



テクニカル レポート

ONTAPを使用したOracleマルチテナンシー 解決策と導入のベストプラクティス

NetApp

Ebin Kadavy / Jeffrey Steiner

2021年1月 | TR-4876

概要

このテクニカルレポートでは、NetApp® ONTAP® ストレージを使用してOracleマルチテナントデータベースをプロビジョニング、管理、保護し、OracleマルチテナントデータベースとONTAPソフトウェアの機能のメリットを最大限に活用する方法について、解決策のベストプラクティスの概要を紹介します。

<<本レポートは機械翻訳による参考訳です。公式な内容はオリジナルである英語版をご確認ください。>>

目次

はじめに.....	4
Oracleマルチテナントアーキテクチャ	4
Oracleマルチテナントアーキテクチャのメリット.....	5
Oracleのマルチテナントの制限とライセンス.....	5
ストレージのレイアウトとプロビジョニング.....	5
概要.....	5
共有ストレージ.....	5
専用ストレージ.....	6
SANパスとOracleマルチテナント.....	8
Oracle ASMLib	9
Oracle ASMフィルタドライバ (ASMFD)	10
プロビジョニングとデータファイルの場所.....	10
アップグレードと移行.....	12
ONTAP を使用したレガシーデータベースからPDBへの変換	12
例	13
ストレージの移行、データセンターの統合、再配置.....	16
プラガブルデータベースのパフォーマンスに関するベストプラクティス	18
自動ワークロードリポジトリ (AWR) レポートのプラガブルデータベースの有効化.....	18
データ保護とコピーデータ管理.....	19
NetApp SnapCenter	19
PDBリフレッシュとPDBスイッチオーバー.....	19
異機種混在環境でのPDBクローン	21
付録A:スキーマ統合データベース.....	21
付録B : OracleマルチテナントデータベースでのOracle EBS	21
付録C : 文書の分類とラベル	22
追加情報の入手方法.....	22
バージョン履歴.....	22

図一覧

図1) Oracleマルチテナントアーキテクチャのレイアウト	4
図2) 共有ストレージレイアウト	6
図3) マルチテナントSAN専用アーキテクチャ	7
図4) マルチテナント専用NFSアーキテクチャ	7
図5) 空のコンテナデータベースを選択するためのOracle RUNインストーラの導入オプション	11
図6) NetApp SnapMirrorを使用した新しいインフラへのデータベース移行の一般的なアーキテクチャ	14
図7) SnapMirror DR関係を解除してレプリケーションを停止し、DRコピーを本番環境として昇格	15
図8) レガシーインフラの廃止に成功し、新しいインフラで本番運用を開始	16
図9) データファイルのオンライン移動.....	17
図10) PDBの移動	18
図11) Oracle PDBの更新.....	20
図12) 異機種混在ストレージ環境におけるNetApp SnapCenterを使用したPDBクローン	21

はじめに

このテクニカルレポートでは、ONTAPストレージでの最新のOracleマルチテナントアーキテクチャに関する解決策とその導入のベストプラクティスについて簡単に説明します。このドキュメントにはOracleのマルチテナントシーに関する情報が含まれていますが、Oracleの公式ドキュメントに代わるものではありません。Oracleテクノロジの例と説明は、ストレージ構成のベストプラクティスの背後にある理由を示すためにのみ記載されています。

Oracleのマルチテナントアーキテクチャ

Oracleマルチテナントは、Oracle Database 12cR1で導入されたオプションです。統合、プロビジョニング、アップグレードなどを簡易化してITコストを削減できます。これにより、プラガブルデータベース（PDB）と呼ばれる複数の論理データベースを、コンテナデータベース（CDB）と呼ばれる単一の物理データベース内で実行できます。このマルチテナント機能では、Oracle Real Application Clusters（RAC）やOracle Data Guardなど、他のOracleテクノロジもサポートされます。PDBに変更を加えずに既存のデータベースを簡単に変換でき、アプリケーションの他の階層に変更を加える必要はありません。

図1) Oracleマルチテナントアーキテクチャのレイアウト

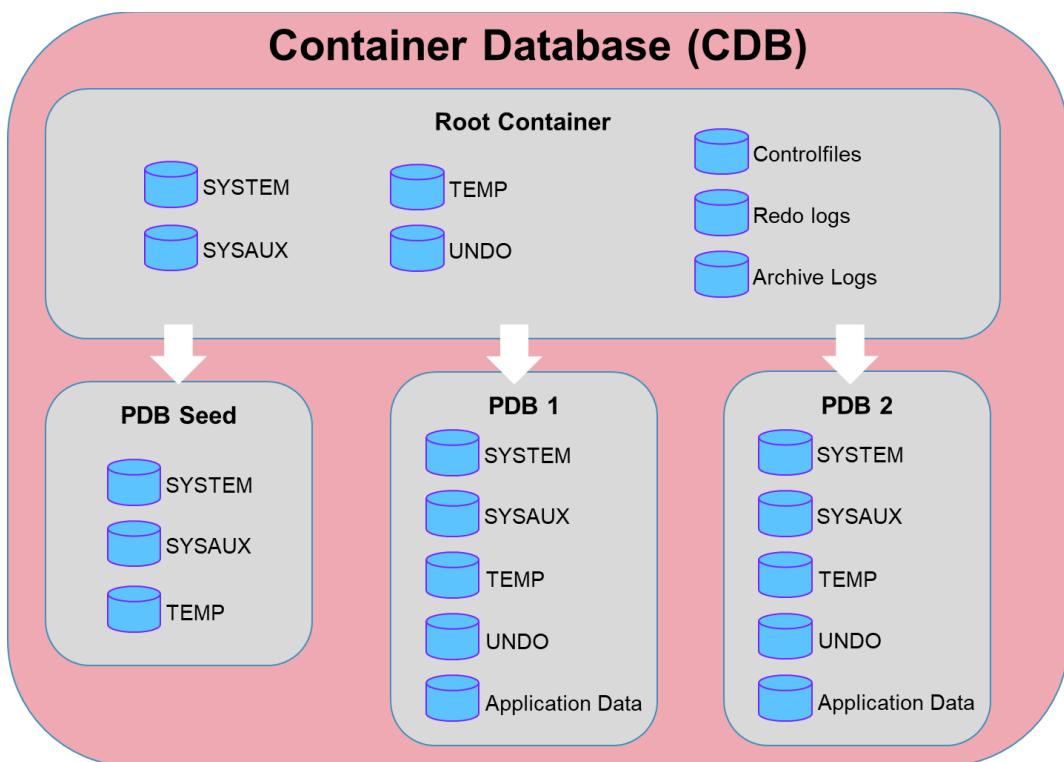


図1に示すように、マルチテナントデータベースには、データファイル、制御ファイル、REDOログなど、使い慣れた作業領域の大部分が含まれています。これらのオブジェクトのデータディクショナリはルートコンテナによって所有され、すべてのPDBに表示されます。CDBにはデータベースの作業部品の大部分が含まれているため、PDBにはそれ自体に固有の情報のみを含める必要があります。制御ファイル、REDOログ、アーカイブログなどのファイルは、格納先データベースに含まれています。PDBは、それ自身のデータファイルと関連オブジェクトだけで構成されます。

Oracleマルチテナントアーキテクチャのメリット

- **高密度の統合**：1つのマルチテナントCDB内の多数のPDBは、メモリとバックグラウンドプロセスを共有します。これにより、多数のデータベースをより小さくシンプルなハードウェアに統合できます。
- **迅速なパッチ適用とアップグレード**：データベースのアップグレードプロセスも簡素化されています。個々のPDBを現在のCDBから取り外し、上位バージョンのOracleのCDBに再接続することで、PDBをアップグレードできます。コンテナデータベース内のホストされているすべてのPDBをアップグレードするには、コンテナデータベースをアップグレードするだけで、ホストされているすべてのPDBが「インプレース」でアップグレードされます。
- **複数のデータベースを1つにまとめて管理**：複数のデータベースをPDBとして統合することで、管理者は多数のデータベースを1つのオブジェクトとして管理できます。たとえば、バックアップやディザスタリカバリなどのタスクはマルチテナントCDBレベルで実行され、すべてのコンステンティチュエントPDBに影響します。

Oracleのマルチテナントの制限とライセンス

非CDBアーキテクチャはOracle 12.1.0.2で廃止されました。PDBは、[Oracleデータベース19C](#)以上のバージョンでサポートされている唯一のアーキテクチャです。

Oracleのマニュアルに記載されている情報に基づき、[Oracle 19C](#)の最新リリースでは、1つのCDB内で最大252個のPDB for Enterprise Edition (EE) と3個のPDB for Standard Edition (SE2) がサポートされます。Oracle Database 19Cでは、CDBごとに最大3つのユーザ作成PDBがサポートされるようになりました。マルチテナントライセンスを追加する必要はありません。

Oracle [ノート742060.1](#)によると、Oracle Database 19Cは、12c、18c、および19Cファミリーのデータベースに対する長期サポートリリースです。つまり、Oracle Database 19Cは12c、18c、19cのイノベーションをすべて備えており、2024年まではプレミアサポート、2027年までは延長サポートが提供されています。こうした変化により、組織はマルチテナントデータベースを詳しく検討し、本番データベースを非CDBからCDB内のPDBに移行する計画を立て始めています。

