

# Oracle Data GuardとOracle GoldenGateによる透過的なロール移行

Oracleホワイト・ペーパー | 2015年2月

はじめに .....	3
Oracle GoldenGateとOracle Active Data Guardを使用したMAAアーキテクチャの例 .....	5
Oracle GoldenGate共有ファイル用のOracle DBFSの構成 .....	7
Oracle GoldenGateの統合キャプチャ .....	8
データ損失を考慮したOracle Data Guardのフェイルオーバー .....	9
Oracle GoldenGate Data Pump .....	11
Oracle GoldenGate Replicat .....	11
データ損失を考慮したOracle Data Guardのフェイルオーバー .....	12
Oracle GoldenGate Manager .....	12
Oracle Cluster Resource (Oracle CRS) の構成 .....	12
Oracle GoldenGateに適したOracle Clusterwareのロールベース・サービスの作成 .....	13
Oracle DBFSリソースの作成 .....	14
Oracle GoldenGateアプリケーションの仮想IP (VIP) アドレスの作成 (Oracle RAC使用時) .....	15
Oracle Grid Infrastructure Agentの構成 .....	15
結論 .....	18
付録：Oracle Grid Infrastructure Agentを使用しないOracle GoldenGateの構成 .....	19
Oracle DBFSの自動マウント/アンマウントのためのデータベース構成 .....	19
Oracle GoldenGate Data Pumpの構成とロール移行 .....	24

## はじめに

計画外停止とすべての計画保守作業の両方で、停止時間がゼロまたはほぼゼロになる最高レベルの可用性を実現するために、Oracle Active Data Guard (Oracle ADG) と Oracle GoldenGateの組み合わせがよく使用されます。Oracle Data Guardを使用すると、データ損失ゼロのフェイルオーバーと包括的なデータ保護、およびデータベース障害、クラスタ障害、およびデータ破損の際の非常に高速なフェイルオーバーが可能になります。また、Oracle Active Data Guardを使用すると読取り専用モードでスタンバイ・データベースを開けるという利点もあり、クエリを実行すると、プライマリ・データベースから返される結果と同じ結果が返されます。Oracle Active Data Guardのその他の重要なHA上の利点は、物理的なブロック破損の自動ブロック修復です。Oracle GoldenGateにはアドバンスド・レプリケーション機能が搭載されています。この機能には、停止時間なしの計画保守や移行を簡単に実行したり、データベースやアプリケーションをアップグレードしたりするための、双方向レプリケーションなどがあります。Oracle Active Data GuardとOracle GoldenGateは、ゴールドおよびプラチナのMAAリファレンス・アーキテクチャの必須コンポーネントです。詳しくは、『Oracle MAA Architecture – The Foundation for Database as a Service』と『Oracle Database高可用性概要12c』ガイドを参照してください。

このMaximum Availability Architecture (MAA) のベスト・プラクティスの資料では、Data Guardスタンバイによって保護されるデータベースを使用してOracle GoldenGateレプリケーションを有効にし、Data Guard保護モードの構成（最大パフォーマンス、最大可用性、または最大保護）に関係なく、Oracle Data Guardのロール移行後に透過的かつシームレスに動作できるようにするための、構成のベスト・プラクティスについて説明しています。

おもな前提条件は次のとおりです。

- Oracle Data GuardやOracle Active Data Guardで保護されるOracle GoldenGateレプリケーションのソース・データベースまたはターゲット・データベースであるOracle GoldenGateデータベース。

Oracle GoldenGateデータベースには、このデータベースに対して実行されるOracle GoldenGateコンポーネント（Extract、Data Pump、およびReplicat）のサブセットまたは全部が含まれる場合があります。これは、Oracle Data GuardまたはOracle Active Data Guardをフィジカル・スタンバイ・データベースで使用する、Oracle Database 11g以降のリリースのみに適用できます。

- Oracle Grid Infrastructure 11g Release 2 (11.2.0.4) 以降。

Oracle Grid Infrastructureには、あらゆるビジネス・クリティカル・アプリケーションの高可用性の管理に必要なコンポーネントが含まれます。Oracle Clusterware (Oracle Grid Infrastructureのコンポーネント) ネットワークを使用すると、データベースとOracle GoldenGateのリソースを管理して、障害発生時に可用性を確保できます。

- Oracle Grid Infrastructure Agent/バージョン6.1以降。

Oracle Grid Infrastructure Agentは、Oracle Grid Infrastructureコンポーネントを利用して、Oracle GoldenGateとその依存リソース（データベース、ネットワーク、ファイル・システムなど）を統合します。また、Oracle GoldenGateとOracle Data Guardを統合して、ロール移行後にOracle GoldenGateが新しいプライマリ・データベースで再起動されるようにします。

- Oracle Database 11g Release 2 (11.2.0.4) 以降。

Oracle Database 11g Release 2 (11.2.0.4) 以降を使用して、Oracle GoldenGateの統合Extractと統合Replicatの機能を利用します。Oracle GoldenGateを使用する場合に推奨されるすべてのOracleデータベース・パッチの一覧については、My Oracle Support Note 1557031.1を参照してください。

- Oracle GoldenGateバージョン12.1.2.0以降。

Oracle GoldenGate Extractはローカルにデプロイし、統合キャプチャ・モードで使用します。また、Oracle GoldenGateの統合Replicatの使用も推奨します。これは、Oracle Database 11g Release 2 (11.2.0.4) 以降で使用できます。

- 重要なGoldenGateファイルを保護およびレプリケートするためのOracle Database File System (Oracle DBFS) 。

Oracle Database File System (Oracle DBFS) は、MAAが検証し、推奨している唯一のファイル・システムです。その理由は、重要なOracle GoldenGateファイル (チェックポイント・ファイルや証跡ファイルなど) のストレージをOracle Data Guardで保護されている同じデータベース内に置くことができることから、Oracle GoldenGateファイルとデータベースの間の整合性が維持されるためです。Oracle DBFSディスク・グループは、Oracle Exadata Database Machineでも事前構成されています。

Oracle ClusterwareとOracle Grid Infrastructure Agentを使用して、Oracle GoldenGateとOracle Data Guardの統合を簡素化および自動化することを強く推奨します。Oracle ClusterwareとOracle Grid Infrastructure Agentを使用できない場合は、このホワイト・ペーパーの付録の手順を参照してください。

すべての汎用MAAおよびOracle GoldenGate MAAのベスト・プラクティスについては、次のURLを参照してください。 <http://www.oracle.com/technetwork/jp/content/maa-094615-ja.html>

## Oracle GoldenGateとOracle Active Data Guardを使用したMAAアーキテクチャの例

Oracle GoldenGateによる計画外/計画保守の利点を生かすには、アクティブ/アクティブHAデータベース・アーキテクチャが必要な場合があります。ただし、Oracle GoldenGateだけでは、データベース、クラスタ、データ破損などの障害が発生した場合に、データ損失ゼロを保証することはできません。このような障害から保護するには、次の図のようなMAAアーキテクチャを推奨します。

