

Oracle Cloud Infrastructure上のOracle GoldenGateマイクロサービス・アーキテクチャ 構成ガイド

ORACLE WHITE PAPER | 2019年4月



免責事項

下記事項は、弊社の一般的な製品の方向性に関する概要を説明するものです。また、情報提供を唯一の目的とするものであり、いかなる契約にも組み込むことはできません。以下の事項は、マテリアルやコード、機能を提供することをコミットメント（確約）するものではないため、購買決定を行う際の判断材料になさらないで下さい。オラクルの製品に関して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。Oracle GoldenGateのライフサイクル管理は、Oracle Cloud Infrastructure上のOracle Databaseから独立しており、クライアントで処理する必要があります。

改訂履歴

本ホワイトペーパーは、初版の公開後、次の改訂がありました。

日付	改訂内容
2019年4月18日	初版発行

Oracle Cloud Infrastructureホワイトペーパーの最新版は、
<https://cloud.oracle.com/iaas/technical-resources>でご覧いただけます。



目次

概要	4
Oracleソフトウェア	4
コンポーネントの概要	4
構成の概要	6
Oracle Cloud Infrastructure上でのOracle GoldenGateの構成	7
Oracle Cloud Infrastructure DBシステムの設定	7
VCNの構成	7
Oracle Clientのインストール	8
ファイル・システムの設定	9
データベースの構成	19
Oracle GoldenGateの構成	20
TNS別名の作成	31
Oracle GoldenGateのHA用の構成	32
リバース・プロキシとHAの構成	36
データ・レプリケーションの構成	38
まとめ	41

概要

Oracle GoldenGateは、異種IT環境におけるリアルタイムのデータ統合およびレプリケーションのための包括的なソフトウェア・パッケージです。高可用性ソリューション、リアルタイム・データ統合、トランザクション変更データ・キャプチャ、データ・レプリケーション、そして運用システムと分析エンタープライズ・システムの間での変換および検証が可能になります。

Oracle GoldenGateは、Classic ArchitectureとMicroservices Architectureの2つのアーキテクチャをサポートしています。Microservices Architectureでは、Oracle GoldenGate環境の一環としてREST対応のサービスが提供されます。REST対応のサービスでは、HTML5のWebページ、コマンドライン、APIを通じたりモートの構成、管理および監視が可能です。

このホワイトペーパーでは、Oracle Cloud Infrastructureの仮想マシン（VM）、ベア・メタル、またはExadataデータベース上でOracle GoldenGate Microservices Architectureを構成する際のベスト・プラクティスについてお伝えします。オンプレミス・データベースでのOracle GoldenGateの構成については扱いません。詳細は、[『Oracle GoldenGate Performance Best Practices』](#)と[『Oracle GoldenGate Microservices Architecture with Oracle RAC Configuration Best Practices』](#)を参照してください。

Oracleソフトウェア

追加された機能と、強化されたパフォーマンス特性を利用するには、Oracle GoldenGateリリース18.1以上を使用してください。[Oracle GoldenGateの最新版は、こちらからダウンロードできます。](#)パッチとアップデートは、[My Oracle Support](#)でご利用いただけます。推奨の最新パッチは、[最新GoldenGate/Database（OGG/RDBMS）パッチの推奨情報（Doc ID 2193391.1）](#)でご確認ください。

パフォーマンス強化のために、Oracle GoldenGateはOracle Database Cloud Enterprise Editionでを使用することをお勧めします。統合されたGoldenGateのExtractおよびReplicat機能を使用するには、Oracle Database 11gリリース2（11.2.0.4）以上が必要です。詳細は、[『Oracle GoldenGate Performance Best Practices』](#)を参照してください。

コンポーネントの概要

Oracle Cloud InfrastructureでOracle GoldenGateを構成するには、次のコンポーネントを使用します。

Oracle Database Cloud Service

Oracle Cloud Infrastructureで提供されるOracle Database Cloud Serviceを使用すると、ベア・メタル、VM、およびExadata DBシステム上のデータベースにアクセスできます。

Oracle GoldenGate

Oracle GoldenGateは、エンタープライズの複数の異機種プラットフォーム間において、トランザクション・レベルでデータの交換および操作を可能にするデータ・レプリケーション・ソリューションです。トランザクションの整合性を維持したまま、既存インフラストラクチャへのオーバーヘッドを最小限に抑えて、コミットされたトランザクションを移動します。一方向、双方向、P2P、ブロードキャスト、統合カスケードなど、さまざまなトポロジにインストールできます。詳細は、[Oracle GoldenGateのWebサイト](#)を参照してください。

Oracle Grid Infrastructure Bundled Agent

Oracle Grid Infrastructure Bundled Agent (XAG) は、Oracle Grid Infrastructure のコンポーネントで、エージェント制御 (AGCTL) によって管理されるアプリケーション・リソースおよびリソース・タイプに対する高可用性 (HA) フレームワークを構成します。このフレームワークは、使用準備の整った完全なアプリケーション (HA) ソリューションであり、アプリケーションを統合して完全なアプリケーションHAを実現するための事前定義されたOracle Grid Infrastructureリソースの構成とエージェントを含んでいます。エージェントのコマンドライン・ユーティリティであるAGCTLを使用して、Oracle GoldenGateを起動および停止したり、クラスタのノード間でOracle GoldenGateを再配置したりすることができます。

Oracle GoldenGate Microservices Architectureを使用する場合はXAGバージョン9以上が必要です。[最新版のエージェント・ソフトウェアはこちらから](#)ダウンロードできます。

Oracle Database File System

Oracle Database File System (DBFS) は、データベースに格納されたファイルに対するファイル・システム・インタフェースを作成します。DBFSは、ローカル・ファイル・システムのように見える共有ネットワーク・ファイル・システムを提供する点でNFSに似ています。データがデータベース内に格納されるため、このファイル・システムは、Oracle Databaseが提供するHAおよびディザスタ・リカバリ (DR) 機能をすべて継承します。

Oracle Automatic Storage Management Cluster File System

Oracle Automatic Storage Management Cluster File System (ACFS) は、POSIX、X/OPEN、およびWindowsに準拠した業界標準の汎用クラスタ・ファイル・システムで、Oracle Exadata Database Machine、Oracle Database Appliance、および従来のサーバー環境など複数のOSとサーバー・プラットフォームをサポートしています。Oracle ACFSは、Oracle Automatic Storage Management (ASM) ファイルを利用し、ストライプ化、ミラー化、リバランシングなどのASM機能を継承しています。

構成の概要

この項では、Oracle Cloud Infrastructure上でOracle GoldenGateを構成する際に必要な手順の概要を説明します。概要に続いて、詳細なステップを紹介します。

1. Oracle Cloud Infrastructure DBシステムの設定 : Oracle Cloud Infrastructure上でOracle GoldenGateを構成するには、Oracle Cloud Infrastructure上にVM、ベア・メタル、またはExadata DBシステムが必要です。これらのデータベース・システムについては、[「ベア・メタルおよび仮想マシンDBシステム」](#)と[「Exadata DBシステム」](#)を参照してください。
2. VCNの構成 : 仮想クラウド・ネットワーク（VCN）は、Oracle Cloud Infrastructureでデータベースなどのリソースに安全に接続できるソフトウェア定義ネットワークです。Oracle GoldenGateが正常に機能するためには、セキュリティ・リストやファイアウォールなどのVCNコンポーネントを構成する必要があります。作成の手順など、VCNとセキュリティ・リストの詳細は[Oracle Cloud Infrastructure Networkingのドキュメント](#)を参照してください。
3. Oracle Clientのインストール : Oracle GoldenGate用のOracle Clientライブラリへの専用アクセスを確保するために、Oracle Clientをインストールします。Oracle GoldenGateは、クライアント・ライブラリを使用するため、Oracle Databaseのライブラリに依存しません。
4. ファイル・システムの設定 : Oracle GoldenGateを使用してOracle Cloud Infrastructure上でHAを構成するには、Oracle DBFSまたはOracle ACFSを設定する必要があります。
5. データベースの構成 : ベスト・プラクティスに従って、Oracle GoldenGateのレプリケート環境にソースおよびターゲット・データベースを構成します。
6. Oracle GoldenGateの構成 : ベスト・プラクティスに従って、Oracle Cloud InfrastructureにOracle GoldenGateのコンポーネントをインストールして構成します。
7. TNS別名の作成 : Oracle RACのノード間を切り替える際にOracle GoldenGateプロセスのデータベース接続を簡素化するために、TNS別名を作成します
8. Oracle GoldenGateのHA用の構成 : Oracle Cloud Infrastructure上でOracle GoldenGateをHA用に構成します。
9. リバース・プロキシとHAの構成 : Nginxを使用してリバース・プロキシとHAを構成します。
10. データ・レプリケーションの構成 : データ・レプリケーションに必要なOracle GoldenGateのExtract、Replicat、およびPathプロセスを作成して構成します。

