


Hybrid Data Guard to Oracle Cloud Infrastructure

オンプレミスの本番データベースとOracle Cloud
InfrastructureのDBaaS BMまたはVMシェイプを使用した
ディザスタ・リカバリ

Oracle ホワイト・ペーパー | 2019年4月



はじめに.....	3
Data Guard および Active Data Guard による クラウドへのディザスタ・リカバリ	4
クラウドにおけるハイブリッド・スタンバイのメリット	5
OCI DBaaS でのディザスタ・リカバリの有効化.....	5
サービス・レベル要件	6
セキュリティ要件	7
データベース、OS 環境、およびネットワークの前提条件.....	8
スタンバイ・データベースの使用による計画メンテナンス中の停止時間の短縮	8
OCI ネットワークの前提条件.....	9
セキュアな接続.....	9
パブリック・インターネット接続	10
オンプレミス・ネットワークの構成	10
オンプレミスの前提条件	10
プライマリ・データベースでの MAA ベスト・プラクティスのパラメータ設定の実装..	11
オンプレミス・ホストと OCI DBaaS ホストとの間の接続の検証.....	11
デプロイメント・プロセス	12
手順 1：DB システムの VM または BM を作成する	13
手順 2：ツールを使用して、作成済みのデータベースを手動で削除する	13
手順 3：パスワード・ファイルを OCI DBaaS ホストにコピーする.....	14
手順 4：ウォレット・ファイルを OCI DBaaS ホストにコピーする.....	15
手順 5：静的リスナーを構成する	16
手順 6：REDO 転送用のオラクルのネットワーク暗号化と TNS エントリ	17
手順 7：スタンバイ・データベースをインスタンス化する.....	19
手順 8：tnsnames.ora に RAC インスタンスを追加する（RAC のみ）	20
手順 9：Data Guard Broker を構成する.....	21
クライアントのフェイルオーバーを構成する.....	22
ヘルス・チェックと監視	23
Oracle MAA スコアカード.....	23



監視.....	23
DR の準備状況の検証.....	23
スタンバイ・データベースからスナップショット・スタンバイへの変換.....	24
クラウドへのフェイルオーバー/スイッチオーバー.....	25
オンプレミスへのスイッチバック.....	25
結論.....	26
付録 A : RMAN Duplicate の出力例 :	27
付録 B : MAA ベスト・プラクティスのパラメータ設定.....	32
付録 C : RMAN RESTORE FROM SERVICE の出力例.....	34

はじめに

Oracle Maximum Availability Architecture (Oracle MAA) は、プライベート・クラウド、パブリック・クラウドまたはハイブリッド・クラウドにデプロイされる、Oracle データベースのデータを保護し可用性を高めるためのベスト・プラクティス構想です。Oracle Data Guard および Oracle Active Data Guard により、バックアップからのリストアではリカバリ時間目標 (RTO) を達成することのできないデータベースのディザスタ・リカバリ (DR) を実現できます。これらのソリューションを利用して本番データベース (プライマリ・データベース) の同期レプリカ (スタンバイ・データベース) を物理的に離れた場所に 1 つ以上デプロイし、ミッション・クリティカルなデータの高可用性、包括的なデータ保護、およびディザスタ・リカバリを実現します。

有効なディザスタ・リカバリ計画には、リモート・データセンターの設置、装置の整備、および管理が関係しており、かなりのコストがかかる可能性があります。Oracle Cloud は、スタンバイ・データベースをホスティングするための優れた代替手段であり、DR サイトを設置していない場合や、リモート・データセンターの管理に伴うコストや煩雑さを避けたい場合に最適です。既存の本番データベースはオンプレミスに残し、DR 用のスタンバイ・データベースを Oracle Cloud 上にデプロイします。このデプロイメント・モードは、一般にハイブリッド・クラウド実装と呼ばれます。

要件に応じて、Data Guard と Active Data Guard のどちらかのスタンバイをクラウド上にデプロイするよう選択できます。ハイブリッド・クラウド DR 構成の場合には、特有の考慮事項がいくつかありますが、他の Data Guard デプロイメントの場合と同じ Oracle MAA のベスト・プラクティスが適用されます。この Oracle MAA 構想では、Oracle MAA のベスト・プラクティスについて詳しく説明し、Database as a Service を使用して Oracle Cloud 上に DR をデプロイする手順の概要を示します。このホワイト・ペーパーの対象読者は、Oracle Database、Data Guard または Active Data Guard、および Oracle Database のバックアップとリカバリに関する知識を持つ技術者です。またこのホワイト・ペーパーでは、Oracle Cloud Infrastructure で提供されるサービスについての基本的な理解があることも前提としています。

Data GuardおよびActive Data Guardによる クラウドへのディザスタ・リカバリ

Oracle Cloud は、Platform as a Service (PaaS)、Infrastructure as a Service (IaaS)、Software as a Service (SaaS) など、お客様の特定の要件に応じて調整される広範なクラウド・サービスを提供します。オンプレミス・システムのディザスタ・リカバリ (DR) は、Oracle Cloud Infrastructure Database as a Service (OCI DBaaS) を使用して展開されます。

OCI DBaaS を使用したクラウドへの DR 展開には、次の 3 つのオプションがあります。

- » Enterprise Edition Service または High-Performance Service を利用する Data Guard
- » ADG のライセンスがないところで、BYOL の場合に Extreme Performance Service を利用する Data Guard (推奨)
- » Extreme Performance Service または Exadata Service を利用する Active Data Guard (推奨)

Data Guard は、Oracle Database Enterprise Edition に付属しており (オンプレミス・システムの場合、別途ライセンス不要)、OCI DBaaS のすべてのエディション (Enterprise、High Performance および Extreme Performance) でサポートされています。Standard Edition では Data Guard がサポートされないことに留意してください。Enterprise Edition と Standard Edition とを比較して、何がサポートされるかを示すサポート対応マトリックスを確認するには、選択したバージョンの [Oracle Database ドキュメント](#) で“ライセンス情報”を検索してください。

Active Data Guard は、Data Guard の機能を拡張したもので、データ保護および可用性を目的とした高度な機能のほか、読取り専用ワークロードや高速増分バックアップを本番データベースからオフロードする機能も備えています。Active Data Guard は、Extreme Performance Edition と Exadata Service に組み込まれています。ハイブリッド構成で使用する場合は、オンプレミス・システムでも Active Data Guard のライセンスを取得する必要があります。

MAA の推奨事項

1. ロールの移行後、同じパフォーマンス SLA を確実に満たせるように、オンプレミスのプライマリ・データベースと対称になるか類似の構成になるように、OCI スタンバイ・データベースのターゲットを作成する。たとえば、RAC には RAC、Exadata には Exadata を使用する。
2. ピーク REDO レートに対処するためのネットワーク帯域幅を十分に確保する。
3. オンプレミスと OCI クラウドの間のネットワークの信頼性とセキュリティを確保する。
4. 自動ブロック修復、データ保護、およびオフロードによる恩恵を追加で受けるために、Active Data Guard を使用する。
5. プライマリ・データベースとスタンバイ・データベースの双方で、オラクルの透過的データ暗号化 (TDE) を使用する。

クラウドにおけるハイブリッド・スタンバイのメリット

1. クラウド・データセンターとインフラストラクチャはオラクルが管理する。
2. リソースの拡張と縮小などの基本的なシステム・ライフサイクルの運用は、クラウドで行われる。
3. 利用率と投資収益率を高めるため、ディザスタ・リカバリ、データ保護、オフロード機能を Oracle Data Guard が提供する。
4. MAA プラクティスに従って構成を行い、Data Guard ファスト・スタート・フェイルオーバーを使用して構成すると、自動フェイルオーバーした場合、RTO が秒単位になる。また、ASYNC 転送を使用する Data Guard では RPO が 1 秒未満、SYNC 構成または FAR SYNC 構成の Data Guard では RPO がゼロになる。

スイッチオーバー、フェイルオーバー、元の状態への復元といった Data Guard ライフサイクル管理は、Hybrid Data Guard 構成で、手動で行われるプロセスです。

OCI DBaaSでのディザスタ・リカバリの有効化

クラウドで DR を有効にするには、Oracle OCI DBaaS で Data Guard スタンバイ・データベースをインスタンス化する必要があります。インスタンス化されると、Data Guard により、オンプレミスのプライマリ・データベースとクラウドのスタンバイ・データベースの同期が維持されます。

- » オンプレミスの展開と同じ Oracle MAA のベスト・プラクティスの利用。このホワイト・ペーパーで説明されている、ハイブリッド・クラウド・デプロイメントに固有の Oracle MAA の追加ベスト・プラクティスの使用。
- » 計画メンテナンスや計画外停止の期間に、本番データベースをクラウドのスタンバイ・データベースにスイッチオーバー（計画イベント）またはフェイルオーバー（計画外イベント）する機能。障害が発生したオンプレミス・データベースの修復後に、そのデータベースをクラウド内の新しい本番データベースと自動的に再同期させ、本番データベースをオンプレミス・データベースに切り替える機能。

サービス・レベル要件

ハイブリッド・クラウド・デプロイメントは、定義上、ユーザー管理の環境です。管理者は、可用性、データ保護、およびパフォーマンスについて、所定の構成およびアプリケーションの場合に期待する実際的なサービス・レベルを決める必要があります。サービス・レベルはディザスタ・リカバリに関連する次の3つの側面ごとに設定する必要があり、それらはどの Data Guard 構成にでも適用できるものであることが必要です。