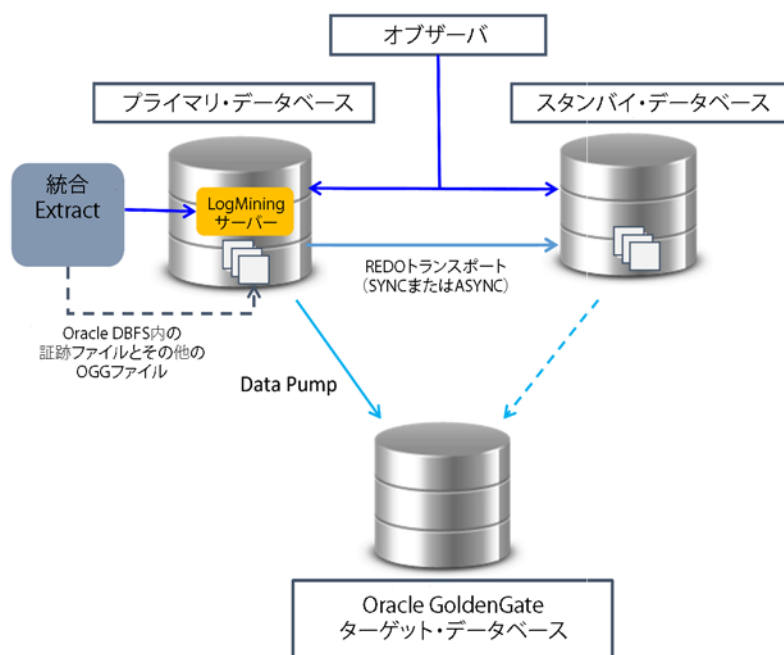


図1：アーキテクチャ例その1

図1では、フィジカル・スタンバイ・データベースへのフェイルオーバーやスイッチオーバーを実行する場合、Oracle Data Guardのロール移行後に、Oracle GoldenGateのExtractデータベースとターゲット・データベースがシームレスに機能します。Oracle Active Data Guardスタンバイ・データベースは、データベースやクラスタの障害時の、プライマリ・フェイルオーバー・ターゲットです。Oracle Active Data Guardは読取り専用でアクティブですが、DMLやDDLが含まれるアプリケーション・トランザクションは、すべてプライマリ・データベースにリダイレクトする必要があります。Oracle GoldenGateターゲット・データベースは、アプリケーション・アップグレードなどの計画保守作業の停止時間を短縮するために使用します。

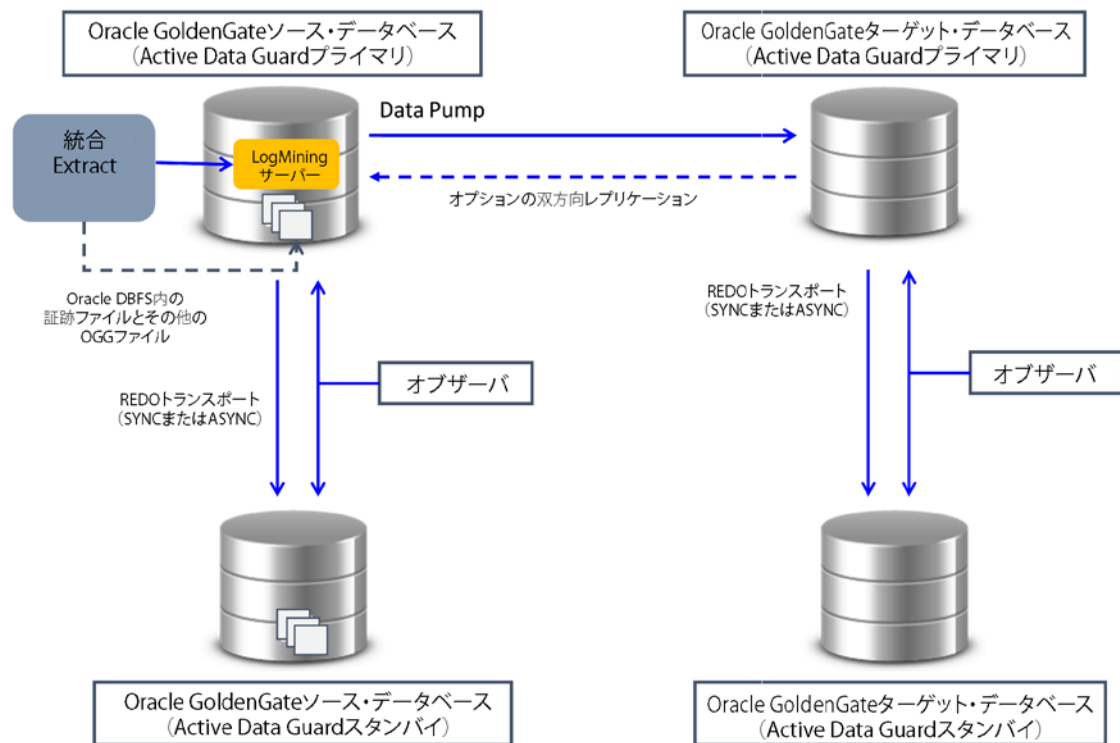


図2：アーキテクチャ例その2

図2では、2つのプライマリ・データベースが、Oracle GoldenGateを使用して相互にデータをレプリケートしています。データ・レプリケーションは、一方向にも双方向にも行うことができます。データ損失を発生させたくない場合は、最大可用性または最大保護モード（SYNCトランスポート）で構成されるOracle Active Data Guardファスト・スタート・フェイルオーバーを、各Oracle GoldenGateデータベースで構成します。ファスト・スタート・フェイルオーバーが有効であると、Oracle Data Guard Brokerとそのオブザーバによって、フェイルオーバーが必要かどうか判断され、指定したスタンバイ・データベースへのフェイルオーバーが自動的に開始されます。DBAが操作する必要はありません。Oracle Data Guardのフェイルオーバーやスイッチオーバーの後、Oracle GoldenGateのレプリケーションは引き続きシームレスに機能します。

図2は、プライマリ・データベースでアクティブなOracle GoldenGateを示しています。図3のフェイルオーバー後の図では、スタンバイ・サイトの新しいプライマリ・データベースで、Oracle GoldenGateが再度有効化されていることを示しています。