Oracle Cloud Infrastructure上でのOracle GoldenGateの構成

この項では、Oracle Cloud InfrastructureでOracle GoldenGateを構成する詳細なステップについて説明します。

Oracle Cloud Infrastructure DBシステムの設定

Oracle GoldenGateのデプロイメントには、まずOracle Cloud Infrastructure上にVM、ベア・メタル、またはExadata DBシステムが必要です。ビジネス上のニーズに応じて、既存のDBシステムにOracle GoldenGateをデプロイすることも、新しいDBシステムを起動することもできます。Oracle Cloud Infrastructure上でDBシステムを起動および管理する手順については、[「ベア・メタルおよび仮想マシンDBシステムの管理」](#)、または[「Exadata DBシステムの管理」](#)を参照してください。

VCNの構成

この項では、Oracle Cloud InfrastructureでOracle GoldenGateを使用するためにVCNを構成する手順について説明します。

セキュリティ・リスト

インGRESS・ポート443を開き、サービス・プロキシとしてNginxを使用して、Oracle GoldenGateサービスに接続します。詳細は、[「DBシステムのセキュリティ・リストの更新」](#)を参照してください。

セキュリティ・リストを更新すると、次のような値のエントリが作成されます。

- **ソース** : ソースCIDRブロック
- **IPプロトコル** : TCP
- **ソース・ポート範囲** : すべて
- **宛先ポート範囲** : 443
- **許可** : 次のポートのTCPトラフィック : 443 HTTPS

Source: 0.0.0.0/0	IP Protocol: TCP	Source Port Range: All	Destination Port Range: 443	Allows: TCP traffic for ports: 443 HTTPS
-------------------	------------------	------------------------	-----------------------------	--

ファイアウォール

ホストDBシステムのファイアウォールを更新し、インGRES・アクティビティ用にポート443を開きます。

```
[ssh into DB System as opc user]
sudo iptables-save > /tmp/iptables.orig
sudo iptables -I INPUT 8 -p tcp -m state --state NEW -m tcp --dport 443 -j
ACCEPT -m comment --comment "for Nginx"
sudo service iptables save
sudo service iptables restart
sudo iptables --list -n
```

Oracle Clientのインストール

次の手順で、Oracle GoldenGate用のOracle Clientライブラリへの専用アクセスを確保するためにOracle Clientをインストールします。

クライアント・ホームでは、/u01マウント・ポイントに約1.7 GBの容量が必要です。使用可能なクラウド・ツールを使用して、追加のホームをパッチまたは作成するなど、ライフサイクル管理の操作をこれらの環境で実行する場合には、マウント・ポイントに十分な空き容量があることを確認してください。

1. Oracle Databaseと同じバージョンの[Oracle Clientソフトウェアをダウンロード](#)します。
(クライアント・ファイルにアクセスするには、「すべて参照」リンクをクリックします。)
2. ファイルをダウンロードしたら、各データベース・ホストにアップロードします。
3. インストール・ディレクトリを作成します。以下はOracle Database 18cクライアントの場合の例です。

```
[ssh into DB System as opc user]
sudo mkdir -p /u01/app/oracle/product/18.0.0.0/client
sudo chown -R oracle:oinstall /u01/app/oracle/product/18.0.0.0/client
```

4. レスポンス・ファイルを編集します。

```
[ssh into DB System as opc user]
sudo su - oracle
cd <installation media>
vi ./response/client_install.rsp
```


以下はレスポンス・ファイルのパラメータの例です。

```
UNIX_GROUP_NAME=oinstall
INVENTORY_LOCATION=/u01/app/oraInventory
ORACLE_HOME=<client home>
ORACLE_BASE=/u01/app/oracle
oracle.install.client.installType=Custom
oracle.install.client.customComponents="oracle.sqlplus:18.0.0.0.0"," oracle
.rdbms.oci:18.0.0.0.0" ," oracle.rdbms.util:18.0.0.0.0"
```

5. Oracle Clientをインストールします。

```
[ssh into DB System as opc user]
sudo su - oracle
cd <installation media>
./runInstaller -silent -nowait -responseFile <path of response file>
```

クライアント・ソフトウェアのインストール中にスワップ容量不足の警告が表示された場合は無視してください。

6. tnsnames.oraファイルを、データベース・ホームからクライアント・ホームにコピーします。

```
[ssh into DB System as opc user]
sudo su - oracle
cp <database home>/network/admin/tnsnames.ora <client home>/network/admin
```

クライアント・ホームでtnsnames.oraファイルにTNS別名が追加され、Oracle GoldenGateで使用されます。

注意：オラクルから提供されるクラウド・ツールは、Oracle Clientホームでのライフサイクル管理操作をサポートしていません。このクライアント・ホームに対しては、お客様の責任でバッチを適用してください。

ファイル・システムの設定

Oracle ACFSまたはOracle DBFSファイル・システムをHA用に設定する必要があります。DBシステムにOracle Data Guardがすでに構成されている場合には、DBFSの使用をお勧めします。

ACFSの設定

1. ASMCMDを使用して、ファイル・システムを作成します。

決定されたサイズ要件に従って、ファイル・システムのサイズを変更します。DATAディスク・グループに十分な容量がない場合は、RECOディスク・グループの使用を検討してください。詳細は、Oracle GoldenGateのドキュメントで[ディスク要件](#)を参照してください。

```
[ssh into DB System as opc user]
sudo su - grid
asmcmd volcreate -G DATA -s 10G acfs_gg
```

2. ボリューム・デバイスを確認します。

```
[ssh into DB System as opc user]
sudo su - grid
asmcmd volinfo -G DATA acfs_gg
```

以下はACFSボリューム・デバイスの出力の例です。

```
Diskgroup Name: DATA
Volume Name: ACFS_GG
Volume Device: /dev/asm/acfs_gg-453
State: ENABLED
Size (MB): 10240
Resize Unit (MB): 512
Redundancy: UNPROT
Stripe Columns: 8
Stripe Width (K): 1024
Usage: ACFS
Mountpath:
```

3. ACFSファイル・システムを作成します。

```
/sbin/mkfs -t acfs <volume device>
```

4. ACFS用のディレクトリを作成します。

```
[ssh into DB System as opc user]
su -
mkdir /mnt/acfs_gg
```

5. グリッド・ユーザー用のORACLE_HOMEおよびPATH環境変数を確認します。

```
[ssh into DB System as opc user]
sudo su - grid
cat .bashrc
```

以下はGrid Infrastructure 12.2の例です。

```
ORACLE_HOME=/u01/app/12.2.0.1/grid; export ORACLE_HOME
PATH=$PATH:/u01/app/12.2.0.1/grid/bin; export PATH
LD_LIBRARY_PATH=/u01/app/12.2.0.1/grid/lib; export LD_LIBRARY_PATH
ORACLE_SID=+ASM1; export ORACLE_SID
```

6. まだ作成していない場合は、新しいACFSファイル・システムのためのCluster Ready Services (CRS) リソースを作成します。

```
[ssh into DB System as opc user]
su -
export ORACLE_HOME=<ORACLE_HOME environmental variable of grid user>
export PATH=<PATH environmental variable of grid user>
export LD_LIBRARY_PATH=<LD_LIBRARY_PATH environmental variable of grid user>
```

```
export ORACLE_SID=<ORACLE_SID environmental variable of grid user>
srvctl add filesystem -device <device name> -volume acfs_gg -diskgroup
DATA -path /mnt/acfs_gg -user oracle -autostart NEVER
```

7. 現在構成されているACFSファイル・システムを確認するために、次のコマンドを使用してファイル・システムの詳細を表示します。

```
[ssh into DB System as opc user]
sudo su - grid
srvctl config filesystem
```

8. ACFSリソースのステータスを確認して、マウントします。

```
[ssh into DB System as opc user]
sudo su - grid
srvctl status filesystem -volume acfs_gg -diskgroup DATA
srvctl start filesystem -volume acfs_gg -diskgroup DATA
```

作成されるCRSリソースは、

ora.<diskgroup name>.<volume name>.acfsという形式で名前が付けられます。前のファイル・システムの例では、CRSリソースの名前はora.datacl.acfs_gg.acfsとなります。

9. ファイル・システムがマウントされたら、Oracle GoldenGateのデプロイメントとサービス・マネージャを格納するディレクトリを作成します。

```
[ssh into DB System as opc user]
sudo su - oracle
cd /mnt/acfs_gg
mkdir deployments
```

DBFSの設定

Oracle DBFSは、Oracle databaseに格納されるファイルへのファイル・システム・インタフェースを提供します。Oracle Databaseに格納されるファイルは通常、SecureFiles LOBとして格納され、パス名、ディレクトリ、その他のファイル・システム情報はデータベース表に格納されます。オープンや作成など、ファイル・システムのアクセス権限は、PL/SQLの一連のプロシージャによって実装されます。Linuxの場合、DBFS_clientはFUSEカーネル・モジュールとのインタフェースを介してファイル・システムのマウント・ポイントを実装し、データベースに格納されるファイルへの透過的なアクセスを実現します。

