



Oracle Zero Downtime Migration

Oracle ZDMを使用した物理的移行のステップ・バイ・ステップ・ガイド

2021年8月 | バージョン1.02

Copyright © 2021, Oracle and/or its affiliates

公開

免責事項

本文書には、ソフトウェアや印刷物など、いかなる形式のものも含め、オラクルの独占的な所有物である占有情報が含まれます。この機密文書へのアクセスと使用は、締結および遵守に同意したOracle Software License and Service Agreementの諸条件に従うものとします。本文書と本文書に含まれる情報は、オラクルの事前の書面による同意なしに、公開、複製、再作成、またはオラクルの外部に配布することはできません。本文書は、ライセンス契約の一部ではありません。また、オラクル、オラクルの子会社または関連会社との契約に組み込むことはできません。

本書は情報提供のみを目的としており、記載した製品機能の実装およびアップグレードの計画を支援することのみを意図しています。マテリアルやコード、機能の提供をコミットメント（確約）するものではなく、購買を決定する際の判断材料になさらないでください。本書に記載されている機能の開発、リリース、および時期については、弊社の裁量により決定されます。

製品アーキテクチャの性質上、コードが大幅に不安定化するリスクなしに、本書に記載されているすべての機能を安全に含めることができない場合があります。

目次

免責事項	2
目次	3
はじめに	3
停止時間ゼロの移行	4
アーキテクチャ	4
サポートされるプラットフォームと構成	4
要件	5
Zero Downtime Migrationサービス・ホストの要件	5
ソース・データベースの要件	5
ターゲット・データベースの要件	6
バックアップの要件	7
Oracle NET Services (Oracle SQL*Net) の接続性の要件	7
ポートの要件	7
Oracle ZDMのインストールと移行前の構成	8
Oracle ZDMサービス・ノードの構成	8
Oracle ZDMサービス・ノードとソースおよびターゲット・データベース・サーバー間の接続の構成	13
ソース・データベース・サーバーとターゲット・データベース・サーバー間の接続の構成	14
オプション1 - SCANを使用した接続	14
オプション2 - SSHトンネル	15
ターゲット・データベースの作成	15
透過的データ暗号化ウォレットの設定	18
データベース・バックアップ用のObject Storageの作成	19
認証トークンの生成	21
レスポンス・ファイルの準備	23
レスポンス・ファイルのパラメータ	23
Oracle ZDMを使用したデータベースの移行	27
データベース移行の事前チェック・プロセスの実行	27
データベース移行の実行	29
よくある質問と一般的な問題	31
F.A.Q	31
既知の問題	34
トラブルシューティングとその他のリソース	34

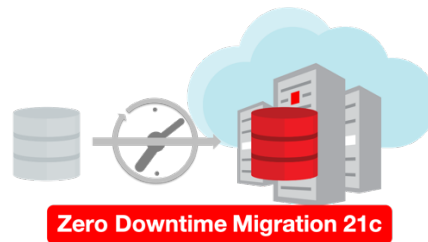


図0 : Oracle Zero Downtime Migration 21cのロゴ

はじめに

オラクルのお客様がOracleのワークロードをOracle Cloud内またはエンジニアド・システム上に移動する動きは益々加速しています。しかし、長年に渡ってワークロードの移行作業は難題となっていました。特に、あるシステムから別のシステムまたはOracle Cloud内へのデータベース・ワークロードの移行は、言うは易し行うは難しです。

オラクルは、Oracleのワークロードの移行における長年に及ぶ経験に基づいて、Oracle Zero Downtime Migration (Oracle ZDM) を開発しました。Oracle ZDMは、簡素化および自動化された移行エクスペリエンスを提供するオラクルが誇るソリューションで、移行シナリオに応じて本番システムの停止時間ゼロからほとんどゼロを実現します。Oracle ZDMにより、オンプレミスのOracle Databaseを、オンプレミスのOracle Exadata Database Machine、Oracle Exadata Cloud at Customer (ExaC@C)、Oracle Cloud Infrastructureなど、オラクルが所有する任意のインフラストラクチャに、およびそれらのインフラストラクチャとの間で、直接かつシームレスに移行できます。Oracle ZDMは広範なOracle Databaseバージョンをサポートし、その名称が意味するように、移行時の本番データベースへの影響を最小限からゼロに抑えています。

Oracle ZDMは、Oracle Maximum Availability Architecture (Oracle MAA) の原則¹に従い、Oracle GoldenGateやOracle Data Guardなどの製品を組み込むことによって、高可用性や、Oracle Recovery Manager、Oracle Data Pump、データベース・リンクなどのテクノロジーを利用するオンライン移行ワークフローを実現しています。

この技術概要は、Zero Downtime Migrationの物理的なワークフローを使用して、オンプレミスのOracle DatabaseをOracle Cloudに移行するためのステップ・バイ・ステップ・ガイドです。ここで使用する移行シナリオは、(オンプレミス環境をエミュレートした) Oracle Cloud InfrastructureのDatabase Cloud Service (DBCS) 仮想マシンで実行中のソース・データベースと、Oracle Cloud InfrastructureのDBCS仮想マシンで実行中のターゲット・データベースで構成されています。

Oracle ZDMは別のノードで実行され、ソースとターゲットの両方に接続して移行を実行します。本書では、Oracle ZDMサービス・ホストのインストール、移行元のソース・データベース、移行プロセスの移行先であるターゲット・データベース、および使用されるバックアップとネットワークに関するすべての要件を取り扱います。移行プロセスをステップ・バイ・ステップで詳しく説明するとともに、製品と全体的な移行プロセスに関するよくある質問についても回答します。

Oracle Zero Downtime Migrationについて詳しくは、Oracle ZDMの製品Webサイト²をご覧ください。

¹ <https://www.oracle.com/jp/database/technologies/high-availability/maa.html>

² <https://www.oracle.com/jp/database/technologies/rac/zdm.html>

停止時間ゼロの移行

アーキテクチャ

Oracle Zero Downtime Migration (Oracle ZDM) は、Oracle DatabaseをOracle Cloudに移行するためのOracle Maximum Availability Architecture (Oracle MAA) 推奨ソリューションです。Oracle ZDM固有の設計により、移行プロセスが可能な限り簡単に保たれ、本番ワークロードにほとんど影響しません。移行するソース・データベースは、オンプレミスの、またはOracle Public Cloud Gen 1やOracle Cloud Infrastructure上にデプロイされているデータベースです。ターゲット・データベースは、Oracle Cloud Infrastructure (OCI) 仮想マシンのOracle Database Cloud Service、Oracle Exadata Cloud Service、Exadata Cloud at Customer、またはOracle Autonomous Databaseにデプロイ可能です。Oracle ZDMでは、移行プロセス全体を自動化することによって人為的エラーの入り込む余地を減らしています。Oracle ZDMでは、Oracle Data GuardやOracle GoldenGateなどのOracle Databaseに統合された高可用性 (HA) テクノロジーを利用し、本番環境の停止時間ゼロを確実に実現するOracle MAAのすべてのベスト・プラクティスに従っています。Oracle ZDMは、物理的移行と論理的移行の両方のワークフローをサポートしています。この技術概要では、バックアップ・ロケーションにOracle Object Storageを使用する物理的移行ワークフローのステップ・バイ・ステップ・ガイドを取り扱います。

Oracle ZDMの物理的移行ワークフローは、必要に応じてスケジュール設定および監視できる8つの簡単なステップで、停止時間ゼロの移行処理を実行します。次のセクションでそれらの各ステップについて詳しく説明します。

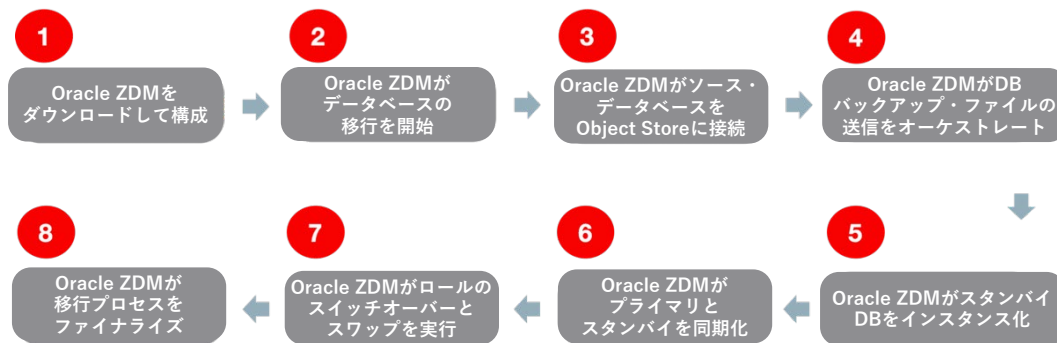


図1: Oracle Zero Downtime Migrationの物理的オンライン移行ステップ

ステップ1: Oracle ZDMをダウンロードして構成する。ステップ2: Oracle ZDMがデータベースの移行を開始する。ステップ3: Oracle ZDMがソース・データベースをOracle Object Storeに接続する。ステップ4: Oracle ZDMがDBバックアップ・ファイルの送信をオーケストレートする。ステップ5: Oracle ZDMがスタンバイ・データベースをインスタンス化する。ステップ6: Oracle ZDMがプライマリとスタンバイを同期化する。ステップ7: Oracle ZDMがロールのスイッチオーバーとスワップを実行する。ステップ8: Oracle ZDMが移行プロセスの最終処理を行う。

サポートされるプラットフォームと構成

Oracle ZDMは次のバージョンのOracle Databaseをサポートしています。11.2.0.4、12.1.0.2、12.2.0.1、18c、19c、21c。ソースとターゲットのデータベースでは、データベースのバージョンが同じになっている必要があります。Oracle ZDMは、Linuxオペレーティング・システムでホストされるOracle Databaseをサポートしています。シングル・インスタンス・データベースであるソース・データベースをシングル・インスタンスまたはRACデータベースに移行することも、RAC One NodeまたはRACデータベースであるソース・データベースをRACデータベースに移行することもできます。

Oracle ZDMは、ソース・データベースとして、Enterprise EditionとStandard EditionのOracle Databaseをサポートしています。Enterprise EditionのOracle DatabaseはOracle Data Guardを利用して移行され、Standard EditionのOracle Databaseはバックアップ/リストア方式を使用してオフラインで移行されます。Oracle ZDMでは、非コンテナ・データベース (非CDB) または1つ以上のプラガブル・データベース (PDB) を含むCDBをソース・データベースとして選択できます。21cより、非CDBからCDBへの移行は、Oracle ZDMの物理的移行と論理的移行の両方のワークフローでサポートされるようになりました。

Oracle ZDMでは、次の環境へのオンプレミス・データベースの移行をサポートしています。

1. Oracle Database Cloud Serviceベアメタル
2. Oracle Database Cloud Service仮想マシン
3. Exadata Cloud Service、Exadata Cloud at Customer、オンプレミスのExadata
4. Autonomous Database (論理的ワークフローのみ)

Oracle ZDMは、次のバックアップ媒体をサポートしています。OCI Object Storage、Oracle Zero Data Loss Recovery Appliance、NFSストレージ。

要件

Zero Downtime Migrationサービス・ホストの要件

Oracle Zero Downtime Migrationバイナリが、以下の要件を満たす別のホストにインストールされている必要があります。

- Oracle 7で実行中のLinuxホスト
- 100 GBの空きストレージ領域
- zdmグループ、およびこのグループのユーザーとしてzdmuser。次のように作成してください
 - `[root@zdm-servicenode ~]# groupadd zdm -g 1001`
 - `[root@zdm-servicenode ~]# useradd zdmuser -g 1001`
- 次のパッケージがインストールされている
 - `glibc-devel`
 - `expect`
 - `unzip`
 - `libaio`
 - `oraclelinux-developer-release-el7`
- 使用するすべてのホスト名とIPアドレスが、`/etc/hosts`にエントリとして存在する
- ソースおよびデータベースにアクセスするための適切な権限（SSH鍵）

ソース・データベースの要件

Oracle ZDMは、Oracle Database 11gリリース2（11.2.0.4）以降をサポートしています。ソースとターゲットの両方が同じリリースのバージョンであることが重要です。