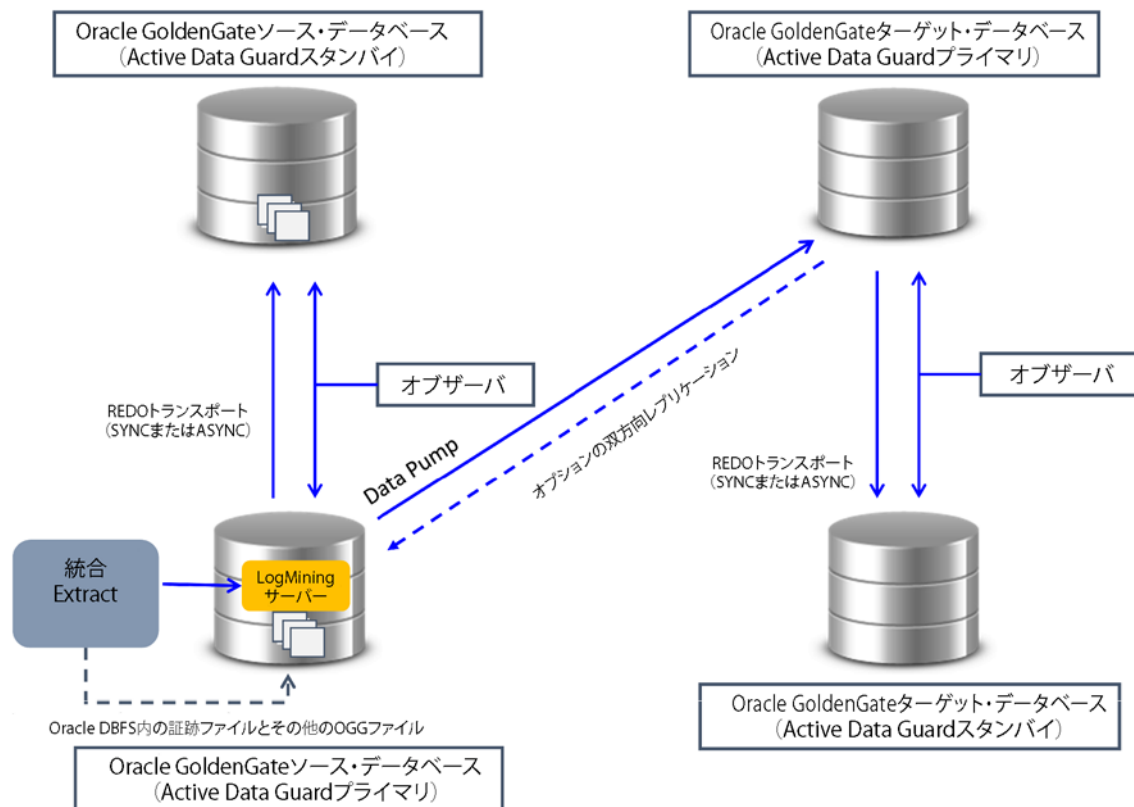


図3：フェイルオーバー後のOracle GoldenGateソース・データベース

## Oracle GoldenGate共有ファイル用のOracle DBFSの構成

Oracle Data Guardのロール移行中は、Oracle GoldenGateでそのチェックポイントとプロセス情報が維持されることが重要です。MAAでは、このようなOracle GoldenGateファイルをOracle DBFSマウント・ポイントに保存することを推奨しています。Oracle DBFS表領域は、Oracle Data Guardで保護されている同じOracle GoldenGateデータベースにあるため、データベースとOracle GoldenGateのファイルは、Oracle Data Guardのロール移行後にも引き続き同期されます。

次のURLのホワイト・ペーパー『*Oracle Real Application Clustersを使用したOracle GoldenGateの構成*』を参照して、Oracle DBFS構成のベスト・プラクティスに従ってください。

<http://www.oracle.com/technetwork/jp/database/features/availability/maa-goldengate-rac-2007111-ja.pdf>

高レベルなOracle DBFS構成の要件は、次のとおりです。

- Oracle Data Guardで保護されているプライマリ・データベースに、2つのOracle DBFS表領域を作成します。1つはOracle GoldenGateチェックポイント・ファイル用で、もう1つはその他のすべてのOracle GoldenGateファイル用です。
- チェックポイント・ファイルのファイル・システム用に作成したOracle DBFSのLOBセグメントはCACHE LOGGINGに設定し、もう1つはNOCACHE LOGGINGに設定します。
- 同じデータベース・ユーザーが両方のファイル・システムを所有するため、1つのmountコマンドで両方のファイル・システムがマウントされます。

MAAのホワイト・ペーパー『Oracle Real Application Clustersを使用したOracle GoldenGateの構成』 (<http://www.oracle.com/technetwork/jp/database/features/availability/maa-goldengate-rac-2007111-ja.pdf>) で推奨されているように、別のホストからアクセスできる必要があるすべてのOracle GoldenGateファイルは、Oracle DBFSに置く必要があります。たとえば、証跡ファイル (dirdat)、チェックポイント・ファイル (dirchk)、パラメータ・ファイル (dirprm)、バウンデッド・リカバリ・ファイル (BR)、資格証明ストア・ファイル (dircrd)、および一時ファイル (dirtmp) などです。

**注：**Oracle DBFSとOracle GoldenGateでOracle Data Guardを使用する場合、プロセス・ファイル・ディレクトリ (dirpcs) は、Oracle DBFSではなくローカル・ディスクに残す必要があります。プロセス・ファイルをOracle DBFSに置くと、Oracle Data Guardのフェイルオーバーやスイッチオーバーの間に、問題が発生する原因となる場合があります。

## Oracle GoldenGateの統合キャプチャ

Oracle Data Guardのロール移行中に抽出を継続するには、統合キャプチャ・モードのExtractが必要です。クラシック・キャプチャではロール移行中に自動マイニングができないため、推奨しません。統合キャプチャの利点は次のとおりです。

- Oracle Data Guardの非対称構成をサポートします。この構成では、プライマリ構成とスタンバイ構成で、Oracle Real Application Clusters (Oracle RAC) のノードやインスタンスの数が異なる場合があります。プライマリ・データベースとスタンバイ・データベースの対称構成を使用することを推奨します。
- データベースのRESETLOGS操作を含んだ抽出をサポートします (Oracle Data Guardのフェイルオーバー操作で一般的)。
- プライマリのオンラインREDOログと、スタンバイ・データベースのスタンバイREDOログの両方にREDOが書き込まれると、プライマリ・データベースからREDOの変更が抽出されます。これにより、Oracle Data Guardフェイルオーバーをデータ損失ゼロで実行した後、レプリケーションは同期されるようになります。

また、統合キャプチャはクラシック・キャプチャより多くのOracleデータベース機能をサポートしています。たとえば、データ圧縮 (Basic、OLTP、Exadata Hybrid Columnar Compression) やOracle RACサポートの改善などです。統合ExtractはRMANファスト・リカバリ領域の保存方針と統合されているため、アーカイブ・ログはExtractで不要になるまで削除されません。

Oracle GoldenGateの統合Extractの構成のベスト・プラクティスについて詳しくは、次のURLを参照してください。

<http://www.oracle.com/technetwork/jp/database/availability/maa-gg-performance-1969630-ja.pdf>

重要な推奨事項：

- データベース強制ロギングを有効にして、REDOのすべての変更を検出およびリカバリできるようにします。
- データベースの最小限のサプリメント・ロギングを有効にして、Oracle GoldenGateのコマンドADD SCHEMATRANDATAの使用中に追加のサプリメント・ログ・グループを作成します。
- 初期化パラメータSTREAMS\_POOL\_SIZEを使用して、Streamsプールを構成します。
- 統合Extractと統合Replicatの構成とパフォーマンス分析のため、UTL\_SPADVパッケージとOracle GoldenGate統合機能のヘルスチェックをインストールします。



プライマリ・データベース・ホストやスタンバイ・データベース・ホストでExtractプロセスを実行するには、Oracle GoldenGate資格証明ストアや暗号化されたパスワードを使用して、Extract/パラメータ・ファイルで明示的な接続文字列を設定することを推奨します。Extract/パラメータ・ファイルでプレーン・テキストのパスワードを使用することは推奨しません。

GoldenGate資格証明ストアのExtractパラメータの例は次のとおりです。

```
USERIDALIAS ggconn
```

次のコマンドを使用して、ggconn資格証明エイリアス名を資格証明ストアに追加します。

```
GGSCI> ALTER CREDENTIALSTORE ADD USER ggadmin@ggtns, password ggadmin, alias ggconn
```

暗号化されたパスワードのExtractパラメータの例は次のとおりです。

```
USERID ggadmin@ggtns, password AACAAAAAAAAAAAMAVJSBMDNBWHDJCAEGDJUIAEHBYIOACAJJ  
BLOWFISH ENCRYPTKEY DEFAULT
```

tnsnames.ora/パラメータ・ファイルで指定したTransparent Network Substrate (TNS) エイリアスが、その特定のノードで実行されるソース・データベースと接続していることを確認します。この例では、TNSエイリアスであるggtnsはノードごとに異なりますが、エイリアス名は同じです。tnsnames.ora/パラメータ・ファイルのTNSエントリには、この後作成するロールベース・サービスのサービス名が含まれる必要があります。ロールベース・サービスは、実行中の現在のプライマリ・データベースのホストのみで開始され、Oracle GoldenGateのExtractプロセスは、ロールベース・サービスが実行されている場合にのみ、データベースで実行されます。