DBFSの詳細は、DBFSの[ドキュメント](#)を参照してください。

1. VMまたはベア・メタルのDBシステムを使用している場合は、Oracle RACクラスタのすべてのノードに次のRPMをインストールしてください。Exadata DBシステムを使用している場合、RPMのインストールは必要ありません。VMまたはベア・メタルDBシステムにyumリポジトリを設定する場合は、[「OSのアップデート」](#)を参照してください。

```
#yum install fuse-libs
#yum install fuse-devel
```

2. DBFSの表領域を作成します。VMまたはベア・メタルDBシステムを使用している場合は、+DATAディスク・グループを使用します。

```
SQL> Alter session set container=<pdb_name>;
SQL> create bigfile tablespace dbfsts datafile '+DATA1' size 32g
autoextend on next 8g maxsize 300g EXTENT MANAGEMENT LOCAL AUTOALLOCATE
SEGMENT SPACE MANAGEMENT AUTO ;
SQL> create user dbfs_user identified by <password> default tablespace
dbfsts quota unlimited on dbfsts;
SQL> grant create session, create table, create view, create procedure,
dbfs_role to dbfs_user;
```

3. rootユーザーとして、Oracle RACクラスタのすべてのノード上にDBFSマウント・ポイントを作成し、その表領域にDBFSを作成します。

```
# cd /mnt
# mkdir dbfs
# chown oracle:oinstall dbfs/
```

4. すべてのノードでDBFSマウント・スクリプトによって使用されるTNS_ADMINディレクトリを作成します。

```
<alias name> =
(DESCRIPTION =
  (SDU=2097152)
  (ADDRESS = (PROTOCOL = TCP) (HOST =<hostname>) (PORT = 1521))
  (CONNECT_DATA =
    (SERVER = DEDICATED)
    (SERVICE_NAME = <pdb service name>)
  )
)
```

5. ユーザーをoracleに切り替えてから、DBFSファイル・システムを作成します。

```
#su - oracle
$cd $ORACLE_HOME/rdbms/admin
$ sqlplus dbfs_user/password@pdb
SQL> start dbfs_create_filesystem dbfsts ogg
```

6. LOBセグメントがNOCACHE LOGGINGを使用していることを確認し、出力を確認します。

```
SQL>alter session set container=pdb;
SQL> SELECT table_name, segment_name, logging, cache FROM user_lobs WHERE
tablespace_name='DBFSTS';
```

出力は次のようになります。

TABLE_NAME	SEGMENT_NAME	LOGGING	CACHE
-----	-----	-----	-----
SFS\$_FST_11	LOB_SFS\$_FST_11	YES	NO

LOBセグメントがNOCACHE LOGGINGを使用していない場合は、次のようにして変更します。

```
SQL> ALTER TABLE DBFS.<TABLE_NAME> MODIFY LOB (FILEDATA) (NOCACHE LOGGING);
```

7. すべてのノードに、シンボリック（ソフト）リンクを作成します。

Oracle Databaseリリース18.xの場合は、次のコマンドを使用してすべてのノードにライブラリ・パスを設定します。他のデータベース・バージョンの場合は、適切なRDBMS ORACLE_HOMEと適切なライブラリlibnnz12.aまたはlibnnz11.aに置き換えます。rootユーザーとして次のコマンドを実行します。

```
#export ORACLE_HOME=<client_home>
#ln -s $ORACLE_HOME/lib/libnnz18.a /usr/local/lib/libnnz18.a
#ln -s $ORACLE_HOME/lib/libclntsh.so /usr/local/lib/libclntsh.so
#ln -s /usr/lib64/libfuse.so.2 /usr/local/lib/libfuse.so.2
#echo /usr/local/lib >> /etc/ld.so.conf.d/usr_local_lib.conf
```

8. 両方のノードでマウント・ポイントが動作していることを確認します。passwd.txtにdbfs_userのパスワードがあることを確認してください。

VMまたはベア・メタルDBシステムの場合は、/usr/bin/fusermountの権限を変更します。詳細は、[Oracle Supportドキュメント1908868.1](#)を参照してください。

```
# chmod +x /usr/bin/fusermount
```

Oracle Databaseバージョン12.2以上の場合は、/etc/fuse.confファイルでuser_allow_otherパラメータのコメントを解除します。詳細は、[Oracle Supportドキュメント2388733.1](#)を参照してください。

次のコマンドを使用して、DBFSをマウントします。

```
nohup $ORACLE_HOME/bin/dbfs_client dbfs_user@<alias name> -o
allow_other,direct_io,failover,nolock /mnt/dbfs < ~/passwd.txt
```

動作することを確認したら、dbfs_clientプロセスを停止し、次のステップでクラスタ・リソースとして再度、登録して起動します。

9. DBFSをクラスタ・リソースとして登録します。

- A. [Oracle Supportドキュメント1054431.1](#)に添付されているmount-dbfs.zipファイルをダウンロードし、一時的な場所（/tmpなど）にあるデータベース・サーバーに配置します。次に、そのファイルを解凍すると、その中にmount-dbfs.shファイルとmount-dbfs.confファイルがあります。

B. お使いの環境用の`mount-dbfs.conf`で変数の設定を編集します。スクリプト・ファイルで次の変数の設定を編集するか、確認してください。変数の値を確認する際は、スクリプトのコメントが役に立ちます。

- DBNAME : DBの一意の名前
- MOUNT_POINT
- DBFS_USER
- ORACLE_HOME : `oracle_client`のホーム・ディレクトリ
- LOGGER_FACILITY : `syslog`がメッセージと、このスクリプトからの出力をログに記録する際に使用
- MOUNT_OPTIONS
- DBFS_PASSWD : `WALLET=false`の場合にのみ使用
- DBFS_PWDFILE_BASE : `WALLET=false`の場合にのみ使用
- WALLET : `true`または`false`
- TNS_ADMIN : `WALLET=true`の場合にのみ使用
- DBFS_LOCAL_TNSALIAS : データベースに接続する際の
<client_home>/network/admin/tnsnames.ora別名
- IS_PDB : プラガブル・データベース (PDB) を使用する場合は`true`に設定
- PDB : PDB名 (該当する場合)
- PDB_SERVICE : 作成したサービスの名前PDBに対してサービス・リソースを作成する必要があります。

```
$ srvctl add service -d <db unique name> -pdb <pdb name> -service  
pdb_service -preferred "<instance name1>,<instance_name2>"  
$ srvctl start service -d <db unique name> -pdb <pdb name>
```

`dbfs_user`のパスワードを格納するためにOracle Walletを作成して使用する場合は、ウォレットの場所を含む`sqlnet.ora`ファイルを<client_home>/network/adminディレクトリに作成します。ウォレットを作成して`dbfs_user`のパスワードを格納する方法については、[Oracle Supportドキュメント1054431.1](#)を参照してください。

mount-dbfs.shスクリプトのサンプル

Oracle ClusterwareにDBFSを登録する際には、サンプルとして次のスクリプトを使用できます。必要に応じてパラメータを変更し、`/tmp/mount-dbfs.conf`にコピーしてください。

```
### This file provides configuration for mount-dbfs.sh.  
### Note 1054431.1 provides information about the setup required to use this  
script
```

```
#####
### Everyone must set these values
#####
### Database name for the DBFS repository as used in "srvctl status database -d
$DBNAME"
### If using PDB/, this should be set to the CDB
name DBNAME=<db unique name>

### Mount point where DBFS should be mounted
MOUNT_POINT=/mnt/dbfs

### Username of the DBFS repository owner in database $DBNAME
DBFS_USER=dbfs_user

### RDBMS ORACLE_HOME directory path → Here this should be client home
ORACLE_HOME=/u01/app/oracle/product/18.0.0.0/client

### GRID HOME directory path
##GRID_HOME=/u01/app/18.1.0.0/grid

### Syslog facility name (default user)
### Changed default from local3 to user for Solaris default support on 17-FEB-
2012
### This will allow us to log messages to the syslog
### (/var/log/messages on Linux, /var/adm/messages on Solaris)
LOGGER_FACILITY=user

### mount options for dbfs_client; these are used for both wallet and non-
wallet mounting
#MOUNT_OPTIONS=allow_other,direct_io
### if tracing is required, maybe consider parameter like the example below
### following example is commented out, only uncomment if directed by Oracle
Support
### for 11.2.0.3 and earlier, fix_control=32 added per bug 13340960 to allow
async statfs response. Use trace only for troubleshooting purpose.
#MOUNT_OPTIONS=allow_other,direct_io,fix_control=32,trace_level=1,trace_file=/t
mp/dbfs_client_trace.$$log,trace_size=100
### for 11.2.0.4 and later, remove fix control. Trace file only required while
troubleshooting.
MOUNT_OPTIONS=allow_other,direct_io,failover,nolock,spool_max=64,trace_level=1,
trace_file=/tmp/dbfs_client_trace.$$log,trace_size=100
### PERL_ALARM_TIMEOUT is number of seconds to wait for response from status
command.
### After this, if no response, the script will run clean.
### NOTE: If this is longer than the clusterware check interval, bad things may
happen.
###     Adjust the CHECK_INTERVAL to ensure it is at least 2x as long as
PERL_ALARM_TIMEOUT.
###     Example:
###     $ crsctl status res dbfs_mount -p|grep ^CHECK
###     CHECK_INTERVAL=30
###     $ crsctl modify res dbfs_mount -attr "CHECK_INTERVAL=32"
###     $ crsctl status res dbfs_mount -p|grep ^CHECK
###     CHECK_INTERVAL=32
PERL_ALARM_TIMEOUT=14
```

```
#####
### If using password-based authentication, set these
#####
### This is the plain text password for the DBFS_USER user
DBFS_PASSWD=xxxxxxx

### The file used to temporarily store the DBFS_PASSWD so dbfs_client can read
it
### This file is removed immediately after it is read by dbfs_client
### The actual filename used will have the PID appended to the name for
uniqueness
### This variable should be a full pathname including a directory and the first
part of a filename.
# DBFS_PWDFILE_BASE=/home/oracle/passwd.txt

#####
### If using wallet-based authentication, modify these
#####
### WALLET should be true if using a wallet, otherwise, false
WALLET=false

### TNS_ADMIN is the directory containing tnsnames.ora and sqlnet.ora used by
DBFS
TNS_ADMIN=/u01/app/oracle/product/18.0.0.0/client/network/admin

### TNS alias used for mounting with wallets
DBFS_LOCAL_TNSALIAS=dest

#####
### If using PDBs, modify these
#####
### Configure the following if this is a Pluggable Database (PDB)
### IS_PDB should be set to "true" or "false"
### PDB should be a TNS alias defined per instructions in Note 1054431.1.
### PDB_SERVICE should be a cluster database service defined per
### instructions in Note 1054431.1.
IS_PDB=true
PDB=pdb
PDB_SERVICE=dbfspdb
```

