

Oracle® GoldenGate

for Teradata インストレーションおよび
セットアップ・ガイド

リリース 11.2.1 for Windows and Unix

B72809-01 (原本部品番号 : E27295-01)

2013 年 5 月

ORACLE®

Oracle GoldenGate for Teradata インストールおよびセットアップ・ガイド リリース 11.2.1 for Windows and Unix

B72809-01 (原本部品番号 : E27295-01)

Copyright © 2010, 2011, 2012 Oracle and/or its affiliates. All rights reserved.

このソフトウェアおよび関連ドキュメントの使用と開示は、ライセンス契約の制約条件に従うものとし、知的財産に関する法律により保護されています。ライセンス契約で明示的に許諾されている場合もしくは法律によって認められている場合を除き、形式、手段に関係なく、いかなる部分も使用、複写、複製、翻訳、放送、修正、ライセンス供与、送信、配布、発表、実行、公開または表示することはできません。このソフトウェアのリバース・エンジニアリング、逆アセンブル、逆コンパイルは互換性のために法律によって規定されている場合を除き、禁止されています。

ここに記載された情報は予告なしに変更される場合があります。また、誤りが無いことの保証はいたしかねます。誤りを見つけた場合は、オラクル社までご連絡ください。

このソフトウェアまたは関連ドキュメントが、米国政府機関もしくは米国政府機関に代わってこのソフトウェアまたは関連ドキュメントをライセンスされた者に提供される場合は、次の Notice が適用されます。

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

このソフトウェアは様々な情報管理アプリケーションでの一般的な使用のために開発されたものです。このソフトウェアは、危険が伴うアプリケーション (人的傷害を発生させる可能性があるアプリケーションを含む) への用途を目的として開発されていません。このソフトウェアを危険が伴うアプリケーションで使用する場合、このソフトウェアを安全に使用するために、適切な安全装置、バックアップ、冗長性 (redundancy)、その他の対策を講じることは使用者の責任となります。このソフトウェアを危険が伴うアプリケーションで使用したことにより起因して損害が発生しても、オラクル社およびその関連会社は一切の責任を負いかねます。

Oracle は Oracle Corporation およびその関連企業の登録商標です。その他の名称は、他社の商標の可能性があります。

このソフトウェアおよびドキュメントは、第三者のコンテンツ、製品、サービスへのアクセス、あるいはそれらに関する情報を提供することがあります。オラクル社およびその関連会社は、第三者のコンテンツ、製品、サービスに関して一切の責任を負わず、いかなる保証もいたしません。オラクル社およびその関連会社は、第三者のコンテンツ、製品、サービスへのアクセスまたは使用によって損失、費用、あるいは損害が発生しても一切の責任を負いかねます。

目次

はじめに	vii
対象読者	vii
ドキュメントのアクセシビリティについて	vii
関連ドキュメント	vii
表記規則	viii
1 システム要件およびインストール前の手順	
1.1 Oracle GoldenGate for Teradata の概要	1-1
1.2 このドキュメントの内容	1-1
1.3 レプリケーション・サーバーとしてサポートされているプラットフォーム	1-2
1.4 オペレーティング・システム要件	1-2
1.4.1 メモリー要件	1-2
1.4.2 ディスクの要件	1-3
1.4.3 Relay Services Gateway(RSG)vprocs	1-3
1.4.4 ネットワーク	1-3
1.4.5 オペレーティング・システムの権限	1-3
1.4.6 コンソール	1-4
1.4.7 その他のプログラム	1-4
1.5 データベースの要件	1-4
1.5.1 データベース構成	1-4
1.5.2 Oracle GoldenGate プロセス用のデータベース・ユーザー	1-4
1.6 サポートされている Teradata データ型	1-5
1.6.1 数値データ型に関するサポートの制限事項	1-6
1.6.2 シングルバイト文字データ型に関するサポートの制限事項	1-7
1.6.3 マルチバイト文字データに関するサポートの条件および制限事項	1-7
1.6.4 バイナリ・データ型に関するサポートの制限事項	1-7
1.6.5 ラージ・オブジェクト・データ型に関するサポートの制限事項	1-7
1.6.6 日付データ型に関するサポートの制限事項	1-7
1.6.7 IDENTITY データ型に関するサポートの制限事項	1-8
1.7 Teradata でサポートされているオブジェクトおよび操作	1-8
1.7.1 DML	1-8
1.7.2 DDL	1-8