ストレージのレイアウトとプロビジョニング

概要

ONTAPでのストレージプロビジョニングの基本単位はボリュームです。他のストレージベンダーとは異なり、ONTAPではLUNの参照に「ボリューム」という用語を使用しません。ONTAPボリュームは、単なるデータのコンテナです。たとえば、Automatic Storage Management (ASM) を使用するOracleデータベースには、特定のASMディスクグループを構成するLUNが16個含まれているとします。通常、これらのLUNは1つのボリュームに同じ場所に配置されます。これにより、ボリュームのバックアップ、リストア、クローニング、複製を1つのユニットとして実行できるため、管理タスクが簡易化されます。さらに、ボリュームにQuality of Service (QoS；サービス品質) ポリシーや効率化ポリシーを設定することもできます。このポリシーは、ボリュームに含まれるすべてのデータに適用されます。

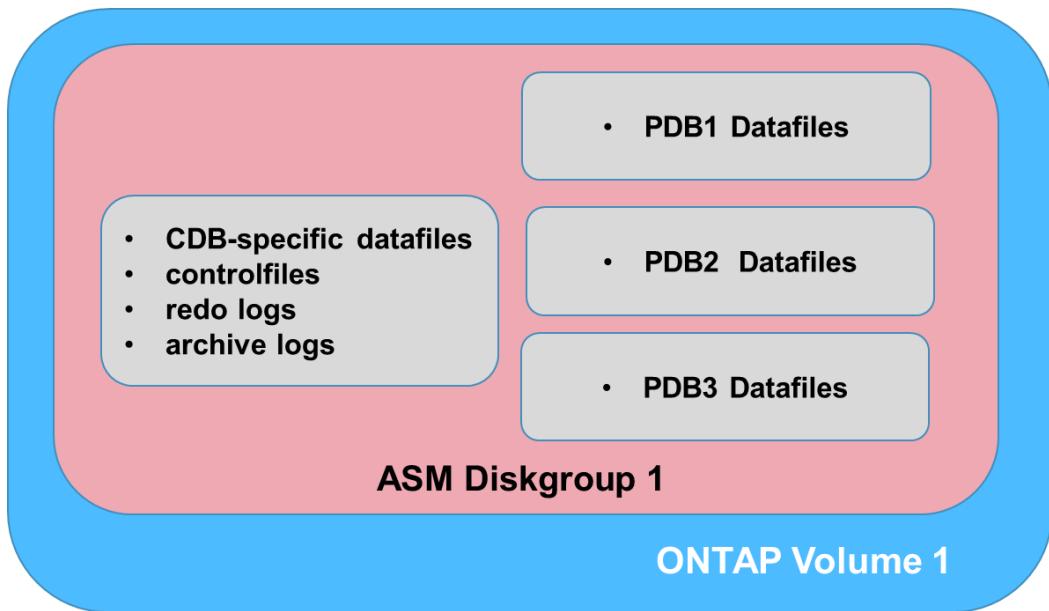
データは主にボリュームレベルで管理されるため、Oracleのマルチテナントデータベースに最適なストレージレイアウトは、データベースの管理要件によって異なります。

共有ストレージ

マルチテナントデータベースには、CDBとそのPDBの両方をホストするためのストレージが必要です。最も簡単な導入方法は、CDBファイルとPDBファイルを含むすべてのデータを同じボリュームにホストすることです。これは共有ストレージ環境と呼ばれ、ストレージの観点からはPDBの使用は無視されます。

たとえば、3つのPDBをホストし、すべてが同じASMディスクグループを共有するコンテナデータベースがあるとします。これは、小規模なデータベースや高度な管理機能を必要としないデータベースに適しています。たとえば、この構成では、図2に示すように、すべてのデータベースファイルがASMディスクグループの基盤となる同じLUNを共有しています。

図2) 共有ストレージレイアウト



このアプローチの主なメリットは、シンプルさです。パフォーマンスと容量に関して管理が必要なLUNのセットは1つだけです。バックアップは、Oracle Recovery Manager (RMAN) またはNetApp Snapshot™コピーを使用して実行できます。ただし、すべてのPDBを1つのユニットとしてリストアできる場合に限ります。QoSを使用してパフォーマンスを制御できますが、すべてのPDBに等しく影響します。

専用ストレージ

ONTAPの高度な機能を活用するには、データタイプに基づいてデータをより分離する必要があります。この場合のベストプラクティスは、図3および図4に示すように、管理目標によって異なります。

たとえば、CDBに3つのPDBがあり、データが次のように4つのグループに編成されているとします。

- CDBデータファイル、PDBシード、制御ファイル、REDOログなどのグローバルデータベースインフラ アーカイブログ
- 財務PDBデータファイル
- HR PDBデータファイル
- Data Warehouse (DW) PDBデータファイル

図3) マルチテナントSAN専用アーキテクチャ

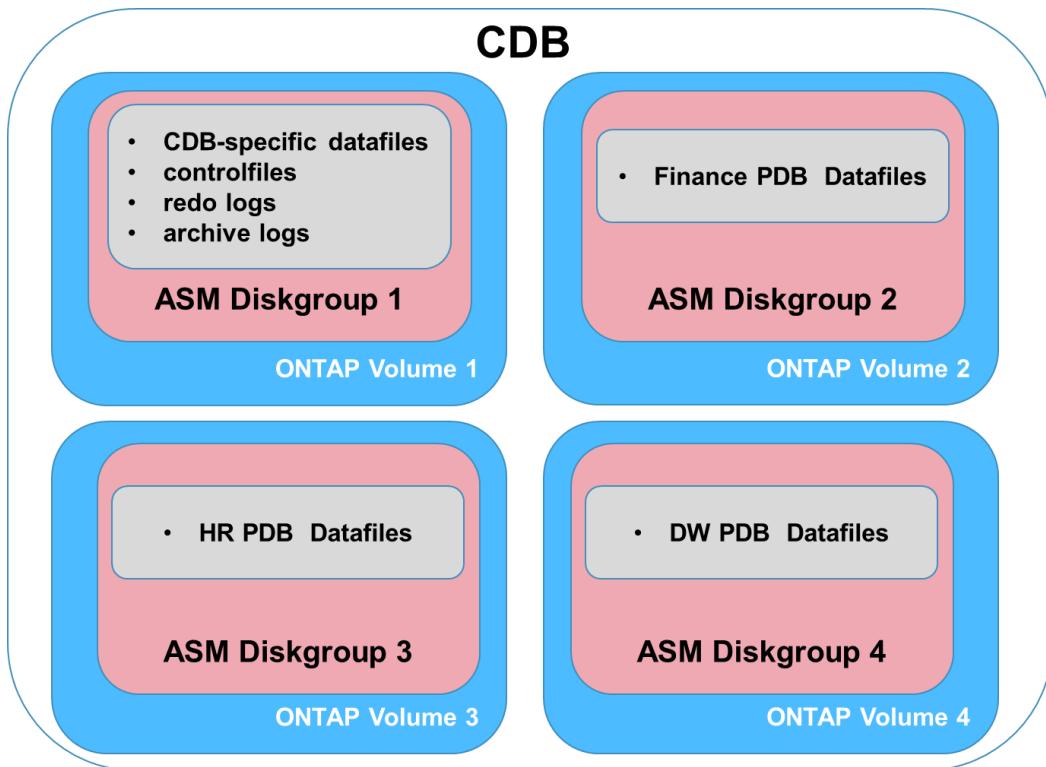
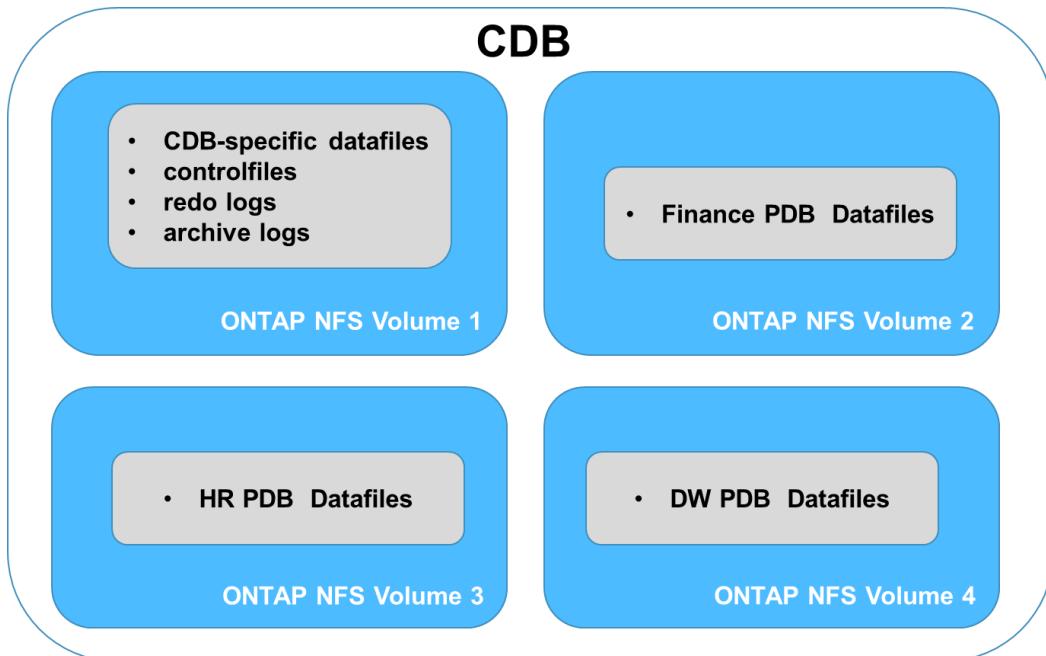