- C. 必要に応じてファイルを編集したら、mount-dbfs.sh（必要な場合は名前を変更）を、データベース・ノード上の適切なディレクトリ（GI_HOME/crs/script、または独自のディレクトリ）にコピーし、rootユーザーとして適切な権限を設定します。mount-dbfs.confについて以上のステップを繰り返し、/etc/oracleにコピーします。
- D. すべてのノードで、mount-dbfs.shファイルの権限/所有権を変更します。rootユーザーとして次のコマンドを実行します。

```
# dos2unix /tmp/mount-dbfs.sh
# dos2unix /tmp/mount-dbfs.conf
# cp -f /tmp/mount-dbfs.sh /home/oracle
# chown oracle: oinstall /home/oracle/mount-dbfs.sh
# chmod 750 /home/oracle/mount-dbfs.sh
# cp -f /tmp/mount-dbfs.conf /etc/oracle
```



```
# chown oracle:oinstall /etc/oracle/mount-dbfs.conf
# chmod 640 /etc/oracle/mount-dbfs.conf
```

E. mount-dbfs.sh ファイルを次のように変更します。

```
1> #SRVCTL=$ORACLE_HOME/bin/srvctl --> Comment this variable value

2> ### set the ORACLE_SID dynamically based on OCR info, if it is
running --> Comment out below line from this section

# if RAC, set SID with following command
# if single instance, set ORACLE_SID based on below command instead
## if [ -z "$ORACLE_SID" ]; then
##     export ORACLE_SID=$(($SRVCTL config db -d $DBNAME | $GREP
'instance' | $AWK '{print $3}'))
## fi
# logit info "ORACLE_SID is $ORACLE_SID"
### further checks required if we are using a PDB
### if the local instance isn't running (because ORACLE_SID is null)
don't bother
## if [ -n "$ORACLE_SID" -a "$IS_PDB" = 'true' ]; then
##     logit info "IS_PDB = true"
##     HOSTNAME=`$HN -s`
### by checking to see if the service is running on the local
instance,
### we can determine whether we should continue or not
##     PDB_SERVICE_STATE=$(($SRVCTL status service -d $CDB -s
$PDB_SERVICE | \
##         $GREP "$ORACLE_SID" )
### if the output from the last command is > 0, service is running
locally
## if [ -n "$PDB_SERVICE_STATE" ]; then
##     logit info "PDB service is online"
## else
##     logit error "PDB service is offline. Exiting"
##     exit 2
## fi
## fi

3> ### if there's no SID defined locally or it isn't running, stop -
-> Comment out following line from this section

## if [ -z "$ORACLE_SID" -a "$WALLET" = 'false' ]; then
##     logit error "No running ORACLE_SID available on this host,
exiting"
##     exit 2
## fi
```

Oracle Clusterwareにリソースを登録する際には、ローカル・リソースではなくクラスター・リソースとして作成します。クラスター・リソースの場合、ファイル・システムは一度に1つのノードにしかマウントできません。そのため、誤って同時ノードからDBFSをマウントしてしまい、同時ファイル書き込みが発生した結果ファイルが破損するという問題が起きるのを防ぐことができます。

- F. 次のように、スクリプトを追加してDBFSリソースを登録します。

```
##### start script add-dbfs-resource.sh
#!/bin/bash
add-dbfs-resource.sh
ACTION_SCRIPT=/home/oracle/mount-dbfs.sh
RESNAME=dbfs_mount
DBNAME=<db_unique_name>
DBNAMEL=`echo $DBNAME | tr A-Z a-z`
ORACLE_HOME=/u01/app/18.0.0.0/grid
PATH=$ORACLE_HOME/bin:$PATH
export PATH ORACLE_HOME crsctl
add resource $RESNAME \
-type cluster_resource \
-attr "ACTION_SCRIPT=$ACTION_SCRIPT, \
CHECK_INTERVAL=30,RESTART_ATTEMPTS=10, \
START_DEPENDENCIES='hard(ora.$DBNAMEL.db)pullup(ora.$DBNAMEL.db)',\
STOP_DEPENDENCIES='hard(ora.$DBNAMEL.db)',\
SCRIPT_TIMEOUT=300"
##### end script add-dbfs-resource.sh
```

- G. add-dbfs-resource.shスクリプトを実行して、DBFSをクラスターに登録します。1つのデータベース・サーバーで、oracleユーザーとしてスクリプトを実行します。

```
$ sh ./add-dbfs-resource.sh
```

- H. DBFSリソースを起動します。

```
$ <GRID_HOME>/bin/crsctl start resource dbfs_mount
```

注意：ファイルのロックはDBFSによって処理されます。詳細は、[『Oracle GoldenGate Microservices Architecture with Oracle RAC Configuration Best Practices』](#)のステップ1：Oracle RACでのファイル・システムの設定と、[Oracle Supportドキュメント1054431.1](#)を参照してください。

データベースの構成

この項では、Oracle GoldenGateのレプリケート環境にソースおよびターゲット・データベースに対する構成のベスト・プラクティスについて説明します。次の手順を実行するときは、[『Oracle GoldenGate Performance Best Practices』](#) ホワイトペーパーで推奨されている構成の手順に従ってください。

- `alter system set enable_goldengate_replication=true`を設定します。
- ARCHIVELOGモードでOracle GoldenGateソース・データベースを実行します
- Extractが実行中の場合は、FORCE LOGGINGモードを有効にします。
- Extractが実行中の場合は、最小限のサブリメンタル・ロギングを有効にします。また、レプリケートされたすべてのオブジェクトに対してスキーマレベルまたは表レベルのロギングを追加します。
- システム・グローバル領域（SGA）でストリーム・プールを構成します。
- Extractに、REDOログ読み取りパフォーマンス・パラメータを構成します。
- Oracle Streams Performance Advisor（UTL_SPADV）PL/SQLパッケージをインストールして、統合Extractと統合Replicatのパフォーマンスを監視します。

注意：FORCE LOGGINGモードとサブリメンタル・ロギングを有効にしている場合は、REDOログの生成が増えます。また、高速リカバリ領域の消費とアーカイブ・ログのバックアップ時間も増えます。

Oracle GoldenGateのデータベース・ユーザーも作成する必要があります。手順については、[「Oracle GoldenGate資格証明の構築」](#)を参照してください。

Oracle Database 11.2の場合の追加手順

1. パッチ20448066を[My Oracle Support](#)から入手し、適用します。詳細は、[Oracle Support ドキュメント2348973.1](#)を参照してください。
2019年2月の時点では、Exadata 11.2.0.4.180116 / 11.2.0.4.180116 and 11.2.0.4.180417 / 11.2.0.4.180417のパッチが利用できます。パッチ・レベルがこれより高い場合は、Oracle Supportへのサービス・リクエストを発行してください。手順については、[「サポートへの連絡」](#)を参照してください。
2. Database Vaultを無効にします。この手順は、Oracle Database Cloud ServiceのEnterprise Edition High PerformanceとEnterprise Edition Extreme Performanceにのみ適用されます。詳細は、[Oracle Support ドキュメント1513366.1](#)と[Oracle Support ドキュメント803948.1](#)を参照してください。
3. Oracle GoldenGateホーム・ディレクトリで、スクリプトを実行します。
事後スクリプトを実行する手順については、[Oracle Support ドキュメント2304095.1](#)を参照してください。