- ソース・データベースがarchive logモードである。
- ソース・データベースが12cリリース2以降で、透過的データ暗号化（Oracle TDE）が有効化されていない場合は以下が必要です。
 - 透過的データ暗号化ウォレットを構成する。ウォレットの構成方法の手引きについては、「透過的データ暗号化ウォレットの設定」³を参照してください。
 - Oracle TDEウォレットの設定時に以下を確認する。
 - `WALLET_TYPE`が`AUTOLOGIN`または`PASSWORD`に設定されている。
 - ウォレットの`STATUS`が`OPEN`である。
 - ソース・データベースがコンテナ・データベースの場合、ウォレットがすべてのプラガブル・データベースで開かれている。すべてのプラガブル・データベースとコンテナ・データベースでマスター鍵が設定されている。
- ソース・データベースはOracle RACデータベースでも構いません。その場合、`SNAPSHOT CONTROLFILE`がすべてのクラスタ・ノードの共有ロケーションを指すように構成する必要があります。

³ <https://docs.oracle.com/en/database/oracle/zero-downtime-migration/>

- ソース・データベース・サーバーとターゲット・データベース・サーバーの各SCANリスナー・ポートで、互いのサーバーからの着信接続要求が許可されている。
- パラメータDB_NAMEがソース・データベースとターゲット・データベースの両方で同じである。
- パラメータDB_UNIQUE_NAMEがソース・データベースとターゲット・データベースで異なる。これにより、Oracle Data Guardはソースとターゲットが異なることを認識できます。
- パラメータSYSPASSWORDがソース・データベースとターゲット・データベースの両方で同じである。
- データベース・バージョンがソース・データベースとターゲット・データベースの両方で一致する。

ターゲット・データベースの要件

- 移行前にターゲット・データベースが作成されている。このプロセスについては、本書の後のセクションで説明します。
- ターゲット・データベースのバージョンがソース・データベースのバージョンと同じである。ターゲット・データベースのパッチ・レベルもソース・データベースと同じである（またはそれよりも高い）。ターゲット・データベースのパッチ・レベルがソース・データベースよりも高い場合は、顧客はデータベースの移行後にデータパッチを実行する必要があります。
- Oracle RACデータベースでは、oracleユーザーのためにOracle RACノード間のSSH接続を設定することが重要です。
- 移行中に不足しないよう、ターゲット・データベースのサイジングが考慮されている。経験則から、ソース・データベースと同じ、またはそれ以上のサイズが推奨されます。
- パラメータDB_NAMEがソース・データベースとターゲット・データベースの両方で同じである。
- パラメータDB_UNIQUE_NAMEがソース・データベースとターゲット・データベースで異なる。これにより、Oracle Data Guardはソースとターゲットが異なることを認識できます。
- ターゲット・データベースで自動バックアップが無効になっている。無効にする方法は、次のようにターゲットのクラウド・プラットフォームによって異なります。
 - OCI DBCS BM/VM：ターゲット・データベースの作成時に、オプション「Enable automatic backups」を選択しない。
 - Exadata Cloud Service：ターゲット・データベースの作成時に、オプション「Enable automatic backups」を選択しない。
 - Exadata Cloud at Customer：データベース・オプションの「Configure Backups」セクションで、「Backup Destination」オプションの「Type」サブオプションに「None」を選択する。
- ターゲット・データベースで透過的データ暗号化が有効化されている。また、ウォレットのステータスがopenであり、ウォレット・タイプがautologinまたはpasswordのいずれかである。

バックアップの要件

- ソース・データベース・サーバーとターゲット・データベース・サーバーの両方が、バックアップ媒体（DBCS BM/VMおよびOracle ExaCSの場合はObject Store、Oracle ExaCCの場合はRecovery ApplianceまたはNFSストレージ）にアクセスできる。
- ソース・データベースのバックアップがバックアップ媒体に問題なく保存されるように、サイジングが考慮されている。

Oracle NET Services（Oracle SQL*Net）の接続性の要件

- すべてのデータベース・ポートとSCANポートが、ソース・データベース・サーバーとターゲット・データベース・サーバーの両方で開かれている。

ポートの要件

- Oracle ZDMサービス・ノード
 - ポート22が開かれている。このポートはSSHで使用され、サービス・ノードとソース・データベース・サーバー間、およびサービス・ノードとターゲット・データベース・サーバー間の接続を可能にします。
- ソース・データベース・サーバー
 - ポート1521が開かれており、ファイアウォールによってブロックされていない。このポートは、ソース・データベース・サーバーとターゲット・データベース・サーバー間のOracle NET Services接続に使用されます。この接続により、Data Guardの適切な同期が可能になります。
 - ポート443が開かれており、ファイアウォールによってブロックされていない。このポートは、Object Storeにアクセスするために使用されます。
- ターゲット・データベース・サーバー
 - ポート1521が開かれており、ファイアウォールによってブロックされていない。このポートは、ソース・データベース・サーバーとターゲット・データベース・サーバー間のOracle NET Services接続に使用されます。この接続により、Data Guardの適切な同期が可能になります。
 - ポート443が開かれており、ファイアウォールによってブロックされていない。このポートは、Object Storeにアクセスするために使用されます。

Oracle ZDMのインストールと移行前の構成

Oracle ZDMサービス・ノードの構成

www.oracle.com/database/technologies/rac/zdm-downloads.htmlにアクセスします。

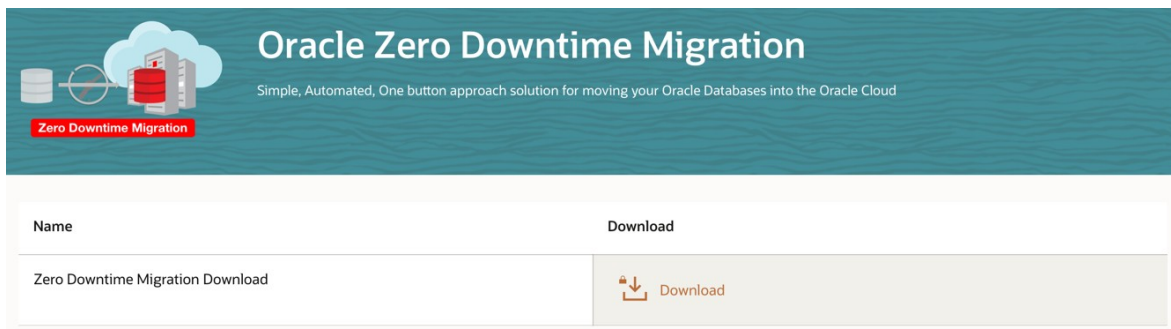


図2：“Oracle Zero Downtime Migration Downloads”のスクリーンショット (www.oracle.com/goto/zdm)

オラクルのライセンス同意書 (<https://www.oracle.com/downloads/licenses/sqldev-license.html> - licenseContent) に同意し、Oracle ZDMサービス・ホスト上のOracle ZDMバイナリをダウンロードします。ダウンロード後、以前作成したzdmuserユーザーに切り替えます。unzipコマンドを使用してバイナリを解凍します。Oracle ZDMサービス・ノードは別のホストで実行されることに留意してください。移行のベスト・プラクティスは以下の図のとおりです。

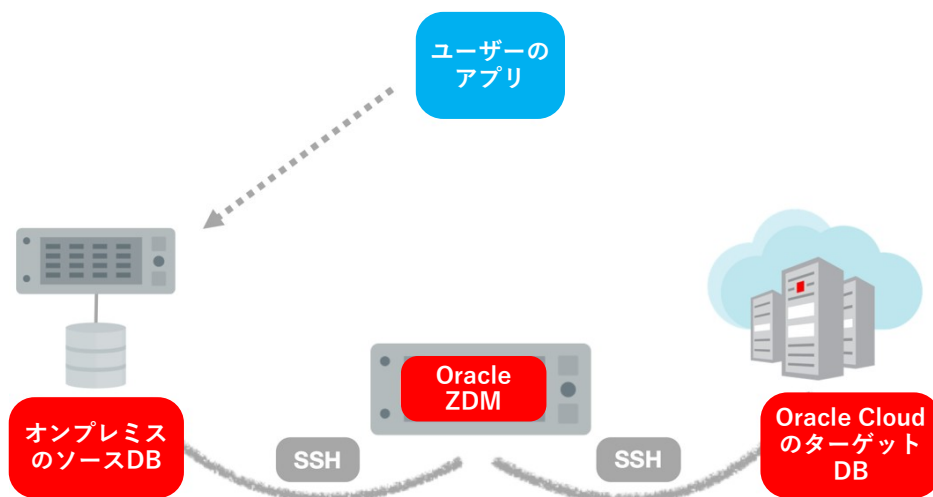


図3：Oracle Zero Downtime Migrationの物理的オンライン移行アーキテクチャ。

Oracle ZDMサービス・ノードは、SSH経由でオンプレミスのソース・データベースとOracle Cloudのターゲット・データベースに接続します。ユーザー・アプリケーションは、オンプレミスのソース・データベースに接続しています。

バイナリの抽出後、以下のディレクトリを作成します。

- oraclehome：Oracle ZDMをインストールするディレクトリ
- oraclebase：Oracle ZDMのすべての構成ファイルとログを保管するディレクトリ

次のように、glibc-develやexpectなど、必要なパッケージをインストールします。

- [root@zdm-servicenode ~]# yum install glibc-devel*
- [root@zdm-servicenode ~]# yum install expect

zdmuserとして、次のようにOracle ZDMのインストール・スクリプトを実行します。

```
./zdminstall.sh setup oraclehome=zdm_oracle_home oraclebase=zdm_base_directory ziploc=zdm_loc-zdm
```

ziplocは、配置されているzdm_home.zipの完全なパスであることに留意してください。このファイルは、www.oracle.com/goto/zdmからダウンロードできるバイナリ(non-compressed bundle)の一部です。

Oracle ZDMとGrid Infrastructureの固有の関係性から、インストールが完了すると、スクリプトをrootユーザーとして実行することを求める一連のメッセージが表示されます。このメッセージは無視してください。Oracle ZDMのインストール・プロセスには、rootとしての実行が求められる操作はありません。さらに、以下に示すように、特定のスクリプトをrootとして実行するようインストーラに求められますが、これも無視してください。

すでにOracle ZDMをインストール済みで、アップグレードを実行する場合は、オラクルの製品ページ (www.oracle.com/goto/zdm) からアクセスできる最新の製品ドキュメントを参照して、Oracle ZDMへのアップグレード方法についてのステップ・バイ・ステップ・ガイドを確認してください。

インストールが正常に終了したら、Oracle Databaseの移行を実施する前に、Oracle ZDMサービスを起動して、インストールが成功したことを確認する必要があります。これを行うには、zdmuserとして以下を実行します。

```
/zdm_oracle_home/bin/zdmservice start
```

```
/zdm_oracle_home/bin/zdmservice status
```

zdmservice startコマンドによってzdmserviceが起動されます。zdmservice statusコマンドによってサービス・ステータスが戻され、running:trueと表示されるはずですが、これにより、zdmserviceが正常にインストールされ、稼働中であることが保証されます。インストール・プロセスが失敗に終わった場合、あるいはエラーが表示された場合は、本書の最後の「よくある質問」セクションを参照してください。

```
[zdmuser@zdm-servicenode media]$ ./zdminstall.sh setup oraclehome=/u01/app/oracle/zdmhome
oraclebase=/u01/app/oracle/zdmbase ziploc=/u01/media/zdm_home.zip -zdm
```

```
-----
Unzipping shiphome to gridhome
```

```
Unzipping shiphome...
```

```
Shiphome unzipped successfully..
```

```
-----
##### Starting GridHome Software Only Installation #####
```

```
-----
Launching Oracle Grid Infrastructure Setup Wizard...
```

ここに警告が表示されますが、無視してください

The response file for this session can be found at:

```
/u01/app/oracle/zdmhome/install/response/grid_.rsp
```

You can find the log of this install session at:

```
/tmp/GridSetupActions/gridSetupActions.log
```

As a root user, execute the following script(s):

1. /u01/app/oracle/zdmhome/inventory/orainstRoot.sh
2. /u01/app/oracle/zdmhome/root.sh

Execute /u01/app/oracle/zdmhome/inventory/orainstRoot.sh on the following nodes:

```
[zdm-servicenode]
```

Execute /u01/app/oracle/zdmhome/root.sh on the following nodes:

```
[zdm-servicenode]
```

Successfully Setup Software with warning(s).