図4) マルチテナントNFS専用アーキテクチャ



このアプローチには、次のような複数の利点があります。

- ターゲットファイルが含まれているボリュームでONTAP QoS制御を使用して、ストレージパフォーマンス制限とパフォーマンスリザベーションを個々のPDBに選択的に適用できます。
- ボリュームのSnapshotコピーやボリュームクローンに不要なデータをキャプチャすることなく、個々のPDBを選択的にバックアップ、クローニング、またはレプリケートできます。
- 実行中の他のPDBに影響を与えることなく、個々のPDBを選択的にリストアできます。
- 実行中の他のPDBに影響を与えることなく、個々のPDBを他の物理サーバに再配置できます。
- 効率化ポリシーと暗号化をPDB単位で選択的に制御可能

他のオプションも存在します。たとえば、一時表領域のSnapshotコピーを作成する理由はありませんが、通常はそのデータをメインデータベースから分離する価値はありません。ただし、一部のデータベースでは大量の一時表領域が必要になるため、Snapshotポリシーが設定されていないファイルについては、PDBごとに追加のボリュームを作成することをお勧めします。

ベスト プラクティス

1. Oracleデータベースの保護に関する一般的な説明については、[TR-4591](#)を参照してください。
2. LUNの数が多すぎる場合や少なすぎる場合は、作成しないでください。LUNの数が多いと管理が複雑になりますが、数が少ないとパフォーマンスや管理上の制限が生じる可能性があります。一般に、SANファイル用のLUNの最小数は4、最大数は16です。非常に大規模なデータベースや非常にアクティブなデータベースには例外があります。
3. 事前にパフォーマンスSLAを設定し、QoSポリシーはボリュームレベルでもLUNレベルでも設定できることに留意してください。
4. データ保護に関するSLAを事前に設定し、Snapshot、クローニング、NetApp SnapRestore® の処理はフルボリュームレベルで実行するのが最適であることを忘れないでください。
5. Snapshotコピーを使用した個々のPDBのポイントインタイムリカバリが必要な場合は、CDBデータベースをPDBが専用のストレージリソースを使用する1つのPDBに制限する必要があります。
6. Snapshotコピーでも複数のPDBを含むCDBを効率的に保護できますが、ポイントインタイムリカバリプロセスではコピー処理が必要です。Snapshotコピーは、ポイントインタイムやSCNベースのリカバリではなく、完全なリカバリシナリオの場合はインプレースで瞬時にリストアできます。NetApp SnapCenter® / NetApp SnapManager®などの製品では、リストアライフサイクルに関連するエンドツーエンドのプロセスを自動化できます。

SANパスとOracleマルチテナンシー

共有SANストレージを使用してOracleのマルチテナント環境を構築すると、LUNの総数が増加します。これまでデータベースは、ストレージアレイで回転式ドライブを使用する方法に制限があるため、多数のLUNで構成されることが多くありました。ただし、これは最新のアレイ、特にオールSSDアレイではありません。PDBでこのような従来の手法を使用すると、原因でLUNとパス数に問題が発生する可能性があります。

LUNパスの数が多すぎると、管理が複雑になり、ストレージ制御コードのアップグレード、SANスイッチのリブート、または一時的なSANの問題の発生時にパスのフェイルオーバーが遅くなる可能性があります。ONTAPオールSANアレイ（ASA）は、対称アクティブ/アクティブ構成のSAN専用システムです。つまり、ASAは、両方のコントローラを経由するパスを基盤となるすべてのLUNにアドバタイズして使用します。このアーキテクチャの変更により、HAペアのいずれかのコントローラでホストされているすべてのLUNへのアクティブパスが常にホストに確保されます。

非ASA設定では、プライオリティが異なる2つのパスグループがアドバタイズされます。優先度が高いパスはアクティブ/最適化されます。つまり、LUNが使用するドライブを所有するコントローラによって処理されます。優先度が低いパスはアクティブですが、別のコントローラから提供されるため最適化されていません。最適化されていないパスが使用されるのは、最適化されたパスがない場合だけです。

ベスト プラクティス

1. Oracleデータベースのストレージ構成に関する一般的な説明については、[TR-3633](#)を参照してください。
2. LUNパスを大量に作成しないでください。

-
3. LUNごとに4つのアクティブパスがあれば、ほぼすべてのOracle SAN環境に対応でき、8個を超える必要はありません。
 4. 通常、ドライブを所有するコントローラ（アグリゲート）とそのHAパートナーの2台以上のコントローラでLUNがアドバタイズされることはありません。移行作業以外で他のコントローラにLUNを提供することは避けてください。
 5. LUNを使用できるポートのリストを簡単に定義する方法については、ポートグループに関するONTAPのドキュメントを参照してください。これにより、必要に応じて、ストレージシステムの進化に応じて、LUNをアドバタイズするコントローラを簡単かつ無停止で変更できます。
-

マルチテナント環境に対するマルチストレージベンダーのシナリオ

PDBは移植性が高いため、異なるベンダーのストレージアレイでホストされたPDBでデータベースを実行しなければならない場合があります。たとえば、一部のPDBを従来のSANフレームアレイで実行し、徐々にデータをONTAPに移行しているとします。そのためには、マルチパス要件が異なるベンダーのLUNを1台のサーバで使用する必要があります。

NetApp Host Utilitiesのドキュメントに詳細が記載されていますが、正しいHost Utilitiesパッケージがインストールされていれば、LUNは共存できます。

注意：Linuxではmultipath.conf構成ファイルに注意してください。マルチパスバイナリは、ONTAP LUNを適切に使用するために必要なすべてまたはほぼすべてのデフォルトでコンパイルされていますが、multipath.confファイルには、以前のストレージアレイの設定がデフォルトセクションに含まれていることがあります。

次に、上書きされたデフォルトを修正する例を示します。この場合 path_checker 、multipath.confファイルに定義されている、およびの値は no_path_retry ONTAP LUNと互換性がありません。他のSANアレイがホストに接続されているために削除できない場合は、デバイススタンザを使用してONTAP LUN専用にこれらのパラメータを修正できます。

```
defaults {  
    path_checker readsector0  
    no_path_retry fail  
}  
  
devices {  
    device {  
        vendor "NETAPP"  
        product "LUN.*"  
        path_checker queue  
        no_path_retry tur  
    }  
}
```

Oracle ASMLib

Oracle ASMを使用して複数のPDBを統合および管理するには、ASMのブロックサイズを把握しておく必要があります。ブロックサイズはASMディスクグループの作成時に設定され、あとから変更することはできません。

ASMLibはASMデバイス管理を容易にするオプションのライブラリですが、ブロックサイズには重要な依存関係があります。最新のストレージアレイのほとんどは、論理ブロックサイズが512バイト、物理ブロックサイズが4KのLUNをアドバタイズします。通常、論理ブロックサイズは512バイトです。これは、データセンターの運用履歴のほとんどでブロックサイズだったためです。ここ数年、通常の物理ブロックサイズが512バイトから4Kに変更されました。これは、パフォーマンスと効率性が向上するためです。データを少数のブロックで管理することを推奨します。

ほとんどのOracle環境では、512バイトのASMディスクグループと4KのASMディスクグループが混在しています。これにより、ASMLibが使用されている他のサーバにデータベースをプラグインする際に問題が発生する可能性があります。デフォルトの動作で構築されたASMディスクグループでPDBが実行されているコンテナデータベースでは4Kブロックが使用されますが、多くのお客様がデフォルトの動作を上書きし、ブロックサイズを512バイトに強制しています。これは通常、ASMのリバランスングベースの移動を有効にするためです。

/etc/sysconfig/oracleasm構成ファイルは1つだけあり、デバイスの論理ブロック・サイズまたは物理ブロック

ク・サイズを、そのサーバに接続されているすべてのデバイスに使用するかどうかを制御します。

たとえば、CDBがASMLibでネイティブの4KブロックLUNデバイスを使用するサーバ上に構築されており、既存の4K ASMディスクグループがあるとします。環境内の他の場所には、4Kデバイスを使用するASMディスクグループが存在する可能性がありますが、ASMLibのブロックサイズよりも優先されるため、512バイトのブロックサイズを使用します。これらのPDBをCDBに接続することはできません。上書きしない場合、ASMディスクグループのブロックサイズは512バイトになり、基盤となるデバイスは4Kになります。オンラインにすることはできません。この問題は、CDBサーバのブロックサイズを上書きすることで解決できますが、既存のASM 4Kディスクグループへのアクセスが切断されます。そのため、新しく作成したASMディスクグループにデータをコピーする必要があります。

可能であれば、将来のNVMeへの移行も含めて、将来のニーズに対応できる解決策を実現するために、4Kブロックデバイスを使用することを推奨します。

ASMLibとブロックサイズの詳細については、[TR 3633 : 『Oracle Databases on ONTAP』](#) を参照してください。

Oracle ASMフィルタドライバ (ASMFd)

Oracle ASM Filter Driver (Oracle ASMFd) は、Oracle ASMディスクのI/Oパスに存在するカーネルモジュールです。ASMFdは、Oracle ASMディスクへの書き込みI/O要求の検証に使用されます。また、Oracle ASMFdは、システムを再起動するたびにOracle ASMで使用するディスク・デバイスを再バインドする必要がないため、ディスク・デバイスの構成と管理を簡素化します。

