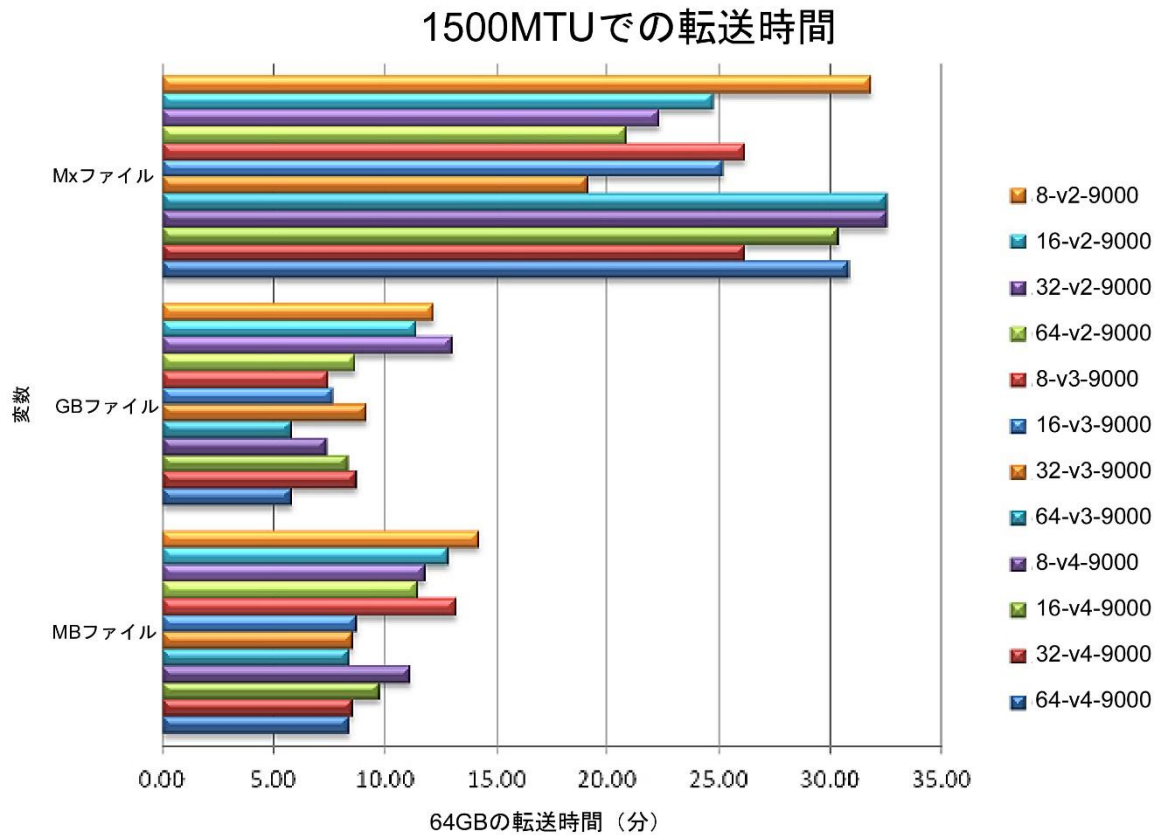


図12. MTUが1500の場合に、シャドウ・マイグレーションを使用した64GBのデータ・マイグレーションにかかった時間

これらの図が示すように、ファイル・サイズの分布とネットワーク構成によって、性能が異なります。ネットワーク・インタフェースでMTUを変更すると、転送時間が向上しますが、MTUの増加がスループットの向上につながるという保証はありません。特に、MTUを増やすことで、1500バイトではなく9000バイトのパケットの再送信が必要になるような頻りに競合が発生する環境では、その保証はありません。

次の図は、MTUが1500の場合と9000の場合での転送レートの差をパーセンテージで示しています。

1500から9000MTUに増加した際の スループットの増加率 (%)