Moved the install session logs to:

```
/u01/app/oracle/zdmhome/inventory/logs/GridSetupActions
```

```
-----
スクリプトをrootユーザーとして実行するという上記のメッセージは無視してください。
```

```
Oracle ZDMのインストールにはrootとしてのアクションは必要ありません。
```

```
-----
making dir /u01/app/oracle/zdmbase/crsdata/zdm-servicenode/rhp/conf
```

```
-----
Generating Preference file
```

```
/u01/app/oracle/zdmbase/crsdata/zdm-servicenode/rhp/conf/rhp.pref
```

```
-----
Generating Root Certificate
```

```
-----
Cluster root certificate generated successfully.
```

```
-----
Generating CA CERTS file
```

```
spawn /u01/app/oracle/zdmhome/bin/crskeytoolctl -copycacerts -filestore
```

```
/u01/app/oracle/zdmbase/crsdata/zdm-servicenode/security/
```

Enter JRE cacerts truststore password:

```
JRE cacerts copied to file [/u01/app/oracle/zdmbase/crsdata/zdm-servicenode/security/cacerts].
```

```
-----
Generating nogi.enabled file
```

```
-----
nogi.enabled file generated successfully
```

```
-----
Generating standalone_config.properties file
```

```
-----
Setting base folder permissions
```

```
-----
Copying service script to bin folder in Oracle Home
```

```
ZDM setup without GI finished successfully...
```

```
[zdmuser@zdm-servicenode bin]$ ./zdm-service start
```

```
No instance detected, starting zdm-service
```

```
[jwcctl debug] Environment ready to start JWC
```

```
[jwcctl debug] Return code of initialization: [0]
```

```
[jwcctl debug] ...BEGIN_DEBUG [Action= start] ...
```

```
Start JWC
```

```
[jwcctl debug] Loading configuration file: /u01/app/oracle/zdmbase/crsdata/zdm-servicenode/rhp/conf/jwc.properties
```

```
[jwcctl debug] oracle.jmx.login.credstore = CRSCRED
```

```
[jwcctl debug] oracle.jmx.login.args = DOMAIN=rhp
```

```
[jwcctl debug] oracle.rmi.url = service:jmx:rmi://{0}:{1,number,#}/jndi/rmi://{0}:{1,number,#}/jmxrmi
```

```
[jwcctl debug] oracle.http.url = http://{0}:{1,number,#}/rhp/gridhome [jwcctl debug] oracle.jwc.tls.clientauth = false
```

```
[jwcctl debug] oracle.jwc.tls.rmi.clientfactory = RELOADABLE
```

```
[jwcctl debug] oracle.jwc.lifecycle.start.log.fileName = JWCStartEvent.log
```

```
[jwcctl debug] Get JWC PIDs
```

```
[jwcctl debug] Done Getting JWC PIDs
```

```
[jwcctl debug] ...JWC containers not found ...
```

```
[jwcctl debug] Start command:-server -Xms128M -Xmx384M -Djava.awt.headless=true -Ddisable.checkForUpdate=true -  
Djava.util.logging.config.file=/u01/app/oracle/zdmbase/crsdata/zdm-servicenode/rhp/conf/logging.properties -  
Djava.util.logging.manager=org.apache.juli.ClassLoaderLogManager -DTRACING.ENABLED=true -DTRACING.LEVEL=2 -  
Doracle.wlm.dbwlmlogger.logging.level=FINEST -Duse_scan_IP=true
```

```
-Djava.rmi.server.hostname=zdm-servicenode -Doracle.http.port=8896 -Doracle.jmx.port=8895 -Doracle.tls.enabled=false -
```

```
Doracle.jwc.tls.http.enabled=false -Doracle.rhp.storagebase=/u01/app/oracle/zdmbase -Djava.security.egd=file:/dev/urandom -
```

```
Doracle.jwc.wallet.path=/u01/app/oracle/zdmbase/crsdata/zdm-servicenode/security -Doracle.jmx
```

```
.login.credstore=WALLET -Dcatalina.home=/u01/app/oracle/zdmhome/tomcat -
```

```
Dcatalina.base=/u01/app/oracle/zdmbase/crsdata/zdm-servicenode/rhp -Djava.io.tmpdir=/u01/app/oracle/zdmbase/crsdata/zdm-  
servicenode/rhp/temp -Doracle.home=/u01/app/oracle/zdmhome -Doracle.jwc.mode=STANDALONE -classpath
```

```
/u01/app/oracle/zdmhome/jlib/cryptoj.jar:/u01/app/oracle/zdmhome/jlib/ora
```

```
clepki.jar:/u01/app/oracle/zdmhome/jlib/osdt_core.jar:/u01/app/oracle/zdmhome/jlib/osdt_cert.jar:/u01/app/oracle/zdmhome/tom
```

```
cat/lib/tomcat-juli.jar:/u01/app/oracle/zdmhome/tomcat/lib/bootstrap.jar:/u01/app/oracle/zdmhome/jlib/jwc-logging.jar
```

```
org.apache.catalina.startup.Bootstrap start
```

```
[jwcctl debug] Get JWC PIDs
```

```
[jwcctl debug] Done Getting JWC PIDs
```

```
[jwcctl debug] ...JWC Container (pid=26626) ...
```

```
[jwcctl debug] ...JWC Container running (pid=26626) ...
```

```

[jwcctl debug]          Check command:-Djava.net.preferIPv6Addresses=true -
Dcatalina.base=/u01/app/oracle/zdmbase/crsdata/zdm-servicenode/rhp -
Doracle.wlm.dbwlmlogger.logging.level=FINEST-
Doracle.jwc.client.logger.file.name=/u01/app/oracle/zdmbase/crsdata/zdm-
servicenode/rhp/logs/jwc_checker_stdout_err_%g.log -Doracle.jwc.client.logger.file.number=10 -
Doracle.jwc.client.logger.file.size=1048576 -
Doracle.jwc.wallet.path=/u01/app/oracle/zdmbase/crsdata/zdm-servicenode/security -
Doracle.jmx.login.credstore=WALLET -Doracle.tls.enabled=false -
Doracle.jwc.tls.http.enabled=false -classpath /u01/app/oracle/zdmhome/jlib/jwc-logging.jar:/u01/app/oracle/zdmhome/jlib/jwc-
security.jar:/u01/app/oracle/zdmhome/jlib/jwc-
client.jar:/u01/app/oracle/zdmhome/jlib/jwc-
cred.jar:/u01/app/oracle/zdmhome/jlib/srv.jar:/u01/app/oracle/zdmhome/jlib/srvmhas.jar:/u01/app/orac
le/zdmhome/jlib/crypto.jar:/u01/app/oracle/zdmhome/jli
b/oraclepki.jar:/u01/app/oracle/zdmhome/jlib/osdt_core.jar:/u01/app/oracle/zdmhome/jlib/osdt_cert.jar:/u01/app/oracle/zdmhom
e/tomcat/lib/tomcat-juli.jar oracle.cluster.jwc.tomcat.client.JWCChecker localhost 8896 -1
[jwcctl debug] ...JWC Container is ready ...
[jwcctl debug] ...START - Return code = 0 ...
[jwcctl debug] ...END_DEBUG [Action=start] ...
[jwcctl debug] Return code of AGENT: [0]

```

Return code is 0

Server started successfully.

```
[zdmuser@zdm-servicenode bin]$
```

イメージに対する問合せを試行することで、Oracle ZDMサービスが正常に設定されたことを確認できます。これを行うには、zdmuserとして以下を実行します。

```
./zdm_oracle_home/bin/zdmcli query image
```

```

[zdmuser@tool bin]$ ./zdmcli query image
x.x.x.x.host.com:Audit ID:2
No image has been configured

```

Oracle ZDMサービス・ノードとソースおよびターゲット・データベース・サーバー間の接続の構成

Oracle ZDMサービス・ノードは、迅速な移行プロセスを確保するために、ソース・データベース・サーバーとターゲット・データベース・サーバーにセキュアに接続する必要があります。それには、作成したばかりのzdmuserが、SSH認証鍵ペアを（パスフレーズなしで）使用できなければなりません。鍵がない場合、あるいは鍵にパスフレーズが必要な場合は、オラクルのドキュメントの「パスフレーズなしの秘密SSH鍵の生成」セクションに従って鍵を生成できます。

<https://docs.oracle.com/en/database/oracle/zero-downtime-migration/>

秘密SSH鍵ファイルを作成したら、id_rsaファイルとid_rsa.pubファイルが利用できるようになります。ベスト・プラクティスとして、id_rsa.pubファイルの名前を<zdm_service_node_name>.ppkに変更することが推奨されます。最後に、公開鍵をソース・データベース・サーバーとターゲット・データベース・サーバーの各ファイル

（opc_user_home/.ssh/authorized_keys）に追加する必要があります。

次のように、Oracle ZDMサービス・ノードの/etc/hostsファイルを編集し、ソースおよびターゲット・データベース・サーバーのホスト名とIP情報を追加します。

```
#Source
111.1.1.1 source.node-1234 source

#OCI Target
129.1.2.3 target.node-1234-oci target
```

ソースとターゲットのどちらの別名も、ドメイン情報を含まない名前で行わなければならないことに留意してください。最後に、ssh -iコマンドを使用して、Oracle ZDMサービス・ノードからソース・データベース・サーバーおよびターゲット・データベース・サーバーへの接続をテストします。

```
[zdmuser@zdm-servicenode ~]# ssh -i zdm_service_tool.ppk opc@source
[zdmuser@zdm-servicenode ~]# ssh -i zdm_service_tool.ppk opc@target
```

```
[zdmuser@tool .ssh]$ ssh -i zdm_service_tool.ppk opc@source
The authenticity of host 'source (X.X.X.X)' can't be established.
RSA key fingerprint is .....
RSA key fingerprint is .....
Are you sure you want to continue connecting (yes/no)? yes
警告： Permanently added 'source,X.X.X.X' (RSA) to the list of known hosts.
-- WARNING -- This system is for the use of authorized users only. Individuals using this computer system without authority or in excess of their authority are subject to having all their activities on this system monitored and recorded by system personnel. Anyone using this system expressly consents to such monitoring and is advised that if such monitoring reveals possible evidence of criminal activity system personnel may provide the evidence of such monitoring to law enforcement officials.
```

```
Last login: from inet-x-x-x.x.com
*** ALERTS::INFO BEGIN ***
ALERTS::System Alert: [Database User accounts check Failed]
ALERTS::Please check /var/opt/oracle/log/monitor_alerts/monitor_alerts.log
*** ALERTS::INFO END ***
[opc@source ~]$ exit
```

```
[zdmuser@tool .ssh]$ ssh -i zdm_service_tool.ppk opc@target
The authenticity of host 'target (x.x.x.x)' can't be established.
ECDSA key fingerprint is .....
ECDSA key fingerprint is .....
Are you sure you want to continue connecting (yes/no)? yes
警告： Permanently added 'target,x.x.x.x' (ECDSA) to the list of known hosts.
Last login: from x.x.x.x
[opc@target ~]$ exit
```

ソース・データベース・サーバーとターゲット・データベース・サーバー間の接続の構成

ソース・データベース・サーバーとターゲット・データベース・サーバーは、Data Guardの操作を保証するために、Oracle NET Services (SQL*Net) 経由で接続できなければなりません。ソースとターゲット間の接続には次の2つのオプションがあります。

- 1つ目のオプションでは、SCANを使用して、ソース・データベース・サーバーとターゲット・データベース・サーバーの両方から解決できる必要があるソース/ターゲットSCANポート経由の接続を確保します。
- 2つ目のオプションでは、ソース・データベース・サーバーからターゲット・データベース・サーバーへのSSHトンネルの設定が必要な接続を確保します。

オプション1 - SCANを使用した接続

ソース・データベース・サーバーとターゲット・データベース・サーバー間でSCAN経由の接続を保証するには、双方の/etc/hostsファイルを編集する必要があります。

ソース・データベース・サーバーの/etc/hostsファイルに追加するエントリは以下のとおりです。

```
#Source
111.1.1.1 source.node-1234-oci source
111.1.1.1 source-priv.node-1234-oci source-priv
111.1.1.1 source-vip.node-1234-oci source-vip
111.1.1.1 source-scan.node-1234-oci source-scan
#Target
129.1.2.3 target.node-1234-oci target
129.1.2.3 target-scan.node-1234-oci target-scan
```

ターゲット・データベース・サーバーの/etc/hostsファイルに追加するエントリは以下のとおりです。

```
#Target
129.1.2.3 target.node-1234-oci target
129.1.2.3 target-priv.node-1234-oci target-priv
129.1.2.3 target-vip.node-1234-oci target-vip
129.1.2.3 target-scan.node-1234-oci target-scan
#Source
111.1.1.1 source.node-1234 source
111.1.1.1 source-scan.node-1234 source-scan
```