ASMFdでは、「Oracle ASMLib」の項で説明したPDB管理のブロックサイズに関する考慮事項と同じです。

ASMFdはOracleのマルチテナンシーに直接関連付けられているわけではありませんが、多くの場合、データベースをPDBに変換したり、PDBを使用して新しいOracle環境に導入したりするときに設定されます。

ASMFdには、最近のLinuxの多くのバージョンとの互換性がありません。最近のカーネルでは、マルチパスデバイスに送信されるI/Oサイズ制限が適用されますが、ASMFdではこれらの制限が適用されません。その結果、ASMFdで構成されたデータベースサーバはドライブを読み取ることができなくなります。

解決策では、ONTAP LUNのmultipath.confファイルに転送長を手動で指定します。

```
devices {
    device {
        vendor "NETAPP"
        product "LUN.*"
        max_sectors_kb 4096
    }
}
```

注：現在問題がない場合でも、このパラメータはASMFdを使用するように設定する必要があります。

Oracle、Linux、またはONTAPをアップグレードすると、マルチパスデバイスが変更されて問題が発生する可能性があります。

マルチプレックスネットワーク

パフォーマンスが重大な問題になる場合は、Redoログを多重化しないことを推奨します。RAID-DPまたはRAID-TECはメディア障害からの保護を提供するため、冗長性を確保する必要はありません。Redoログを多重化することで、OLTPワークロードの負荷が高い場合、パフォーマンスが最大で約5~10%低下しました。運用ポリシーでREDOログの多重化が必要な場合は、グループメンバーを別々のディスクグループまたはファイアウォールシステムに配置することを推奨します。たとえば、2つのディスクグループ、REDOSSD1 および REDOSSD2を作成し、それらを多重化できます。

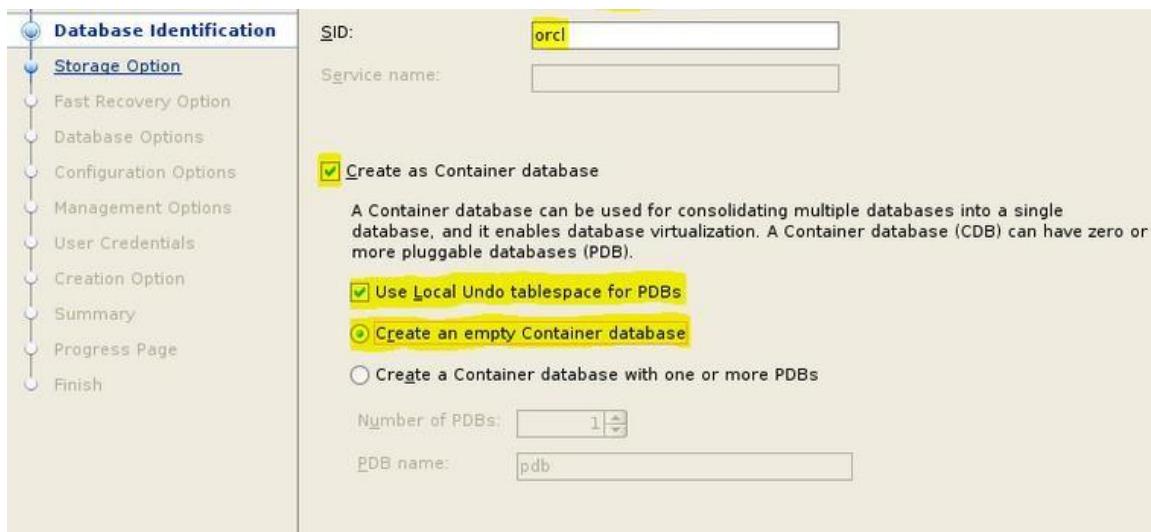
プロビジョニングとデータファイルの場所

PDBの柔軟性と管理性を最大限に高めたい場合は、すべてのデータファイルが想定される場所に配置されるようにプロビジョニングに注意する必要があります。

を dbca プロビジョニングに使用する場合、またはバイナリのインストール時にデータベースを作成する場合は、高度な設定を使用することが重要です。標準構成を使用する場合、一部のPDBデータファイルがCDBデータファイルと一緒に配置されるため、パフォーマンス、バックアップ/リカバリ、およびプラグの取り外し/プラグのタスクがより困難になります。

最適なオプションは、空のコンテナデータベースを作成し、図5に示すようにPDB用のローカルテーブルスペースを元に戻すを選択することです。その後、必要な場所にPDBを作成すると、すべてのデータファイルが正しく配置されます。

図5) 空のコンテナデータベースを選択するためのOracle RUNインストーラの導入オプション



PDB\$シードテンプレートからの新しいPDBのプロビジョニング (PDB分離)

Oracle 12.2以降では、`create_file_dest pdb create`コマンドとともに使用するオプションが用意されています。これにより、データファイルの作成対象を特定のディレクトリまたはディスクグループに限定し、OMFなどのデータファイルの作成を自動化できます（次の例を参照）。

```
SQL> CREATE PLUGGABLE DATABASE pdb2 ADMIN USER admin2 IDENTIFIED BY p2
ROLES=(DBA) CREATE_FILE_DEST = 'PDB2DIRECTORY';
```

この方法では、PDBローカルユーザは STORAGE、以外の場所に新しいデータファイルを作成できません。FILE_DEST。CDB管理者ユーザがこのパラメータを設定した場合、ローカルユーザは変更できません。これは、PDBファイルを配置できる場所に制限を適用する場合に特に役立ちます。これにより、データベースのパフォーマンス、バックアップ/リカバリのSLA、およびその他の管理性の側面をPDB単位で簡単に制御できます。

ユーザの表領域

PDBシードまたはアプリケーションシード以外の方法（PDBのクローニングや未接続PDBのプラグインなど）を使用してPDBを作成する場合、デフォルトの表領域はソースPDBにすでに存在する表領域である必要があります。これは、複数のスキーマと表領域を持つ非CDBから単一のPDBに移行する場合には推奨されません。その場合は、`USER_TABLESPACE`句を使用する必要があります。

PDBとSTORAGE句

PDBの容量使用率は、NFSファイルシステムまたは特定のPDBに割り当てられたLUNのサイズで制御するのが理想的です。これが不可能で、共有ストレージレイアウトが必要な場合は STORAGE、CREATE PLUGGABLE DATABASE ステートメントのSQL Plus句を使用してPDBのストレージ制限を指定できます。MAXSIZE基盤となるLUNおよびボリュームに十分なスペースがある場合でも、PDBに属するすべての表領域で使用できるストレージ容量がこれ以上増加することはできません。MAXSIZE 句を UNLIMITED、またはに設定し STORAGE

UNLIMITED た場合、またはが設定されて STORAGE いない場合、または句が設定されていない場合、PDBにストレージ制限は設定されません。

- ストレージ(MAXSIZE 2G)
- ストレージ(MAXSIZE無制限)

MAX_AUDIT_SIZEおよびMAX_DIAG_SIZE

また、MAX_AUDIT_SIZE MAX_DIAG_SIZE パラメータとパラメータを使用して、各PDBの統合監査OSスピルオーバー (.bin形式) ファイル、診断トレースファイル、またはADR内のインシデントダンプのストレージ容量を制限することもできます。

アップグレードと移行

ONTAPを使用したレガシーデータベースからPDBへの変換

CDB以外のデータベースをマルチテナントアーキテクチャに移行する方法は複数あります。ほとんどの方法では、データポンプ、DBリンク、または移動可能な表領域を使用して、ストレージレベルまたはネットワーク経由でデータを物理的にコピーします。これは、データベースのサイズが小さい場合は問題になりませんが、データベースのサイズがギガバイトまたはテラバイトを超える場合は、移行を完了するために必要な時間、ネットワーク帯域幅、および追加の容量が課題になります。大量のデータを物理的にコピーするには時間がかかります。このような長時間のダウンタイムは望ましくありません。多くの場合、ミッションクリティカルな本番環境のOracleデータベースでは実現できない可能性もあります。

ONTAPでは、Snapshot機能とNetApp SnapMirror®機能を使用して、データベースの移行プロセスを簡易化し、プロセスをより安全にすることができます。次のセクションでは、いくつかのオプションについて説明し、Oracleの基本的なプロセスについて説明します。最新情報については、お使いのOracleソフトウェア・バージョンのOracle公式ドキュメントを参照してください。

Snapshotホコラショウシタインプレースヘンカン

最も効率的な移行オプションは、実際にデータを移動することなく、古いデータベースをコンテナデータベースにプラグインするだけで、移行完了までの時間を大幅に短縮することです。最も簡単なオプションは、現在のレガシーデータベースの使用を停止してPDBインプレースに変換することですが、これには同じサーバを使用する必要があります。多くの場合、PDBは新しいハードウェア上の新しいサーバ、新しいバージョンのオペレーティングシステム、およびOracleの新規インストールに移動されます。

少なくとも、ONTAP上のソースデータベースを、Snapshotコピーによってバックアウトパスとして保護する必要があります。移行プロセス中に問題が発生しても、元のデータはSnapshotコピー内で引き続き使用でき、移行を中止する必要がある場合は、数秒でアクティブデータとして再アクティブ化できます。Snapshotコピーを作成するために追加のスペースは必要なく、ペタバイト規模の場合でもほぼ瞬時にリストアプロセスが完了します。