## データ損失を考慮したOracle Data Guardのフェイルオーバー

常にデータ損失の可能性があるData Guard構成 (ASYNC REDO転送を使用したData Guardの最大パフォーマンス・モード) の場合、統合Extractのプロセス・パラメータ・ファイルに次のパラメータを追加する必要があります。

```
TRANLOGOPTIONS HANDLEDLFAILOVER
```

このパラメータを指定すると、ExtractによってREDOデータが抽出されたり、Oracle Data Guardスタンバイ・データベースにまだ適用されていない証跡ファイル・データに書き込まれたりすることがなくなります。これは、Oracle GoldenGateで、スタンバイ・データベースに存在しないターゲット・データベースにデータがレプリケートされないようにする場合に重要です。このパラメータを指定しないと、データ損失を伴うフェイルオーバー後に、ソース・データベースに存在しないターゲット・データベース内のデータで、論理データの非一貫性が発生する場合があります。

ロール移行後に古いプライマリ (または新しいスタンバイ) データベースを使用できない場合は、Oracle GoldenGateのターゲット・スタンバイ・データベースが使用可能にならないと、ExtractでREDOデータをマイニングできません。次のエラーが発生して、Extractプロセスが異常終了する可能性があります。

```
2015-01-23 18:28:00 ERROR OGG-02077 Extract encountered a read error in the  
asynchronous reader thread and is abending:Query to retrieve applied SCN of the target  
standby database failed.
```

「Oracle GoldenGate Manager」の項で後述するように、Managerプロセスの再開パラメータを正しく指定すれば、Extractプロセスの再開が試行されます。

古いプライマリ・データベースが長期間使用できなかつたり完全になくなつたりした場合、Oracle GoldenGateのレプリケーションを続行するには、Extract/パラメータ・ファイルからHANDLEDLFAILOVER/パラメータを削除する必要があります。データ抽出の前にREDOがスタンバイ・データベースに適用されるまで、Extractが待機す

ることはなくなります。スタンバイ・データベースが再度オンラインになり、プライマリ・データベースからすべてのREDOが適用されるまでの間、スタンバイ・データベースとOracle GoldenGateのターゲット・データベースの間でデータが一致しません。これは、スタンバイ・データベースが最新の状態になり、HANDLEDLFAILOVERパラメータを統合Extractのプロセス・パラメータ・ファイルに再度追加すれば解決します。

また、SYNC REDO転送による最大可用性モードを使用して、このパラメータをData Guard構成に追加することも推奨します。通常の操作では、最大可用性モードの場合、停止が発生してもフェイルオーバー時のデータ損失はゼロとなります。このような場合、このパラメータは不要です。ただし、停止が複数回発生すれば、データが失われる可能性は常にあります。たとえば、プライマリ・データベースとスタンバイ・データベースの間でネットワークが停止したのに、プライマリ・データベースで新しいトランザクションの処理が継続されるような場合、最初の問題が解決する前に2回目の停止が発生すると、少し後にプライマリ・データベースに障害が発生する原因となります。最初の停止の発生後に新しいトランザクションで生成されたデータは、すべて失われます。このようなユースケースに対応するために、このパラメータの設定が必要です。

プライマリ・データベースの損失が発生した場合に、以前に選択したスタンバイ・データベースにブローカが自動的にフェイルオーバーできるように、ファスト・スタート・フェイルオーバー (FSFO) の構成を推奨します。FSFOがない場合、スタンバイ・データベースへのフェイルオーバーは手動プロセスになります。

Oracle Data Guardファスト・スタート・フェイルオーバーが無効な場合、追加の統合Extract/パラメータを指定する必要があります。次に例を示します。

```
TRANLOGOPTIONS _FAILOVERTARGETDESTID n
```

このパラメータでは、Oracle Data Guardスタンバイ・データベースに適用されていないREDOデータを抽出しないことに関して、Oracle GoldenGateのExtractプロセスで後に残す必要があるスタンバイ・データベースが識別されます。\_FAILOVERTARGETDESTIDの正しい値を特定するには、archive\_log\_dest データベース初期化パラメータを使用します。nをアーカイブ・ログの保存先の正しい識別子に置き換えます。

次に例を示します。

NAME	TYPE	VALUE
log_archive_dest_1	string	location=USE_DB_RECOVERY_FILE_DEST, valid_for=(ALL_LOGFILES, ALL_ROLES)
log_archive_dest_2	string	service="ggs2d", ASYNC NOAFFIRM delay=0 optional compression=disable max_failure=0 max_connections=1 reopen=300 db_unique_name="GGS2D" net_timeout=30, valid_for=(online_logfile, all_roles)

この例では、Extract/パラメータを次のように設定します。

```
TRANLOGOPTIONS _FAILOVERTARGETDESTID 2
```

## Oracle GoldenGate Data Pump

Oracle GoldenGate Data Pumpプロセスでは、ソースの証跡ファイルが読み取られ、フィルタリングまたは変換された後で、ターゲット・ホストに送信されます。ターゲット・ホストでは、Collectorサーバー・プロセスによってデータがローカル証跡ファイルに書き込まれます。これを、オラクルのData Pump (impdpまたはexpdp) ユーティリティと混同しないでください。デフォルトでは、ソースからターゲット・ホストへの証跡ファイルのコピーは、ターゲット・ホストで証跡ファイルの書き込みを実行する前に、小さなパケットを送信するという方法で行われます。(Linuxでのfsyncを使用した) ディスクへのデータ書き込みの成功の確認は、それほど頻繁には行われません。このためロール移行の場合、1つ以上のReplicatプロセスからのチェックポイントと証跡ファイルが同期されない可能性があります。この問題が発生する可能性を減らすには、次のパラメータを使用してData Pumpを構成する必要があります。

```
RMTHOST <hostname>, MGRPORT <port#>, PARAMS -f -B1048576, TCPBUFSIZE 1048576,  
TCPFLUSHBYTES 1048576
```

TCPBUFSIZEパラメータとTCPFLUSHBYTESパラメータを組み合わせると、ソースとターゲットの間の送信バッファが1MBに増加します。このとき、ターゲット上のコレクタ・プロセスでは、1MB (-B書き込みバッファ・サイズ・パラメータ) を受信するたびに書き込みが実行され、その後すぐに書き込み確認 (-fフラッシュと同期) が行われます。

Replicatプロセスが実行されているOracle GoldenGate環境のターゲット・データベースが、Oracle Data Guard構成で保護されている場合、ソース環境で実行されているData Pumpに関する重要な考慮事項があります。Data Pumpによってソース証跡ファイルが送信されるホストは、RMTHOSTパラメータで決定されます。リモート・ホスト名には、プライマリ・ホストとスタンバイ・ホストの間を移動できる仮想IP (VIP) アドレスを使用することを推奨します。Oracle Data Guard環境でプライマリ・ホストとスタンバイ・ホストのVIPを定義できない場合、ソースData Pumpが常に現在のプライマリ・ホストに証跡ファイルを送信するように追加構成を行う必要があります。このような追加構成の手順について詳しくは、My Oracle Support Note 1950121.1を参照してください。