- » 停止状態に陥った場合に許容できる最大停止時間を表すのが**リカバリ時間目標 (RTO)** です。これには、停止を検出し、データベースとアプリケーションの両方の接続をフェイルオーバーしてサービスが再開されるまでに必要な時間が含まれます。
- » 許容可能な最大データ損失量を表すのが**リカバリ・ポイント目標 (RPO)** です。望ましい RPO を達成できるかどうかは、次の要因に左右されます。
 - » ネットワークのデータ量に対する使用可能な帯域幅
 - » 信頼性が高く中断することのない送信を実現するネットワークの能力
 - » Data Guard で使用される転送方式（データ損失をほぼゼロに抑える非同期方式、またはデータ損失をゼロに抑える同期方式のいずれか）
- » **データ保護**：Active Data Guard と MAA 構成により、ユーザーは**ブロック破損検知、防止、自動修復**を非常に包括的に構成できます。
- » **パフォーマンス**：スタンバイ・システムでプロビジョニングされている計算、メモリ、I/O などの能力がオンプレミスの本番システムよりも劣っていると、データベースの応答時間がフェイルオーバー後に変わる可能性があります。管理者がコストを削減するために意図的に少ないリソースでスタンバイ・システムを構成し、DR モード中はサービス・レベルが低下することを容認している場合に、この状態が発生します。Oracle MAA のベスト・プラクティスでは、プライマリとスタンバイの両方を同容量のリソースで構成し、フェイルオーバー後も応答時間が変わらないようにすることを推奨しています。クラウドでは高速プロビジョニングが使用可能なため、安定状態では少ない容量でデプロイされ、フェイルオーバーが必要になった場合には新しいプライマリが高速にスケールアップされるようにする、という中間的な構成が可能です。

セキュリティ要件

Oracle MAA のベスト・プラクティスでは、オラクルの透過的データ暗号化（TDE）を使用してプライマリ・データベースとスタンバイ・データベースの両方を暗号化することで、すべてのデータを暗号化するよう推奨しています。移行プロセス中にデータを変換することはできますが、もっとも安全な Data Guard 環境を実現するために、移行前に TDE に変換することを強く推奨します。詳しくは、『Oracle Database Tablespace Encryption Behavior in Oracle Cloud』（ドキュメント ID：2359020.1）を参照してください。TDE によって暗号化されていない他のデータベース・ペイロード（データファイルや REDO ヘッダーなど）の送信中の暗号化には、VPN 接続または Oracle Net の暗号化も必要です。

TDE を使用してデータを保護することは、システムのセキュリティを強化する上で重要です。ただしユーザーは、いずれの暗号化ソリューションを使用する場合であっても、以下に示す特定の考慮事項があることを認識している必要があります。

- » CPU オーバーヘッドの増大：暗号化では、暗号化された値と復号された値を計算するための追加の CPU サイクルが必要となります。ただし TDE は、データベースのキャッシング機能を利用し、Oracle Exadata 内のハードウェア・アクセラレーションを利用することによって、オーバーヘッドが最小限に抑えられるように最適化されています。大半の TDE ユーザーは、TDE を有効にした後に、本番システムのパフォーマンスへの影響はほとんど発生しないことを認めています。パフォーマンスのオーバーヘッドに関して詳しくは、『Oracle Database Advanced Security ガイド』を参照してください。
- » データ圧縮率の低下：暗号化されたデータでは、元の平文データの情報が一切開示されないようにする必要があります。そのため、圧縮率が低くなります。そのため、TDE で暗号化されたデータの圧縮率は、どのような圧縮方式を適用した場合でも低くなります。したがって、TDE 暗号化は、REDO 転送で使用することは勧められません。ただし、Oracle Advanced Compression や Hybrid Columnar Compression といったオラクルのデータベース圧縮テクノロジーと TDE を併用する場合は、暗号化の前に圧縮が実行されるため、圧縮と暗号化の両方のメリットが得られます。
- » 鍵の管理：暗号化の強度は、暗号化に使用される鍵の強度によって決まります。さらに、暗号化鍵を失うことは、その鍵によって保護されているすべてのデータを失うことと同じです。暗号化が有効になっているデータベースの数が少ない場合は、鍵とそのライフサイクルを比較的簡単に追跡できます。しかし、暗号化されるデータベースの数が増えれば、鍵の管理もより難しくなります。暗号化データベースを多数使用する場合は、オンプレミスで Oracle Key Vault を使用して、TDE マスター鍵を保管および管理することを推奨します。

データベース、OS環境、およびネットワークの前提条件

表1: 前提条件

	オンプレミス	Oracle Cloud Infrastructure DBaaS (OCI DBaaS)
オペレーティング・システム	Linux、Windows、またはSolaris X86 (Data Guardのクロス・プラットフォームの互換性については、My Oracle Support Note 413484.1を参照)	Oracle Enterprise Linux (64ビット)
Oracle Database*	Oracle Database Enterprise Edition 11.2.0.4 (64ビット) Oracle Database Enterprise Edition 12.1.0.2 (64ビット) Oracle Database Enterprise Edition 12.2.0.1 (64ビット) Oracle Database Enterprise Edition 18c (64ビット) Oracle Database Enterprise Edition 19c (64ビット)	Database as a Service : Enterprise、High PerformanceおよびExtreme Performanceの各エディションActive Data Guard : Database as a Service (または) 仮想イメージ : Extreme Performance Edition (または) Exadata Cloud Service
RAC	RACまたは非RAC	RACまたは非RAC
マルチテナント	12.1以上では、プライマリ・データベースはCDB/PDBデータベースとする必要がある。	マルチテナント・データベース
物理/仮想	物理または仮想	物理 (BMシェイプ) または仮想 (VMシェイプ)
データベース・サイズ	あらゆるサイズ	あらゆるサイズ。シェイプの制限についてはドキュメントを参照
TDE暗号化	プライマリおよびスタンバイの双方で必須	プライマリおよびスタンバイの双方で必須

オンプレミスで TDE がまだ有効になっていない場合は、マスター・ノート『[Master Note For Transparent Data Encryption \(TDE\)](#)』（ドキュメント ID : 1228046.1）に従って、TDE を有効にし、ウォレット・ファイルを作成してください。

* プライマリ・データベースとスタンバイ・データベース上の Oracle Database バージョンは、初期インストールの間、一致している必要があります。Standby-First と互換性があるデータベース・ソフトウェアの更新の場合、プライマリ・データベースとスタンバイ・データベースの Oracle ホーム・ソフトウェアは同じである必要はありません。『[Oracle Patch Assurance - Data Guard Standby-First Patch Apply](#)』（ドキュメント ID : 1265700.1）を参照してください。

スタンバイ・データベースの使用による計画メンテナンス中の停止時間の短縮

クラウド上のスタンバイ・データベースを使用し、いくつかの方法でプライマリ本番データベースの計画停止時間を短縮できます。

Standby-First Patch Apply

徹底的な検証のため、多くのパッチをまずフィジカル・スタンバイ・データベースに適用できます。停止時間を最小限に抑えるために、まずスタンバイにパッチを適用し、次に本番データベースをスタンバイ・データベースに切り替え、それから元のプライマリ・データベースにパッチを適用する作業が行われることがよくあります。プライマリとスタンバイが Oracle RAC で、ソフトウェアの更新が RAC ローリングである場合、スイッチオーバーは必要ありませんが、それでも検証と保護をさらに強化するため、ソフトウェアを Standby-First で更新することをお奨めします。Standby-First 対象のパッチ (パッチの readme に記載) を適用する場合、異なるパッチ・バージョンで稼働しているプライマリとスタンバイの間の Data Guard による物理レプリケーションがサポートされます。パッチに関して予期せぬ問題が発生した場合には、パッチが適用されていないバージョンに直ちにフォールバックできるよう、一定期間は異なるパッチ・バージョンのプライマリとスタンバイを稼働

させることもできます。Standby-First プロセスを対象とするパッチの詳細については、My Oracle Support Note 1265700.1『Oracle Patch Assurance - Data Guard Standby-First Patch Apply』を参照してください。

データベースのローリング・アップグレード

OCI DBaaS でスタンバイの利点を活かすことができるもう 1 つのユースケースは、データベースのローリング・アップグレードを行う場合で、オラクルがリリースする、Standby-First 互換ではない新しいデータベースへのアップグレードの際に、停止時間を短縮できます。Oracle 11g と Oracle 12c で使用される一時ロジカル・プロセスでは、フィジカル・スタンバイ・データベースを一時的にロジカル・スタンバイに変換し、ロジカル・スタンバイを新しいバージョンにアップグレードして検証し、準備ができれば Data Guard スイッチオーバーを実行します。スイッチオーバーが完了すると、元のプライマリ・データベースは、同様に新しいリリースで動作している同期されたフィジカル・スタンバイに変換されます。詳細については、『[Oracle Data Guard フィジカル・スタンバイ・データベースの使用によるデータベース・ローリング・アップグレードの簡略化](#)』または [Oracle 12c の DBMS_Rolling](#) を参照してください。12.2 以降の Data Guard 環境の場合、スタンバイ・データベースを使ったさらに効率的なデータベース・ローリング・アップグレード・プロセスがあります。Oracle Data Guard ドキュメントの「[Using DBMS_ROLLING to Perform a Rolling Upgrade](#)」のセクションを参照してください。

オンプレミスから Oracle Cloud へのデータ転送では、Oracle FastConnect によって提供されるパブリック・ネットワーク、VPN、高帯域オプションが使用されます。

OCIネットワークの前提条件

Data Guard 構成では、プライマリとスタンバイで双方向の通信ができる必要があります。そのため、システム間のポートにアクセスするために、ネットワーク構成を追加する必要が生じます。

セキュアな接続

クラウド・ネットワークをオンプレミスのネットワーク、IPSec VPN、FastConnect にプライベート接続する場合、2 つの選択肢があります。いずれの方法でも、仮想クラウド・ネットワーク (VCN) に接続するには動的ルーティング・ゲートウェイ (DRG) が必要です。DRG の作成について詳しくは、[このリンクのドキュメント](#)を参照してください。

IPSec VPN

IPSec は Internet Protocol Security または IP Security の略です。IPSec は、パケットが送信元から宛先に送られる前に IP トラフィック全体を暗号化する、プロトコル・スイートです。OCI での IPSec の概要については、[こちら](#)のドキュメントを参照してください。

FastConnect

Oracle Cloud Infrastructure FastConnect によって、データセンターと Oracle Cloud Infrastructure の間に、専用のプライベート接続を作成できます。FastConnect には高帯域幅のオプションが用意されており、インターネットベースの接続に比べて、より信頼性の高い一貫したネットワーク接続を体験できます。FastConnect について詳しくは、[こちら](#)を参照してください。