Snapshotによる保護を使用したデータ再配置

場合によっては、レガシーデータベースを別の物理サーバ上のPDBに変換する必要があります。たとえば、さまざまなサーバ上の古いデータベースの集まりを、より高速なCPUとより多くのRAMで実行されている新しいデータベースサーバに移動できます。

NFS共有またはLUNを1台のサーバからディスマウントし、データを別のサーバに提供するプロセスは簡単ですが、Oracleデータベースにはパスの依存関係があります。ファイルシステムまたはASMディスクグループを新しいサーバに別の名前でマウントする必要がある場合があります。これには、PDB記述子ファイルの編集が必要になります。これについては、次のセクションで説明します。

インプレース変換と同様に、ソースデータベースをシャットダウンしたあとにSnapshotコピーを保護してバックアウトパスを維持する必要があります。

SnapMirrorによるデータ再配置

場合によっては、データベースを別のストレージシステム上のPDBに変換しなければならないことがあります。たとえば、データベースのアップグレードは一般的なITインフラストラクチャのアップグレードで行われることが多く、レガシーデータベースを別の物理データセンターのデータベースサーバにリホストする必要がある場合があります。

ONTAPを使用すると、このプロセスがシンプルになり、ダウンタイムも削減されます。現在のデータベースのミラーコピーは、現在の処理を中断することなく、ターゲットストレージシステムに作成できます。これは、長距離の移行で特に便利です。帯域幅が非常に限られている一部のONTAPのお客様は、最初のデータレプリケーションの完了に数週間を要した移行を実施しています。

初回のデータコピーが完了したら、都合の良い時間にカットオーバーをスケジュールできます。リモートコピーをソースと同期するために必要なデータ転送量が少ないため、システム停止も最小限に抑えられます。

例

以降のセクションでは、SnapMirrorを使用した変換プロセスの基本的な例を示します。基本原則は、あらゆる変換アプローチに適用できます。

前提条件

- ソース非CDBおよびターゲットCDBホストは、互換性のある文字セットおよび各国の文字セットを使用する必要があります。
- マルチテナントを使用してターゲットCDBホストにOracle Database 19C（またはそれ以降）をインストールしておく必要があります。
- エンディアンネスは一致する必要があります。データベースを新しいCPUアーキテクチャに再プラットフォーム化する必要がある場合（AIXからLinux x86への移行など）は、PDBに変換してから再プラットフォーム化することをお勧めします。

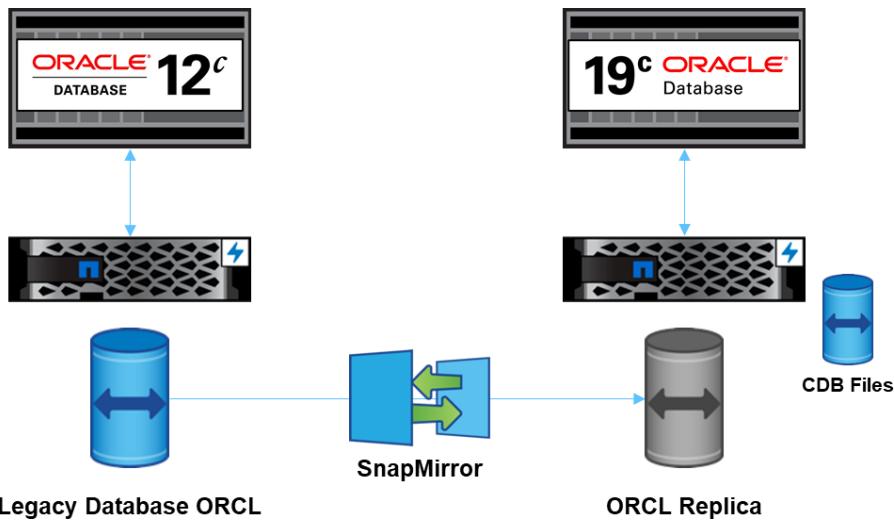
初期レプリケーション

まず、SnapMirror機能を使用して、ソースデータベースファイルをホストしているONTAPボリュームをレプリケートします。ベースラインが完了すると、以降の更新では差分のみが必要になります。スケジュールされた更新を一定の間隔で設定して、カットオーバーまでデータの2つのコピーをほぼ同期した状態に保つことができます。

同時に、Oracle Real Application Clusters（RAC）またはOracle ASMを使用している場合は、Gridバイナリを含む新しいOracle 19Cバイナリをインストールできます。空のコンテナを使用して新しいCDBデータベースを設定し、RAC/ASMの場合はクラスタを使用してASMインスタンスを設定し、ディスクグループを投票します。

図6に示すような構成が表示されます。

図6) NetApp SnapMirrorを使用した新しいインフラへのデータベース移行の一般的なアーキテクチャ



PDB概要ファイル

カットオーバーの時間が来たら、ソースデータベースをシャットダウンし、読み取り専用として再度開く必要があります。

次のようにPDB概要ファイルを作成します。このファイルには、データベース内の各データファイルの基本的な説明（パスなど）が格納されます。データベースの状態をフリーズしてこのファイルを生成するには、データベースが読み取り専用モードになっている必要があります。

```
BEGIN
DBMS_PDB.DESCRIBE(
  pdb_descr_file => '/tmp/ORCL.xml');
END;
/
```

この出力ファイルを新しいデータベースサーバにコピーします。

データベースのシャットダウン

次に、データベースをシャットダウンし、データベースのファイルシステムをディスマウントし、移行の反転が必要になった場合に備えてSnapshotコピーを作成します。

注： Snapshotコピーを作成する前に、データが完全に休止されていることを確認してください。これには通常、データベース自体をシャットダウンするだけでなく、ファイルシステムをディスマウントする必要があります。ASMディスクグループをディスマウントする必要があります。データベースがシャットダウンされても、動作中のASMディスクグループでメタデータが変更され、Snapshotコピーを使用できなくなる可能性があります。

SnapMirrorの同期と解除

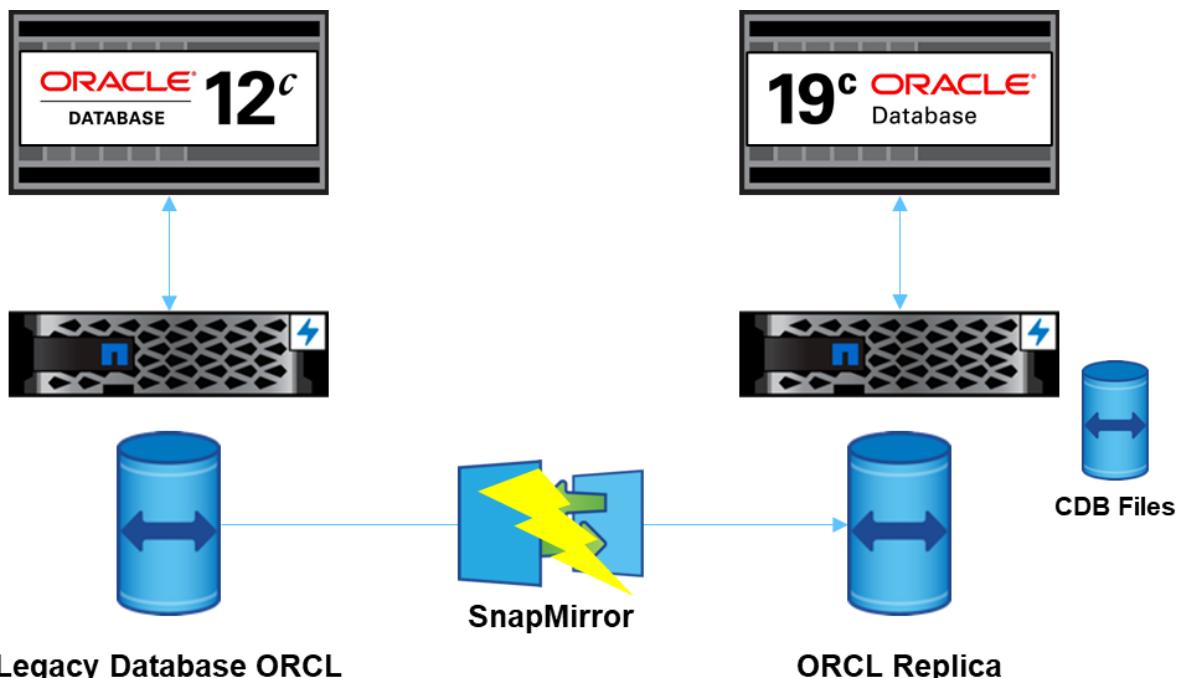
データを休止して保護Snapshotコピーを作成したら、SnapMirrorの最後の更新を1回実行して、デスティネーションのデータがソースのデータと一致するようにします。これには、次の2つの方法があります。

- データベースを含むソースボリュームにマイグレーション対象のデータだけが保持され、それ以外のデータは保持されていない場合は、ソースデータに対してSnapMirror更新を実行できます。これにより、ソースボリュームとまったく同じ状態でレプリケートされたデータが生成されます。
- ソースボリュームが他のデータと共有されている場合もあります。その結果、ボリューム全体のデータは引き続き変更されている可能性があります。この場合、前の手順で作成した特定のSnapshotコピーのレプリケートが必要になることがあります。ソースボリューム上的一部のデータは引き続き変更される可能性がありますが、ターゲットデータベースは完全に休止されており、デスティネーションでも使用できません。