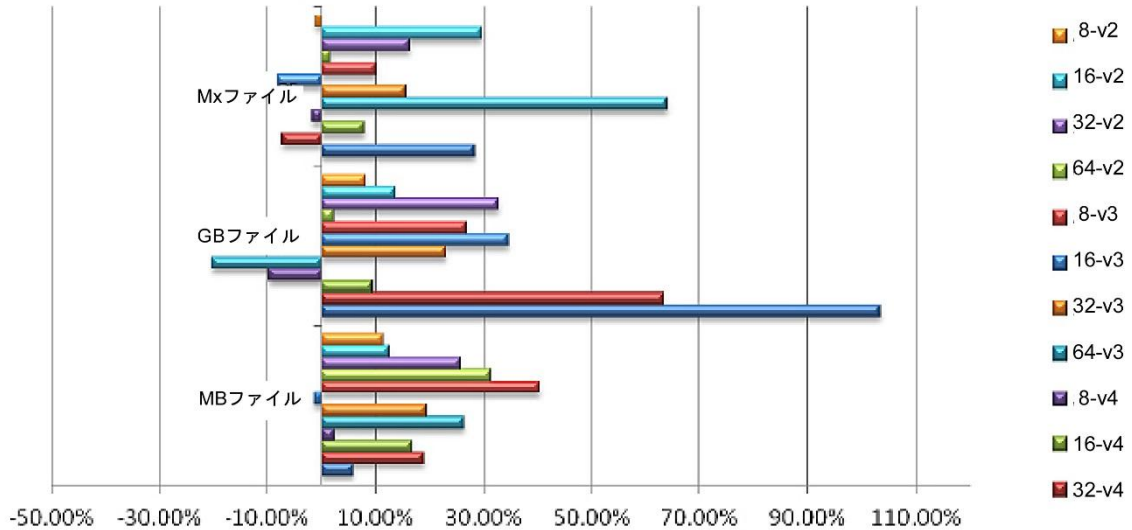


図13. 1500MTUと9000MTUでの差 (%)

もっとも優先する必要があるのは、NFSv4用にレガシー・プラットフォームを構成することです。特にこの環境では、複雑なACL（アクセス制御リスト）を展開しており、維持する必要があります。これらのACLを意味のあるものにするために、レガシー・ストレージとOracle ZFS Storage Applianceには、同じディレクトリ・サービス、つまり、LDAP、Active Directory、Network Information Service (NIS) などに対するアクセス権が必要です。

Oracle ZFS Storage Appliance内のログ・デバイスとキャッシュ・デバイスは、マイグレーション性能だけでなく、移行中、クライアントからのデータ・アクセスの性能にも大きな効果を発揮できます。シャドウ・マイグレーションは、すべての書き込みを同期的に実行し、ログ・デバイスのアプリケーションから大きなメリットを生み出します。