Oracle GoldenGateの構成

この項の手順に従って、Oracle GoldenGateをインストールし、構成します。

Oracle GoldenGateのインストール

1. [GoldenGateソフトウェア](#)をダウンロードし、システムにアップロードします。
2. Oracle GoldenGateソフトウェアを、Oracle GoldenGate構成の一部になるOracle RACのすべてのノードでローカルにインストールします。インストール・ディレクトリがすべてのノードで同じであることを確認してください。

以下は、サイレント・インストールのレスポンス・ファイルの例です。インストール・ユーザーとしてOSのoracleユーザーを使用し、インストール・ディレクトリ（たとえば /u01/app/oracle/product/ogg181）を作成します。

次のインストール・コマンドを実行します。

```
[ssh into DB System as opc user]
sudo su - oracle
cd <directory installation media is
uploaded>/fbo_ggs_Linux_x64_services_shiphome/Disk1
./runInstaller -silent -nowait -responseFile <path of response file>
```

Oracle Database 12cの場合のレスポンス・ファイルの例は、次のとおりです。インストール・メディアによるレスポンス・ファイルのパスは、<directory installation media is uploaded>/fbo_ggs_Linux_x64_services_shiphome/Disk1/response/oggcore.rspです。

INSTALL_OPTIONとSOFTWARE_LOCATIONのパラメータを適宜編集します。

```
#####
## Copyright(c) Oracle Corporation 2018. All rights reserved.      ##
##                                                                    ##
## Specify values for the variables listed below to customize      ##
## your installation.                                              ##
##                                                                    ##
## Each variable is associated with a comment. The comment        ##
## can help to populate the variables with the appropriate        ##
## values.                                                         ##
##                                                                    ##
## IMPORTANT NOTE: This file should be secured to have read      ##
## permission only by the oracle user or an administrator who    ##
## own this installation to protect any sensitive input values.   ##
##                                                                    ##
#####

#-----
#-----
```

```

# Do not change the following system generated value.
#-----
-----
oracle.install.responseFileVersion=
/oracle/install/rspfmt_ogginstall_response_schema_v12_1_2

#####
#####
##
##
## Oracle GoldenGate installation option and details
##
##
##
#####
#####

#-----
-----
# Specify the installation option.
# Specify ORA18c for installing Oracle GoldenGate for Oracle Database 18c
or
#          ORA12c for installing Oracle GoldenGate for Oracle Database 12c
or
#          ORA11g for installing Oracle GoldenGate for Oracle Database 11g
#-----
-----
INSTALL_OPTION=<version>

#-----
-----
# Specify a location to install Oracle GoldenGate
#-----
-----
SOFTWARE_LOCATION=<installation location>

#####
#####
##
##
## Specify details to Create inventory for Oracle installs
##
## Required only for the first Oracle product install on a system.
##
##
##
#####
#####

```

```
#-----
# Specify the location which holds the install inventory files.
# This is an optional parameter if installing on
# Windows based Operating System.
#-----
INVENTORY_LOCATION=/u01/app/oraInventory
#-----
# Unix group to be set for the inventory directory.
# This parameter is not applicable if installing on
# Windows based Operating System.
#-----
UNIX_GROUP_NAME=oinstall
```

Oracle GoldenGateの構成

Oracle GoldenGateソフトウェアをインストールしたら、Oracle GoldenGate Configuration Assistant（OGGCA）を使用してデプロイメントを作成します。

以下は、Oracle GoldenGateのデプロイメントを作成する際の推奨事項です。

- 新しいサービス・マネージャを作成する際は、次の値を指定します。
 - サービス・マネージャのACFS用のデプロイメント・ホームの場所 :
/mnt/acfs_gg/deployments/ggsm01
 - サービス・マネージャのDBFS用のデプロイメント・ホームの場所 :
/mnt/dbfs/gg/oggsm
 - リスニング・ホスト名/アドレスのフィールドにはlocalhostを指定します。
localhostを使用すると、仮想IPアドレス（VIP）を必要とせずにOracle RACの全ノードでデプロイメントを開始することができます。
 - XAGとの統合を選択します。
- 共有のDBFSまたはACFSファイル・システムで、デプロイメント・ホームのディレクトリを指定します。
 - ACFS : /mnt/acfs_gg/deployments/gg01
 - DBFS : /mnt/dbfs/gg/deployments

サイレント構成の場合は、次のサンプル・ファイルをコピーし、oracleユーザーがアクセスできる任意の場所に貼り付けます。次の値を適宜編集します。

- ADMINISTRATOR_PASSWORD
- SERVICEMANAGER_DEPLOYMENT_HOME

- OGG_SOFTWARE_HOME
- OGG_DEPLOYMENT_HOME
- ENV_ORACLE_HOME（前の手順でインストールしたOracle Clientのホーム・ディレクトリを指定）
- ENV_LD_LIBRARY_PATH（Oracle Clientホーム・ディレクトリ下）
- ENV_TNS_ADMIN（Oracle Clientホーム・ディレクトリ下）

サンプルのレスポンス・ファイル

```
#####
## Copyright(c) Oracle Corporation 2016, 2018. All rights reserved.      ##
##                                                                    ##
## Specify values for the variables listed below to customize your      ##
## installation.                                                         ##
##                                                                    ##
## Each variable is associated with a comment. The comments can help to  ##
## populate the variables with the appropriate values.                  ##
##                                                                    ##
## IMPORTANT NOTE: This file should be secured to have read permission  ##
## by the Oracle user or an administrator who owns this configuration to ##
## protect any sensitive input values.                                   ##
##                                                                    ##
#####

#-----
# Do not change the following system generated value.
#-----
oracle.install.responseFileVersion=/oracle/install/rspfmt_oggca_response_schema_
v18_1_0

#####
##                                                                    ##
## Oracle GoldenGate deployment configuration options and details        ##
##                                                                    ##
#####

#####
##                                                                    ##
## Instructions to fill out this response file                          ##
## -----                                                                ##
## Fill out section A, B, and C for general deployment information      ##
## Additionally:                                                         ##
## Fill out sections D, E, F, G, H, I, and J for adding a deployment    ##
## Fill out section K for removing a deployment                        ##
##                                                                    ##
#####
```

```

#####
#
#                               SECTION A - GENERAL                               #
#
#####

#-----
# Specify the configuration option.
# Specify:
# - ADD      : for adding a new GoldenGate deployment.
# - MODIFY   : for modifying an existing GoldenGate deployment.
# - REMOVE   : for removing an existing GoldenGate deployment.
#-----
CONFIGURATION_OPTION=ADD

#-----
# Specify the name for the new or existing deployment.
#-----
DEPLOYMENT_NAME=ogg_deployment

#####
#
#                               SECTION B - ADMINISTRATOR ACCOUNT                               #
#
#####

#-----
# Specify the administrator account username.
#-----
ADMINISTRATOR_USER=oggadmin

#-----
# Specify the administrator account password.
#-----
ADMINISTRATOR_PASSWORD=<password for oggadmin>

#####
#
#                               SECTION C - SERVICE MANAGER                               #
#
#####

#-----
# Specify the location for the Service Manager deployment.
# This is only needed if the Service Manager deployment doesn't exist already.
#-----
SERVICEMANAGER_DEPLOYMENT_HOME=<location for service manager>

```



```

#-----
# Specify the host for the Service Manager.
#-----
HOST_SERVICEMANAGER=localhost

#-----
# Specify the port for the Service Manager.
#-----
PORT_SERVICEMANAGER=9100

#-----
# Specify if SSL / TLS is or will be enabled for the deployment.
# Specify true if SSL / TLS is or will be enabled, false otherwise.
#-----
SECURITY_ENABLED=false

#-----
# This option is only needed when CONFIGURATION_OPTION is ADD
#
# Specify if a new Service Manager should be created.
# Specify true if a new Service Manager should be created, false otherwise.
#-----
CREATE_NEW_SERVICEMANAGER=true

#-----
# This option is only needed when CONFIGURATION_OPTION is ADD
# This option does not apply to Windows platform
#
# Specify if Service Manager should be registered as a service/daemon. This
option is mutually exclusive with the 'INTEGRATE_SERVICEMANAGER_WITH_XAG'
option.
# Specify true if Service Manager should be registered as a service, false
otherwise.
#-----
REGISTER_SERVICEMANAGER_AS_A_SERVICE=false

#-----
# This option is only needed when CONFIGURATION_OPTION is ADD
#
# Specify if Service Manager should be integrated with XAG. This option is
mutually exclusive with the 'REGISTER_SERVICEMANAGER_AS_A_SERVICE' option.
# Specify true if Service Manager should be integrated with XAG, false
otherwise.
#-----
INTEGRATE_SERVICEMANAGER_WITH_XAG=true

#-----
# This option is only needed when CONFIGURATION_OPTION is ADD
#
# If using an existing Service Manager, specify if it is integrated with XAG.