ソース・データベースがGrid Infrastructureを基盤としていない場合、/etc/hostsターゲット・ファイルのsource scanエントリは必要ありません。

これらのエントリの一部は、すでに/etc/hostsファイルにデフォルトで存在する場合もあることに留意してください。これは、Oracle ZDMを適切に機能させるためのファイルのサンプルにすぎません。

その後、tnspingユーティリティを使用して、ポート1521経由でソース・データベース・サーバーからターゲット・データベース・サーバーへの接続を試み、さらにその逆方向の接続を試みることで、接続をテストできます。

```
[oracle@source ~]$ tnsping target:1521
```

```
[oracle@target ~]$ tnsping source:1521
```

オプション2 - SSHトンネル

ソース・データベース・サーバーとターゲット・データベース・サーバー間のSSHトンネルを設定するには、Oracle ZDMの製品ドキュメントの「ソース・データベース・サーバーとターゲット・データベース・サーバー間の接続の構成」セクションにある「オプション2」サブセクションで概説されている手順に従ってください。

<https://docs.oracle.com/en/database/oracle/zero-downtime-migration/>

ターゲット・データベースの作成

プレースホルダ・ターゲット・データベースを作成するには、Oracle Cloudアカウントにログインし、「Bare Metal, VM and Exadata」メニューをクリックして「DB systems」タブにアクセスします。



図4 : Oracle Cloudの「Bare Metal, VM and Exadata」メニューのスクリーンショット

「Create DB System」オプションをクリックします。

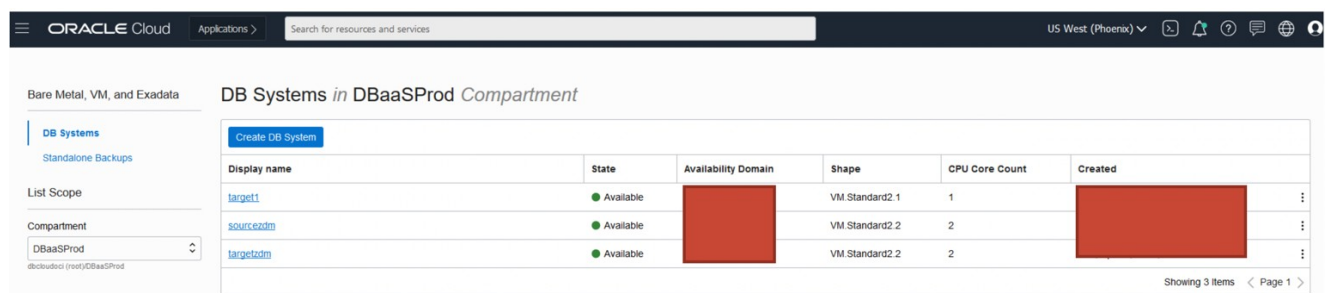
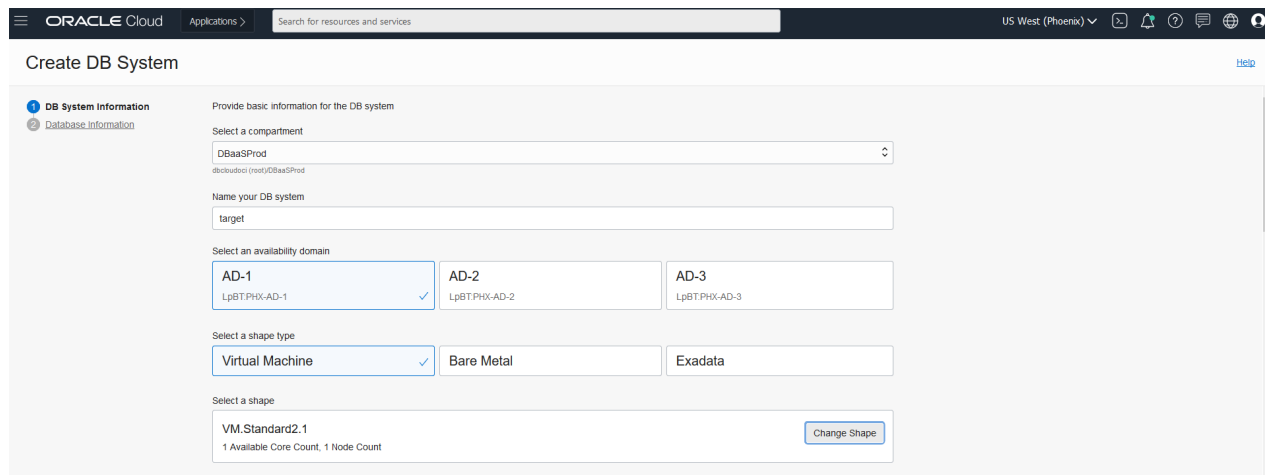


図5 : Oracle Cloudの「Create DB System」のスクリーンショット

コンパートメント、DBシステム名、可用性ドメイン、シェイプに適切な値を選択します。

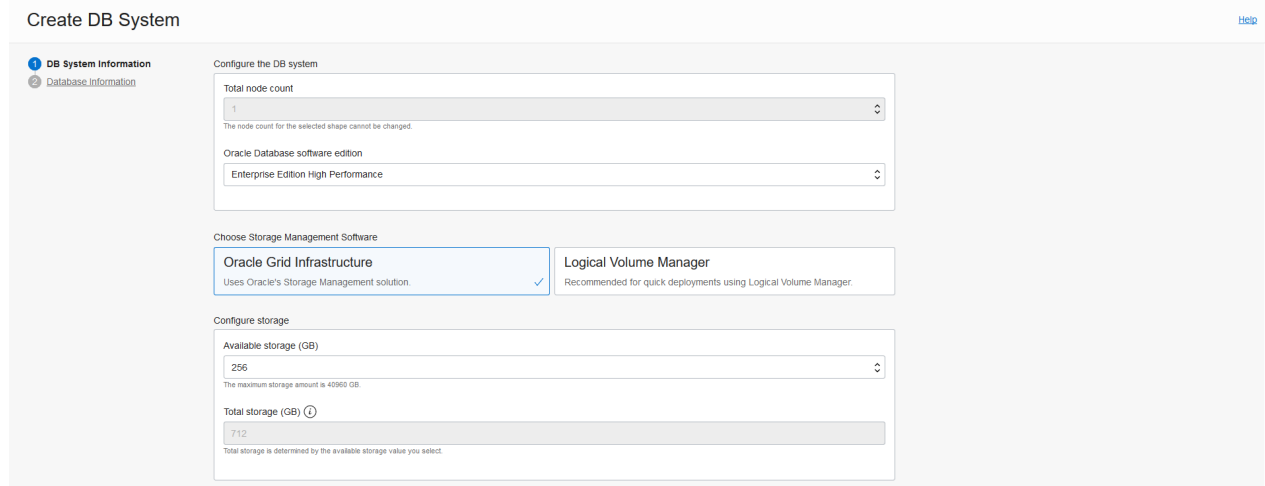


The screenshot shows the 'Create DB System' interface in Oracle Cloud. The 'DB System Information' step is active, and the user has configured the following options:

- Compartment: DBaaSProd
- DB System Name: target
- Availability Domain: AD-1 (LpBT:PHX-AD-1)
- Shape Type: Virtual Machine
- Shape: VM.Standard2.1 (1 Available Core Count, 1 Node Count)

図6：Oracle Cloudの「Create DB System」ウィンドウでDB System Informationのオプションを設定している画面のスクリーンショット

「Total node count」と「Oracle Database software edition」を選択してDBシステムを構成します。Storage Management Softwareを選択します。Oracle ZDMは、Oracle Grid Infrastructureを使用するデータベースへの移行のみをサポートしていることに留意してください。Logical Volume Managerはサポートされていません。



The screenshot shows the 'Create DB System' interface in Oracle Cloud, specifically the 'Database Information' step. The user has configured the following options:

- Total node count: 1
- Oracle Database software edition: Enterprise Edition High Performance
- Storage Management Software: Oracle Grid Infrastructure
- Available storage (GB): 256
- Total storage (GB): 712

図7：Oracle Cloudの「Create DB System」ウィンドウでDBシステム構成、ストレージ管理ソフトウェア、ストレージ構成の各オプションを設定している画面のスクリーンショット

「Upload SSH key files」を選択し、ライセンス・タイプを選択します。

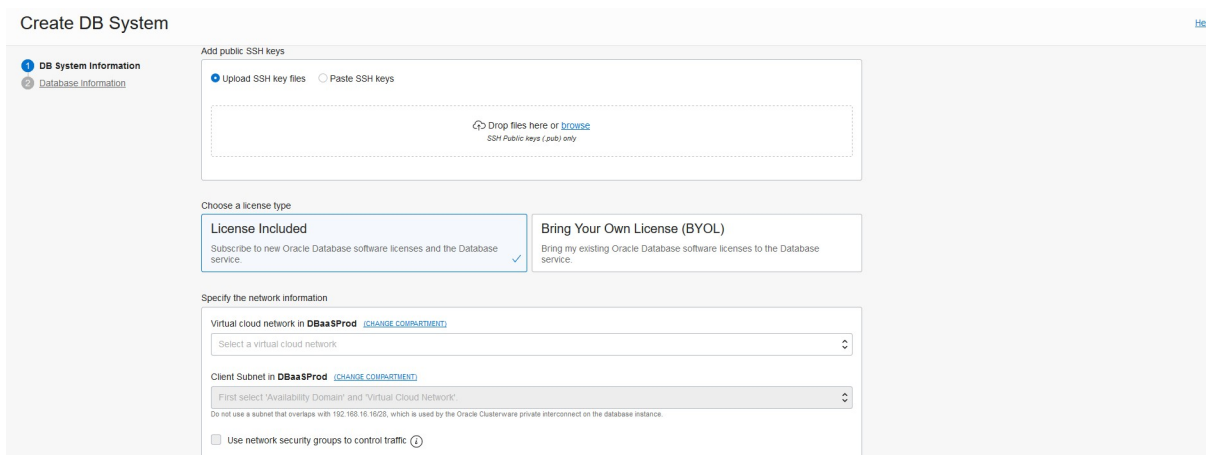


Figure 8 : Oracle Cloudの“Create DB System”ウィンドウでSSH鍵のオプションを設定している画面のスクリーンショット

ご自身の設定に応じて、ネットワーク情報を指定します。これには、Hostname prefix、Host domain name、Host and domain URLが含まれます。Host and domain URLの値は、Hostname prefixとHost domain nameによって決定され、自動生成されます。

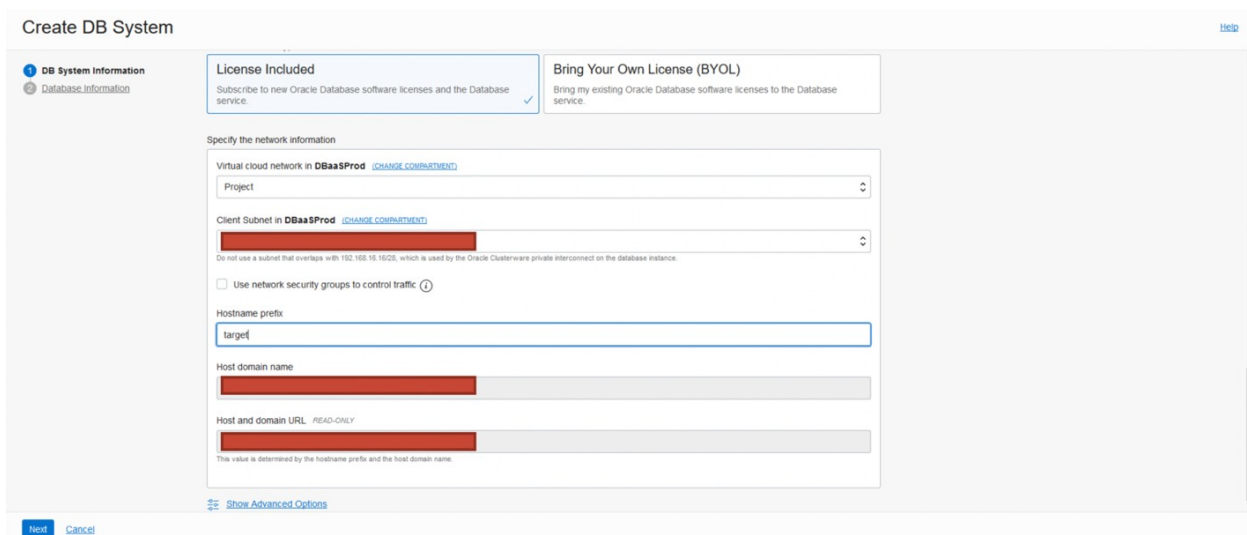


図9 : Oracle Cloudの“Create DB System”ウィンドウでネットワーク構成情報を設定している画面のスクリーンショット

ターゲット・データベースに関する情報を入力します。これには、Database name、Database version、および administrator credentialsが含まれます。Database nameの値は、ソース・データベース名と同じでなければならないことに留意してください。