図14は、ログ・デバイスの使用の有無のみを変えた条件で、シャドウ・マイグレーションを実行した場合の比較を示しています。

64GBの移行時間（分）

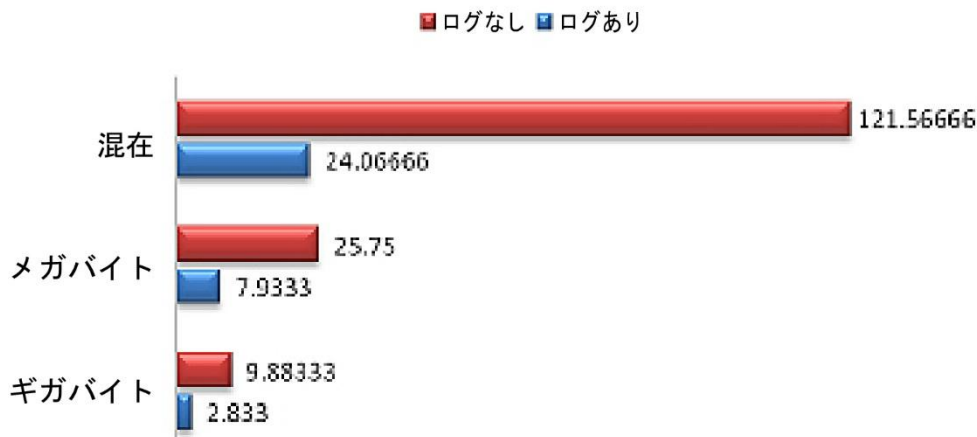


図14. ログ・デバイスを使用した場合と使用しなかった場合に、64MBのデータ・マイグレーションにかかった時間

シャドウ・マイグレーションを使用した、ローカル・ソースからのデータの移動

シャドウ・マイグレーションは、Oracle ZFS Storage Appliance内で1つのプールから別のプールにデータを移行する目的でも使用できます。シャドウ・マイグレーションは、たとえば、RAID 10プールからRAID Z2への移動など、高性能のプールから大容量のプールにデータを移動するための非常に有用なツールとなる可能性があります。データを引き続き保持する必要がある一方で、データに高性能のサポートが不要となるなど、アクセスのパターンが変化した場合に役立つ可能性があります。

また、シャドウ・マイグレーションを使用して、他の方法では変更できない設定を変更することもできます。たとえば、ファイル・システムのレコード・サイズやファイル・システムの圧縮バージョンの作成などに使用できます。

ユースケースに関係なく、データ・マイグレーションの方法は同じです。シャドウ・マイグレーションは動的なソース・ボリュームを処理するには設計されていないため、（アクセスする必要がある場合でも）クライアントは、ソース・ボリュームに読み込み専用モードでのみアクセスする必要があります。

NFSベースのシャドウ・マイグレーションに使用したのと同じファイル・サイズの分布を使用して、性能テストを続行したところ、いずれのボリューム設定も変更することなく、ソースと同じプールに対してローカル・マイグレーションが実行されました。

この場合も、以下の結果は、さまざまなファイル・サイズの分布に対する相対的な時間であり、絶対的な性能の例ではありません。これらのグラフは、NFSベースの移行のグラフと同じ尺度で示しています。