```

```

# Specify true if the existing Service Manager is integrated with XAG, false
otherwise.
#-----
EXISTING_SERVICEMANAGER_IS_XAG_ENABLED=false

#####
#
# SECTION D - SOFTWARE HOME
#
#####

#-----
# Specify the existing OGG software home location.
#-----
OGG_SOFTWARE_HOME=<GoldenGate home directory>

#####
#
# SECTION E - DEPLOYMENT DIRECTORIES
#
#####

#-----
# Specify the location of the new or existing OGG deployment.
#-----
OGG_DEPLOYMENT_HOME=<GoldenGate deployment home>

#-----
# Specify the location for OGG_ETC_HOME.
#-----
OGG_ETC_HOME=

#-----
# Specify the location for OGG_CONF_HOME.
#-----
OGG_CONF_HOME=

#-----
# Specify the location for OGG_SSL_HOME.
#-----
OGG_SSL_HOME=

#-----
# Specify the location for OGG_VAR_HOME.
#-----
OGG_VAR_HOME=

#-----
# Specify the location for OGG_DATA_HOME.
#-----
OGG_DATA_HOME=

```

```

#####
#
#           SECTION F - ENVIRONMENT VARIABLES           #
#
#####

#-----
# Specify the value for the ORACLE_HOME environment variable.
#-----
ENV_ORACLE_HOME=<Oracle Client home>

#-----
# Specify the value for the LD_LIBRARY_PATH environment variable.
#-----
ENV_LD_LIBRARY_PATH==<Oracle Client home>/lib

#-----
# Specify the value for the TNS_ADMIN environment variable.
#-----
ENV_TNS_ADMIN=<Oracle Client home>/network/admin

#-----
# Specify the value for the ORACLE_SID environment variable.
#-----
ENV_ORACLE_SID=

#-----
# This option is only needed when Sharding will be enabled.
# Specify the value for the STREAMS_POOL_SIZE environment variable.
#-----
ENV_STREAMS_POOL_SIZE=

#-----
# Specify any additional environment variables.
#-----
ENV_USER_VARS=

#####
#
#           SECTION G - SECURITY           #
#           This section is only needed if Security will be enabled           #
#
#####

#-----
# Specify the Cipher Suites to be used for SSL.
#-----

```

```

CIPHER_SUITES=TLS_RSA_WITH_AES_256_CBC_SHA,TLS_RSA_WITH_AES_128_CBC_SHA,SSL_RSA_
WITH_3DES_EDE_CBC_SHA,
SSL_RSA_WITH_RC4_128_MD5,SSL_RSA_WITH_RC4_128_SHA

#-----
# If SSL / TLS will be enabled, specify the server Wallet to be copied.
#-----
SERVER_WALLET=

#-----
# If SSL / TLS will be enabled, specify the Oracle GoldenGate server certificate
# to be imported.
#-----
SERVER_CERTIFICATE=

#-----
# If the Oracle GoldenGate server certificate is encrypted, specify the key file
#-----
SERVER_CERTIFICATE_KEY_FILE=

#-----
# If the Oracle GoldenGate server certificate is encrypted, specify the key file
# password
#-----
SERVER_CERTIFICATE_KEY_FILE_PWD=

#-----
# If SSL / TLS will be enabled, specify the client Wallet to be copied.
(Optional)
#-----
CLIENT_WALLET=

#-----
# If SSL / TLS will be enabled, specify the Oracle GoldenGate client certificate
# to be imported. (Optional)
#-----
CLIENT_CERTIFICATE=

#-----
# If the Oracle GoldenGate client certificate is encrypted, specify the key file
#-----
CLIENT_CERTIFICATE_KEY_FILE=

#-----
# If the Oracle GoldenGate server certificate is encrypted, specify the key file
# password
#-----
CLIENT_CERTIFICATE_KEY_FILE_PWD=

#####

```

```

#                                                                 #
#              SECTION H - SHARDING                               #
#                                                                 #
#####

#-----
# Specify if Sharding will be enabled for this deployment.
# Specify true if Sharding will be enabled for this deployment, false otherwise.
#-----
SHARDING_ENABLED=false

#-----
# This option is only needed when Sharding will be enabled.
# Specify the Sharding user.
#-----
SHARDING_USER=

#####
#                                                                 #
#              SECTION I - SERVICES                               #
#                                                                 #
#####

#-----
# Specify the port for Administration Server.
#-----
PORT_ADMINSRVR=9101

#-----
# Specify the port for Distribution Server.
#-----
PORT_DISTSRVR=9102

#-----
# If security is disabled, specify if this non-secure deployment will be used
# to send trail data to a secure deployment.
#-----
NON_SECURE_DISTSRVR_CONNECTS_TO_SECURE_RCVRSRVR=false

#-----
# Specify the port for Receiver Server.
#-----
PORT_RCVRSRVR=9103

#-----
# Specify if Performance Metrics server will be enabled.
# Specify true if Performance Metrics server will be enabled, false otherwise.
#-----
METRICS_SERVER_ENABLED=true

```

```

#-----
# Specify if Performance Metrics server is a critical service.
# Specify true if Performance Metrics server is a critical service, false
otherwise.
# This is optional and only takes effect when Performance Metrics server will be
enabled.
# Also, this option should only be set when the Service Manager is integrated
with XAG.
# The default value is false.
#-----
METRICS_SERVER_IS_CRITICAL=true

#-----
# This option is only needed when Performance Metrics server will be enabled.
# Specify the port for Performance Metrics server (TCP).
#-----
PORT_PMSRVR=9104

#-----
# This option is only needed when Performance Metrics server will be enabled.
# Specify the port for Performance Metrics server (UDP).
#-----
UDP_PORT_PMSRVR=9105

#-----
# This option is only needed when Performance Metrics server will be enabled.
# Specify the DataStore type for Performance Metrics server.
# Valid values are: BDB, LMDB
#-----
PMSRVR_DATASTORE_TYPE=BDB

#-----
# Specify the DataStore home location for Performance Metrics server.
# This is optional and only takes effect when Performance Metrics server will be
enabled.
#-----
PMSRVR_DATASTORE_HOME=

#####
#
#                               SECTION J - REPLICATION OPTIONS
#
#####

#-----
# Specify the value for the GoldenGate schema.
#-----
OGG_SCHEMA=c##ggadmin

```

```
#####
#
#          SECTION K - REMOVE DEPLOYMENT OPTIONS          #
#
#####

#-----
# Specify if the deployment files should be removed from disk.
# Specify true if the deployment files should be removed, false otherwise.
#-----
REMOVE_DEPLOYMENT_FROM_DISK=
```

次のコマンドで、サイレント構成を実行します。

```
[ssh into DB System as opc user]
sudo su - oracle
export OGG_HOME=/u01/app/oracle/product/ogg181
export TNS_ADMIN=$ORACLE_HOME/network/admin
export PATH=$OGG_HOME/bin:$ORACLE_HOME/bin:$PATH
export OGG_BASE_DIR=/mnt/acfs_gg
export OGG_DEPLOYMENT=$OGG_BASE_DIR/deployments/gg01
export JAVA_HOME=$OGG_HOME/jdk
oggca.sh -silent -responseFile <path of response file>
```

詳細は、[「Oracle RACを使用したOracle GoldenGate Microservices Architectureの構成のベスト・プラクティス」](#)の「ステップ4：Oracle GoldenGateデプロイメントの作成」を参照してください。

TNS別名の作成

DBシステムの全ノードのtnsnames.oraファイルに、CDBまたはCDB以外のローカル・インスタンスに接続するためのTNS別名を追加します。tnsnames.oraファイルは<client_home>/network/admin/ディレクトリにあります（client_homeは、前の手順でインストールしたOracle Clientのインストール場所）。

```
<Alias name> =
(DESCRIPTION =
  (SDU=2097152)
  (ADDRESS = (PROTOCOL = TCP) (HOST =<hostname>) (PORT = 1521))
  (CONNECT_DATA =
    (SERVER = DEDICATED)
    (SERVICE_NAME =<db_unique_name>.<domain>)
  )
)
```

Oracle Databaseリリース12c以上の場合は、PDBにもTNS別名を作成する必要があります。以下は、ローカルPDBインスタンスに接続するためのTNS別名の例です。

```
<Alias name> =
(DESCRIPTION =
  (SDU=2097152)
  (ADDRESS = (PROTOCOL = TCP) (HOST =<hostname>) (PORT = 1521))
  (CONNECT_DATA =
```

```
(SERVER = DEDICATED)
(SERVICE_NAME = oggpdbserv.<domain>)
)
```

詳細は、[「Oracle RACを使用したOracle GoldenGate Microservices Architectureの構成のベスト・プラクティス」](#)の「ステップ5：Oracle GoldenGateデータベース接続のためのCreate Oracle Net 別名」を参照してください。

Oracle GoldenGateのHA用の構成

HAを構成するには、[「Oracle RACを使用したOracle GoldenGate Microservices Architectureの構成のベスト・プラクティス」](#)の「ステップ6：Oracle Clusterware構成」と、この項に示したOracle Cloud Infrastructure固有のトピックを参照してください。