2 Oracle GoldenGate のインストール

2.1	インストールの概要	2-1
2.2	Oracle GoldenGate のダウンロード	2-1
2.3	UNIX 上での動的ビルドのライブラリ・パスの設定	2-2
2.4	クラスタへの Oracle GoldenGate のインストール準備	2-3
2.4.1	Oracle GoldenGate のバイナリおよびファイルをインストールするクラスタ内の場所の決定	2-3
2.5	Linux および UNIX 上での Oracle GoldenGate のインストール	2-4
2.5.1	Oracle GoldenGate ファイルのインストール	2-4
2.6	Windows への Oracle GoldenGate のインストール	2-4
2.6.1	Windows クラスタへの Oracle GoldenGate のインストール	2-4
2.6.2	Oracle GoldenGate ファイルのインストール	2-5
2.6.3	Manager のカスタム名の指定	2-5
2.6.4	Windows サービスとしての Manager のインストール	2-6
2.7	クラスタへの Oracle GoldenGate の統合	2-7
2.7.1	クラスタの一般的な要件	2-7
2.7.2	Windows クラスタ・リソースとしての Oracle GoldenGate の追加	2-8

3 Oracle GoldenGate システムの準備

3.1	処理に向けた表の準備	3-1
3.1.1	トリガーおよびカスケード制約を無効にする	3-1
3.1.2	行識別子の割当て	3-1
3.1.2.1	使用する行識別子の種類を決定する Oracle GoldenGate の仕組み	3-1
3.1.2.2	KEYCOLS を使用したカスタム・キーの指定	3-2
3.2	Tearadata のレプリケーション・グループの作成	3-2
3.3	Teradata TAM による DDL 取得のアクティブ化	3-3
3.4	DDL の取得の無効化	3-4
3.5	TAM 初期化ファイルの構成	3-4

4 Oracle GoldenGate の構成

4.1	最大保護モードでの Oracle GoldenGate の構成	4-1
4.1.1	推奨される最大保護の構成	4-1
4.1.2	最大保護モードでの Extract の構成	4-2
4.1.3	最大保護モードをサポートするための Replicat の構成	4-4
4.2	最大パフォーマンス・モードでの Oracle GoldenGate の構成	4-5
4.2.1	推奨される最大パフォーマンスの構成	4-5
4.2.2	最大パフォーマンス・モードでの Extract の構成	4-5
4.2.3	最大パフォーマンス・モードをサポートするための Replicat の構成	4-7
4.3	Oracle GoldenGate のその他の構成ガイドライン	4-8
4.3.1	DDL サポートの構成	4-8
4.3.2	大量の更新操作および削除操作の処理	4-8
4.3.3	複数の接続の回避	4-8
4.3.4	ODBC 経由での Replicat パフォーマンスの改善	4-8
4.3.5	初期同期の実行	4-8

5 Teradata データベースへの DDL 同期の構成

5.1	このドキュメントについて	5-1
5.2	DDL 同期の概要	5-1
5.3	Oracle GoldenGate における DDL サポートの制限事項	5-1
5.3.1	DDL 文の長さ	5-2
5.3.2	サポートされているトポロジ	5-2
5.3.3	フィルタリング、マッピングおよび変換	5-2
5.3.4	DDL レスポンス時間	5-2
5.4	DDL サポートの構成ガイドライン	5-2
5.4.1	データベース権限	5-2
5.4.2	並列処理	5-2
5.4.3	データ・ポンプ Extract および VAM ソートによる Extract の DDL と DML	5-3
5.4.4	オブジェクト名	5-3
5.4.5	データ定義	5-3
5.4.6	初期同期	5-4
5.5	DDL スコープについて	5-4
5.5.1	MAPPED スコープ	5-4
5.5.1.1	ALTER INDEX のマッピング	5-5
5.5.2	UNMAPPED スコープ	5-6
5.5.3	OTHER スコープ	5-6
5.6	DDL サポートの有効化	5-6
5.7	DDL レプリケーションのフィルタリング	5-6
5.7.1	DDL パラメータ・オプションの組合せ	5-10
5.7.2	DDL EXCLUDE ALL	5-10
5.8	Oracle GoldenGate における導出オブジェクト名の処理方法	5-10
5.8.1	導出オブジェクトのマッピング	5-11
5.8.1.1	MAP がベース・オブジェクトに存在して導出オブジェクトに存在しない場合	5-11
5.8.1.2	MAP がベース・オブジェクトと導出オブジェクトに存在する場合	5-11
5.8.1.3	MAP が導出オブジェクトに存在してベース・オブジェクトに存在しない場合	5-12
5.8.2	導出オブジェクトとしての新しい表	5-12
5.8.2.1	RENAME	5-12
5.8.2.2	CREATE TABLE AS SELECT	5-12
5.8.3	導出オブジェクトのマッピングの無効化	5-13
5.9	DDL 文字列置換の使用	5-14
5.10	処理で DDL が評価される仕組み	5-14
5.10.1	Extract が DDL を評価する仕組み	5-14
5.10.2	Replicat が DDL を評価する仕組み	5-15
5.11	DDL 処理エラーの処理	5-16
5.12	DDL レポート情報の表示	5-16
5.12.1	Extract DDL レポート	5-16
5.12.2	Replicat DDL レポート	5-16
5.12.3	プロセス・レポートの統計	5-17
5.13	DDL 処理のトレース	5-17