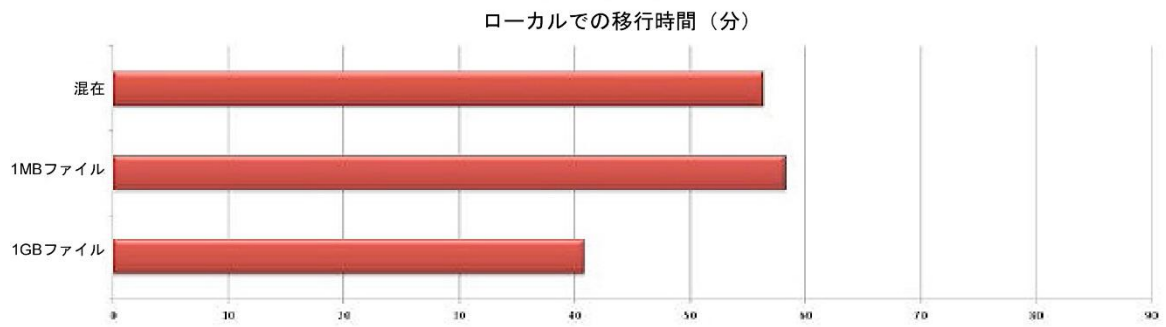


図15. Oracle ZFS Storage Applianceで64GBのデータのローカル・マイグレーションにかかった時間

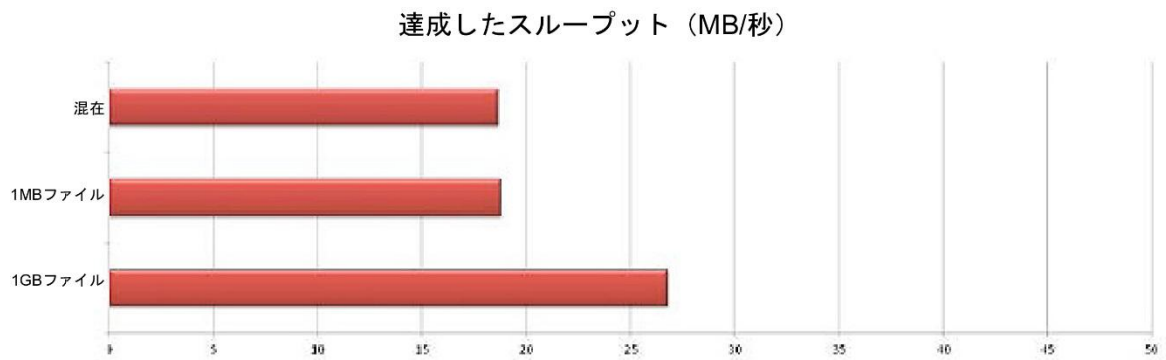


図16. Oracle ZFS Storage Appliance上のテスト・ファイルのローカル・マイグレーションにおけるスループット速度

分析を利用したシャドウ・マイグレーションの監視

Oracle ZFS Storage Applianceでは、ストレージ管理者がOracle ZFS Storage Applianceの性能とクライアントのI/Oリクエストの現在の状態や、ネットワーク、ディスクおよびCPU負荷を表示できる、強力な分析機能を提供します。ブラウザ・ユーザー・インタフェース（BUI）による詳細な性能メトリックにより、Oracle ZFS Storage Appliance内の各コンポーネントに対するドリルダウン・ビューを利用できるため、エンド・ツー・エンドの性能特性をきわめて詳細に把握できます。

シャドウ・マイグレーションには、さまざまな性能メトリックがあります。

- シャドウ・マイグレーションのバイト数
 - ファイル名別
 - プロジェクト別
 - シェア別
 - 生の統計情報として
- シャドウ・マイグレーションの運用
 - ファイル名別
 - プロジェクト別
 - シェア別
 - レイテンシ別
 - 生の統計情報として
- シャドウ・マイグレーションのリクエスト
 - ファイル名別
 - プロジェクト別
 - シェア別
 - レイテンシ別
 - 生の統計情報として

これらの性能インジケータと連動し、ネットワーク負荷も重要な要因となる場合があります。

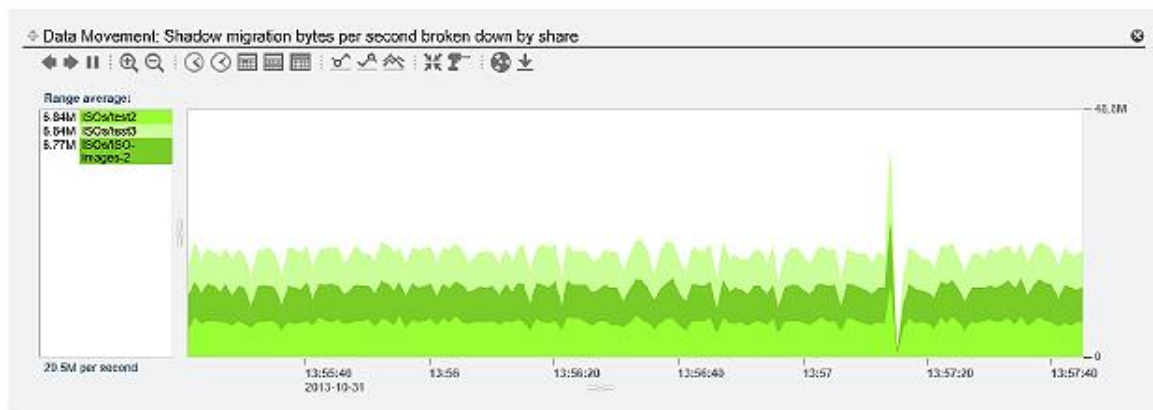


図17. Oracle ZFS Storage Appliance BUIでのシャドウ・マイグレーション分析の画面

図17に、同時に移行した3つのボリュームに関する分析のトレースを示します。

結論

Oracle ZFS Storage Applianceの機能であるシャドウ・マイグレーションは、管理者ツールの選択肢を増やす、価値あるツールです。このツールを使用すると、自動的に測定可能な方法で、レガシー・プラットフォームからOracle ZFS Storage Applianceへの移行を実行できるだけでなく、必要に応じてOracle ZFS Storage Appliance内での移行も実行できます。