パブリック・インターネット接続

OCI とオンプレミスとの間の接続には、パブリック・インターネットを使用することもできます。この方法は、デフォルトの状態では安全ではなく、送信のセキュリティを確保するために追加の作業が必要になります。このホワイト・ペーパーのこの後の部分は、パブリック・インターネット接続の利用を前提としています。

デフォルトでは、ポート 1521 のクラウド・セキュリティ・ルールは無効です。また、クラウド VM/BM で構成済みのこのデフォルト・ポートには、パブリック・インターネットからアクセスできます。

1. スタンバイ・データベースの仮想クラウド・ネットワークにインターネット・ゲートウェイがない場合は、インターネット・ゲートウェイを追加する必要があります。以下のドキュメントで、インターネット・ゲートウェイの作成方法が説明されています。

<https://docs.us-phoenix-1.oraclecloud.com/Content/Network/Tasks/managingIGs.htm>

2. セキュリティ・リストのイングレスおよびエグレスのルールを編集して、オンプレミス・データベースと接続します。以下のリンクから追加情報を参照できます。

<https://docs.us-phoenix-1.oraclecloud.com/Content/Network/Concepts/securitylists.htm>

オンプレミス・ネットワークの構成

Data Guard 構成では、プライマリとスタンバイのデータベース間で情報が双方向に伝送されます。これには、基本設定を行い、ネットワークをチューニングし、プライマリとスタンバイの両方のデータベースでポートを開く必要があります。

オンプレミスの前提条件

スタンバイ・データベースをインスタンス化するには、以下の前提条件を満たす必要があります。

- » OCI DBaaS VM への名前解決を設定する必要があります。この設定は、/etc/hosts のような静的ファイルを使用するか、OCI インスタンスのパブリック IP アドレスが正しく解決されるようにオンプレミス DNS を設定することによって行うことができます。また、オンプレミスのファイアウォールでは、オンプレミス・システムと OCI DBaaS VM との間の SSH および Oracle Net アクセスを許可するように適切に設定された、アクセス制御リストが必要になります。
- » DR シチュエーションの Data Guard では、クラウド・インスタンスからオンプレミスのデータベースにアクセスするため、iptables などの機能を使用してクラウドの IP アドレスからのアクセスには制限を設けつつ、プライマリ・データベースのリスナー・ポートを開いておく必要があります。ネットワーク・セキュリティ・ポリシーは企業ごとに異なるため、ネットワーク管理者は、以下のセクションに示すクラウド側のネットワーク構成に類似した操作を実行することが必要になります。
- » OCI DBaaS VM からオンプレミス・マシンへの、プロンプトレス SSH。これはオンプレミスからクラウド、クラウドからオンプレミスの両方で、プロビジョニング・プロセス中に構成されます。
- » OCI DBaaS VM からオンプレミス・マシンへのインバウンド SSH 接続を許可する、オンプレミス・ファイアウォールの構成。

» オンプレミス・データベースの Oracle ホームは、スタンバイ・データベースの Oracle パッチ・セットと同じである必要があります。OCI 環境が別のバンドル・パッチ・レベルにあり、オンプレミス・データベースが別のバンドル・パッチまたは PSU にある場合、クラウド環境のデータベース・ホームと同じデータベース・バンドル・パッチ・レベルに、ソース環境をパッチ適用することをお奨めします（ソース環境とターゲット環境の両方にインストールされているパッチを確認するには、コマンド“\$ORACLE_HOME/OPatch/patch lspatches”を実行します）。

» このドキュメントで概要を示す手順では、オンプレミスのプライマリ・データベースがまだ既存の Data Guard Broker 構成の一部になっていないことを前提としています。オンプレミス・データベースの既存のブローカ構成が存在する場合は、管理者がブローカについての情報を事前に得ており、既存のブローカ構成に新しいスタンバイ・データベースを追加する方法を知っていることを前提とします。次の問合せに対する戻り値が“NOCONFIG”以外の場合は、既存のブローカ構成を意味します。

```
SQL> select decode(count(1),0,'NOCONFIG') from v$DG_BROKER_CONFIG;
```

» LISTENER という名前のデフォルトのリスナーを使用します。このドキュメントで概説する手順では、デフォルトのリスナー名 (LISTENER) を使用することを前提としています。検証するには、オンプレミスのマシンから次のコマンドを実行します。予期される結果が表示されます。

```
$lsnrctl show current_listener | grep 'Current Listener' Current Listener is LISTENER
```

» オンプレミスのマシンから次のコマンドを実行してリスナー・ポートを検証します。予期される結果が表示されます。

```
$lsnrctl status | grep 'Connecting to'
Connecting to (ADDRESS=(PROTOCOL=tcp)(HOST=)(PORT=(1521)))
```

プライマリ・データベースでのMAAベスト・プラクティスのパラメータ設定の実装

ベスト・プラクティスのリストについては、付録 B を参照してください。インスタンス化の前に、プライマリ・データベースでこのプロセスを完了しておくことをお奨めします。プロセス中に複製される REDO ログを構成する場合は特にそのようにしてください。

オンプレミス・ホストとOCI DBaaSホストとの間の接続の検証

上記の手順が完了したら、以下のコマンドを実行して、ソースからターゲットへ、およびターゲットからソースへの接続に問題がないことを確認します。telnet に成功したら、次の手順に進みます。

オンプレミス・ホスト上

```
[root@onpremise1 ~]# telnet xxx.xxx.xxx.xxx 1521
Trying xxx.xxx.xxx.xxx...
Connected to xxx.xxx.xxx.xxx.
Escape character is '^]'.
^C^]q telnet> q
Connection closed.
```

OCI DBaaSホスト上

```
[root@oci2 ~]# telnet xxx.xxx.xxx.xxx 1521
Trying xxx.xxx.xxx.xxx...
Connected to xxx.xxx.xxx.xxx.
Escape character is '^]'.
^]q telnet> q
Connection closed.
```

デプロイメント・プロセス

この手順は基本的に、オンプレミスから OCI へのデータベースの移行、およびシングル・インスタンス (SI) または RAC での Data Guard のセットアップと同じです。SI オンプレミスからクラウド・インフラストラクチャの 2 ノード RAC に、または RAC オンプレミスからクラウド・インフラストラクチャの SI に Data Guard をセットアップするオプションがあります。このホワイト・ペーパーでは、SI から RAC への変換手順については説明しません。

<https://docs.us-phoenix-1.oraclecloud.com/Content/Database/Tasks/mig-rman-duplicate-active-database.htm>

前提条件

以下の手順は、ベア・メタルと VM DBaaS シェイプの両方に適用されます。唯一の違いは、VM サービスは VM シェイプごとに1つのデータベースしかサポートしないということです。オンプレミスのデータベースと同じ名前でデータベースを作成することを推奨します。DB 名は同じにする必要がありますが、db_unique_name はプライマリ・システムとは異なるものにしてください。ソース・データベースの Oracle ホームのパッチ・レベルは、VM DBaaS のデータベースと同じにする必要があります。

表2: DBaaSのエディション

OCI DBaaSのOracle Databaseソフトウェア 含まれるライセンスのエディション	
Standard Edition	Oracle Database Standard Edition 2。 *Data GuardではStandard Editionはサポートされません
Enterprise Edition	Oracle Database Enterprise Edition、Data MaskingおよびSubsetting Pack、DiagnosticsおよびTuning Pack、Real Application Testing。
Enterprise High Performance	Enterprise Editionに以下のオプションをプラス: Multitenant、Partitioning、Advanced Compression、Advanced Security、Label Security、Database Vault、OLAP、Advanced Analytics、Spatial & Graph、Database Lifecycle Management Pack、Cloud Management Pack for Oracle Database。
Enterprise High Performance	High Performanceパッケージに以下のオプションをプラス: In-Memory Database、Active Data Guard、RAC (2台のVMでそれぞれ2個以上のOCPUが必要)

手順1：DBシステムのVMまたはBMを作成する

以下のリンクで説明されている手順に従って、DB システムを起動します。プライマリのオンプレミス・データベースと同じ名前をデータベースに付けます。

<https://docs.cloud.oracle.com/iaas/Content/Database/Tasks/launchingDB.htm>

スタンバイ・データベースとしてベア・メタルを使用する場合は、新しいデータベースを作成し、オンプレミスと同じ名前にすることができます。データベースの名前とバージョンは、ソース・データベースと同じにする必要があります。

手順2：ツールを使用して、作成済みのデータベースを手動で削除する

データベースがプロビジョニングされて準備ができたなら、以下の操作で、初期データベースのファイルを削除してください。そうすると、RMAN を使用して、オンプレミス・データベースがリストアされます。RMAN duplicate を開始する手順は、このドキュメントで後ほど説明します。

初期データベースを削除するには、ASM ディスク・グループから、データベース・ファイルを手動で削除する方法を使用してください。DBCA を使用すると、スタンバイのために保持しておくべき `srvctl` の登録や `/etc/oratab` のエントリも削除されてしまうため、使用しないでください。

OCI DBaaS ホストのデータベースを手動で削除するには、以下の手順を実行します。

1. OCI DBaaS データベースの現在の `db_unique_name` を確認します。これは、これ以降の手順で全体を通して使用されます。

```
$ srvctl config database
```

2. スクリプトを作成して、すべてのデータベース・ファイルを削除します。

```
SQL> set heading off linesize 999 pagesize 0 feedback off trimspool on
SQL> spool /tmp/files.lst
SQL> select 'asmcmd rm '||name from v$datafile
union all
select 'asmcmd rm '||name from v$tempfile
union all
select 'asmcmd rm '||member from v$logfile;
SQL> spool off
SQL> create pfile='/tmp/<standby DB_UNIQUE_NAME>.pfile' from spfile; #Backup spfile
$chmod 777 /tmp/files.lst
```

3. データベースをシャットダウンします。

後で参照するために、データベースの構成を最初に収集しておきます。

```
$ srvctl config database -d <db_unique_name> /tmp/<standby db_unique_name>.config
```

次に、データベースを停止させます。

```
$ srvctl stop database -d <db_unique_name> -o immediate
```