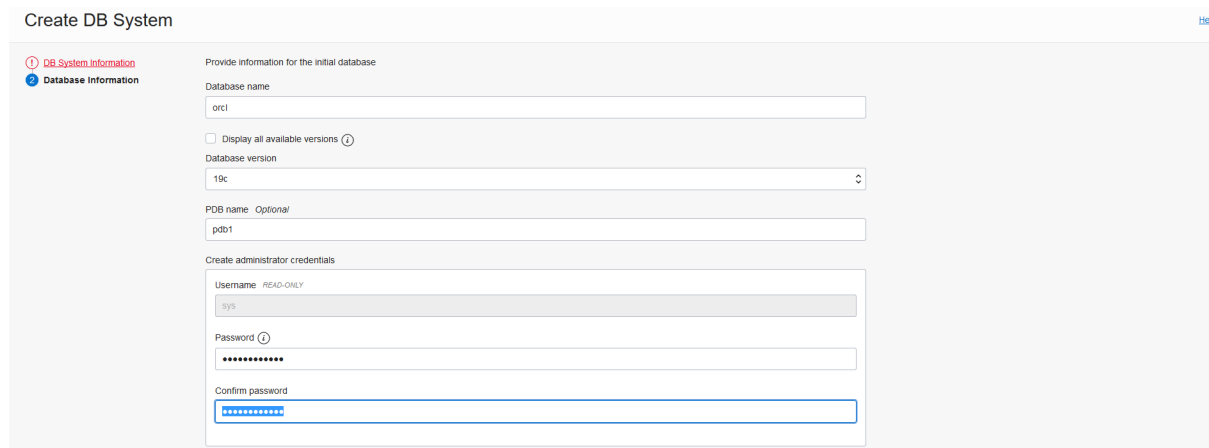


図10 - Oracle Cloudの“Create DB System”ウィンドウでデータベース構成情報を設定している画面のスクリーンショット

最後に、データベース・ワークロード・タイプに「Transaction Processing」を選択し、必要に応じて「Character set」を希望のキャラクタ・セットに変更します。準備が整ったら、「Create DB System」をクリックします。

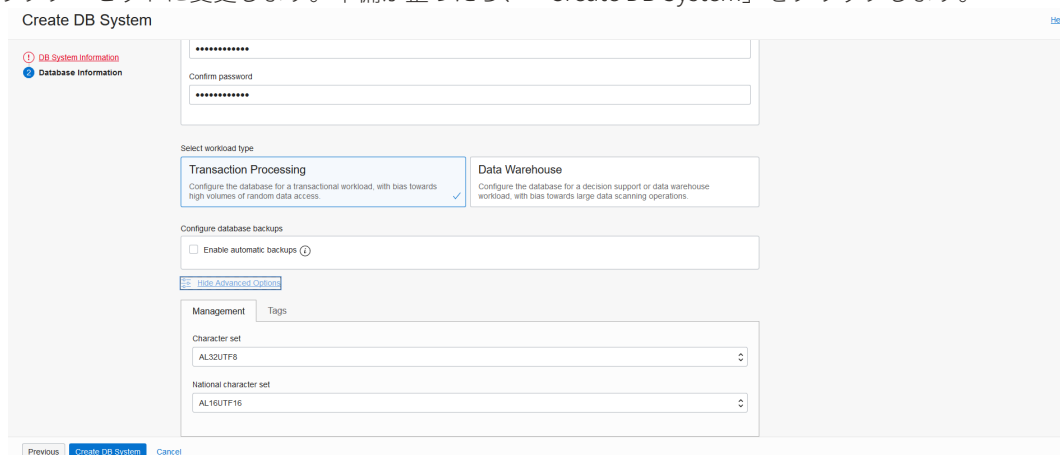


図11 : Oracle Cloudの“Create DB System”ウィンドウでワークロード・タイプ情報を設定している画面のスクリーンショット

透過的データ暗号化ウォレットの設定

Oracle Cloudでは、すべてのデータベースがデフォルトで暗号化されます。オラクルは、透過的データ暗号化テクノロジーを使用してこれを実現しています。つまり、顧客がデータベースをOracle Cloudに移行する場合、ソース・データベースが暗号化されていなくても、ターゲット・データベースはOracle TDEを使用して暗号化されます。

Oracle Zero Downtime Migrationは、ソース・データベースとして非暗号化データベースをサポートしており、移行プロセス中にターゲット・データベースの暗号化を行っています。すべてのOracleソース・データベースがデフォルトで暗号化されるわけではないため、特にソース・データベースが12cリリース2以降の場合は、実行しなければならない複数の手順が存在します。その場合、移行を実施する前に、透過的データ暗号化ウォレットを構成する必要があります。

このタスクを実行する方法についてのステップ・バイ・ステップ・ガイドについては、Oracle Zero Downtime Migrationドキュメントの「[透過的データ暗号化ウォレットの設定](#)」セクションを参照してください。

<https://docs.oracle.com/en/database/oracle/zero-downtime-migration/>

データベース・バックアップ用のObject Storageの作成

Oracle Zero Downtime Migrationでは、Oracle Recovery Manager（Oracle RMAN）が生成したソース・データベースのバックアップを配置するためにObject Storageを利用します。このバックアップは、https接続を介してソース・データベースからObject Storageに送信されます。Oracle ZDMはこの処理を完了すると、このバックアップ・ファイルを利用するスタンバイ・データベースをインスタンス化します。次の手順に従って、データベース・バックアップ用のObject Storageを作成してください。

Oracle Cloudアカウントにログインし、「Object Storage」メニュー、「Object Storage」オプションを順にクリックします。

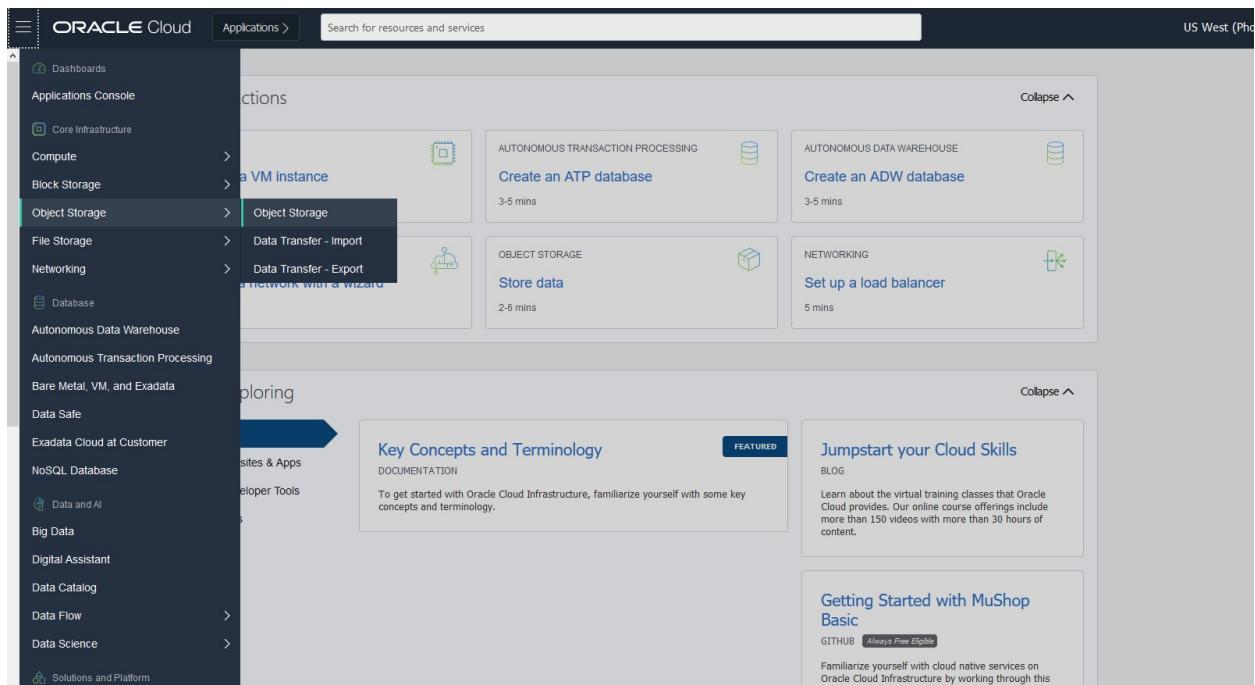


図12 : Oracle Cloudの“Object Storage”ウィンドウのスクリーンショット

コンパートメントを選択し、「Create Bucket」オプションをクリックします。コンパートメントは、ターゲット・プレースホルダ・データベースを作成したローカル・リージョンと同じリージョンになければならないことに留意することが重要です。

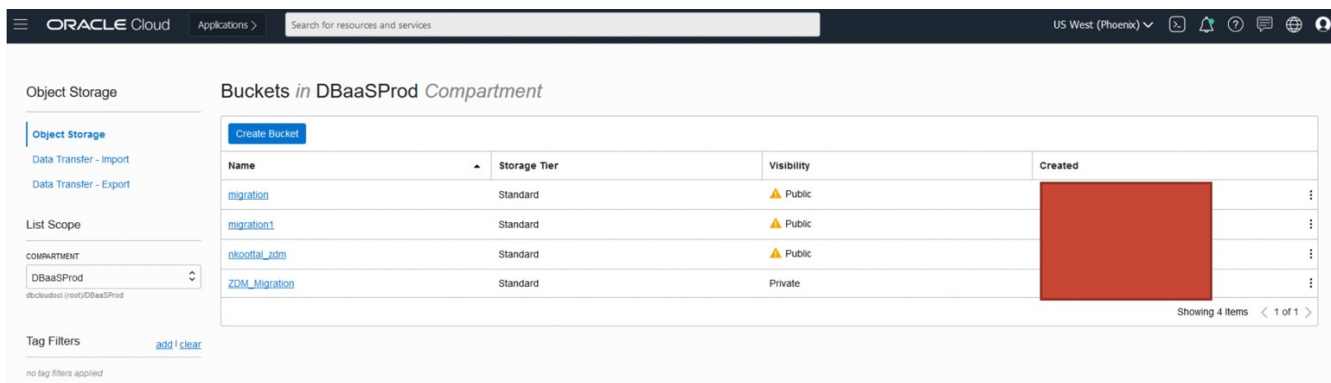


図13 : Oracle Cloudの“Create Bucket”ウィンドウのスクリーンショット

「Bucket Name」に希望の名前を入力し、「Storage Tier」に「Standard」を選択し、「Encrypt using Oracle Managed Keys」を選択します。最後に、「Create Bucket」オプションをクリックします。

The screenshot shows the 'Create Bucket' configuration window in Oracle Cloud. At the top right, there are links for 'Help' and 'Cancel'. The 'BUCKET NAME' field contains 'ZDM_Migration'. Under 'STORAGE TIER', 'STANDARD' is selected with a radio button. Below that, 'OBJECT EVENTS' and 'OBJECT VERSIONING' are both unchecked. In the 'ENCRYPTION' section, 'ENCRYPT USING ORACLE MANAGED KEYS' is selected. A section for 'TAGGING' explains that tags are metadata for organizing resources and includes a link to 'Learn more about tagging'. Below this, there are three input fields: 'TAG NAMESPACE' (set to 'None (add a free-form tag)'), 'TAG KEY', and 'VALUE'. A '+ Additional Tag' button is located to the right of these fields. At the bottom, there are 'Create Bucket' and 'Cancel' buttons.