Oracle Grid Infrastructure Standalone Agentのインストール

[Oracle Grid Infrastructure Standalone Agentソフトウェア](#)をインストールします。Oracle GoldenGate Microservices Architectureには、Oracle Grid Infrastructure Agent (XAG) バージョン9以上が必要です。

ソフトウェア・パッチとの競合を避けるために、XAGソフトウェアはOracle Grid Infrastructure ホーム・ディレクトリ以外の場所にインストールする必要があります。次のコマンドを使用すると、XAGを各ノードに別々にインストールできます。ORACLE_BASE変数を設定すれば、agctl configコマンドに対するエラーが回避されます。

```
[ssh into DB System as opc user]
sudo mkdir /u01/app/xag
sudo chown grid:oinstall /u01/app/xag
sudo su - grid
cd <xag media directory>
export ORACLE_BASE=/u01/app
./xagsetup.sh --install --directory /u01/app/xag
```

PATH環境変数の更新

ユーザーがマシンにログオンしたときにagctlの場所が認識されるように、新しくインストールしたXAGソフトウェアの場所をoracleユーザーのPATH環境変数に追加します。

/home/oracle/.bashrcファイルに次の行を追加します。

```
export PATH=/u01/app/xag/bin:$PATH
```

詳細は、[『Oracle GoldenGate Microservices Architecture with Oracle RAC Configuration Best Practices』](#)の「ステップ6：Oracle Clusterware構成」のステップ2を参照してください。

アプリケーションの仮想IPアドレスの作成

Oracle GoldenGateのアプリケーションの仮想IPアドレス（VIP）として、VCNにフローティングIPアドレスを追加する必要があります。フローティングIPアドレスを使用すると、各クラスター・ノードのOracle GoldenGateに、同じIPアドレスを使用してアクセスできます。次の例では、フローティングIPの作成に、Oracle Cloud Infrastructure APIコールを使用しています。各APIリクエストには、SSH秘密鍵が割り当てられ、Oracleは公開鍵を使用してそのリクエストの信憑性を確認します。前提条件として、API割当て鍵を作成し、リクエストを実行する特定のユーザーにアップロードする必要があります。[API認証を設定する手順](#)は、ドキュメントに記載されています。このタスクは、データベース・サーバーではなくクライアント・マシンから実行する必要があります。

1. APIをコールするマシンでUNIXシェル関数を定義します。

具体的には、DB システムでフローティングIPを管理するリクエストを簡単にコールできるように、oci-curl関数を作成します。詳細は、[「リクエストの署名」](#)を参照してください。

https://docs.cloud.oracle.com/iaas/Content/Resources/Assets/signing_sample_bash.txtにあるスクリプトをコピーし、oci-curl.shとして保存すれば、oci-curl関数を作成するスクリプトを作成できます。

```
curl -X GET
https://docs.cloud.oracle.com/iaas/Content/Resources/Assets/signing_sample
_bash.txt >> oci-curl.sh
```

関数を作成する前に、次の値を編集します。

- tenancyId : Webコンソールで確認できるテナンシのOCID（Oracle Cloud Identifierで呼び出される、オラクルが署名した一意のID）
- authUserId : Webコンソールで確認できるユーザーOCID
- keyFingerprint : ユーザーにAPI鍵を追加すると、Webコンソールでフィンガープリントを確認できます。
- privateKeyPath : クライアント上のAPI秘密鍵のファイル・パス

例 :

```
$cat oci-curl.sh
.
.
    local tenancyId="ocidl.tenancy.oc1..exampleuniqueID";
    local authUserId="ocidl.user.oc1..exampleuniqueID";
    local
keyFingerprint="6c:3e:67:5c:2e:a2:dd:ea:7e:14:f8:07:f8:f7:53:03";
    local privateKeyPath="/Users/ygoto/keys/ygoto_api_key.pem";
```

oci-curl.shコマンドをソーシングして、oci-curl関数を定義します。

```
$ . oci-curl.sh
or
$ source oci-curl.sh
```

typeset -Fコマンドを使用して、oci-curl関数が現在のシェルに定義されていることを確認します。

```
# typeset -F |grep oci-curl
declare -f      oci-curl
```

2. クライアント・ネットワークのVNIC IDを問い合わせます。

フローティングIPが、システムのクライアント・ネットワークに付加されます。VNIC IDは、次の例のように、DBシステムのインスタンス・メタデータを問い合わせて確認できます。

```
# curl -s "http://169.254.169.254/opc/v1/vnics/0/vnicId"
ocidl.vnic.oc1.uk-london-
1.abwgiljtopawtklo3tgiqv7cekhwokpalsrww2bgv6w4mzchfbf22kqrr5vq
```

3. フローティングIPを作成します。

A. プライベートIPを作成するJSONリクエスト本文を作成します ([APIドキュメントを参照](#))。サブネットのどこでもフローティングIPとして使用されていないIPアドレスを指定します。以下はリクエスト本文の例です。

```
{"displayName": "<your own display name>", "hostnameLabel": "<your own display name >", "ipAddress": "<IP address for floating IP>", "vnicId": "<vnicId you confirmed in preceding step>"}
```

例：

```
$cat createVip.json
{"displayName": "ggappvip2", "hostnameLabel": "ggappvip2", "ipAddress": "10.0.1.32",
"vnicId": "ocidl.vnic.oc1.iad.exampleuniqueID"}
```

B. 次のoci-curlコマンドを実行してフローティングIPを作成します。

```
oci-curl <iaas host> post ./createVip.json "/20160918/privateIps" -k
```

IaaSホストを確認するには、[APIリファレンスおよびエンドポイント](#)でコア・サービスAPIを参照してください。

例：

```
$ oci-curl iaas.us-ashburn-1.oraclecloud.com post ./createVip.json
"/20160918/privateIps" -k
.
.
```

```
{
  "availabilityDomain" : "KDFA:US-ASHBURN-AD-2",
  "compartmentId" : "ocidl.compartment.oc1..exampleuniqueID",
  "displayName" : "ggappvip2",
  "hostnameLabel" : "ggappvip2",
  "id" : "ocidl.privateip.oc1.iad.exampleuniqueID",
  "ipAddress" : "10.0.1.32",
  "isPrimary" : false,
  "subnetId" : "ocidl.subnet.oc1.iad.exampleuniqueID",
  "timeCreated" : "2017-09-19T02:57:01.869Z",
  "vnicId" : "ocidl.vnic.oc1.iad.exampleuniqueID"
}
```

4. VIPのリストを確認します。

作成したフローティングIPは、次のコマンドを使用してVIPのリストで確認できます。

```
oci-curl <iaas host> get "/20160918/privateIps?vnicId=<vnid>" -k |grep
ipAdd
```

例 :

```
$ oci-curl iaas.us-ashburn-1.oraclecloud.com get
"/20160918/privateIps?vnicId=ocidl.vnic.oc1.iad.exampleuniqueID" -k |grep
ipAdd
"ipAddress" : "10.0.1.32",
"ipAddress" : "10.0.1.27",
"ipAddress" : "10.0.1.23",
"ipAddress" : "10.0.1.25"
```

アプリケーション・サービスの作成

次のように、Oracle GoldenGateサービスと統合されるアプリケーション・サービスを作成します。

```
[ssh into DB system as opc user]
sudo su - oracle
srvctl add service -d $ORACLE_UNQNAME -service oggserv -role PRIMARY -preferred
<preferred instance name> -available <available instance name>
srvctl start service -d $ORACLE_UNQNAME -service oggserv
```

Oracle Database 12c以上の場合は、PDBのサービスを作成します。

```
[ssh into DB system as opc user]
sudo su - oracle
srvctl add service -d $ORACLE_UNQNAME -service oggpdbserv -role PRIMARY -
preferred <preferred instance name> -available <available instance name> -pdb
<pdb name>
srvctl start service -d $ORACLE_UNQNAME -service oggpdbserv
```

詳細は、[『Oracle GoldenGate Microservices Architecture with Oracle RAC Configuration Best Practices』](#)の「ステップ6：Oracle Clusterware構成」のステップ4を参照してください。

Oracle Grid Infrastructure Agentの構成

Oracle GoldenGateサービスの高可用性を実装するには、次のようにOracle GoldenGateをOracle Grid Infrastructure Agent（XAG）に統合します。

```
[ssh into DB system as opc user]
sudo su -
cd /u01/app/xag/bin
./agctl add goldengate ggscal --gg_home <GoldenGate Home directory> --
service_manager --config_home <GoldenGate Service Manager Deployment Home
directory>/etc/conf --var_home <GoldenGate Service Manager Deployment Home
directory>/var --port 9100 --adminuser oggadmin --user oracle --group oinstall -
-network 1 --ip <floating ip address> --filesystems ora.data.acfs_gg.acfs --
db_services ora.<ORACLE_UNQNAME>.<service name>.svc --use_local_services

(You will be required to enter password of admin user)

./agctl start goldengate ggscal --node <hostname of RAC node1>
```

詳細は、[『Oracle GoldenGate Microservices Architecture with Oracle RAC Configuration Best Practices』](#)の「ステップ6：Oracle Clusterware構成」のステップ5を参照してください。