す。データの最新コピーと特定のSnapshotコピーのレプリケート手順については、SnapMirrorのドキュメントを参照してください。

更新が完了したら、SnapMirror関係を解除し、デスティネーションレプリカをデータベースサーバに提供します。これには、図7に示すように、論理ボリュームマネージャ（LVM）ボリュームグループまたは重要なASMディスクグループを検出する追加の手順が含まれる場合があります。

図7) SnapMirror DR関係を解除してレプリケーションを停止し、DRコピーを本番環境として昇格



互換性の確認

ORCL.xml処理を続行する前に、前の手順()で作成したPDB記述子ファイルの編集が必要になる場合があります。元のデータベースのデータファイルのパスがOS上で同じ場合、ファイルはそのまま動作します。ファイルシステムのマウントポイントに変更があった場合、またはASMディスクグループの名前が変更された場合は、このファイルを編集する必要があります。構文は全体的に直感的です。このファイルの編集の詳細については、Oracleの公式ドキュメントを参照してください。

レプリケートされたデータが使用可能になり、記述ファイルが正確になったら、CDBデータベースを開き、次のPL/SQL手順を使用してプラグインの互換性を確認します。PDBに互換性がない場合は PDB_PLUG_IN_VIOLATIONS、違反がビューに表示されます。

```
SET SERVEROUTPUT ON
DECLARE
  l_result BOOLEAN;
BEGIN
  l_result := DBMS_PDB.check_plug_compatibility(
    pdb_descr_file => '/tmp/ORCL.xml',
    pdb_name       => 'pdb2');

  IF l_result THEN
    DBMS_OUTPUT.PUT_LINE('compatible');
  ELSE
    DBMS_OUTPUT.PUT_LINE('incompatible');
  END IF;
END;
/
compatible
```

```
PL/SQL procedure successfully completed.
```

移行の完了

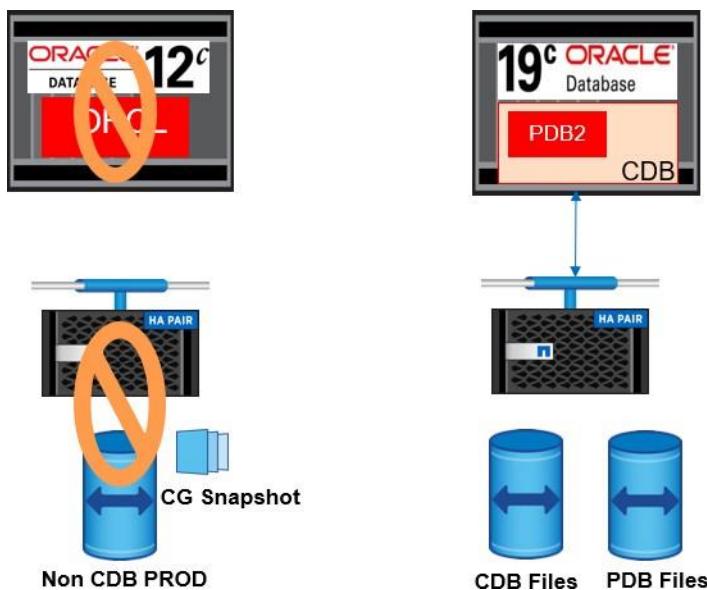
チェックが成功したら、次のスクリプトを使用してXML記述子ファイルを使用してPDBを作成します。

```
CREATE PLUGGABLE DATABASE pdb2 USING '/tmp/orcl.xml'  
NOCOPY  
TEMPFILE REUSE;
```

次に、それぞれのコンテナに移動し、アップグレードスクリプトを起動し、次のように読み取り/書き込みモードでPDBを起動します(図8を参照)。

```
SQL> alter session set container=PDB2;  
SQL> @$ORACLE_HOME/rdbms/admin/noncdb_to_pdb.sql;  
SQL> alter pluggable database PDB2 open;
```

図8) レガシーインフラの廃止に成功し、新しいインフラで本番運用を開始



ストレージの移行、データセンターの統合、移転

通常、ストレージの移行には次の6つの方法があります。

- データの単純なコピー
- Oracle Data Guard / Oracle GoldenGate
- ログ配布
- ASMによる移行
- LVMによる移行
- FLI (Foreign LUN Import) :

詳細については、[TR-4534 : 『Migration of Oracle Databases to NetApp Storage Systems』](#)を参照してください。

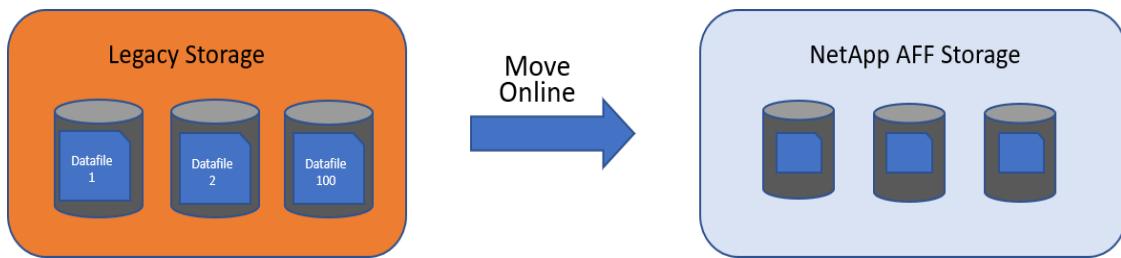
FLIはONTAPに組み込まれている機能で、外部アレイLUNからNetApp LUNにデータを簡単かつ効率的にインポートできます。FLIでは、NetApp FlexArray® テクノロジを活用して外部RAIDアレイLUNを検出し、LUNからデータを取得します。FlexArrayはNetApp ONTAPコントローラの動作を可能にします。

1つ以上の他社製アレイの前にイニシエータとして配置します。FlexArrayでは、これらのアレイのLUNをバッケンドストレージとしてマウントし、LUNのスペースをNetAppユニファイドストレージ (FCP、FCoE、iSCSI、NFS、およびCIFS / SMBプロトコルアクセスを許可) としてSAN環境またはNAS環境のホストに提供できます。詳細については、[TR 4380 : 『SAN Migration Using Foreign LUN Import』](#) および [TR 3633 : 『Oracle Databases on ONTAP』](#) を参照してください。

ストレージアレイ間でデータファイルをオンラインで移動

Oracle 12cより前のバージョンでは、ASM以外のデータファイルをストレージアレイ上またはストレージアレイ間で移動するには、データファイルをオフラインにし、ビジネス部門とシステムの停止について交渉する必要がありました。Oracle 12c以降では、この状況は解消されています。Oracle ALTER DATABASE MOVEデータファイル機能を使用すると、新しいストレージプラットフォームにデータを簡単にオンラインで移行できます。OSレベルでのデータファイルの名前変更や移動、Oracle制御ファイル内の参照の変更など、面倒な作業をすべて実行できます。図9に示すように、

図9) データファイルのオンライン移動



データファイルの移動中は v\$SESSION_LONGOPS ビューで進捗状況を監視できます。報告された実行済みバイト数は、移動が完了するまで増加します。

注：開始する前に、ターゲットストレージプラットフォームに十分なスペースがあることを確認してください。Oracleはターゲットシステムに必要なストレージスペースを割り当て、移動が完了するまでソースのスペースを解放しません。

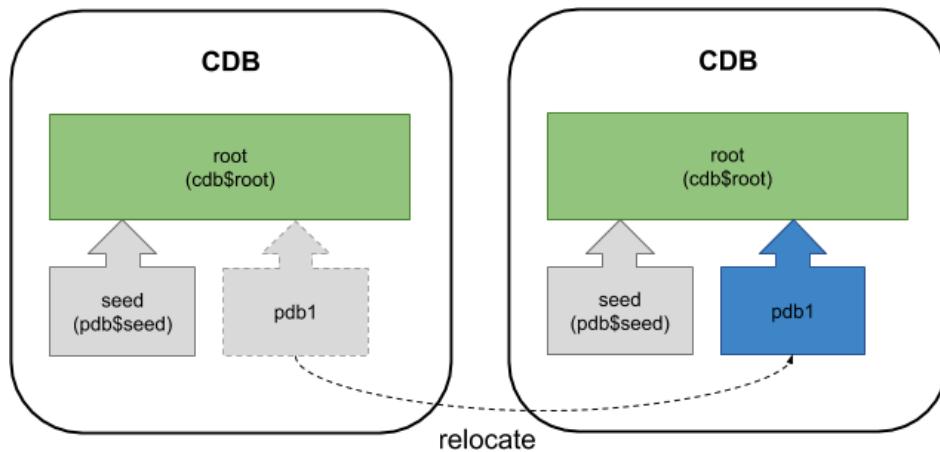
KEEP パラメータを指定すると、データファイルが新しい場所にコピーされ、古いデータファイルは元の場所に保持されます。データファイルは制御ファイルで参照されなくなり、不要になったときは削除できます。この[アプローチ](#)により、Oracleのデータファイルを他のストレージベンダーや従来型ストレージからNetApp AFF ONTAPストレージにダウンタイムなしで移動できます。このアプローチを使用して、優先度の低いアプリケーションまたはテーブルをロードまたはハイエンドのストレージに移動することができますが、ファイルの整合性を維持するには時間がかかる場合があります。その場合は、別のアグリゲートへのストレージボリュームの移動利用するのが最善の方法です。