Figure 14 – Oracle Cloudの“Create Bucket”ウィンドウでバケット構成情報を設定している画面のスクリーンショット

認証トークンの生成

Oracle Cloudアカウントにログインし、「Identity」→「Users」オプションをクリックします。

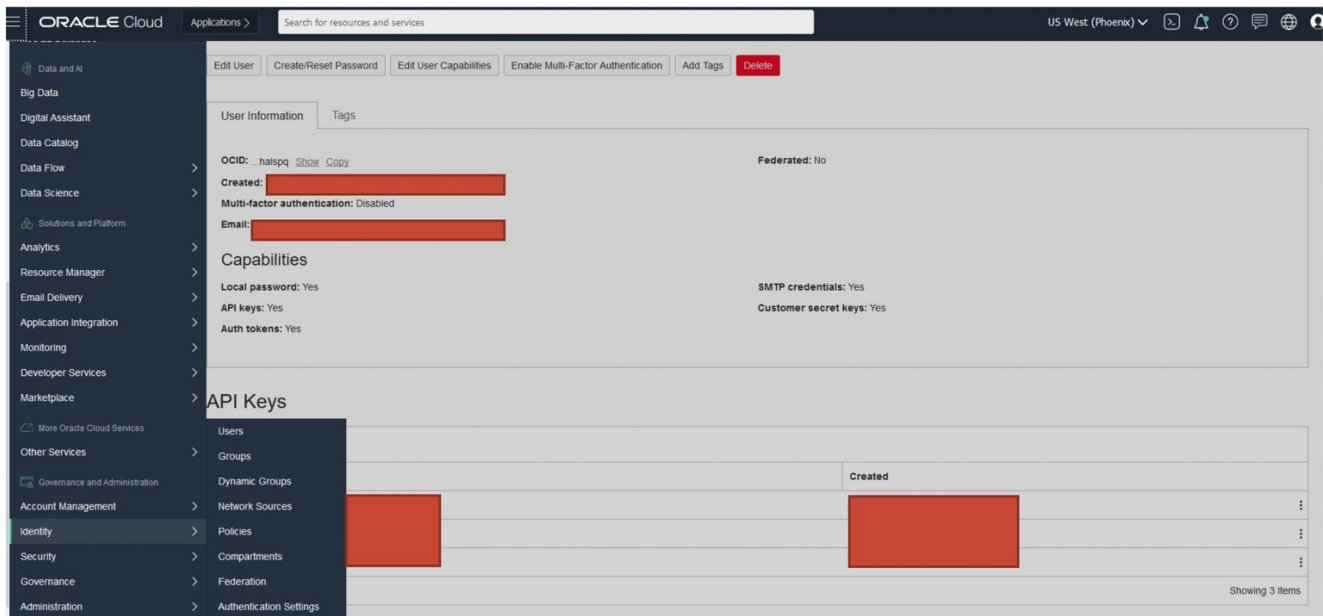


図15 : Oracle Cloudの「Identity>Users」ウィンドウのスクリーンショット

Object Storageの認証に使用するユーザーを選択し、「Auth Tokens」オプションをクリックし、「Generate Token」オプションをクリックします。

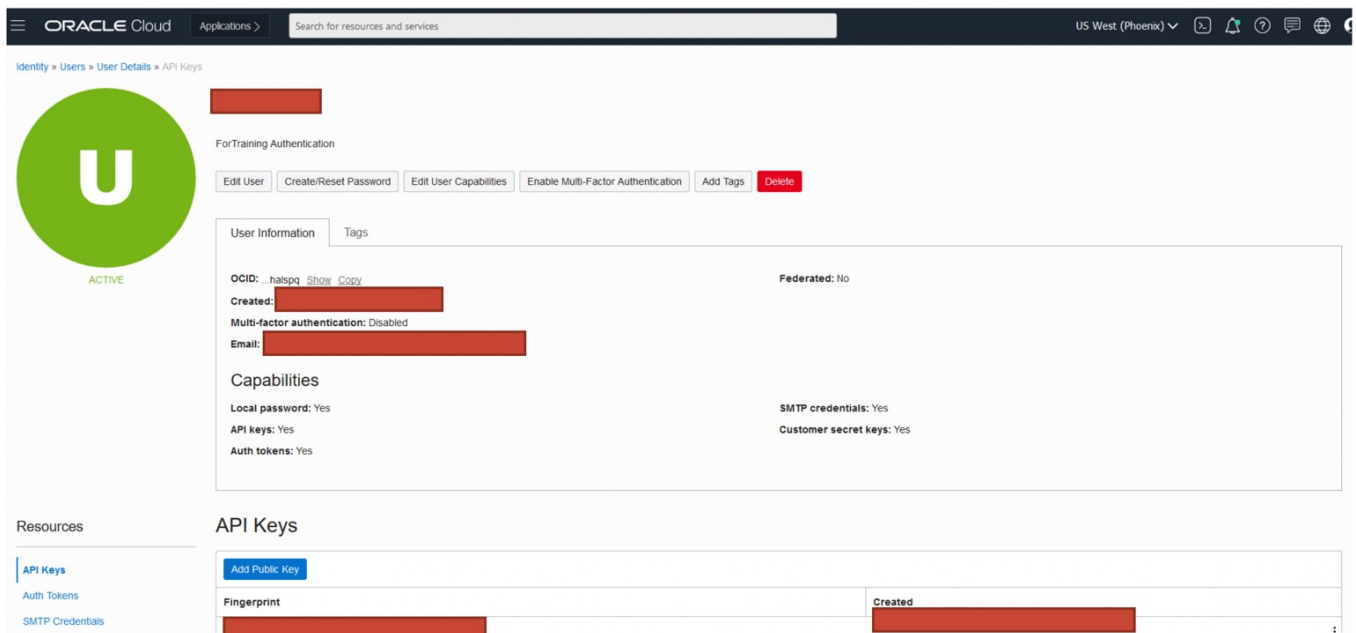


図16 : Oracle Cloudの「Auth Token」ウィンドウのスクリーンショット

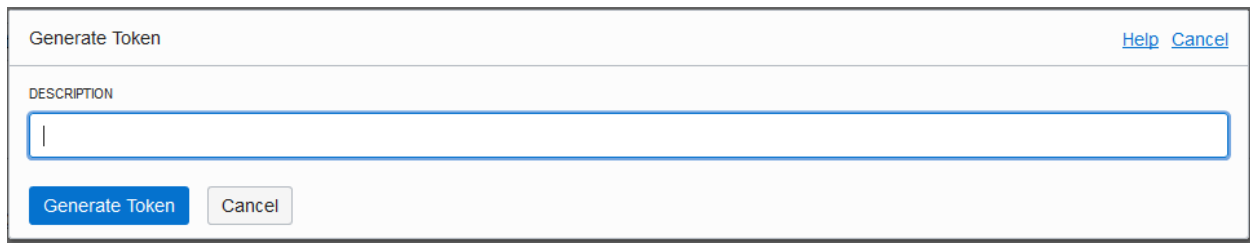


図17 : Oracle Cloudの“Auth Token”ウィンドウのスクリーンショット

トークンのコピーを促されたら、コピーする必要があることに留意してください。トークンを後でコピーすることはできません。ここでコピーしなかった場合は、新しいトークンを後で作成する必要があります。

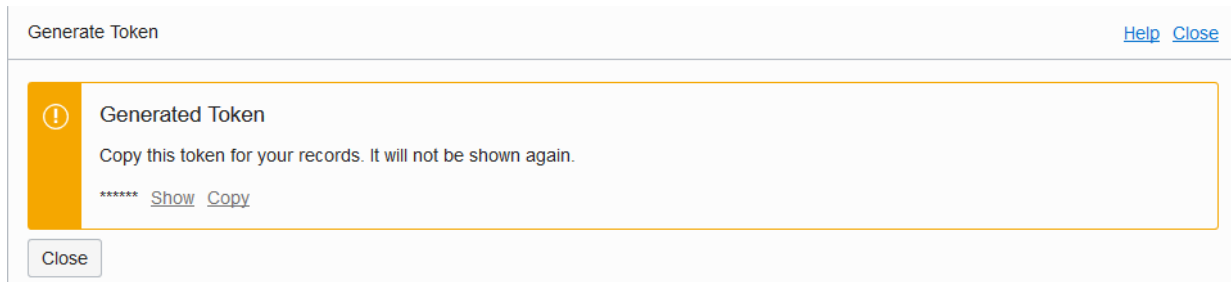


図18 : Oracle Cloudの“Generate Token”ウィンドウのスクリーンショット

レスポンス・ファイルの準備

レスポンス・ファイルのパラメータ

パラメータ	説明
TGT_DB_UNIQUE_NAME	Oracle ZDMサービス・ノードがターゲット・データベースのdb_unique_nameの値を認識できるようになります。
MIGRATION_METHOD	<ul style="list-style-type: none">ONLINE_PHYSICAL OracleZDMは移行プロセスにData Guardを使用します。OFFLINE_PHYSICALOracle ZDMは移行にバックアップおよびリストアの方法を使用します。
PLATFORM_TYPE	移行プロセスのターゲット・プラットフォームを指定します。 <ul style="list-style-type: none">VMDB・このオプションは、仮想マシンまたはベアメタルで有効です。EXACC・このオプションは、Exadata Cloud at Customerで有効です。EXACS・このオプションは、Exadata Cloud Serviceで有効です。NON_CLOUD・このオプションはオンプレミスのExadataで有効です。
SRC_HTTP_PROXY_URL	必要に応じて、ソース・データベース・サーバーのSSHプロキシURLを指定します。
SRC_HTTP_PROXY_PORT	必要に応じて、ソース・データベース・サーバーのSSHプロキシ・ポートを指定します。
SRC_CONFIG_LOCATION	ソース・データベース・サーバーのSSH構成ファイルのロケーションがデフォルト・ロケーションの<user_home>/.ssh/configでない場合に、そのロケーションを指定します。
SRC_TIMEZONE	ソース・データベースのタイムゾーン値。Grid Infrastructureが構成されていないシングル・インスタンスのソース・データベースでのみ必要です。
SRC_OSS_PROXY_HOST	Object Storeへの接続に必要な場合に、ソース・データベース・サーバーのプロキシURLを指定します。
SRC_OSS_PROXY_PORT	Object Storeへの接続に必要な場合に、ソース・データベース・サーバーのプロキシ・ポートを指定します。
TGT_HTTP_PROXY_URL	必要に応じて、ターゲット・データベース・サーバーのSSHプロキシURLを指定します。
TGT_HTTP_PROXY_PORT	必要に応じて、ターゲット・データベース・サーバーのSSHプロキシ・ポートを指定します。

TGT_CONFIG_LOCATION	ターゲット・データベース・サーバーのSSH構成ファイルのロケーションがデフォルト・ロケーションの<user_home>/.ssh/configでない場合に、そのロケーションを指定します。
TGT_SSH_TUNNEL_PORT	このパラメータは、トンネリング構成がある場合に、Oracle NET Services (SQL*Net) の接続設定におけるソースとターゲット間のSSHトンネルのポートを指定します。
TGT_OSS_PROXY_HOST	Object Storeへの接続に必要な場合に、ターゲット・データベース・サーバーのプロキシURLを指定します。
TGT_OSS_PROXY_PORT	Object Storeへの接続に必要な場合に、ターゲット・データベース・サーバーのプロキシ・ポートを指定します。
TGT_DATADG、TGT_REDODG、TGT_RECODG	ターゲット・データベースがデータファイルにOracle ASMを使用している場合、この値でデータベース・データファイルのプロパティを指定します。
TGT_DATAACFS、 TGT_REDOACFS、TGT_RECOACFS	ターゲット・データベースがデータファイルにOracle ACFSを使用している場合、この値でデータベース・データファイルのプロパティを指定します。
HOST	Object Storageネームスペースを指定します。この値は、cloud consoleメニュー→administration→tenancy detailで確認できます。tenancy informationセクションに、Object Storageネームスペースの有効なURLが表示されます。
OPC_CONTAINER	Object Storageバケット名
SRC_ZDLRA_WALLET_LOC	Zero Data Loss Recovery Applianceがバックアップ媒体として使用されている場合のソース・データベース・サーバーのウォレット・ロケーション。
TGT_ZDLRA_WALLET_LOC	Zero Data Loss Recovery Applianceがバックアップ媒体として使用されている場合のターゲット・データベース・サーバーのウォレット・ロケーション。
ZDLRA_CRED_ALIAS	Zero Data Loss Recovery Applianceがバックアップ媒体として使用されている場合のウォレット資格証明エイリアス。
BACKUP_PATH	Oracle ZDMはバックアップ・ロケーションがどこにあるかを知ることができます。 <ul style="list-style-type: none"> • STORAGEPATH • EXTBACKUP
SKIP_FALLBACK	ロールのスワップ後に、REDOログがターゲット・データベースからソース・データベースに送信されないかどうかを指定します。デフォルト値はFALSEです。
TGT_RETAIN_DB_UNIQUE_NAME	REDOログをOracle Cloudからオンプレミスのスタンバイに送信するかどうかを指定し、一定期間環境を監視し、後でフォールバックを取り消します