Data Guard構成で最大パフォーマンス・モードまたは最大可用性保護モードが使用されている場合は、Bug19870326用のGoldenGateパッチを適用します。Data Guardフェイルオーバーでデータ損失が発生した場合、新しいプライマリ・データベースのサイトでは、Oracle GoldenGateターゲットの証跡ファイルが予想より小さくなる可能性があります。このパッチ (19870326) を適用すると、Oracle GoldenGateターゲット・データベースのData Guardデータ損失のフェイルオーバー後に、Oracle GoldenGate Data Pumpを再起動できます。

プライマリ・データベース・ホストやスタンバイ・データベース・ホストでData Pumpプロセスを実行するには、「Oracle GoldenGateの統合キャプチャ」の項で前述したように、Oracle GoldenGate資格証明ストアを使用して、Data Pumpパラメータ・ファイルで明示的な接続文字列を設定することを推奨します。

## Oracle GoldenGate Replicat

Oracle GoldenGate Release 12.1およびOracle Database 11g Release 2 (11.2.0.4) で導入された統合Replicatを使用することを推奨します。統合Replicatでは、データベース内部で使用できる適用プロセス機能を利用します。障害発生後のプロセス・リカバリに必要な統合Replicatのチェックポイント情報は、データベースで自動的に維持されます。このため、チェックポイント表の構成は不要です。

非統合のReplicatプロセスを使用する場合、このプロセスでチェックポイント表が使用されるように構成して、チェックポイント・ファイルを使用する代わりに、データベースにチェックポイントが記録されるようにする必要があります。チェックポイント表の作成方法については、次のURLにあるホワイト・ペーパー『*Installing and Configuring Oracle GoldenGate for Oracle Database*』を参照してください。

[http://docs.oracle.com/goldengate/1212/gg-winux/GIORA/config\\_apply.htm#GIORA931](http://docs.oracle.com/goldengate/1212/gg-winux/GIORA/config_apply.htm#GIORA931)

Replicatのパフォーマンスを上げる構成については、次のURLにあるMAAのホワイト・ペーパー『Oracle GoldenGate Performance Best Practices』を参照してください。

<http://www.oracle.com/technetwork/jp/database/availability/maa-gg-performance-1969630-ja.pdf>

Oracle Data Guardのフェイルオーバー後に、チェックポイントが証跡ファイルより大きくなったことによるエラーが原因で非統合Replicatを起動できない場合は、次のエラー・メッセージが報告されます。

```
11:44:04      ERROR   OGG-01705      Input checkpoint position 388369723 for input
trail file '/u01/goldengate/dirdat_os/bb000028' is greater than the size of the file
(387583957). Please consult Oracle Knowledge Management Doc ID 1138409.1. for
instructions.
```

非統合Replicatで推奨されているチェックポイント表を使用する場合、次のコマンドを使用してプロセスを再開できます。

```
GGSCI> START REPLICAT <replicat name> FILTERDUPTRANSACTIONS
```

プロセス再開の手順について詳しくは、My Oracle Support Note 1536741.1を参照してください。Oracle Data Guardのロール移行後に、統合Replicatで自動的にリカバリが処理されるため、このエラーは報告されません。

プライマリ・データベース・ホストやスタンバイ・データベース・ホストでReplicatプロセスを実行するには、「Oracle GoldenGateの統合キャプチャ」の項で前述したように、Oracle GoldenGate資格証明ストアを使用して、Data Pump/パラメータ・ファイルで明示的な接続文字列を設定することを推奨します。

## データ損失を考慮したOracle Data Guardのフェイルオーバー

データ損失が発生したOracle Data Guardのフェイルオーバー後は、データベースでデータが失われた場合でも、Oracle GoldenGate Replicatプロセスによって更新される表のデータは失われません。これは、ターゲット・データベースでOracle DBFSに格納されている証跡ファイル・データが失われると、ソースのData Pumpプロセスによって再送信されるためです。Oracle GoldenGateによってレプリケートされないオブジェクトと、レプリケートされるオブジェクトの間で、データの非一貫性が存在する場合があります。

## Oracle GoldenGate Manager

Managerプロセスを構成して、このプロセスでOracle GoldenGateプロセスが自動的に再開されるようにする必要があります。mgr.prm/パラメータ・ファイルで、次のManagerパラメータを設定します。

```
AUTOSTART ER *
AUTORESTART ER *, RETRIES 20, WAITSECONDS 15, RESETMINUTES 60
```

AUTOSTARTを指定すると、Managerプロセスの開始後に、Extract、Data Pump、Replicatの各プロセスが自動的に開始されます。AUTORESTARTを指定すると、Oracle GoldenGateプロセスに障害が発生した後に、その再開が試行されます。ロール移行後に、Oracle GoldenGateプロセスの再開にかかる時間を短縮するには、再試行の最大回数を20、各再試行間の待機時間を15秒に設定することを推奨します。

## Oracle Cluster Resource (Oracle CRS) の構成

Oracle Data Guardを使用してOracle GoldenGateプロセスのシャットダウンと起動をすべて自動化するには、次のOracle CRSリソースを作成します。

## Oracle GoldenGateに適したOracle Clusterwareのロールベース・サービスの作成

ロールベース・サービスを使用すると、Oracle Data Guardのロール移行後に、Oracle GoldenGateと、Oracle GoldenGateで必要なリソース (Oracle DBFSなど) を起動できます。サービスは、実行中の現在のプライマリ・データベースのホストでのみ開始されます。

Oracleユーザーとして、次のコマンドを使用してOracle GoldenGateのロールベース・サービスを作成します (Oracle Database 12cの例)。

```
% srvctl add service -db GGS1 -service oggserv -role PRIMARY -preferred GGS21  
-available GGS22
```

注：

- この例では、データベース・サービスであるGGS1が、データベースですでに作成されています。次のURLにある『Oracle Database管理者ガイド』を参照してください。

[https://docs.oracle.com/cd/E57425\\_01/121/ADMIN/restart.htm#BABGIGDB](https://docs.oracle.com/cd/E57425_01/121/ADMIN/restart.htm#BABGIGDB)

- role PRIMARYは、データベースがOracle Data Guardプライマリ・データベースの場合にのみ、このサービスを開始できることを示しています。
- preferredは、データベースとロールベース・サービスが、使用可能なクラスタ・ノードのサブセットで実行される場合にのみ必要です。-preferredを使用してデータベース・インスタンスに名前を付けると、サービスがOracle RACクラスタ内のこのデータベース・インスタンスで開始されるようになります。
- availableを使用すると、ロールベース・サービスを実行できる他のOracle RACデータベース・インスタンスが表示されます。

Oracle Database 11g Release 2 (11.2.0.4) を使用している場合は、次のコマンドを使用して、Oracle GoldenGateのロールベース・サービスを作成します。

```
% srvctl add service -d GGS1 -s oggserv -l PRIMARY -r GGS21 -a GGS22
```

次のコマンドを使用して、新しいロールベース・サービスを開始し、そのステータスを確認します (Oracle Database 12cの例)。

```
% srvctl start service -db GGS1 -service oggserv  
% srvctl status service -db GGS1 -service oggserv
```

Oracle Database 11g Release 2 (11.2.0.4) を使用している場合は、次のコマンドを使用して、新しいロールベース・サービスを開始して確認します。

```
% srvctl start service -d GGS1 -s oggserv  
% srvctl status service -d GGS1 -s oggserv
```