4. データベース・ファイルを削除します。

既存のデータファイル、ログ・ファイル、一時ファイルを削除します。パスワード・ファイルが置き換えられ、spfile が再利用されます。

grid ユーザーとして (opc ユーザーから sudo を実行して grid ユーザーに切り替える)、以下のようにします。

作成済みの/tmp/files.lst を編集して、sqlplus から不要な行をすべて削除します。'asmcmd'で始まる行はすべて残します。

スクリプトを保存して実行します。

```
[grid@<host> ~]$ . /tmp/files.lst
```

これで、初期データベースの全ファイルが削除されました。

手順3：パスワード・ファイルをOCI DBaaSホストにコピーする

オンプレミスのパスワード・ファイルを、OCI DBaaS ホストの\$ORACLE_HOME/dbs/orapw<ORACLE_SID>にコピーします。

パスワード・ファイルの場所を確認する

Oracle Grid Infrastructure がオンプレミスのホストで実行されている場合、"srvctl config database -db <db_name>"でパスワード・ファイルの場所を確認します。Oracle Grid Infrastructure が構成されていない、またはパスワード・ファイルの場所が NULL の場合は、パスワード・ファイルはデフォルトの場所 (\$ORACLE_HOME/dbs/orapw<ORACLE_SID>) にあります。

```
$ srvctl config database -db testdbname Database unique name: testdbname
Database name:
Oracle home: /u02/app/oracle/product/12.1.0/dbhome_2
Oracle user: oracle
Spfile: +DATA/testdbname/spfiledbtestdbname.ora
Password file: +DATA/testdbname/PASSWORD/orapw<sid> <===== password file location
Domain: domainname.xxxxx.xxxxx
```

オンプレミスのパスワード・ファイルをOCI DBaaSホストにコピーする

OCI DBaaS が RAC データベースである場合は、OCI DBaaS の全ノードにパスワード・ファイルをコピーします。パスワード・ファイルの場所が ASM ではない場合は、以下のようにしてファイルをコピーします。

```
$ scp -i <ssh key for OCI DBaaS Host> $ORACLE_HOME/dbs/orapw<SID> opc@<Public-IP-OCI- HOST>:~
```

パスワード・ファイルの場所が ASM である場合は、ユーザーを"grid"または ASM オーナーに切り替え、以下の
ように環境変数を取得して、パスワード・ファイルをコピーします。

```
$ sudo su - grid
$ export ORACLE_SID=<ASM ORACLE_SID>
$ export ORACLE_HOME=<GRID_HOME>
$ asmcmd
ASMCMD> cd +<DISKGROUP_NAME>/<DB_UNIQUE_NAME>/PASSWORD
ASMCMD> cp orapw<SID> /tmp
copying +DATA/<DB_UNIQUE_NAME>/PASSWORD/orapw<sid> -> /tmp/orapw<sid>

scp -i <ssh key for OCI DBaaS Host> /tmp/orapw<SID> opc@<Public-IP-OCI-HOST>:/tmp
```

OCI DBaaS ホストの\$ORACLE_HOME/dbs/orapw<SID>にパスワード・ファイルをコピーする

OCI DBaaS ホストの opc ユーザーとして、以下のようになります。

```
$ chmod 777 /tmp/<password file name>
$ sudo su - oracle
```

シングル・インスタンスの OCI DBaaS では、\$ORACLE_HOME/dbs にパスワード・ファイルを保存します。

```
$ cp /tmp/<password file name> <ORACLE_HOME>/dbs/orapw<SID>
```

RAC DBaaS 構成では、ASM にパスワード・ファイルを保存します。grid ユーザーとして、以下のようになります。

```
ASMCMD> pwcopyp --dbuniquename <standby DB_UNIQUE_NAME>
/tmp/pwdvictor.858.1001423135 +DATA/victor_phx2w5/orapwvictor_phx2w5 -f
```

注: ASMCMD-9453: failed to register password file as a CRS resource と表示された場合は、oracle ユーザーとして以下を実行します。

```
$ srvctl modify database -d <standby DB_UNIQUE_NAME> -pwfile
'+DATA/victor_phx2w5/orapwvictor_phx2w5'
```

パスワード・ファイルが正しく登録されていることを確認します (oracle として)。

```
$ srvctl config database -d <standby DB_UNIQUE_NAME>
```

手順4: ウォレット・ファイルをOCI DBaaSホストにコピーする

\$ORACLE_HOME/network/admin/sqlnet.ora に以下の行が含まれていること、ウォレット・ファイルの場所が
sqlnet.ora で ENCRYPTION_WALLET_LOCATION パラメータとして定義されていることを確認します。

オンプレミス・ホストの SQLNET.ORA

```
ENCRYPTION_WALLET_LOCATION=(SOURCE=(METHOD=FILE)
(METHOD_DATA=(DIRECTORY=/opt/oracle/dcs/commonstore/wallets/tde/$ORACLE_UNQNAME)))
```

ORACLE_UNQNAME 環境変数はクラスタウェアで設定され、スタンバイ・データベースの *DB_UNIQUE_NAME* と同じになります。oracle ユーザーの環境では、この変数の設定を推奨します。

オンプレミスの ewallet.p12 と cwallet.sso ファイルを、上記の OCIDBaas ホストのディレクトリにコピーします。

オンプレミス・ホスト上

```
scp -i ~/<ssh_key> ewallet.p12 opc@<Public-IP-OCI-HOST>:/tmp
scp -i ~/<ssh_key> cwallet.sso opc@<Public-IP-OCI-HOST>:/tmp
```

/opt/oracle/dcs/commonstore/wallets/tde/\$ORACLE_UNQNAME の古いウォレット・ファイルを削除します。OCI DBaaS が RAC データベースである場合は、ノード 1 のみでコマンドを実行します (ACFS を使用して /opt/oracle/dcs/commonstore で共有される VM RAC ノードのストレージ)。

OCI DBaaS ホスト上

```
$ chmod 777 /tmp/ewallet.p12
$ chmod 777 /tmp/cwallet.sso
$ sudo su - oracle
$ cp /tmp/ewallet.p12 /opt/oracle/dcs/commonstore/wallets/tde/$ORACLE_UNQNAME
$ cp /tmp/cwallet.sso /opt/oracle/dcs/commonstore/wallets/tde/$ORACLE_UNQNAME chmod
600 /opt/oracle/dcs/commonstore/wallets/tde/$ORACLE_UNQNAME/*wallet*
```

手順5：静的リスナーを構成する

スタンバイ・データベースの初回のインスタンス化には静的リスナーが必要です。静的リスナーにより、所定のインスタンスを開始するためにデータベースが停止している間でも、インスタンスにリモート接続することができます。詳しくは、MOS 1387859.1 を参照してください。

grid ユーザーとして、変数を置き換えてから、クラウド DBaaS とオンプレミスのホストの両方の listener.ora に、次のエントリを追加します。listener.ora は \$ORACLE_HOME/network/admin にあります。

listener.ora

```
SID_LIST_LISTENER =
  (SID_LIST = (SID_DESC =
    (GLOBAL_DBNAME = <DB_UNIQUE_NAME of the OCI database>) (ORACLE_HOME = <Local
Oracle Home>)
    (SID_NAME = <ORACLE SID of the local instance>)
  ))
```

11.2 の構成では、Data Guard Broker に静的リスナーも必要です。変数を置き換えてから、オンプレミスの listener.ora に次のエントリを追加します。RAC の場合は両ノードについてこれを行う必要があります。

```
SID_LIST_LISTENER =
  (SID_LIST = (SID_DESC =
    (GLOBAL_DBNAME = <DB_UNIQUE_NAME of the OCI database>_DGMGRL)
    (ORACLE_HOME = <Local Oracle Home>) (SID_NAME = <ORACLE SID of the local
instance>))
  ))
```

最後に、リスナーを再ロードします (grid ユーザーとして)。

```
$ORACLE_HOME/bin/lsnrctl reload
```

スタンバイ・インスタンスを起動する

```
srvctl start instance -d <standby DB_UNIQUE_NAME> -i <standby instance name> -o mount
```

手順6：REDO転送用のオラクルのネットワーク暗号化とTNSエントリ

WAN 上でプレーンテキストまたは暗号化されていない表領域の REDO が可視状態にならないよう保護するには、オンプレミスとクラウドのすべてのマシンで、\$ORACLE_HOME/network/admin に格納されている sqlnet.ora ファイルに次のエントリを入力します。

オンプレミス・ホスト上の sqlnet.ora

```
SQLNET.ENCRYPTION_SERVER=REQUIRED
SQLNET.CRYPTO_CHECKSUM_SERVER=REQUIRED
SQLNET.ENCRYPTION_TYPES_SERVER=(AES256,AES192,AES128)
SQLNET.CRYPTO_CHECKSUM_TYPES_SERVER=(SHA1)
SQLNET.ENCRYPTION_CLIENT=REQUIRED
SQLNET.CRYPTO_CHECKSUM_CLIENT=REQUIRED
SQLNET.ENCRYPTION_TYPES_CLIENT=(AES256,AES192,AES128)
SQLNET.CRYPTO_CHECKSUM_TYPES_CLIENT=(SHA1)
```

REDO転送が正常に行われるようにするには、プライマリとスタンバイの両方の `tnsnames.ora` ファイルに各データベースのエントリが含まれている必要があります。以下の例を、構成に沿った値で置き換えて使用してください。

注：プライマリ・データベースのオンプレミス `tnsnames.ora` には、サーバー名が `HOST` になっている `TNS` エントリがすでに含まれていることがあります。その場合は、ホストの `IP` アドレスを使用してエントリのサーバー名を変更します。

注：オンプレミス環境とクラウド環境の間にはサーバー名を `IP` アドレスに解決する `DNS` が存在しないため、`IP` アドレスが使用されます。

注：`RAC` 構成では、スキャン・リスナー名を解決できないため、全ノードが記載されたアドレス・リストを使用する必要があります。それらは後から追加され、初期インスタンスの `TNS` エントリでリストに入れられるのは1つの `IP` のみです。これで、`RMAN` は常に同じノードに接続されます。