SHUTDOWN_SRC	移行後にソース・データベースをシャットダウンするかどうかを指定します。この値がTRUEに設定されている場合はシャットダウンを実行し、FALSEに設定されている場合は実行しません。
SRC_RMAN_CHANNELS	Oracle RMANバックアップの実行に使用されるソース・データベース・サーバーのRecovery Managerチャンネル数を指定します。デフォルト値は10です。
TGT_RMAN_CHANNELS	Oracle RMANバックアップの実行に使用されるターゲット・データベース・サーバーのRecovery Managerチャンネル数を指定します。デフォルト値は10です。
ZDM_BACKUP_FULL_SRC_MONITORING_INTERVAL	ZDM_BACKUP_FULL_SRC移行ジョブ・フェーズの進捗状況を監視して報告する時間隔（分）を指定します。
ZDM_BACKUP_INCREMENTAL_SRC_MONITORING_INTERVAL	ZDM_BACKUP_INCREMENTAL_SRC移行ジョブ・フェーズの進捗状況を監視して報告する時間隔（分）を指定します。
ZDM_BACKUP_DIFFERENTIAL_SRC_MONITORING_INTERVAL	ZDM_BACKUP_DIFFERENTIAL_SRC移行ジョブ・フェーズの進捗状況を監視して報告する時間隔（分）を指定します。
ZDM_CLONE_TGT_MONITORING_INTERVAL	ZDM_CLONE_TGT移行ジョブ・フェーズの進捗状況を監視して報告する時間隔（分）を指定します。
ZDM_OSS_RECOVER_TGT_MONITORING_INTERVAL	ZDM_OSS_RECOVER_TGT移行ジョブ・フェーズの進捗状況を監視して報告する時間隔（分）を指定します。
ZDM_OSS_RESTORE_TGT_MONITORING_INTERVAL	ZDM_OSS_RESTORE_TGT移行ジョブ・フェーズの進捗状況を監視して報告する時間隔（分）を指定します。
SKIP_SRC_SERVICE_RETENTION	ソース・データベース・サービスの保存をユーザーが省略できるようにします。デフォルト値はFALSEです。
ZDM_BACKUP_RETENTION_WINDOW	Zero Downtime Migrationによって作成されたデータベース・バックアップが廃止されるまでの日数をユーザーが指定できるようにします。デフォルトは60（日）です。
ZDM_OPC_RETRY_WAIT_TIME	Object Storeの再試行待機時間（秒）をユーザーが指定できるようにします。デフォルト値は529（秒）です。
ZDM_OPC_RETRY_COUNT	Object Storeの再試行回数をユーザーが指定できるようにします。デフォルト値は0（ゼロ）です。
ZDM_CURL_LOCATION	ソース上のCURLバイナリのカスタム・ロケーションをユーザーが指定できるようにします。デフォルト値は/usr/bin/curlです。
ZDM_LOG_OSS_PAR_URL	ログ・ファイルをアップロードする事前認証済みのURLをユーザーが指定できるようにします。ログは、現在のフェーズとフェーズの実行ステータスも取得します。

ZDM_SRC_TNS_ADMIN	Grid Infrastructureが実装されていない場合に、ソース・データベースのTNS_ADMINのカスタム・ロケーションをユーザーが指定できるようにします。
SRC_SSH_RETRY_TIMEOUT	Oracle ZDMがソースへのSSH接続の試行を停止するまでのタイムアウト値（分）を指定します。
TGT_SSH_RETRY_TIMEOUT	Oracle ZDMがターゲットへのSSH接続の試行を停止するまでのタイムアウト値（分）を指定します。
TGT_SKIP_DATAPATCH	移行後のタスクとして、Zero Downtime Migrationがターゲット・データベースでデータパッチ・ユーティリティを実行するかどうかをユーザーが指定できるようにします。デフォルト値はFALSEです。FALSEに設定されている場合、Zero Downtime Migrationはデータパッチを実行します。
MAX_DATAPATCH_DURATION_MINS	データパッチ・ユーティリティが完了しなかった場合に操作を停止させるタイムアウト値（分）をユーザーが構成できるようにします。デフォルト値はありません。デフォルトでは、Zero Downtime Migrationはデータパッチが完了するまで待ちます。
DATAPATCH_WITH_ONE_INSTANCE_RUNNING	TRUEに設定されている場合、データパッチ・ユーティリティが実行されている間にターゲット・データベース・サーバーで実行中の1つを除いて、Zero Downtime Migrationはすべてのインスタンスを停止できます。データパッチが完了したら、停止中のすべてのインスタンスは起動されます。デフォルト値はFALSEです。
NON_CDBTOPDB_CONVERSION	移行プロセスの一貫として、非CDBからCDBへの変換を実施するかどうかを指定します。デフォルト値はFALSEです。

Oracle ZDMを使用したデータベースの移行

データベース移行の事前チェック・プロセスの実行

データベース移行の事前チェック・プロセスを実行するには、-evalフラグを指定してzdmcli migrate databaseコマンドを実行する必要があります。以下はこのコマンドの例です。

```
./zdmcli migrate database -sourcedb db1_phx -sourcename sourcezdm -srcauth zdmauth  
-srcarg1 user:opc -srcarg2 identity_file:/home/zdmuser/.ssh/zdm-servicenode.ppk  
-srcarg3 sudo_location:/usr/bin/sudo -targetnode targetzdm  
-targethome /u01/app/oracle/product/18.01  
-backupuser bkpuser -rsp /u01/response/zdm_template.rsp -tgtauth zdmauth -tgtarg1 user:opc  
-tgtarg2 identity_file:/home/zdmuser/.ssh/zdm-spk -tgtarg3 sudo_location:/usr/bin/sudo  
-ignore ALL -eval
```

フラグ	説明
-sourcedb または- sourcesid	ソース・データベース名。この例ではデータベース名はdb1_phxです。ソース・データベースが、非Grid Infrastructure環境にデプロイされたシングル・インスタンス・データベースの場合、データベース名と一緒に-sourcesidフラグを指定するのが適切です。
-sourcename	ソース・データベース・サーバーのホスト名。この例ではsourcezdmです。
-srcauth	ソース・データベース・サーバーにアクセスするために必要な認証プラグイン名。このフラグにより、Oracle ZDMは、使用される特定の認証を指定したn個のプラグイン引数が渡されることを知ることができます。
-srcarg1	プラグイン引数。この例では、opc userが指定されています。
-srcarg2	プラグイン引数。この例では、公開鍵のロケーション、/home/zdmuser/.ssh/zdm-servicenode.ppkが指定されています。
-srcarg3	プラグイン引数。この例では、sudoロケーション、/usr/bin/sudoが指定されています。
-backupuser	データベースのバックアップ/リストア権限を持つユーザー名。移行が開始されると、Oracle ZDMはこのユーザーのパスワードを求めます。Object Storageの認証トークンを提供してください。この例では、ユーザーはbkpuserです。
-rsp	レスポンス・ファイルのロケーション。この例では、ロケーションは/u01/response/zdm_template.rspです。
-tgtauth	ターゲット・データベース・サーバーにアクセスするために必要な認証プラグイン。このフラグにより、Oracle ZDMは、使用される特定の認証を指定したn個のプラグイン引数が渡されることを知ることができます。
-tgtarg1	プラグイン引数。この例では、opc userが指定されています。

-tgtarg2	プラグイン引数。この例では、公開鍵のロケーション、 /home/zdmuser/.ssh/zdm-servicenode.ppkが指定されています。
-tgtarg3	プラグイン引数。この例では、sudoロケーション、/usr/bin/sudoが指定されています。
-ignore	チェックを無視します (ALL WARNING PATCH_CHECK) 。
-eval	データベースを実際に移行せずに、テスト実行、すなわち移行の評価を行います。これにより、すべての移行フェーズが円滑に実行されることを確認できます。移行前に行うべき修正がある場合は、ユーザーにアラートが表示されます。

コマンドが実行され、パスワードの入力を促された後に、ジョブIDが生成されます。

```
./zdmcli migrate database -sourcedb db1_phx -sourcnode sourcezdm -srcauth zdmauth
-srcarg1 user:opc -srcarg2 identity_file:/home/zdmuser/.ssh/zdm-servicenode.ppk
-srcarg3 sudo_location:/usr/bin/sudo -targetnode targetzdm
-targethome /u01/app/oracle/product/18.01
-backupuser bkpuser -rsp /u01/response/zdm_template.rsp -tgtauth zdmauth -tgtarg1 user:opc
-tgtarg2 identity_file:/home/zdmuser/.ssh/zdm-spk -tgtarg3 sudo_location:/usr/bin/sudo
-ignore ALL -eval
Enter source database db1_phx1fn SYS password:
Enter user "xxxx" password:
Operation "zdmcli migrate database" scheduled with the job ID "4".
```

入力を求められた2つ目のパスワード、つまりバックアップ・ユーザーのパスワードは、Object Storeの認証トークンのことです。以下のコマンドを使用して、コマンドの進捗状況を問い合わせることができます。

```
./zdmcli query job -jobid 4
```

```

[zdmuser@zdm-servicenode bin]$ ./zdmcli query job -jobid 4 zdm-servicenode.x.x.x.x:Audit ID:27
Job ID:4
User: zdmuser
Client: zdm-servicenode
Scheduled job command: "zdmcli migrate database -sourcedb db1_phx1fn -sourcenode sourcezdm -srcauth zdmauth -srcarg1
user:opc -srcarg2 identity_file: /zdmuser/.ssh/zdm-servicenode.ppk -srcarg3 sudo_location:/usr/bin/sudo -targetnode targetzdm -
targethome /u01/app/oracle/product/18.0.0user xxxx -rsp /u01/response/zdm_template.rsp -tgtauth zdmauth -tgtarg1 user:opc -
tgtarg2 identity_file:/home/zdmuser/.ssh/zdm-serg3 sudo_location:/usr/bin/sudo -ignore ALL -eval"
Scheduled job execution start time:2020-X. Equivalent local time:2020-X
Current status:SUCCEEDED
Result file path: "/u01/app/oracle/zdmbase/chkbase/scheduled/job-4-2020-05.log"
Job execution start time:2020-05-06 20:11:49
Job execution end time:2020-05-06 20:15:15
Job execution elapsed time:3 minutes 25 seconds
ZDM_GET_SRC_INFO.....COMPLETED
ZDM_GET_TGT_INFO.....COMPLETED
ZDM_SETUP_SRC.....COMPLETED
ZDM_SETUP_TGT.....COMPLETED
ZDM_GEN_RMAN_PASSWD.....COMPLETED
ZDM_PREUSERACTIONS.....COMPLETED
ZDM_PREUSERACTIONS_TGT.....COMPLETED
ZDM_VALIDATE_SRC.....COMPLETED
ZDM_VALIDATE_TGT.....COMPLETED

```

データベース移行の実行

データベースの移行プロセスを実行するために、今回は-evalフラグを使用せずに、zdmcli migrate databaseコマンドを実行する必要があります。以下はこのコマンドの例です。

```

./zdmcli migrate database -sourcedb db1_phx -sourcenode sourcezdm -srcauth zdmauth
-srcarg1 user:opc -srcarg2 identity_file:/home/zdmuser/.ssh/zdm-servicenode.ppk
-srcarg3 sudo_location:/usr/bin/sudo -targetnode targetzdm
-targethome /u01/app/oracle/product/18.01
-backupuser bkpuser -rsp /u01/response/zdm_template.rsp -tgtauth zdmauth -tgtarg1 user:opc
-tgtarg2 identity_file:/home/zdmuser/.ssh/zdm-spk -tgtarg3 sudo_location:/usr/bin/sudo
-ignore ALL
Enter source database sourcedb db1_SYS password:
Enter user "xxxxxx" password:
Operation "zdmcli migrate database" scheduled with the job ID "5".