Oracle Data Guardのプライマリ・ホストとスタンバイ・ホストの両方で、oggservロールベース・サービスを作成します。

ロールベース・サービス名を使用して、tnsnames.oraパラメータ・ファイルで定義されているTNSエイリアスをデータベースと接続する必要があります。次に例を示します。

```
ggconn= (DESCRIPTION =
  (ADDRESS = (PROTOCOL = TCP)(HOST = gghost1)(PORT = 1521)) (CONNECT_DATA =
    (SERVER = DEDICATED)
      (SERVICE_NAME = oggserv)
    )
  )
)
```

データベース・サービスの作成について詳しくは、次のURLにある『*Oracle Database*管理者ガイド』を作成してください。

[https://docs.oracle.com/cd/E57425\\_01/121/ADMIN/restart.htm#BABGIGDB](https://docs.oracle.com/cd/E57425_01/121/ADMIN/restart.htm#BABGIGDB)

## Oracle DBFSリソースの作成

Oracle DBFSのOracle CRSリソースを作成すると、Oracle GoldenGateとOracle Grid Infrastructure Agentで必要なOracle DBFSマウント・ポイントが自動的にマウントおよびアンマウントされるようになります(詳しくは、後述の手順を参照してください)。

`mount-dbfs.sh`アクション・スクリプトを使用して、My Oracle Support Note 1054431.1の手順に従います。すべてのOracle RACクラスター・ノードとすべてのOracle Data Guardスタンバイ・ホストで、同じ場所にアクション・スクリプトを保存します。

**注:** Oracle DBFSのアンマウントを強制するには、`mount-dbfs.sh`アクション・スクリプトを変更します。この変更により、Oracle Data Guardのスイッチオーバーのハングや失敗がなくなります。すべての `fusermount -u` コマンドを `fusermount -uz` に変更します。

Oracle DBFSをマウントするための `mount-dbfs.sh` アクション・スクリプトをテストします。次に例を示します。

```
% ./mount-dbfs.sh start
% ./mount-dbfs.sh status
```

```
Checking status now
Check - ONLINE
```

```
% df -k
```

Oracle DBFSのマウント・ポイントが表示されない場合は、My Oracle Support Note 1054431.1を参照してトラブルシューティングを行います。Oracle DBFSのアンマウントをテストします。

次に例を示します。

```
% ./mount-dbfs.sh stop
% ./mount-dbfs.sh status
```

```
Checking status now
Check - OFFLINE
```

```
% df -k
```

次のコマンドを使用して、Oracle DBFSのOracle CRSリソースを作成します。

```
#!/bin/bash

ACTION_SCRIPT=/u01/oracle/scripts/mount-dbfs.sh

RESNAME=dbfs_mount

DEPNAME=ora.ggs1.oggserv.svc

ORACLE_HOME=/u01/app/12.1.0.2/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($DEPNAME)pullup($DEPNAME)', \

      STOP_DEPENDENCIES='hard($DEPNAME)', \

      SCRIPT_TIMEOUT=300"
```

次の変数を変更します。

- ACTION\_SCRIPT - mount-dbfs.shアクション・スクリプトの場所と名前を指定します。
- DEPNAME - 前述の手順で作成したoggservロールベース・サービスのフルネームを指定します。フルネームを特定するには、crsctl stat res|grep oggservを実行します。
- ORACLE\_HOME - Grid Infrastructureのホーム・ディレクトリを指定します。

### Oracle GoldenGateアプリケーションの仮想IP (VIP) アドレスの作成 (Oracle RAC使用時)

アプリケーションの仮想IP (VIP) アドレスは、Oracle Clusterwareによって管理されるクラスタ・リソースです。VIPは1つのクラスタ・ノードに割り当てられ、ノード障害が発生するとクラスタ内の別のノードに移行されます。このため、Oracle GoldenGateのData Pumpは、新しく割り当てられたOracle RACのターゲット・ノードに対して、データ送信を続行できます。

次の項で説明するように、Oracle Grid Infrastructure Agentを使用してOracle GoldenGateを登録する場合、アプリケーションVIPは自動的に作成されます。同じIPアドレスを持つアプリケーションVIPがすでに存在する場合は、最初に削除する必要があります。

## Oracle Grid Infrastructure Agentの構成

Oracle Grid Infrastructure Agentは、Oracle Grid Infrastructureコンポーネントを利用して、Oracle GoldenGateとその依存リソース（データベース、ネットワーク、ファイル・システムなど）を統合します。また、Oracle GoldenGateとOracle Data Guardを統合して、ロール移行後にOracle GoldenGateプロセスを新しいプライマリ・データベースで再起動できるようにします。

Oracle Grid Infrastructureバージョン12cには、Bundled Agentと呼ばれるエージェントのバージョンがすでに含まれています。これは、入手できる最新バージョンではありません。次のURLから、スタンドアロン・エージェントの最新バージョン（v6.1）をダウンロードすることを推奨します。

<http://www.oracle.com/technetwork/jp/database/database-technologies/clusterware/downloads/index-2212905-ja.html>

Oracle Grid Infrastructureのホーム・ディレクトリ以外に古いバージョンのエージェントがインストールされている場合は、スタンドアロン・エージェントをインストールする前に、これを削除する必要があります。

**注：**ダウンロード可能なスタンドアロン・エージェントを、Oracle Grid Infrastructure Bundled Agentのホーム（\$GRID\_HOME/xagなど）に適用することはできません。

Oracle Grid Infrastructure Agentのドキュメントで、必ずバージョン・サポート・マトリックスを確認してください。エージェントの最新バージョン（6.1）は、Oracle Grid Infrastructureバージョン11.2.0.3以降のバージョン（12.1を含む）で認定されています。

新しくインストールしたエージェントのbinディレクトリの場所が、Oracle Grid Infrastructureのホーム・ディレクトリの前のPATH環境変数に表示されていることを確認します。次に例を示します。

```
% export PATH=/u01/oracle/bundledagent/xag/bin:$PATH
```

rootユーザーとして、次のコマンドを使用してOracle GoldenGateのエージェント構成を作成します。

```
agctl add goldengate GG_PRMY --gg_home /home/oracle/goldengate/ ¥
--oracle_home /u01/app/oracle/product/12.1.0.2/dbhome_1 ¥
--db_services ora.ggs1.oggserv.svc --monitor_extracts ext_1a,dpump_1a ¥
--monitor_replicats rep_1a,rep_2a ¥
--environment_vars
'TNS_ADMIN=/u01/app/oracle/product/12.1.0.2/dbhome_1/network/admin' ¥
--filesystems dbfs_mount ¥
--nodes RACnode1,RACnode2 --dataguard_autostart yes --user oracle ¥
--group oinstall --network 1 --ip 192.168.0.54
```

**注：**

- ・ リソース名GG\_PRMYは、必要に応じて、よりわかりやすい名前に変更できます。
- ・ gg\_homeとoracle\_homeは、ご使用の正しい環境設定に設定する必要があります。
- ・ db\_services/パラメータを使用して、以前に作成したoggservロールベース・サービスを指定します。
- ・ 前述の手順で作成したfilesystemsパラメータを使用して、Oracle DBFSのOracle CRSリソース名を指定します。
- ・ nodes/パラメータでOracle GoldenGateを実行できるOracle RACノードを表示します。



- `dataguard_autostart`を指定すると、Oracle Data Guardのロール移行後に、Oracle GoldenGateが自動的に起動および停止します。Oracle GoldenGateオペレーティング・システム・ユーザーの`user`と`group`も指定する必要があります。
- `network`パラメータを使用して、アプリケーションVIPのネットワーク番号を指定します。次のコマンドを使用して、ネットワーク番号を特定できます。

```
% crsctl stat res -p |grep -ie .network -ie subnet |grep -ie name -ie subnet
```