オンプレミス・ホスト上の `tnsnames.ora`

```
<standby db_unique_name> = (DESCRIPTION =
(SDU=65536) (RECV_BUF_SIZE=134217728)
(SEND_BUF_SIZE=134217728)
(ADDRESS_LIST =
(ADDRESS = (PROTOCOL = TCP) (HOST = <standby IP address>) (PORT = {1521|<port#>}))
)
(CONNECT_DATA =
(SERVER = DEDICATED)
(SERVICE_NAME = <standby db_unique_name>[.<standby domain name>])
(UR=A)
))
```

OCI DBaaS ホスト上の `tnsnames.ora`

```
<primary db_unique_name> = (DESCRIPTION =
(SDU=65536) (RECV_BUF_SIZE=134217728)
(SEND_BUF_SIZE=134217728)
(ADDRESS_LIST =
(ADDRESS = (PROTOCOL = TCP) (HOST = <primary IP address>) (PORT = {1521|<port#>}))
)
(CONNECT_DATA =
(SERVER = DEDICATED)
(SERVICE_NAME = <primary db_unique_name>[.<primary domain name>])
(UR=A)
))
```

TCPソケット・サイズを設定する

次のコマンドを root として実行して、オンプレミス・システムおよびクラウド・インスタンス用の TCP ソケットのサイズを確認します (OCI DBaaS の TCP ソケット・サイズは 128 MB)。

オンプレミス・ホスト上

```
# /sbin/sysctl -a | egrep net.core.[w,r]mem_max
net.core.wmem_max = 2097152
net.core.rmem_max = 4194304
# /sbin/sysctl -a | egrep net.core.[w,r]mem_max
net.core.wmem_max = 1048576
net.core.rmem_max = 4194304
```

必要に応じて、すべてのソケットの最大サイズを 128 MB (134217728) に調整します。オンプレミス・システムの場合のソケットの調整方法について詳しくは、オペレーティング・システム・ガイドを参照してください。クラウド・インスタンスの場合、net.core.wmem_max および net.core.rmem_max の/etc/sysctl.conf ファイル設定を編集します。オンプレミスと OPC 間の値が一致しない場合、ネットワーク・プロトコルは、2 つの値のうち、低い方をネゴシエートします。したがって、サイト間の値は一致する必要はありませんが、最適な転送パフォーマンスを達成するには一致させることをお勧めします。

```
net.core.rmem_max = 134217728 net.core.wmem_max = 134217728
```

手順7：スタンバイ・データベースをインスタンス化する

スタンバイ・データベースは、アクティブなプライマリ・データベースから、またはプライマリ・データベースのバックアップから作成できます。このセクションでは、Oracle 12.1 以降の機能である RMAN 'RESTORE...FROM SERVICE'を使用して、アクティブなプライマリ・データベースから複製を実行する方法を説明します。

バックアップは、データベースのインスタンス化に使用することもできますし、データベースのサイズおよびシステム間の転送レートによっては、効率を上げることもできます。

RDBMS 11.2 では、RMAN RESTORE FROM SERVICE はサポートされません。複製または RMAN DUPLICATE をベースとするバックアップを使用する必要があります。

詳しくは [ドキュメント](#) を参照してください。

スタンバイのインスタンス化について詳しくは、[MOS 2275154.1 の『Creating a Physical Standby Database in an 11.2, 12.1, 12.2 or later environment』](#) を参照してください。

スタンバイ・インスタンスを起動する (RAC で1インスタンス)

```
$ srvctl stop database -d <standby DB_UNIQUE_NAME> -o immediate
$ rman target /
RMAN> startup nomount
```

```
RMAN> restore standby controlfile from service 'primary';
```

```
RMAN> alter database mount;
```

```
RMAN> restore database from service 'primary' section size 5G;
```

```
RMAN> shutdown immediate
```

Restart the standby database

```
$ srvctl start database -d <standby DB_UNIQUE_NAME> -o mount
```

すべてのオンラインおよびスタンバイ REDO ログを消去します。

```
$ sqlplus "/ as sysdba"
SQL> alter system set db_create_online_log_dest_1=<DATA Disk group>;
SQL> set pagesize 0 feedback off linesize 120 trimspool on
SQL> spool /tmp/clearlogs.sql
SQL> select distinct 'alter database clear logfile group '||group#||';' from
v$logfile;
SQL> spool off
SQL> @/tmp/clearlogs.sql

SQL> select member from v$logfile;
```

注：すべての REDO ログは、スタンバイ DB_UNIQUE_NAME ディレクトリの DATA ディスク・グループにある必要があります。

手順8：tnsnames.oraにRACインスタンスを追加する (RACのみ)

ハイブリッド構成では、いずれの方向でもスキャン名を解決できないため、アドレス・リストを構成して高可用性を確保する必要があります。

注：各ノードで異なる順番で、アドレス・リストにアドレスを入れ、ノード全体で REDO 転送のバランスを保ちます。

オンプレミス・ホスト上の tnsnames.ora

```
<standby db_unique_name> = (DESCRIPTION =
  (DESCRIPTION=
    (ADDRESS_LIST=
      (FAILOVER=on)
      (CONNECT_TIMEOUT=3) (RETRY_COUNT=3)
      (ADDRESS = (PROTOCOL = TCP) (HOST = <standby node1 IP address>) (PORT =
{1521|<port#>}))
    )
    (ADDRESS = (PROTOCOL = TCP) (HOST = <standby node2 IP address>) (PORT =
{1521|<port#>}))
    )
  )
  (CONNECT_DATA=
    (SERVER=DEDICATED)
    (SERVICE_NAME= standby db_unique_name>[.<standby domain name>])
  )
)
```

OCI DBaaS ホスト上の tnsnames.ora

```
<primary db_unique_name> =
(DESCRIPTION=
  (ADDRESS_LIST=
    (FAILOVER=on)
    (CONNECT_TIMEOUT=3) (RETRY_COUNT=3)
    (ADDRESS = (PROTOCOL = TCP) (HOST = <primary node1 IP address>) (PORT =
{1521|<port#>}))
  )
  (ADDRESS = (PROTOCOL = TCP) (HOST = <primary node2 IP address>) (PORT =
{1521|<port#>}))
  )
  (CONNECT_DATA =
    (SERVER = DEDICATED)
    (SERVICE_NAME = <primary db_unique_name>[.<primary domain name>])
  )
))
```

手順9：Data Guard Brokerを構成する

プライマリ・データベースとスタンバイ・データベースで、dg_broker_config_file のパラメータを有効にします。

注：ASM では、別々のディスク・グループにブローカの設定ファイルを入れます。RAC では、共有ストレージにブローカの設定ファイルを入れる必要があります。

オンプレミスおよび OCI DBaaS ホスト上

```
SQL> alter system set dg_broker_config_file1=<broker_config_file location>;
SQL> alter system set dg_broker_config_file2=<broker_config_file location>;
```

プライマリ・データベースとスタンバイ・データベースで Data Guard Broker のプロセスを開始する

オンプレミスおよび OCI DBaaS ホスト上

```
SQL> alter system set dg_broker_start=true; SQL> show parameter dg_broker_start
```

NAME	TYPE	VALUE
-----	-----	-----
dg_broker_start	boolean	TRUE

```
SQL> select pname from v$process where pname like 'DMON%';
```

```
PNAME
-----
DMON
```

プライマリ・サイトで DGMGRL を使用してデータベースを登録する

オンプレミス・ホスト上

```
$ dgmgrl sys/<sys password>@<net service name for primary database>
DGMGRL> CREATE CONFIGURATION <configuration_name> AS PRIMARY DATABASE IS <primary
database_name> CONNECT IDENTIFIER IS <Net Service name for primary database>;
DGMGRL> ADD DATABASE <standby database_name> AS CONNECT IDENTIFIER IS <Net Service
name for standby database> MAINTAINED AS PHYSICAL;
DGMGRL> enable configuration;
```

クライアントのフェイルオーバーを構成する

クライアント・フェイルオーバーの自動化とは、障害の後にアクティブなプライマリ・データベースにクライアントを再接続するプロセスのことです。Data Guard フェイルオーバーの一部として新しいプライマリ・データベースにデータベース・サービスを再配置する処理、TCP のタイムアウトが起こらないよう、障害が発生したことをクライアントに通知する処理、新しいプライマリ・データベースにクライアントをリダイレクトする処理が含まれます。構成については、Oracle [Database 11g](#) および Oracle [Database 12c](#) のクライアント・フェイルオーバーに関する MAA ベスト・プラクティスのホワイト・ペーパーで詳しく説明されています。Oracle Database 12c 以降を使用しているユーザーは、非常に包括的なアプリケーションの高可用性を継続して維持するために、[アプリケーション・コンティニューイティ](#)を構成することを選択するかもしれません。これらのホワイト・ペーパーを参照して環境を適切に構成してください。

ヘルス・チェックと監視

スタンバイをインスタンス化したら、ヘルス・チェックを実行して、Data Guard データベース（プライマリとスタンバイ）が Oracle MAA のベスト・プラクティスに準拠していることを確認します。ヘルス・チェックは、毎月実施するとともに、データベースのメンテナンスの前後にも実施するようお奨めします。Data Guard 構成のヘルス・チェックは、いくつかの方法で実施できます。

Oracle MAAスコアカード

オラクルでは、複数の自動化されたヘルス・チェック・ツールを提供しており、ハードウェア・プラットフォームのタイプごとに My Oracle Support からダウンロードすることができます。

- » [ORAchk](#) : 汎用プラットフォームに適用可能（Database Cloud Service に最適）
- » [exachk](#) : Oracle Exadata Database Machine に適用可能（Exadata Cloud Service に最適）

自動化された各チェック機能には、Data Guard 構成の多数の主要なベスト・プラクティスおよびその他多くのチェック項目について報告する Oracle MAA スコアカードが組み込まれています。

これらの自動化ツールは、Data Guard 構成だけでなく、システム全体の包括的なヘルス・チェックでも使用することを強く推奨します。ヘルス・チェックの結果は、最新の情報に基づいて定期的に更新されます。必ず、ご使用のプラットフォームに該当する最新バージョンのヘルス・チェック・ツールをダウンロードしてください。