Oracle PDBの再配置

Oracle 12.2以降では、PDB（クローン）を2つのルートコンテナ間で移動することで、ダウンタイムをほぼゼロに抑えることができます。これにより、PDBを1つのホストCDB /ストレージから別のCDBホスト/ストレージに移動する際に、ダウンタイムやデータ損失をゼロにすることができます。また、新しいリストナーに接続を転送するため、アプリケーションに対して透過的です。PDBの再配置は、図10に示すように、2つのホスト間のネットワーク回線（dbリンク）に完全に依存します。

```
CREATE PLUGGABLE DATABASE PDB1 FROM PDB1@clone_link RELOCATE;
Pluggable database created.
```

図10) PDBの移動



PDBの再配置処理には2つのストレージスペースが必要ですが、カットオーバー後に古いPDBが閉じられ、手動で削除できます。このアプローチでは、PDBのあるCDBから別のCDBに移行するのに時間がかかります。PDBを下位のCDBバージョンから上位のCDBバージョンに再配置することで、PDBアップグレードにもこのアプローチを使用できますが、AVAILABILITY_MAX アップグレードには時間がかかり、手動で行う必要があるため、パラメータとしては使用しないでください。したがって、それは失敗します。

代わりに、単純なrelocateコマンドを使用して、PDBのクローンを完全に作成できます。PDBコピーが終了したら、UPGRADE モードでデータベースを起動し、次の簡単な手順でアップグレードを完了します。

```
ALTER PLUGGABLE DATABASE PDB1 OPEN UPGRADE;
```

プラガブルデータベースのパフォーマンスに関するベストプラクティス

NFSとSANのマウントオプション、およびストレージパフォーマンスに関する他のベストプラクティスについては、[TR 3633 : 『Oracle Databases onONTAP』](#)を参照してください。

プラガブルデータベースのAutomatic Workload Repository (AWR) レポートの有効化

AWRレポートはPDBレベルでイネーブルにできます。各PDBのパフォーマンスを確認するには、次の手順を実行します。

1. ALTER SYSTEM SET awr_pdb_autoflush_enabled= TRUE SID='*' SCOPE=BOTH;
2. execute dbms_workload_repository.modify_snapshot_settings(interval => 30, retention=>64800);
3. ALTER SYSTEM SET awr_snapshot_time_offset=1000000 SID='*' SCOPE=BOTH;
4. exec dbms_workload_repository.create_snapshot();

リソーススマネージャを使用したパフォーマンスプロファイル

CDBでは複数のPDBを柔軟にホストできるため、サーバのコンピューティングとI/Oの使用を制御することが不可欠です。開発/テストホストとPRODホストは、完全に分離された状態で同じホスト上に隣接して配置することも、重要度やパフォーマンスのニーズが異なる複数の製品を同じCDB上に配置することもできます。PDBレベルの計算 (SGA / PGAなど) とI/Oパラメータは、それぞれのコンテナに切り替えることで設定できます。

同じストレージデバイス (ボリューム/LUN) (物理または仮想) で複数のPDBをホストしており、きめ細かな制御が必要な場合は、次のパラメータが必要になることがあります。

- **MAX_IOPS** : PDBの1秒あたりの最大I/O処理数。デフォルトは0です。100 IOPS未満の値は推奨されません。
- `ALTER SYSTEM SET MAX_IOPS = 1000 SCOPE = BOTH;`
- **MAX_MBPS** : PDBの1秒あたりのI/Oの最大メガバイト数。デフォルトは0です。25Mbps未満の値は推奨されません。
- `ALTER SYSTEM SET MAX_MBPS = 5 SCOPE = BOTH;`
- **DBA_HIST_RSRC_PDB_METRIC**: このビューは制限を計算するのに役立ちます。`rsmgr:io rate limit` 待機イベントは、制限に達したことを示します。

ベスト プラクティス

コンピューティング管理にリソースマネージャを使用して、PDBのリソース計画を作成することをお勧めします。I/O管理のほとんどは、アダプティブQoSポリシーを使用してストレージレベルで実行できます。

アダプティブQoSベースの管理

アダプティブQoSポリシーグループを使用すると、ボリュームサイズの変更に合わせてスループットの上限や下限を自動的に調整し、TB / GBあたりのIOPSを一定に維持することができます。これは、大規模な環境で数百ものPDBワークロードを管理する場合に大きなメリットです。PDBをホストする各ボリュームでアダプティブQoSを有効にする方法の詳細については、ONTAPの次のドキュメント「[アダプティブQoSポリシーグループの使用](#)」を参照してください。

データ保護とコピーデータ管理

災害やデータ破損などのシナリオからPDBまたはCDB全体を保護することが重要です。また、プロジェクトの迅速な導入、新規ビジネスのコミッショニング、修正パッチの迅速なテストにおいて、DevOpsライフサイクルの高速化が重要な役割を果たすことは避けられません。PDBのアーキテクチャの変更により、PDBのバックアップ、リストア、クローニングの処理は従来の処理と若干異なります。

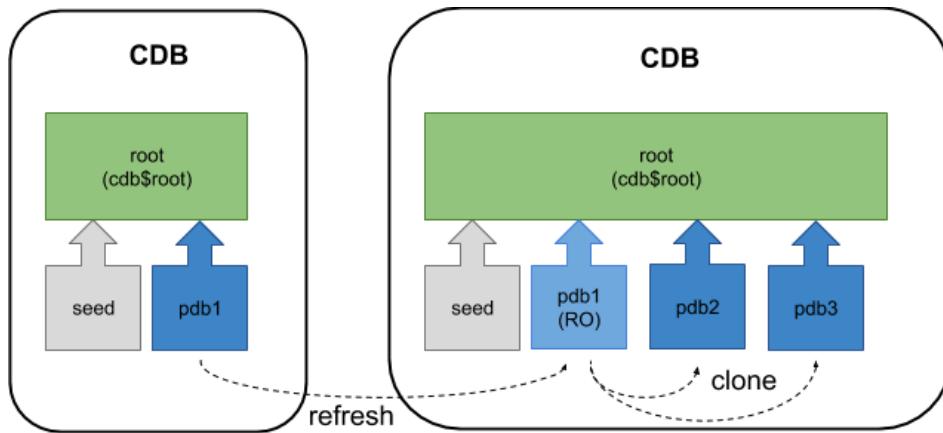
NetApp SnapCenter

NetApp SnapCenterは、マルチテナント環境のバックアップ、マウント、リストア、クローニングの処理をサポートしています。SnapCenter 4.4以降では、個々のPDBまたはPDB表領域レベルでフルリストアまたはポイントインタイムリストアを実行できます。また、ストレージSnapshotコピーを使用して、あるCDBから別のCDBにPDBをクローニングすることもできます。NetAppは、ストレージベンダーとして初めて、ストレージSnapshotコピーを使用して、あるCDBから別のCDBへのPDBクローンをサポートしました。ベストプラクティスについては[TR4700 SnapCenter Plug-in for Oracle Databases](#)を、詳細については[YouTubeのビデオ](#)をご覧ください。

PDBリフレッシュとPDBスイッヂオーバー

Oracle Database 12.2以降では、リモートでホットクローニングされたPDBが更新可能なPDBとして作成され、読み取り専用モードでのみ開かれていれば、その内容を更新できます。読み取り専用PDBを他のクローンのソースとして使用して、複数の最新クローンが定期的に必要な場合に本番システムへの影響を最小限に抑えることができます。図11に示すように、更新可能なPDBを別のCDBにホストするには、ストレージを個別にプロビジョニングする必要があります。

図11) Oracle PDBの更新



DBリンクを使用すると、更新可能なPDBを簡単に作成できます。また、リフレッシュモードを手動または自動に指定し、時間または頻度を指定する必要があります。

```
CREATE PLUGGABLE DATABASE pdb_ro FROM pdb@clonedb_link
  REFRESH MODE MANUAL;

Pluggable database created.

ALTER PLUGGABLE DATABASE pdb_ro OPEN READ ONLY;

SELECT last_refresh_scn
  FROM dba_pdbs
 WHERE pdb_name = 'PDB_RO';

LAST_REFRESH_SCN
-----
1255484
```

自動更新を実行するには、PDBを手動で閉じるか、閉じたままにする必要があります。自動的には閉じません。

```
-- Automatically refresh every 120 minutes.
CREATE PLUGGABLE DATABASE pdb1_ro FROM pdb@clonedb_link
  REFRESH MODE EVERY 120 MINUTES;
```

更新は、PDBが閉じられたときにのみ実行されます。更新が終了したら、レポート作成のために読み取り専用モードで起動したり、自動更新のために閉じたままにしたりできます。前の章で説明したように、PDB Snapshotコピーを使用して複数のクローンを実行できます。そのためには、を clonedb に設定する必要があります。ほとんどの場合、PDBを読み取り専用モードで開くには、PDBを閉じて更新し、再度開く独自のジョブを定義する必要があります。

```
ALTER SESSION SET CONTAINER=pdb_ro;
ALTER PLUGGABLE DATABASE CLOSE IMMEDIATE;
ALTER PLUGGABLE DATABASE REFRESH;
ALTER PLUGGABLE DATABASE OPEN READ ONLY;
```