リバース・プロキシとHAの構成

Nginxを使用して、データベース・サーバーとHAの両方にリバース・プロキシを構成します。リバース・プロキシを構成すると、Oracle GoldenGate Microservices Architectureのデプロイメントに伴う各種マイクロサービスのコンタクトを1点にまとめることができます。リバース・プロキシはオプションですが、マイクロサービスに簡単にアクセスし、セキュリティを強化するために、構成することをお勧めします。

Nginxのインストール

1. Oracle GoldenGateが稼働しているホストにログオンし、yumを使用して、Nginxをインストールするリポジトリの場所を設定します。

```
[ssh into DB system as opc user]
sudo su -
cat <<EOF > /etc/yum.repos.d/nginx.repo
[nginx]
name=nginx repo
baseurl=http://nginx.org/packages/rhel/6/\$basearch/
gpgcheck=0
enabled=1
EOF
```

2. Nginxをインストールします。

```
[ssh into DB system as opc user]
sudo su -
yum install -y nginx
```

Nginxベースのリバース・プロキシの構成

1. Oracle GoldenGateにNginx構成ファイルを作成します。

```
[ssh into DB system as opc user]
sudo su - oracle
export JAVA_HOME=<GoldenGate home>/jdk
<GoldenGate home>/lib/utl/reverseproxy/ReverseProxySettings -u oggadmin -P
<password of oggadmin> -o ogg.conf http://localhost:9100
```

2. Nginx構成ファイルをコピーします。ターゲット・ディレクトリには他のファイルがないようにしてください。

```
[ssh into DB system as opc user]
sudo cp /home/oracle/ogg.conf /etc/nginx/conf.d/
```

3. 証明書を作成します。

```
[ssh into DB system as opc user]
sudo sh /etc/ssl/certs/make-dummy-cert /etc/nginx/ogg.pem
```

4. Nginx構成をテストします。

```
[ssh into DB system as opc user]
sudo nginx -t
```

5. エラーがなければ、check configを無効にしてNginxをClusterwareにと統合します。手順は後述します。

```
[ssh into DB system as opc user]
sudo chkconfig nginx off
```

6. XAGサービスをOracle RACノード2に複製し、そのノードにNginxを設定します。

```
[ssh into DB system as opc user]
sudo su -
cd /u01/app/xag/bin
./agctl relocate goldengate ggscal
```

Oracle RACノード2で「Nginxのインストール」と「Nginxベースのリバース・プロキシの構成」の手順を繰り返します。詳細は、Oracle GoldenGateのドキュメントで[「リバース・プロキシのサポート」](#)を参照してください。

ClusterwareおよびXAGとのNginxの統合

ClusterwareおよびXAGにNginxを統合し、統合後の高可用性をOracle GoldenGateサービスに統合します。

1. Oracle Grid Infrastructureのホーム・ディレクトリを確認します。

```
[ssh into DB system as opc user]
sudo su - grid
echo $ORACLE_HOME
```

2. ClusterwareソフトウェアとしてNginxを追加します。

```
[ssh into DB system as opc user]
sudo su -
cd <Grid Infrastructure home directory>/bin
./crsctl add resource nginx -type generic_application -attr
"EXECUTABLE_NAMES=named,START_PROGRAM='/etc/rc.d/init.d/nginx
start',STOP_PROGRAM='/etc/rc.d/init.d/nginx
stop',CHECK_PROGRAMS='/etc/rc.d/init.d/nginx status'"
./crsctl setperm resource nginx -u user:oracle:r-x
```

3. XAG GoldenGateサービスにNginxを追加します。

```
[ssh into DB system as opc user]
sudo su -
cd /u01/app/xag/bin/
./agctl modify goldengate ggscal --filesystems ora.data.acfs_gg.acfs,nginx
--db_services ora.<ORACLE_UNQNAME>.<service name>.svc --use_local_services
./agctl stop goldengate ggscal
./agctl start goldengate ggscal --node <hostname of RAC node1>
```

GoldenGate Service Managerには、<https://<public IP address>>からアクセスできます。

データ・レプリケーションの構成

Oracle GoldenGateのExtract、Replicat、およびPathプロセスを作成して、データ・レプリケーションを構成します。詳細は、次のリソースを参照してください。

- **一般的な構成** : Oracle GoldenGateのドキュメントで次のトピックを参照してください。
 - [データ・レプリケーションの操作](#)
 - [パスの操作](#)
 - [トレイルの操作](#)

- **HAのための追加設定：** [「Oracle RACを使用したOracle GoldenGate Microservices Architectureの構成のベスト・プラクティス」](#)で、次の項を参照してください。
 - ステップ7：共有ファイル・システムを使用するためのOracle GoldenGateプロセスの構成
 - ステップ8：ExtractおよびReplicatプロセスの自動起動の構成

Ngixリバース・プロキシによるOracle GoldenGateパス・プロセスの構成

この項では、Ngixリバース・プロキシを使用してターゲット・システムにデータを転送するOracle GoldenGateパス・プロセスの例を示します。

1. 分散サーバーでBasic認証の資格証明を作成します。管理サーバーにログインし、右側のメニューで「**構成**」をクリックします。次の値で資格証明を追加します。

The screenshot shows a web form for adding a credential. It contains the following fields and values:

Field	Value
Credential Domain: ?	distsvr
* Credential Alias: ?	dist1
* User ID: ?	oggadmin
* Password:	
* Verify Password:	

At the bottom of the form are two buttons: "Cancel" and "Submit".

2. パス・プロセスを作成し、Ngix固有の設定を入力します。分散サーバーにログインして[+]をクリックします。次のように値を入力し、「**作成して実行**」をクリックします。

注意： ターゲット側にはフローティングIPを指定します。

* Path Name: ?

path1

Description: ?

* Source: ?

EXT1

t1

Generated Source URI:

trail://132.145.52.107/services/ogg_deployment/distsvr/v2/sources?trail=t1

Reverse proxy enabled? ?

☒

wss

10.0.2.111

443

r1

ogg_deployment

/services/ogg_deployment/v2

distsvr

dist1

* Target: ?

Use Basic Authentication: ?

☒

Generated Target URI:

wss://distsvr+dist1@10.0.2.111:443/services/ogg_deployment/v2/targets?trail=r1

Sequence Length: ?

9

Trail Size (MB): ?

500

▼

▲

Configure Trail Format:

☐

Begin: ?

Position in Log

Source Sequence Number:

0

Source RBA Offset:

0

Critical: ?

☐

Auto Restart: ?

☒

Auto Restart Options

Retries: ?

10

Delay: ?

2

Rule-set Configuration

Filtering ?

☐

► More Options

Cancel

Create Path

Create and Run

まとめ

このホワイトペーパーでは、Oracle Cloud Infrastructure上でOracle GoldenGateを構成する際の手順とベスト・プラクティスについてお伝えしました。特に、以下のベスト・プラクティスについて検討してください。

- パフォーマンス強化のために、Oracle GoldenGateリリース18.1以上をOracle Enterprise Editionで使用する。
- Nginxの各データベースで、VCNセキュリティ・リストとファイアウォールにポート443を開く。
- XAGの設定で、`oci-curl`を使用してフローティングIPアドレスを作成する。
- ファイル・システムの設定で、デフォルトのオプションとしてACFSを使用する。DBFSは、DBシステムにOracle Data Guardが構成されている場合にのみ使用する。
- [「Oracle RACを使用したOracle GoldenGate Microservices Architectureの構成のベスト・プラクティス」](#)に記載されている手順に違い、Oracle GoldenGate、共有ファイル・システム（ACFSまたはDBFS）、およびXAGを設定する。
- 安全なデプロイメントとユーザー・エクスペリエンス向上のために、リバース・プロキシとしてNginxを使用する。
- HAのために、NginxをClusterwareおよびXAGに統合する。



Oracle Corporation, World Headquarters

500 Oracle Parkway
Redwood Shores, CA 94065, USA

Worldwide Inquiries

Phone: +1.650.506.7000
Fax: +1.650.506.7200

CONNECT WITH US



blogs.oracle.com/oracle



facebook.com/oracle



twitter.com/oracle



oracle.com

Integrated Cloud Applications & Platform Services

Copyright © 2019, 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、記載内容は予告なく変更されることがあります。本文書は一切間違いがないことを保証するものではなく、さらに、口述による明示または法律による黙示を問わず、特定の目的に対する商品性もしくは適合性についての黙示的な保証を含み、いかなる他の保証や条件も提供するものではありません。オラクル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はオラクルおよびその関連会社の登録商標です。その他の社名、商品名等は各社の商標または登録商標である場合があります。

Intel、Intel Xeonは、Intel Corporationの商標または登録商標です。すべてのSPARCの商標はライセンスをもとに使用し、SPARC International, Inc.の商標または登録商標です。AMD、Opteron、AMDロゴ、AMD Opteronロゴは、Advanced Micro Devices, Inc.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

Oracle Cloud Infrastructure上のOracle GoldenGateマイクロサービス・アーキテクチャ

2019年4月
著者：Oracle Corporation



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