監視

Data Guard 構成の定期的な監視は、Hybrid Data Guard 構成では行われなため、手動で実行する必要があります。監視に関する MAA ベスト・プラクティスについては、MOS Note 『[Monitoring a Data Guard Configuration \(Doc ID 2064281.1\)](#)』を参照してください。

DRの準備状況の検証

ベスト・プラクティスは、Active Data Guard を使用して読取り専用ワークロードをスタンバイ・データベースにオフロードし、スタンバイで本番稼働の準備ができていることをアプリケーションレベルで検証し続けることです。これにより、Data Guard の適用プロセスで実行される継続的な Oracle ブロックレベルの検証に加えて、一定レベルの保証が得られます。また、（Data Guard スナップショット・スタンバイを使用して）スタンバイを定期的に読取り/書き込みモードにし、読取り/書き込みの本番ワークロードにいつでも対応できるようになっているか検証することもベスト・プラクティスです。DR システムは本番システムと同様にサイジングされるので、パッチやアップグレードの本番前機能テストやパフォーマンス・テストの最終段階でスナップショット・スタンバイを使用することもできます。スナップショット・スタンバイはプライマリ・データベースから REDO を受信し続け、後で使用するために REDO をアーカイブし、それにより常時データを保護します。ただし、テストの進行中にフェイルオーバーが必要になった場合のリカバリ時間（RTO）は、スナップショット・スタンバイを元のスタンバイ・データベースに変換するために必要とされる時間だけ長くなります。スタンバイがスナップショット・モードになっている場合は、（プライマリ本番データベースから受信し後で使用するためにアーカイブされた REDO ログと、スナップショット・スタンバイによって生成される最新の REDO ログおよびフラッシュバック・ログを保持するため、）ファスト・リカバリ領域として追加のストレージが必要になります。スタンバイをスナップショット・スタンバイに変換し、戻すための手順は、以下の項に記載されています。Data Guard スナップショット・スタンバイの詳細については、Oracle ドキュメントを参照してください。必要に応じ

て、完全なエンド・ツー・エンドの DR テストとして、クラウドへの実際のスイッチオーバーまたはフェイルオーバー操作を実行することもできます。詳しくは、「クラウドへのフェイルオーバー/スイッチオーバー」を参照してください。

スタンバイ・データベースからスナップショット・スタンバイへの変換

スナップショット・スタンバイは、フィジカル・スタンバイ・データベースから作成される、すべてを更新可能なスタンバイ・データベースです。スナップショット・スタンバイ・データベースでは、REDO データを受信しますが、スナップショット・スタンバイ・データベースが変換されてフィジカル・スタンバイ・データベースに戻されるまで適用されません。

スナップショット・スタンバイ・データベースを使用することには、次のような利点があります。

1. 常時データが保護された状態を維持しつつ、開発およびテスト目的で本番データベースの正確なレプリカとして機能します。Oracle Real Application Testing オプションを使用して、プライマリ・データベースのワークロードを捕捉し、スナップショット・スタンバイで、テスト目的で再生することができます。
2. フィジカル・スタンバイへの変換と再同期化により、容易にリフレッシュして現在の本番データを含めることができます。

フィジカル・スタンバイ・データベースをスナップショット・スタンバイに変換するには、以下の手順に従います。

スタンバイをスナップショット・スタンバイに変換して検証する

Data Guard Broker から、次のコマンドを発行します。

```
DGMGRL> convert database 'stby' to snapshot standby;
DGMGRL> SHOW CONFIGURATION;
Configuration - DRSolution
Protection Mode:MaxPerformance Databases:
prmy - Primary database stby - Snapshot standby database
Fast-Start Failover:DISABLED
Configuration Status:SUCCESS
```

注：スナップショット・スタンバイ・データベースをスイッチオーバーまたはフェイルオーバーのターゲットにすることはできません。スナップショット・スタンバイ・データベースは、そのデータベースへのロール移行を実行する前に、まず変換してフィジカル・スタンバイ・データベースに戻す必要があります。

スナップショット・スタンバイから元のフィジカル・スタンバイ・データベースに変換します。

Data Guard Broker から、次のコマンドを発行します。

```
DGMGRL> CONVERT DATABASE 'stby' to PHYSICAL STANDBY;
```

クラウドへのフェイルオーバー/スイッチオーバー

Data Guard のスイッチオーバー（計画イベント）やフェイルオーバー（計画外イベント）は、いつでも手動で実行することができます。また、ファスト・スタート・フェイルオーバーを設定することによる Data Guard のフェイルオーバーの自動化も選択できます。スイッチオーバーとフェイルオーバーにより、Data Guard 構成におけるデータベースのロールが逆転し、クラウドのスタンバイがプライマリになり、元のオンプレミスのプライマリがスタンバイ・データベースになります。Data Guard のロール移行について詳しくは、Oracle MAA のベスト・プラクティスを参照してください。

スイッチオーバーは常に、データを失わないことが保証された計画イベントとなります。スイッチオーバーを実行するには、Data Guard Broker で以下のコマンドを実行します。

```
DGMGRL> validate database stby;
Database Role:Physical standby database Primary Database: pri
Ready for Switchover:Yes
Ready for Failover:Yes (Primary Running)
DGMGRL> switchover to <target standby>;
```

フェイルオーバーは、プライマリ・データベースで障害が発生することを想定した計画外イベントです。使用可能なプライマリのすべての REDO が適用された後、スタンバイ・データベースが即時にプライマリ・データベースに変換されます。フェイルオーバーの後、古いプライマリ・データベースはフィジカル・スタンバイとして復旧する必要があります。これは、フラッシュバック・データベースと Data Guard Broker を有効にすることによって簡単に行うことができます。フェイルオーバーと復旧を実行するには、Data Guard Broker で以下のコマンドを実行します。

```
DGMGRL> failover to stby;
Performing failover NOW, please wait...
Failover succeeded, new primary is "stby"
Execute startup mount on one instance of the old primary before reinstating.
SQL> shutdown abort
SQL> startup mount
DGMGRL> reinstate database pri
Reinstating database "pri", please wait...
```

Data Guard Broker を使用したロール移行について詳しくは、Oracle Database 11g または 12c 用の Broker のドキュメントを参照してください。

オンプレミスへのスイッチバック

本番データベースをオンプレミス・データベースに移行する準備ができれば、フェイルオーバー/スイッチオーバーのプロセスで述べられているのと同じロール移行手順を再び適用します。



結論

OCI DBaaS システムを使った Hybrid Data Guard は、ディザスタ・リカバリを準備する経済的な方法です。Maximum Availability Architecture のベスト・プラクティスは、データの保護と可用性へのベスト・ソリューションを保証します。