6 Oracle GoldenGate 構成でのオブジェクトの変更

6.1	Extract グループの削除	6-1
6.2	既存の Extract グループへの表の追加	6-1
6.3	新しい Extract グループへの表の追加	6-2
6.4	表列の変更	6-3

7 Oracle GoldenGate のアンインストール

7.1	Linux または UNIX からの Oracle GoldenGate のアンインストール	7-1
7.2	Windows (非クラスタ) からの Oracle GoldenGate のアンインストール	7-2
7.3	Windows クラスタからの Oracle GoldenGate のアンインストール	7-2

A インストールされる Oracle GoldenGate のコンポーネント

A.1	Oracle GoldenGate のプログラムおよびユーティリティ	A-1
A.2	Oracle Goldengate のサブディレクトリ	A-2
A.3	その他の Oracle GoldenGate ファイル	A-4
A.4	Oracle GoldenGate のチェックポイント表	A-8

はじめに

このガイドでは、Teradata データベース環境への Oracle GoldenGate のインストールと、Oracle GoldenGate の取得および配信をサポートするための Teradata データベースの構成について説明します。

対象読者

このガイドは、インストール担当者、データベース管理者、および Oracle GoldenGate のインストール、構成、実行を行うシステム管理者を対象としています。

ドキュメントのアクセシビリティについて

Oracle のアクセシビリティについての詳細情報は、Oracle Accessibility Program の Web サイト (<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>) を参照してください。

Oracle サポートへのアクセス

Oracle サポート・サービスでは、My Oracle Support を通して電子支援サービスを提供しています。詳細情報は <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> か、聴覚に障害のあるお客様は <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> を参照してください。

関連ドキュメント

Oracle GoldenGate の完全なドキュメント・セットは、次の内容で構成されています。

HP NonStop プラットフォーム

- *Oracle GoldenGate for NonStop リファレンス・ガイド*
- *Oracle GoldenGate for NonStop 管理者ガイド*

Windows、UNIX および Linux プラットフォーム

- *サポートされているデータベースごとの Oracle GoldenGate インストールおよびセットアップ・ガイド*
- 『Oracle GoldenGate Windows and UNIX 管理者ガイド』
- 『Oracle GoldenGate Windows and UNIX リファレンス・ガイド』
- 『Oracle GoldenGate Windows and UNIX トラブルシューティングおよびチューニング・ガイド』
- 『Oracle GoldenGate アップグレード・ガイド』

その他の Oracle GoldenGate 製品

- 『Oracle GoldenGate フラット・ファイル・アダプタ管理者ガイド』
- 『Oracle GoldenGate Java アダプタ管理者ガイド』
- 『Oracle GoldenGate Director 管理者ガイド』
- 『Oracle GoldenGate Monitor 管理者ガイド』
- 『Oracle GoldenGate Veridata 管理者ガイド』

表記規則

このドキュメントでは、次のテキスト表記規則が使用されています。

表記規則	意味
太字	太字は、「[ファイル]」メニューから「保存」を選択します」など、操作に関連するグラフィカル・ユーザー・インタフェース要素を示します。また、太字は、本文中や用語集で定義されている用語にも使用されます。
イタリック	イタリックは、TABLE table_name のようなパラメータ文など、ユーザーが特定の値を指定するプレースホルダ変数を示します。イタリックは、マニュアル名や強調にも使用されます。
固定幅フォント 大文字の固定幅フォント	固定幅フォントは、ユーザー・イグジットなどのコード・コンポーネントやスクリプト、ファイルやデータベース・オブジェクトの名前、URL パス、画面に表示される入出力テキストを示します。大文字の固定幅フォントは、通常、Oracle GoldenGate のパラメータ名、コマンド、ユーザーが構成可能な関数、SQL コマンドやキーワードを表すために使用されます。
大文字	通常テキスト・フォント中の大文字は、名前が特定のケースである場合を除き、ユーティリティの名前を示します。
{ }	構文内の中かっこは、{option1 option2 option3} のように、パイプ記号で区切られた一連のオプションを囲んでおり、その中の 1 つを選択する必要があります。
[]	構文内の大かっこはオプション要素を示します。たとえば、CLEANUP REPLICAT group_name [, SAVE count] という構文では、SAVE 句がオプションです。1 つのオプション要素内に複数のオプションがある場合は、[option1 option2] のように、パイプ記号で区切られます。