次のサンプル出力を検討してみましょう。

```
NAME=ora.net1.network
USR_ORA_SUBNET=10.1.41.0
```

`NAME=ora.net1.network`の`net1`は、これがネットワーク1であることを示しています。2行目はVIPが作成されるサブネットを示しています。

- `ip`パラメータを使用して、アプリケーションVIPのIPアドレスを指定します。アプリケーションVIPは、ネーミング形式 `xag.<resource_name>-vip.vip` で作成されます。たとえば、`xag.GG_PRMV-vip.vip`となります。

エージェントを使用してOracle GoldenGateを登録したら、Oracle GoldenGateプロセスの停止と開始に`agctl`を使用する必要があります。

Oracle GoldenGateと依存リソース（アプリケーションVIP、`oggserv`ロールベース・サービス、およびOracle DBFS）を起動するには、Oracleユーザーとして次のコマンドを実行します。

```
% agctl start goldengate GG_PRMV
```

Oracle GoldenGateと依存リソース（アプリケーションVIP、`oggserv`ロールベース・サービス、およびOracle DBFS）をクラスタ内の別のOracle RACノードに再配置するには、次のコマンドを実行します。

```
% agctl relocate goldengate GG_PRMV --node <node_name>
```

Oracle GoldenGateのステータスを確認するには、次のコマンドを実行します。

```
% agctl status goldengate GG_PRMV
Goldengate instance 'GG_PRMV' is running on RACnode1
```

Managerを含むすべてのOracle GoldenGateプロセスを停止するには、次のコマンドを実行します。

```
% agctl stop goldengate GG_PRMV
```

Oracle GoldenGateが停止しても、依存リソースは自動的に停止しません。Oracle GoldenGateのシャットダウン後にすべての依存リソースをシャットダウンするには、次のコマンドを実行します。

```
% srvctl stop service -d <DB name> -s oggserv
```

```
# Run the following two commands as the Oracle Grid Infrastructure user:

% crsctl stop resource xag.GG_PRMV-vip.vip

% crsctl stop resource dbfs_mount
```

これでOracle Grid Infrastructure Agentを使用してOracle GoldenGateが構成されるため、Oracle GoldenGateは、ロール移行後にプライマリ・データベースで（またはノード障害後に使用可能なOracle RACで）自動的に再起動されます。

## 結論

Oracle GoldenGateとOracle Data Guardを組み合わせると、プラチナMAAサービス・レベルの構成を実現できます。この構成では、すべての計画停止/計画外停止で、停止時間をゼロまたはほぼゼロにすることができます。このソリューションで不可欠なのは、Oracle Data Guardのデータ損失ゼロのロール移行の後で、Oracle GoldenGateが透過的に動作できるという点です。本書の構成内容と操作手順に従えば、データ損失ゼロやデータ損失ありのロール移行の後で、Oracle GoldenGateがOracle Data Guardとシームレスに連携するように構成できます。重要なOracle GoldenGateファイル（証跡ファイル、チェックポイント・ファイル、バインド・リカバリ・ファイルなど）のファイル・システムとしてOracle DBFSを使用すると、ロール移行後も、Oracle GoldenGateのExtractプロセスとReplicatプロセスがデータベースと引き続き同期できます。

## 付録：Oracle Grid Infrastructure Agentを使用しないOracle GoldenGateの構成

Oracle Grid Infrastructure Agentを使用できない場合、Oracle GoldenGateとOracle Data Guardの間のシームレスな統合はデータベースのロール移行トリガーを使用して処理されます。

次の項は、前述の「Oracle Cluster Resource (Oracle CRS) の構成」の項の小節の代わりとなるものです。Oracle GoldenGateプロセス (Manager、Extract、Data Pump、Replicat) の構成は、Oracle Grid Infrastructure Agentの使用の有無に関係なく同じです。

### Oracle DBFSの自動マウント/アンマウントのためのデータベース構成

データベースを開いたり、シャットダウンしたりする場合の、Oracle DBFSの自動的なマウント/アンマウントの有効化に必要なトリガーがいくつかあります。

**注：**この項で作成されるすべてのシェル・スクリプトは、スタンバイ・ホストとプライマリ・ホストの同じ場所にも存在する必要があります。この条件が満たされない場合、スクリプトが見つからないとロール移行トリガーが失敗します。次の手順を実行します。

1. データベースのシャットダウン時にOracle DBFSをアンマウントします。

データベースのシャットダウン前に、Oracle DBFSマウントをアンマウントして、データベースが開られない場合にシステムに古いマウント・ポイントが残らないようにする必要があります。

- a. Oracle DBFSマウント・ポイントをアンマウントするために、次のシェル・スクリプトを作成します (データベース・ユーザー (通常はoracle) が所有し、権限は6571に設定されます)。

```
#!/bin/sh
# unmount_dbfs.sh - stops all GoldenGate processes and unmounts DBFS
# First stop all GoldenGate processes:

/home/oracle/scripts/stop_all_GG.sh

# Now unmount DBFS mount point: fusermount -uz /mnt/dbfs
exit 0

#!/bin/sh
# stop_all_GG.sh - stops all the GoldenGate processes, including Manager

export ORACLE_HOME=/u01/app/oracle/product/12.1.0.2/dbhome_1
export ORACLE_SID=GG1
OGG_HOME=/home/oracle/goldengate
export PATH=$ORACLE_HOME/bin:$PATH

export TNS_ADMIN=$ORACLE_HOME/network/admin
export LD_LIBRARY_PATH=$ORACLE_HOME/lib

cd $OGG_HOME
./ggsci << !EOT
```

```
stop er *!  
kill er *  
stop mgr!  
exit  
!EOT  
exit 0
```

- b. これらのスクリプトの権限を6571に設定します。スクリプトを手動で実行して、機能することを確認します。

```
% df -k|grep dbfs  
% ./umount_dbfs.sh  
% df -k|grep dbfs
```

- c. (SYSDBAとして) 次のデータベース・ジョブとデータベース・シャットダウン・トリガーを作成します。

```
SQL> exec  
dbms_scheduler.create_job(job_name=>'dbfs_unmount', job_type=>'EXECUTABLE',  
job_action=>'/home/oracle/scripts/umount_dbfs.sh', enabled=>FALSE);
```

- d. Oracle DBFSがマウントされていることを確認し、ジョブを実行してから、Oracle DBFSがアンマウントされていることを確認します。

```
SQL> exec dbms_scheduler.run_job(job_name=>'dbfs_unmount');
```