付録A : RMAN Duplicateの出力例 :

```
RMAN> duplicate target database for standby from active database ;
```

```
Starting Duplicate Db at 11-JAN-18
using target database control file instead of recovery catalog
allocated channel:ORA_AUX_DISK_1
channel ORA_AUX_DISK_1:SID=771 device type=DISK
```

```
contents of Memory Script:
```

```
{
    backup as copy reuse
        targetfile '+DATA1/racddb/PASSWORD/passwd' auxiliary format
        '/u01/app/oracle/product/12.1.0.2/dbhome_1/dbs/orapwracddb' ;
}
```

```
executing Memory Script
```

```
Starting backup at 11-JAN-18
allocated channel:ORA_DISK_1
channel ORA_DISK_1:SID=785 instance=racddb1 device type=DISK
Finished backup at 11-JAN-18
```


```
contents of Memory Script:
```

```
{
    sql clone "alter system set control_files =
        '+RECO/RACCCDB_PHX34Z/CONTROLFILE/current.263.965119865'" comment=
        ''Set by RMAN'' scope=spfile";
    backup as copy current controlfile for standby auxiliary
    format '+RECO/RACCCDB_PHX34Z/CONTROLFILE/current.264.965119865';
    sql clone "alter system set control_files =
        '+RECO/RACCCDB_PHX34Z/CONTROLFILE/current.264.965119865'" comment=
        ''Set by RMAN'' scope=spfile";
    shutdown clone immediate;
    startup clone nomount;
}
```

```
executing Memory Script
```

```
sql statement: alter system set control_files
= '+RECO/RACCCDB_PHX34Z/CONTROLFILE/current.263.965119865' comment= ''Set by
RMAN'' scope=spfile
```

```
Starting backup at 11-JAN-18
using channel ORA_DISK_1
channel ORA_DISK_1: starting datafile copy
copying standby control file
```



```
output file name=+RECO/RACEDB_PHX34Z/CONTROLFILE/current.264.965119865
tag=TAG20180111T085105
channel ORA_DISK_1: datafile copy complete, elapsed time:00:00:03
Finished backup at 11-JAN-18
```

```
sql statement: alter system set control_files
= '+RECO/RACEDB_PHX34Z/CONTROLFILE/current.264.965119865' comment= 'Set by
RMAN' scope=spfile
```

```
Oracle instance shut down
```

```
connected to auxiliary database (not started)
Oracle instance started
```

```
Total System Global Area      15032385536 bytes
```

```
Fixed Size          3728304 bytes
Variable Size       2348813392 bytes
Database Buffers    12616466432 bytes
Redo Buffers        63377408 bytes
```

```
contents of Memory Script:
```

```
{
  sql clone 'alter database mount standby database';
}
```

```
executing Memory Script
```

```
sql statement: alter database mount standby database
```

```
contents of Memory Script:
```

```
{
  set newname for clone tempfile 1 to new;
  set newname for clone tempfile 2 to new;
  set newname for clone tempfile 4 to new;
  switch clone tempfile all;
  set newname for clone datafile 1 to new;
  set newname for clone datafile 2 to new;
  set newname for clone datafile 3 to new;
  set newname for clone datafile 4 to new;
  set newname for clone datafile 5 to new;
  set newname for clone datafile 6 to new;
  set newname for clone datafile 7 to new;
  set newname for clone datafile 8 to new;
  set newname for clone datafile 9 to new;
  set newname for clone datafile 10 to new;
  backup as copy reuse
```


```
datafile 1 auxiliary format new
datafile 2 auxiliary format new
datafile 3 auxiliary format new
datafile 4 auxiliary format new
datafile 5 auxiliary format new
datafile 6 auxiliary format new
datafile 7 auxiliary format new
datafile 8 auxiliary format new
datafile 9 auxiliary format new
datafile 10 auxiliary format new
;
sql 'alter system archive log current';
}
executing Memory Script
executing command:SET NEWNAME
executing command:SET NEWNAME
executing command:SET NEWNAME

renamed tempfile 1 to +DATA in control file
renamed tempfile 2 to +DATA in control file
renamed tempfile 4 to +DATA in control file

executing command:SET NEWNAME
executing command:SET NEWNAME
executing command:SET NEWNAME
executing command:SET NEWNAME
executing command:SET NEWNAME
executing command:SET NEWNAME
executing command:SET NEWNAME
executing command:SET NEWNAME
executing command:SET NEWNAME
executing command:SET NEWNAME

Starting backup at 11-JAN-18
using channel ORA_DISK_1
channel ORA_DISK_1: starting datafile copy
input datafile file number=00009 name=+DATAC1/raccdb/tokyo/tokyo_sysaux.dbf
output file
name=+DATA/RACCDDB_PHX34Z/627B610857F86EA9E0537701000A1DB6/DATAFILE/sysaux.266.96511
99 11 tag=TAG20180111T085150
channel ORA_DISK_1: datafile copy complete, elapsed time:00:00:45
channel ORA_DISK_1: starting datafile copy
input datafile file number=00001 name=+DATAC1/raccdb/DATAFILE/system.dbf
output file name=+DATA/RACCDDB_PHX34Z/DATAFILE/system.265.965119957
tag=TAG20180111T085150
channel ORA_DISK_1: datafile copy complete, elapsed time:00:00:25
```

```
channel ORA_DISK_1: starting datafile copy
input datafile file number=00003 name=+DATAC1/racddb/DATAFILE/sysaux.dbf
output file name=+DATA/RACCDDB_PHX34Z/DATAFILE/sysaux.272.965119983
tag=TAG20180111T085150
channel ORA_DISK_1: datafile copy complete, elapsed time:00:00:25
channel ORA_DISK_1: starting datafile copy
input datafile file number=00005 name=+DATAC1/racddb/DATAFILE/undotbs1.dbf
output file name=+DATA/RACCDDB_PHX34Z/DATAFILE/undotbs1.263.965120007
tag=TAG20180111T085150
channel ORA_DISK_1: datafile copy complete, elapsed time:00:00:25
channel ORA_DISK_1: starting datafile copy
input datafile file number=00006 name=+DATAC1/racddb/DATAFILE/undotbs2.dbf
output file name=+DATA/RACCDDB_PHX34Z/DATAFILE/undotbs2.262.965120033
tag=TAG20180111T085150
channel ORA_DISK_1: datafile copy complete, elapsed time:00:00:25
channel ORA_DISK_1: starting datafile copy
input datafile file number=00007 name=+DATAC1/racddb/DATAFILE/users.dbf
output file name=+DATA/RACCDDB_PHX34Z/DATAFILE/users.261.965120057
tag=TAG20180111T085150
channel ORA_DISK_1: datafile copy complete, elapsed time:00:00:15
channel ORA_DISK_1: starting datafile copy
input datafile file number=00002 name=+DATAC1/racddb/PDBSEED/DATAFILE/system.dbf
output file
name=+DATA/RACCDDB_PHX34Z/4D8C4492AD829E0DE0530C4F800ADE8D/DATAFILE/system.274.9651200
73 tag=TAG20180111T085150
channel ORA_DISK_1: datafile copy complete, elapsed time:00:00:07
channel ORA_DISK_1: starting datafile copy
input datafile file number=00004 name=+DATAC1/racddb/PDBSEED/DATAFILE/sysaux.dbf
output file
name=+DATA/RACCDDB_PHX34Z/4D8C4492AD829E0DE0530C4F800ADE8D/DATAFILE/sysaux.275.9651200
79 tag=TAG20180111T085150
channel ORA_DISK_1: datafile copy complete, elapsed time:00:00:07
channel ORA_DISK_1: starting datafile copy
input datafile file number=00008 name=+DATAC1/racddb/tokyo/tokyo_system.dbf
output file
name=+DATA/RACCDDB_PHX34Z/627B610857F86EA9E0537701000A1DB6/DATAFILE/system.276.9651200
87 tag=TAG20180111T085150
channel ORA_DISK_1: datafile copy complete, elapsed time:00:00:07
channel ORA_DISK_1: starting datafile copy
input datafile file number=00010 name=+DATAC1/racddb/tokyo/tokyo_users01.dbf
output file
name=+DATA/RACCDDB_PHX34Z/627B610857F86EA9E0537701000A1DB6/DATAFILE/users.277.96512009
3 tag=TAG20180111T085150
channel ORA_DISK_1: datafile copy complete, elapsed time:00:00:01
Finished backup at 11-JAN-18
```



sql statement: alter system archive log current

contents of Memory Script:

```
{  
    switch clone datafile all;  
}
```

executing Memory Script

datafile 1 switched to datafile copy

input datafile copy RECID=1 STAMP=965120096 file
name=+DATA/RACCCDB_PHX34Z/DATAFILE/system.265.965119957

datafile 2 switched to datafile copy

input datafile copy RECID=2 STAMP=965120096 file
name=+DATA/RACCCDB_PHX34Z/4D8C4492AD829E0DE0530C4F800ADE8D/DATAFILE/system.274.9651200
73

datafile 3 switched to datafile copy

input datafile copy RECID=3 STAMP=965120096 file
name=+DATA/RACCCDB_PHX34Z/DATAFILE/sysaux.272.965119983

datafile 4 switched to datafile copy

input datafile copy RECID=4 STAMP=965120096 file
name=+DATA/RACCCDB_PHX34Z/4D8C4492AD829E0DE0530C4F800ADE8D/DATAFILE/sysaux.275.9651200
79

datafile 5 switched to datafile copy

input datafile copy RECID=5 STAMP=965120096 file
name=+DATA/RACCCDB_PHX34Z/DATAFILE/undotbs1.263.965120007

datafile 6 switched to datafile copy

input datafile copy RECID=6 STAMP=965120096 file
name=+DATA/RACCCDB_PHX34Z/DATAFILE/undotbs2.262.965120033

datafile 7 switched to datafile copy

input datafile copy RECID=7 STAMP=965120096 file
name=+DATA/RACCCDB_PHX34Z/DATAFILE/users.261.965120057

datafile 8 switched to datafile copy

input datafile copy RECID=8 STAMP=965120096 file
name=+DATA/RACCCDB_PHX34Z/627B610857F86EA9E0537701000A1DB6/DATAFILE/system.276.9651200
87

datafile 9 switched to datafile copy

input datafile copy RECID=9 STAMP=965120096 file
name=+DATA/RACCCDB_PHX34Z/627B610857F86EA9E0537701000A1DB6/DATAFILE/sysaux.266.9651199
11


datafile 10 switched to datafile copy

input datafile copy RECID=10 STAMP=965120096 file
name=+DATA/RACCCDB_PHX34Z/627B610857F86EA9E0537701000A1DB6/DATAFILE/users.277.96512009
3

付録B：MAAベスト・プラクティスのパラメータ設定

MAA ベスト・プラクティスに従ってデータの最大限の可用性と保護を達成するには、以下の設定を推奨します。以下のパラメータをプライマリ・データベースとスタンバイ・データベースの両方に設定する必要があります。

- ARCHIVELOG enabled
- Flashback database on
- FORCE LOGGING enabled
- Use SPFILE
- Use Data Guard Broker
- COMPATIBLE uses 4 decimals and is the same on both databases
- DB_FILES=1024
- オンライン REDO ログの特性
 - Exadata 以外のデータベースではサイズが 1G 以上、Exadata では 4G 以上
 - 通常の冗長ストレージでは多重化のみ。高冗長性を使用する場合は単一メンバー・グループ
 - スレッドあたり最小 3 つのオンライン・ログ・グループ
 - DATA ディスク・グループ上に存在
- スタンバイ REDO ログの特性
 - オンライン REDO ログと同一サイズ
 - Oracle RAC の場合、SRL グループをスレッドに割り当て
 - 単一メンバーのみ
 - スレッドあたり、オンライン REDO ログ・グループと同じ数のグループ
 - DATA ディスク・グループ上に存在
- LOG_BUFFER = 128M for 11.2; 256M for 12.1+
- DB_BLOCK_CHECKING=MEDIUM または FULL 注：この設定はパフォーマンスに影響する可能性があり、アプリケーションを適切にテストしてから有効にする必要があります。
- DB_BLOCK_CHECKSUM=TYPICAL
- STANDBY_FILE_MANAGEMENT=AUTO
- DB_LOST_WRITE_PROTECT=TYPICAL
- DB_FLASHBACK_RETENTION_TARGET=minimum of 120
- FAST_START_MTTR_TARGET=300
- USE_LARGE_PAGES=HugePages が構成されており、オンプレミス・システムで適切なサイズに調整される場合は ONLY
- CLUSTER_INTERCONNECTS (gv\$cluster_interconnects ごとに設定) # Exadata でのみ設定が必要
- PARALLEL_THREADS_PER_CPU=1

- 
- DB_CREATE_ONLINE_LOG_DEST_1= DATA disk group
 - DB_CREATE_ONLINE_LOG_DEST_n other than 1 should only be set when DATA is not high redundancy
 - DB_CREATE_FILE_DEST uses DATA disk group
 - DB_RECOVERY_FILE_DEST uses RECO disk group
 - Recyclebin is on