```

入力を求められた2つ目のパスワード、つまりバックアップ・ユーザーのパスワードは、Object Storeの認証トークンのことです。

コマンドが実行され、パスワードの入力を促された後に、ジョブIDが生成されます。以下のコマンドを使用して、コマンドの進捗状況を問い合わせることができます。

```
./zdmcli query job -jobid 5
```

移行が完了したら、次のような出力が表示されます。

```
zdmuser@zdm-servicenode bin]$ ./zdmcli query job -jobid 5
zdm-servicenode.x.x.x.x:Audit ID:52
Job ID:5
User: zdmuser
Client: zdm-servicenode
Scheduled job command: "zdmcli migrate database -sourcedb db1_phx1fn -sourcenode sourcezdm -srcauth zdmauth -srcarg1
user:opc -srcarg2
identity_file:/home/zdmuser/.ssh/zdm-servicenode.ppk -srcarg3 sudo_location:/usr/bin/sudo -
targetnode targetzdm -targethome /u01/app/oracle/product/18.0.0.0/dbhome_1 -
backupuser xxxxx -rsp /u01/response/zdm_template.rsp -tgtauth zdmauth -tgtarg1 user:opc -tgtarg2 id
entity_file:/home/zdmuser/.ssh/zdm-servicenode.ppk -tgtarg3 sudo_location:/usr/bin/sudo -ignore ALL"
Scheduled job execution start time:2020-04-15T20:23:59Z.Equivalent local time:2020-05-06 20:23:59
Current status:SUCCEEDED
Result file path: "/u01/app/oracle/zdmbase/chkbase/scheduled/job-5-2020-05.log"
Job execution start time:2020-05-06 20:24:19
Job execution end time:2020-05-06 20:45:15
Job execution elapsed time:20 minutes 55 seconds
ZDM_GET_SRC_INFO .....COMPLETED
ZDM_GET_TGT_INFO .....COMPLETED
ZDM_SETUP_SRC .....COMPLETED
ZDM_SETUP_TGT .....COMPLETED
ZDM_GEN_RMAN_PASSWD .....COMPLETED
ZDM_PREUSERACTIONS .....COMPLETED
ZDM_PREUSERACTIONS_TGT .....COMPLETED
ZDM_VALIDATE_SRC .....COMPLETED
ZDM_VALIDATE_TGT .....COMPLETED
ZDM_OBC_INST_SRC .....COMPLETED
ZDM_OBC_INST_TGT .....COMPLETED
ZDM_BACKUP_FULL_SRC .....COMPLETED
ZDM_BACKUP_INCREMENTAL_SRC..COMPLETED
ZDM_DISCOVER_SRC .....COMPLETED
ZDM_COPYFILES .....COMPLETED
ZDM_PREPARE_TGT .....COMPLETED
ZDM_SETUP_TDE_TGT .....COMPLETED
ZDM_CLONE_TGT .....COMPLETED
ZDM_FINALIZE_TGT .....COMPLETED
ZDM_CONFIGURE_DG_SRC .....COMPLETED
ZDM_SWITCHOVER_SRC .....COMPLETED
ZDM_SWITCHOVER_TGT .....COMPLETED
ZDM_NONCDBTOPDB_PRECHECK ...COMPLETED
ZDM_NONCDBTOPDB_CONVERSION  COMPLETED
ZDM_POSTUSERACTIONS .....COMPLETED
ZDM_POSTUSERACTIONS_TGT .....COMPLETED
ZDM_CLEANUP_SRC .....COMPLETED
ZDM_CLEANUP_TGT .....COMPLETED
```

よくある質問と一般的な問題

F.A.Q

質問	回答
1.Object Storageの認証にはどのような認証方式を使用すべきですか。	認証トークンは、OCIコンソールから作成できます。以下の方法があります。 main Menu > Identity > Users に移動します。 次に、オブジェクトストレート認証に使用するユーザーを選択し、Auth Token オプションをクリックして、generateをクリックします。 認証に使用するユーザーを選択し、Auth Token オプションをクリックして、generateトークンの生成をクリックします。 トークンの作成時には、トークンをコピーしてください。トークンには後からアクセスできないので、アクセスしたい場合は新しいトークンを生成する必要があります。
2.SE (Standard Edition) のデータベース移行ではどのような移行方法を使用すべきですか。	ZDMでは、ソース・データベースとしてStandard Editionが使用できます。推奨される移行方法はPHYSICAL_OFFLINEです。この方法では、バックアップ・ファイルにオブジェクト・ストアを使用します。また、ZDM 21cからは、SEの移行にも有効なLogical Workflowを活用できるようになりました。
3.Oracle ZDMを使用してデータベースをオンプレミスの物理サーバーに移行できますか。	はい。Oracle ZDMでは、ターゲット・プラットフォームがExadata Database Machineであれば、オンプレミスの移行が可能です。
4.Oracle ZDMを使用して、ターゲットのOSと異なるLinuxバージョンでホストされるデータベースを移行できますか。	はい。Oracle ZDMでは、異なるOSバージョン間の移行が可能です (Linuxのみ)。
5.ソース・データベースとターゲット・データベースのバージョンが違っていても構いませんか。	Oracle ZDMの物理的な移行では、同じバージョンのデータベース間の移行しかできません。移行先のデータベースのパッチ・レベルを上げることはできますが、その場合は移行後にターゲット・データベースでデータパッチを実行する必要があります。 Oracle ZDM の論理的な移行では、バージョンをまたいだ移行が可能のため、インフラでのアップグレードの道が開けます。
6.Oracle ZDMでは、Oracle Cloudへの移行中に非CDBからCDBへの変換が実施されますか。	はい。Oracle ZDMはこの種類の移行をサポートしています。レスポンス・ファイルのパラメータはNONCDBTOPDB_CONVERSIONにする必要があります。
7.Oracle ZDMは異なるプラットフォーム間の移行をサポートしていますか。	はい、Oracle ZDMでは、AIXおよびSolaris上でホストされているデータベースのクロスプラットフォーム移行が可能です。サポートされている方法はLOGICAL_OFFLINEです。
8.Oracle ZDMを使用して、あるOCIリージョンでホストされるデータベースを別のOCIリージョンに移行できますか。	はい。Oracle ZDMでは、異なるOCIリージョン間の移行が可能です。
9.Oracle ZDMを使用して、OCI-CデータベースからOCIに移行できますか。	はい。Oracle ZDMでサポートされるソースの1つは、OCI Classicでホストされるデータベースです。
10.Oracle ZDMはApexアプリケーション・レポートを移行しますか。	いいえ。Oracle ZDMはこのタスクを実行しません。

<p>11. ZDMは基本的にswtichoverを行わず、DGの設定のみを行うように指示することはできますか？</p>	<p>はい、Oracle ZDMでは任意の段階で移行プロセスを一時停止することができます。したがって、ロールスワップやスイッチオーバーの段階の前に移行プロセスを一時停止することができます。Oracle ZDMでmigrate databaseコマンドを実行する際に、-pauseafterフラグを使用して、一時停止したい段階（この場合はZDM_CONFIGURE_DG_SRC）を指定する必要があります。</p>
<p>12.どのOSバージョンがOracle ZDMバイナリでサポートされますか。</p>	<p>Oracle ZDMはOracle Linux 7で実行されます。</p>
<p>13.バグに遭遇した場合は、Oracle ZDMの暫定パッチを提供してくれますか。</p>	<p>Oracle ZDMは、必要に応じて問題に対処するパッチをリリースします。問題に遭遇している場合は、サービス・リクエストを登録して製品管理担当者にお問い合わせください。</p> <p>解決時間を短縮するために、My Oracle SupportのドキュメントSRDC – Data Collection for Database Migration Using Zero Downtime Migration (ZDM) (DOC ID 2595205.1) の指示に従ってください。</p> <p>https://support.oracle.com/epmos/faces/DocContentDisplay?id=2595205.1</p>
<p>14.Oracle ZDMはソース・サーバーまたはターゲット・サーバーにインストールできますか。</p>	<p>はい。Oracle ZDMはどのようなロケーションにもインストールできます。ただし、Oracle ZDMを別のノードで実行することがベスト・プラクティスとして推奨されます。また、Oracle ZDMサービス・ノードがデプロイされるノードにOracle GIが存在してはなりません。</p>
<p>15.Oracle ZDMのインストールにはどのようなOSパッケージが必要ですか。</p>	<p>以下のパッケージが必要です。</p> <ul style="list-style-type: none"> ○ glibc-devel ○ expect ○ unzip ○ libaio ○ oraclelinux-developer-release-el7
<p>16.ソース・データベースとターゲット・データベースには同じデータベース・パッチが適用されている必要がありますか。</p>	<p>これはベスト・プラクティスとして推奨されますが、ターゲット・データベースのパッチ・レベルがソース・データベースより高くても構いません。ただし、その場合は、移行後にターゲット・データベースでデータパッチを実行する必要があります。</p>
<p>17.Oracle ZDMではソースとターゲットとの認証にどのような鍵形式がサポートされますか。</p>	<p>サポートされる形式はRSAです。</p>
<p>18.ターゲット・クラウド・プラットフォームごとのサポートされるストレージ・タイプを教えてください。</p>	<ul style="list-style-type: none"> • Oracle Database Cloud Service Bare Metal <ol style="list-style-type: none"> 1. OCI Object Storage • Oracle Database Cloud Service Virtual Machine <ol style="list-style-type: none"> 1. OCI Object Storage • Exadata Cloud Service <ol style="list-style-type: none"> 1. Oracle Zero Data Loss Recovery Appliance 2. NFSストレージ • Exadata Cloud at Customer <ol style="list-style-type: none"> 1. Oracle Zero Data Loss Recovery Appliance 2. NFSストレージ • Exadata On Premises

	<ol style="list-style-type: none"> 1. Oracle Zero Data Loss Recovery Appliance 2. NFSストレージ <ul style="list-style-type: none"> • Autonomous Database (論理的ワークフローのみ) <ol style="list-style-type: none"> 1. OCI Object Storage
<p>19.Oracle ZDMソフトウェアにはライセンスが必要ですか。</p>	<p>Oracle ZDMは無償で利用でき、次のライセンス契約に従います。 https://www.oracle.com/downloads/licenses/sqldev-license.html#licenseContent</p>
<p>20.どのような移行先がサポートされていますか。</p>	<ul style="list-style-type: none"> • Oracle Database Cloud Serviceベアメタル • Oracle Database Cloud Service仮想マシン • Exadata Cloud Service • Exadata Cloud at Customer • オンプレミスのExadata Database (物理的ワークフローのみ) • Autonomous Database (論理的ワークフローのみ)

既知の問題

あらゆる一般的な問題が、以下のOracle Zero Downtime Migrationのドキュメントに文書化され、定期的に更新されます。製品リリース・ノートの「既知の問題」セクションを参照してください。

<https://docs.oracle.com/en/database/oracle/zero-downtime-migration/>

トラブルシューティングとその他のリソース

Oracle ZDMログの確認について

- 次のOracle ZDMサーバー・ログを確認します
 - o `$ZDM_BASE/crsdata/<zdm_service_node>/rhp/rhpserver.log.0`
- 次のソース・ノード・ログを確認します
 - o `<oracle_base>/zdm/zdm_<src_db_name>_<job_id>/zdm/log`
- 次のターゲット・ノード・ログを確認します
 - o `<oracle_base>/zdm/zdm_<tgt_db_name>_<job_id>/zdm/log`

Zero Downtime Migrationに関連するすべてのOracle Supportサービス・リクエストについては、必ず以下のMy Oracle Supportドキュメントの指示に従ってください。

- 『SRDC – Data Collection for Database Migration Using Zero Downtime Migration (ZDM)』 (DOC ID 2595205.1)
- <https://support.oracle.com/epmos/faces/DocContentDisplay?id=2595205.1>

Zero Downtime Migrationに関連するベスト・プラクティスについては、以下のMaximum Availability Architectureドキュメントを参照してください。

- 『MAA Practices for Cloud Migration Using ZDM』 (Doc ID 2562063.1)
- <https://support.oracle.com/epmos/faces/DocContentDisplay?id=2562063.1>

CONNECT WITH US

+1.800.ORACLE1までご連絡いただくか、oracle.comをご覧ください。
北米以外の地域では、oracle.com/contactで最寄りの営業所をご確認いただけます。

 blogs.oracle.com

 facebook.com/oracle

 twitter.com/oracle

Copyright © 2021, 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、ここに記載されてい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。0120

Oracle Zero Downtime Migration – Oracle ZDMを使用した物理的移行のステップ・バイ・ステップ・ガイド

2021年8月

著者：Ameet Kumar Nihalani、Ricardo Gonzalez

共著者：Oracle RACPackチーム、Oracle ZDM開発チーム