どの標準的なツールも、他のツールよりも優れた方法で、すべてのファイル・サイズやファイルの種類に対応できるわけではありません。同じように、シャドウ・マイグレーションも、移行の問題をすべて解決できるわけではありません。シャドウ・マイグレーションが他のツールよりも勝るのは、マイグレーション性能の監視です。Oracle ZFS Storage Applianceの先進的な分析機能により、個別のファイル・レベルではなく、シェアごとに進行状況を監視できます。また、クライアントのI/Oアクセスのパターンも監視できます。このレベルの情報を得ることで、移行が完了するまでに必要な合計時間を高い精度で見積もることができます。

付録：参考資料

注：Sun ZFS Storage Appliance、Sun ZFS Storage 7000、およびZFS Storage Applianceという名称はすべて、同じファミリのOracle ZFS Storage Appliance製品を指しています。引用したドキュメントまたは画面のコードの中には、従来の命名規則を使用しているものがあります。

- Oracle ZFS Storage Appliance製品ドキュメント
<http://www.oracle.com/technetwork/jp/documentation/oracle-unified-ss-193371.html>
- 『Sun ZFS Storage Appliance Administratin Guide』はSun ZFS Storage Applianceのヘルプ・コンテキストで参照できます。
Oracle ZFS Storage Applianceのヘルプ機能には、ブラウザ・ユーザー・インターフェースからアクセスできます。
- Oracle ZFS Storage Appliance 製品情報
<http://www.oracle.com/jp/storage/nas/overview/index.html>
- Oracle ZFS Storage Applianceのホワイト・ペーパーおよびテーマ別のリソース
<http://www.oracle.com/technetwork/jp/server-storage/sun-unified-storage/documentation/index.html>



Oracle ZFS Storage Applianceへの
データ・マイグレーション
2014年3月、バージョン1.0

著者 : Andrew Ness
Application Integration Engineering

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U. S. A.

海外からのお問い合わせ窓口 :
電話 : +1. 650. 506. 7000
ファクシミリ : +1. 650. 506. 7200

www.Oracle.com



Oracle is committed to developing practices and products that help protect the environment

Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

本文書は情報提供のみを目的として提供されており、ここに記載される内容は予告なく変更されることがあります。本文書は一切間違いがないことを保証するものではなく、さらに、口述による明示または法律による黙示を問わず、特定の目的に対する商品性もしくは適合性についての黙示的な保証を含み、いかなる他の保証や条件も提供するものではありません。オラクル社は本文書に関するいかなる法的責任も明確に否認し、本文書によって直接的または間接的に確立される契約義務はないものとします。本文書はオラクル社の書面による許可を前もって得ることなく、いかなる目的のためにも、電子または印刷を含むいかなる形式や手段によっても再作成または送信することはできません。

OracleおよびJavaはOracleおよびその子会社、関連会社の登録商標です。その他の名称はそれぞれの会社の商標です。

AMD、Opteron、AMDロゴおよびAMD Opteronロゴは、Advanced Micro Devicesの商標または登録商標です。IntelおよびIntel XeonはIntel Corporationの商標または登録商標です。すべてのSPARC商標はライセンスに基づいて使用されるSPARC International, Inc.の商標または登録商標です。UNIXはX/Open Company, Ltd.によってライセンス提供された登録商標です。1010

Hardware and Software, Engineered to Work Together