付録C : RMAN RESTORE FROM SERVICEの出力例

```
[oracle@host1 ~]$ rman target /

Recovery Manager:Release 18.0.0.0.0 - Production on Sat Mar 2 16:48:14 2019
Version 18.3.0.0.0

Copyright (c) 1982, 2018, Oracle and/or its affiliates. All rights reserved.

connected to target database (not started)

RMAN> startup nomount

Oracle instance started

Total System Global Area          32212253688 bytes

Fixed Size                          12454904 bytes
Variable Size                       4160749568 bytes
Database Buffers                    27984396288 bytes
Redo Buffers                         54652928 bytes


RMAN> CONFIGURE DEVICE TYPE DISK PARALLELISM 4;

new RMAN configuration parameters:
CONFIGURE DEVICE TYPE DISK PARALLELISM 4 BACKUP TYPE TO BACKUPSET;
new RMAN configuration parameters are successfully stored

RMAN> restore standby controlfile from service 'victorp';

Starting restore at 02-MAR-19
using target database control file instead of recovery catalog
allocated channel:ORA_DISK_1
channel ORA_DISK_1:SID=10 instance=victor1 device type=DISK

channel ORA_DISK_1: starting datafile backup set restore
channel ORA_DISK_1: using network backup set from service victorp
channel ORA_DISK_1: restoring control file
channel ORA_DISK_1: restore complete, elapsed time:00:00:20
output file name=+RECO/VICTOR_PHX2W5/CONTROLFILE/current.256.1001775751
Finished restore at 02-MAR-19
```



```
RMAN> alter database mount;
```

```
released channel:ORA_DISK_1  
Statement processed
```

```
RMAN> restore database from service 'victorp' section size 2G;
```

```
Starting restore at 02-MAR-19  
Starting implicit crosscheck backup at 02-MAR-19  
allocated channel:ORA_DISK_1  
allocated channel:ORA_DISK_2  
allocated channel:ORA_DISK_3  
allocated channel:ORA_DISK_4  
Crosschecked 1 objects  
Finished implicit crosscheck backup at 02-MAR-19
```

```
Starting implicit crosscheck copy at 02-MAR-19  
using channel ORA_DISK_1  
using channel ORA_DISK_2  
using channel ORA_DISK_3  
using channel ORA_DISK_4  
Finished implicit crosscheck copy at 02-MAR-19
```

```
searching for all files in the recovery area  
cataloging files...  
cataloging done
```

List of Cataloged Files


=====

```
File Name: +RECO/VICTOR_PHX2W5/ARCHIVELOG/2019_03_01/thread_2_seq_1.261.1001776239  
File Name: +RECO/VICTOR_PHX2W5/ARCHIVELOG/2019_03_01/thread_2_seq_2.262.1001776491  
File Name: +RECO/VICTOR_PHX2W5/ARCHIVELOG/2019_03_01/thread_1_seq_1.263.1001776653  
File Name: +RECO/VICTOR_PHX2W5/ARCHIVELOG/2019_03_01/thread_2_seq_3.264.1001776653  
File Name: +RECO/VICTOR_PHX2W5/ARCHIVELOG/2019_03_01/thread_1_seq_2.265.1001776679  
File Name: +RECO/VICTOR_PHX2W5/ARCHIVELOG/2019_03_01/thread_2_seq_4.266.1001776721  
File Name: +RECO/VICTOR_PHX2W5/ARCHIVELOG/2019_03_01/thread_2_seq_5.267.1001776723  
File Name: +RECO/VICTOR_PHX2W5/ARCHIVELOG/2019_03_01/thread_1_seq_3.270.1001791759
```


```
using channel ORA_DISK_1  
using channel ORA_DISK_2  
using channel ORA_DISK_3  
using channel ORA_DISK_4
```



```
channel ORA_DISK_1: starting datafile backup set restore
channel ORA_DISK_1: using network backup set from service victorp
channel ORA_DISK_1: specifying datafile(s) to restore from backup set
channel ORA_DISK_1: restoring datafile 00001 to
+DATA/VICTOR_PHX2W5/DATAFILE/system.263.1001869475
channel ORA_DISK_1: restoring section 1 of 1
channel ORA_DISK_2: starting datafile backup set restore
channel ORA_DISK_2: using network backup set from service victorp
channel ORA_DISK_2: specifying datafile(s) to restore from backup set
channel ORA_DISK_2: restoring datafile 00003 to
+DATA/VICTOR_PHX2W5/DATAFILE/sysaux.274.1001869485
channel ORA_DISK_2: restoring section 1 of 1
channel ORA_DISK_3: starting datafile backup set restore
channel ORA_DISK_3: using network backup set from service victorp
channel ORA_DISK_3: specifying datafile(s) to restore from backup set
channel ORA_DISK_3: restoring datafile 00004 to
+DATA/VICTOR_PHX2W5/DATAFILE/undotbs1.270.1001869497
channel ORA_DISK_3: restoring section 1 of 1
channel ORA_DISK_4: starting datafile backup set restore
channel ORA_DISK_4: using network backup set from service victorp
channel ORA_DISK_4: specifying datafile(s) to restore from backup set
channel ORA_DISK_4: restoring datafile 00005 to
+DATA/VICTOR_PHX2W5/DATAFILE/system.272.1001869507
channel ORA_DISK_4: restoring section 1 of 1
channel ORA_DISK_3: restore complete, elapsed time:00:00:25
channel ORA_DISK_3: starting datafile backup set restore
channel ORA_DISK_3: using network backup set from service victorp
channel ORA_DISK_3: specifying datafile(s) to restore from backup set
channel ORA_DISK_3: restoring datafile 00006 to
+DATA/VICTOR_PHX2W5/DATAFILE/sysaux.271.1001869523
channel ORA_DISK_3: restoring section 1 of 1
channel ORA_DISK_4: restore complete, elapsed time:00:05:02
channel ORA_DISK_4: starting datafile backup set restore
channel ORA_DISK_4: using network backup set from service victorp
channel ORA_DISK_4: specifying datafile(s) to restore from backup set
channel ORA_DISK_4: restoring datafile 00007 to
+DATA/VICTOR_PHX2W5/DATAFILE/users.268.1001869811
channel ORA_DISK_4: restoring section 1 of 1
channel ORA_DISK_4: restore complete, elapsed time:00:00:19
channel ORA_DISK_4: starting datafile backup set restore
channel ORA_DISK_4: using network backup set from service victorp
channel ORA_DISK_4: specifying datafile(s) to restore from backup set
channel ORA_DISK_4: restoring datafile 00008 to
+DATA/VICTOR_PHX2W5/DATAFILE/undotbs1.266.1001869831
channel ORA_DISK_4: restoring section 1 of 1
channel ORA_DISK_4: restore complete, elapsed time:00:01:07
```



```
channel ORA_DISK_4: starting datafile backup set restore
channel ORA_DISK_4: using network backup set from service victorp
channel ORA_DISK_4: specifying datafile(s) to restore from backup set
channel ORA_DISK_4: restoring datafile 00009 to
+DATA/VICTOR_PHX2W5/DATAFILE/undotbs2.264.1001869899
channel ORA_DISK_4: restoring section 1 of 1
channel ORA_DISK_3: restore complete, elapsed time:00:06:41
channel ORA_DISK_3: starting datafile backup set restore
channel ORA_DISK_3: using network backup set from service victorp
channel ORA_DISK_3: specifying datafile(s) to restore from backup set
channel ORA_DISK_3: restoring datafile 00010 to
+DATA/VICTOR_PHX2W5/DATAFILE/system.265.1001869925
channel ORA_DISK_3: restoring section 1 of 1
channel ORA_DISK_4: restore complete, elapsed time:00:00:39
channel ORA_DISK_4: starting datafile backup set restore
channel ORA_DISK_4: using network backup set from service victorp
channel ORA_DISK_4: specifying datafile(s) to restore from backup set
channel ORA_DISK_4: restoring datafile 00011 to
+DATA/VICTOR_PHX2W5/DATAFILE/sysaux.262.1001869939
channel ORA_DISK_4: restoring section 1 of 1
channel ORA_DISK_3: restore complete, elapsed time:00:05:10
channel ORA_DISK_3: starting datafile backup set restore
channel ORA_DISK_3: using network backup set from service victorp
channel ORA_DISK_3: specifying datafile(s) to restore from backup set
channel ORA_DISK_3: restoring datafile 00012 to
+DATA/VICTOR_PHX2W5/DATAFILE/undotbs1.261.1001870237
channel ORA_DISK_3: restoring section 1 of 1
channel ORA_DISK_2: restore complete, elapsed time:00:12:44
channel ORA_DISK_2: starting datafile backup set restore
channel ORA_DISK_2: using network backup set from service victorp
channel ORA_DISK_2: specifying datafile(s) to restore from backup set
channel ORA_DISK_2: restoring datafile 00013 to
+DATA/VICTOR_PHX2W5/DATAFILE/undo_2.260.1001870251
channel ORA_DISK_2: restoring section 1 of 1
channel ORA_DISK_3: restore complete, elapsed time:00:00:27
channel ORA_DISK_3: starting datafile backup set restore
channel ORA_DISK_3: using network backup set from service victorp
channel ORA_DISK_3: specifying datafile(s) to restore from backup set
channel ORA_DISK_3: restoring datafile 00014 to
+DATA/VICTOR_PHX2W5/DATAFILE/users.280.1001870265
channel ORA_DISK_3: restoring section 1 of 1
```



```
channel ORA_DISK_2: restore complete, elapsed time:00:00:23
channel ORA_DISK_3: restore complete, elapsed time:00:00:20
channel ORA_DISK_4: restore complete, elapsed time:00:06:25
channel ORA_DISK_1: restore complete, elapsed time:00:14:13
Finished restore at 02-MAR-19
```



Oracle Corporation, World Headquarters

500 Oracle Parkway
Redwood Shores, CA 94065, USA

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

CONNECT WITH US



Integrated Cloud Applications & Platform Services

Copyright © 2018, 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、その内容に誤りがないことを保証するものではなく、また、口頭による明示的保証や法律による黙示的保証を含め、商品性ないし特定目的適合性に関する黙示的保証および条件などのいかなる保証および条件も提供するものではありません。オラクル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。

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。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。UNIX は、The Open Group の登録商標です。0419

ホワイト・ペーパー・タイトル

2019 年 4 月

著者：Kazuhiro Ikeda、Sebastian Solbach

共著者：Ramachandran Pandrapattahil、Andy Steinorth、Pieter Van Puymbroeck、Lawrence To



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