- e. データベース・シャットダウン・トリガーを作成します。

```
SQL> create or replace trigger dbfs_unmount_trigger  
before shutdown on database  
declare  
    role varchar(30) ;  
begin  
    select database_role into role from v$database ;  
    if role = 'PRIMARY' then  
        dbms_scheduler.RUN_JOB('dbfs_unmount');  
    end if;  
end;  
/
```

f. Oracle DBFSをマウントしてからデータベースをシャットダウンして、トリガーをテストします。

**注:** データベース・シャットダウン・トリガーは、SHUTDOWN NORMALまたはSHUTDOWN IMMEDIATEのコマンドのみで実行されます。SHUTDOWN ABORTコマンドでは、トリガーは実行されません。

2. データベースの起動時に、Oracle DBFSをマウントします。

Oracle DBFSは、Oracle GoldenGateプロセスの開始準備のためにデータベースが開かれた後にも、自動的にマウントされる必要があります。

注: 前述の手順で説明したとおり、MOUNTコマンドを使用してOracle DBFSをマウントできるように、/etc/fstabが構成されていることを確認します。データベースの起動時にOracle DBFSをマウントするには、次の手順を実行します。

a. Oracle DBFSマウント・ポイントをマウントするために、次のシェル・スクリプトを作成します（データベース・ユーザー（通常はoracle）が所有し、権限は6571に設定されます）。

```
#!/bin/sh
# mount_dbfs.sh - mounts DBFS

export ORACLE_HOME=/u01/app/oracle/product/12.1.0.2/dbhome_1
export ORACLE_SID=GG51
OGG_HOME=/home/oracle/goldengate
export PATH=$ORACLE_HOME/bin:$PATH
export TNS_ADMIN=$ORACLE_HOME/network/admin
export LD_LIBRARY_PATH=$ORACLE_HOME/lib

# Force DBFS unmounting incase mount point remains from previous
# failure

fusermount -uz /mnt/dbfs
mount /mnt/dbfs

# If desired, add code in here to restart GoldenGate processes

exit 0
```

b. (SYSDBAとして) 次のデータベース・ジョブとデータベース起動トリガーを作成します。

```
SQL>exec
dbms_scheduler.create_job(job_name=>'dbfs_mount', job_type=>'EXECUTABLE',
job_action=>'/home/oracle/scripts/mount_dbfs.sh', enabled=>FALSE);
```

c. Oracle DBFSがマウントされていることを確認し、ジョブを実行してから、Oracle DBFSがアンマウントされていることを確認します。

```
SQL> exec dbms_scheduler.run_job(job_name=>'dbfs_mount');
```

- d. データベース起動トリガーを作成します。

```
SQL> create or replace trigger dbfs_mount_trigger
after startup on database
declare
    role varchar(30) ;
begin
    select database_role into role from v$database ;
    if role = 'PRIMARY' then
        dbms_scheduler.RUN_JOB('dbfs_mount') ;
    end if ;
end ;
/
```

- e. Oracle DBFSをマウント解除してからデータベースを起動して、トリガーをテストします。

3. スタンバイ・データベースからプライマリ・データベースへのロール移行が発生した場合に、Oracle DBFSをマウントしてOracle GoldenGateプロセスを開始します。

スタンバイ・データベースがプライマリ・データベースになるときのロール移行中に、データベースのシャットダウンや再起動が行われることはなく、シャットダウン・トリガーと起動トリガーは実行されません。このため、別のロール移行トリガーが必要です。これは、スタンバイ・データベースがプライマリになった場合にのみ実行されます。ロール移行が発生した場合に、Oracle DBFSをマウントしてOracle GoldenGateプロセスを開始するには、次の手順を実行します。

- a. Oracle DBFSマウント・ポイントをマウントして、Oracle GoldenGate ManagerとOracle GoldenGateのプロセスを開始するには、次のシェル・スクリプトを作成します（データベース・ユーザー（通常はoracle）が所有し、権限は6571に設定されます）。

```
#!/bin/sh
# failover_actions.sh - Mounts DBFS and starts GoldenGate on a
#                       database role transition.

export ORACLE_HOME=/u01/app/oracle/product/12.1.0.2/dbhome_1
export ORACLE_SID=GG1
OGG_HOME=/home/oracle/goldengate
export PATH=$ORACLE_HOME/bin:$PATH
export TNS_ADMIN=$ORACLE_HOME/network/admin
export LD_LIBRARY_PATH=$ORACLE_HOME/lib

# Now unmount (call twice to be sure DBFS is unmounted) and
# mount DBFS:

fusermount -uz /mnt/dbfs
fusermount -uz /mnt/dbfs
mount /mnt/dbfs

# Start GoldenGate:
cd ${GG_HOME}
./ggsci <<EOFF
stop er *!
```

```
stop mgr!  
pause 2  
start mgr  
exit  
EOFF  
  
exit 0
```

必ずスクリプトをテストしてください。Oracle DBFSがマウントされていること、およびGoldenGateプロセスが開始されることを確認します。

- b. (SYSDBAとして) 次のデータベース・ジョブとデータベース・ロール移行トリガーを作成します。

```
exec  
dbms_scheduler.create_job(job_name=>'ogg_failover', job_type=>'EXECUTABLE',  
job_action=>'/home/oracle/scripts/failover_actions.sh', enabled=>FALSE);
```

- c. スケジュールされたジョブを使用して、プライマリ・データベース上でスクリプト実行をテストします。

```
SQL> exec dbms_scheduler.run_job(job_name=>'ogg_failover');
```

- d. データベース・ロール移行トリガーを作成します。

```
create or replace trigger ogg_failover  
after db_role_change on database  
declare  
  
    role varchar2(30);  
begin  
    select database_role into role from v$database ;  
    if role = 'PRIMARY' then  
        dbms_scheduler.RUN_JOB('ogg_failover');  
    end if;  
end;  
/
```

このトリガーをテストする唯一の方法は、ロールを移行することです。

## Oracle GoldenGate Data Pumpの構成とロール移行

Replicatプロセスが実行されているOracle GoldenGate環境のターゲット・データベースが、Oracle Data Guard構成で保護されている場合、ソース環境で実行されているData Pumpに関する重要な考慮事項があります。Data Pumpによってソース証跡ファイルが送信されるホストは、RMTHOSTパラメータで決定されます。リモート・ホスト名には、プライマリ・ホストとスタンバイ・ホストの間を移動できる仮想IP (VIP) アドレスを使用することを推奨します。Oracle Data Guard環境でプライマリ・ホストとスタンバイ・ホストのVIPを定義できない場合、ソースData Pumpが常に現在のプライマリ・ホストに証跡ファイルを送信するように追加構成を行う必要があります。このような追加構成の手順について詳しくは、My Oracle Support Note 1950121.1を参照してください。

ソースData Pump構成を変更する前に、My Oracle Support Note 1950121.1のロール移行トリガーを変更して、Oracle DBFSのマウント・コマンドとOracle GoldenGateのプロセス開始コマンドも含まれるようにする必要があります。

次に例を示します。

```
create or replace trigger ogg_failover
after db_role_change on database
declare
    role varchar2(30);
    hostname varchar2(64);
begin
    select database_role into role from v$database;
    select host_name into hostname from v$instance;
    if role = 'PRIMARY' then
        dbms_scheduler.RUN_JOB('ogg_failover');
        if hostname = '<primary host name/VIP>'
        then
            change_rmthost@gg_source('PRIMARY');
        elsif hostname = '<standby host name/VIP>'
        then
            change_rmthost@gg_source('STANDBY');
        end if;
    end if;
end;
/
```





Oracle Corporation, World Headquarters

500 Oracle Parkway

Redwood Shores, CA 94065, USA

海外からのお問い合わせ窓口

電話：+1.650.506.7000

ファクシミリ：+1.650.506.7200

Oracle Data GuardとOracle GoldenGateによる透過的なロール移行

2015年2月

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



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

Copyright © 2015, 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。本文書は一切間違いがないことを保証するものではなく、さらに、口述による明示または法律による黙示を問わず、特定の目的に対する商品性もしくは適合性についての黙示的な保証を含み、いかなる他の保証や条件も提供するものではありません。オラクル社は本文書に関するいかなる法的責任も明確に否認し、本文書によって直接的または間接的に確立される契約義務はないものとします。本文書はオラクル社の書面による許可を前もって得ることなく、いかなる目的のためにも、電子または印刷を含むいかなる形式や手段によっても再作成または送信することはできません。

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

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。0215

**Hardware and Software, Engineered to Work Together**