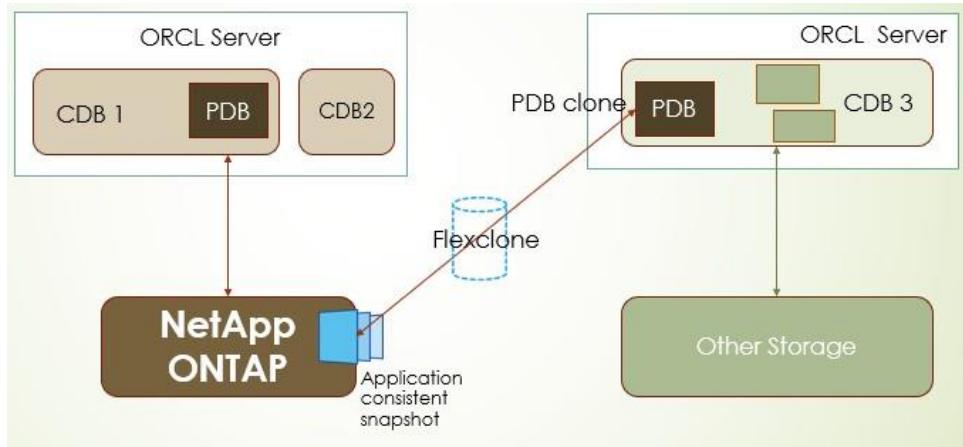
ベストプラクティス

クローニングのみを目的とする場合は、SnapCenterを使用してストレージのSnapshotコピーを活用することを推奨します。この場合、重複するPDBをホストするためのストレージを100%節約できます。PDBアプローチは、Data Guardのようなスイッチオーバー機能を使用すると、ディザスタリカバリシナリオでより役立ちます。Oracleは更新可能なPDBという概念を導入しましたがスイッチオーバー：Oracle ExadataとCloudのバージョンに限定されているため、NetAppのお客様には現時点では推奨されません。

異機種混在環境でのPDBクローン

異なるホストやストレージ上の異なるPDBに分散しているデータから、統合されたデータセットを使用する場合は、NetApp FlexClone®を使用すると、特定のCDBからPDBをクローンングし、そのクローンを別のホストの別のストレージから作成された別のCDBにポートすることができます（ローカルPDBであるかのように）。ターゲットCDBが他社製ストレージアレイ（EMC、Pure、Hitachiなど）にある場合でも、図12に示すように、NetAppストレージおよびポートが所有する個々のPDBをターゲットCDBにクローンングできます。

図12) 異機種混在ストレージ環境でのNetApp SnapCenterを使用したPDBクローン



これらの手順はすべて、NetApp SnapCenter 4.4を使用するか、NetApp検証済みのスクリプトソリューションを使用して自動化できます。

付録A:スキーマ統合データベース

スキーマ統合データベースの場合は、エクスポート/インポートまたはデータポンプのアプローチを検討する必要があります。CDB以外のOracle DatabaseリリースがOracle Database 11g Release 2 (11.2.0.3) 以降の場合は、完全に移動可能なエクスポート/インポートを使用してデータを移動できます。Oracle Database 11g Release 2 (11.2.0.3) 以降のOracle Database 11gデータベースからOracle Database 12c以降にCDB以外を転送する場合は、VERSION Data Pump exportパラメータを12.0.0.0.0以上に設定する必要があります。

非CDBのOracle DatabaseリリースがOracle Database 11g Release 2 (11.2.0.3) より前の場合は、移動可能な表領域を使用してデータを移動したり、データベースのフルエクスポート/インポートを実行したりできます。

付録B : OracleマルチテナントデータベースでのOracle EBS

Database 19C認定により、オンプレミスのEBS 12.2データベースはCDBアーキテクチャ（マルチテナントアーキテクチャ）で認定されました。Database 19Cへのアップグレードの一環として、EBSデータベースを单一のPDBでCDBアーキテクチャに変換します。CDBアーキテクチャへの変換を実行するために必要な手順は、次のドキュメントに記載されています。

[Interoperability Notes: Oracle E-Business Suite Release 12.2 with Oracle Database 19C \(MOS Note 2552181.1\)](#)

次の点に注意することが重要です。

- 1つのPDB（シングルテナント）を備えたCDBは、現在、Oracle E-Business Suite with Database 19Cで唯一認定された導入環境です。
- 複数のPDB（マルチテナント）を持つCDBは、現時点ではOracle E-Business Suiteの認定を受けていません。
- 非CDBアーキテクチャは、データベース19Cを使用するEBSでは認定またはサポートされていません。

現時点では、認定プラットフォームはLinuxとSolarisです。Oracleデータマスキングやデータベースポールトなどのいくつかの機能は、まだ認定されていません。最新の詳細については、Oracleのドキュメントを参照してください。

注：前のセクションで説明したアップグレード手順は、EBSデータベースのアップグレードには適用されません。

付録C：ドキュメントの分類とラベル

NetApp情報資産を適切に分類するには、[NetAppコードレート編集ガイド](#)の「機密情報の保護」セクションを参照してください。

詳細情報の入手方法

このドキュメントに記載されている情報の詳細については、以下のドキュメントやWebサイトを確認してください。

- TR-3633 : 『ONTAPを基盤にしたOracleデータベース』
<https://www.netapp.com/media/8744-tr3633.pdf>
- TR-4700 : 『SnapCenter Plug-in for Oracle Database Best Practices (Oracleのバックアップ、リストア、およびクローン)』
<https://www.netapp.com/media/12403-tr4700.pdf>
- TR-4591 : 『Databases Data Protection : Backup、Recovery、Replication、and DR』
<https://www.netapp.com/media/19666-tr-4591.pdf>
- TR-4695 : 『Database Storage Tiering with NetApp FabricPool』
<https://www.netapp.com/pdf.html?item=/media/9138-tr4695pdf.pdf>

バージョン履歴

バージョン	日付	ドキュメントの改訂履歴
1.0	2021年1月	初版

本ドキュメントに記載されている製品や機能のバージョンがお客様の環境でサポートされるかどうかについては、NetApp サポートサイトで [Interoperability Matrix Tool \(IMT\)](#) を参照してください。NetApp IMT には、NetApp がサポートする構成を構築するために使用できる製品コンポーネントやバージョンが定義されています。サポートの可否は、お客様の実際のインストール環境が公表されている仕様に従っているかどうかによって異なります。

機械翻訳に関する免責事項

原文は英語で作成されました。英語と日本語訳の間に不一致がある場合には、英語の内容が優先されます。公式な情報については、本資料の英語版を参照してください。翻訳によって生じた矛盾や不一致は、法令の順守や施行に対していかなる拘束力も法的な効力も持ちません。

著作権に関する情報

Copyright © 2024 NetApp, Inc. All Rights Reserved. Printed in the U.S. このドキュメントは著作権によって保護されています。著作権所有者の書面による事前承諾がある場合を除き、画像媒体、電子媒体、および写真複写、記録媒体、テープ媒体、電子検索システムへの組み込みを含む機械媒体など、いかなる形式および方法による複製も禁止します。

NetApp の著作物から派生したソフトウェアは、次に示す使用許諾条項および免責条項の対象となります。

このソフトウェアは、NetApp によって「現状のまま」提供されています。NetApp は明示的な保証、または商品性および特定目的に対する適合性の暗示的保証を含み、かつこれに限定されないいかなる暗示的な保証も行いません。NetApp は、代替品または代替サービスの調達、使用不能、データ損失、利益損失、業務中断を含み、かつこれに限定されない、このソフトウェアの使用により生じたすべての直接的損害、間接的損害、偶発的損害、特別損害、懲罰的損害、必然的損害の発生に対して、損失の発生の可能性が通知されていたとしても、その発生理由、根拠とする責任論、契約の有無、厳格責任、不法行為（過失またはそうでない場合を含む）にかかわらず、一切の責任を負いません。

NetApp は、ここに記載されているすべての製品に対する変更を隨時、予告なく行う権利を保有します。NetApp による明示的な書面による合意がある場合を除き、ここに記載されている製品の使用により生じる責任および義務に対して、NetApp は責任を負いません。この製品の使用または購入は、NetApp の特許権、商標権、または他の知的所有権に基づくライセンスの供与とはみなされません。

このマニュアルに記載されている製品は、1つ以上の米国特許、その他の国の特許、および出願中の特許により保護されている場合があります。

本書に含まれるデータは市販の製品および / またはサービス (FAR 2.101 の定義に基づく) に関係し、データの所有権は NetApp, Inc. にあります。米国政府は本データに対し、非独占的かつ移転およびサブライセンス不可で、全世界を対象とする取り消し不能の制限付き使用権を有し、本データの提供の根拠となった米国政府契約に関連し、当該契約の裏付けとする場合にのみ本データを使用できます。前述の場合を除き、NetApp, Inc. の書面による許可を事前に得ることなく、本データを使用、開示、転載、改変するほか、上演または展示することはできません。国防総省にかかる米国政府のデータ使用権については、DFARS 252.227-7015(b) 項で定められた権利のみが認められます。

商標に関する情報

NetApp、NetAppのロゴ、<https://www.netapp.com/company/legal/trademarks/>に記載されているマークは、NetApp, Inc.の商標です。その他の会社名と製品名は、それを所有する各社の商標である場合があります。

TR-4876-0121-JP