システム要件およびインストール前の手順

この章では、Oracle GoldenGate をサポートするシステムおよびデータベース・リソースの要件について説明します。

1.1 Oracle GoldenGate for Teradata の概要

Oracle GoldenGate では、次のデータのレプリケーションをサポートしています。

- ソースの Teradata データベース (ソース・サーバーとして知られている) とターゲットの Teradata データベース (サブスクライバ・サーバーとして知られている) 間
- Teradata データベースとサポートされている他のデータベース・プラットフォーム間

さらに、Oracle GoldenGate によって、同一の Teradata のソース・サーバーとサブスクライバ・サーバー間で DDL 操作がレプリケートされます。

Oracle GoldenGate は、Teradata データベースが存在するサーバーとは別のレプリケーション・サーバー。上で動作します。レプリケーション・サーバーは 1 つにすることも、ソースとターゲットに 1 つずつにすることも可能です。

Oracle GoldenGate は、Teradata Change Data Capture (CDC) ファシリティからトランザクションの変更または表のコピー操作を受信し、ODBC 経由でサブスクライバに送信します。CDC と Oracle GoldenGate 間の通信は、Teradata Access Module (TAM) によって管理されます。

Oracle GoldenGate for Teradata では、特に注記のないかぎり、データのフィルタリング、マッピングおよび変換がサポートされています。

1.2 このドキュメントの内容

このドキュメントには、Teradata 環境内の Oracle GoldenGate ソリューションのセットアップ固有の情報が記載されています。このドキュメントは、Teradata データベースおよび Teradata Replication Solutions の基本的な知識を持つ読者を対象にしています。さらに、次の内容を正しく構成していることを前提とします。

- Relay Services Gateway(RSG)
- Change Data Capture(CDC)
- Teradata Access Module (TAM)
- レプリケーション・グループ

Teradata データベースにレプリケーションを構成するには、Teradata Corporation の『*Teradata Replication Solutions*』のドキュメントを参照してください。

1.3 レプリケーション・サーバーとしてサポートされているプラットフォーム

Teradata 環境では、ソースの Teradata データベースとターゲットの Teradata データベースがインストールされているサーバーとは別のサーバーに Oracle GoldenGate をインストールします。このマシンはレプリケーション・サーバーとなり、Teradata データベース用に Oracle GoldenGate でサポートされているプラットフォームにする必要があります。データベースのバージョンとオペレーティング・システムの特定の組合せに Oracle GoldenGate のどのビルドを使用できるかを調べるには、<http://support.oracle.com> にログインし、「動作保証」タブを選択します。サポートが必要な場合、「動作保証検索のヒント」をクリックします。このサイトに入るには、電子メールとパスワードが必要です。

1 つまたは複数のレプリケーション・サーバーを選択する際の注意点は次のとおりです。

- レプリケーション・サーバーは、ソース・サーバーまたはターゲット・サーバーと同じロケーションに配置するか、それらのサーバーのいずれかまたは両方から離れたロケーションに配置することができます。離れた場所にある Teradata のシステム間のレプリケーションでは、ソースの場所にあるレプリケーション・サーバーに Oracle GoldenGate の Extract プロセスをインストールし、ターゲットの場所にある別のレプリケーション・サーバーに Oracle GoldenGate の Replicat プロセスをインストールします。
- 可能な場合、複数ノードのクラスタ・サーバーに Oracle GoldenGate をインストールし、サーバーの故障による Oracle GoldenGate の停止の影響を最小限に抑えます。

追加構成に関する考慮事項については、Oracle GoldenGate をインストールする前に、Teradata Corporation の『*Teradata Replication Services Using Oracle GoldenGate*』のドキュメントを参照してください。

1.4 オペレーティング・システム要件

この項では、Oracle GoldenGate のサポートに必要なオペレーティング・システム・リソースについて説明します。これらのリソースは、データベース・サーバー、レプリケーション・サーバーまたはすべてのサーバーに適用されます。

1.4.1 メモリー要件

Oracle GoldenGate に必要なレプリケーション・サーバーのメモリー量は、そのシステムで実行される同時プロセスの数によって異なります。ソース・レプリケーション・サーバー上には少なくとも、ソース・データを取得するプライマリ Extract プロセスと、ネットワーク全体にデータを転送するセカンダリ Extract データ・ポンプ・プロセスがあります。ターゲット・レプリケーション・サーバー上には、レプリケートされたデータをターゲット・データベースに適用する Replicat プロセスが少なくとも 1 つ存在します。実装する構成によっては、これらのプロセスがすべて同じシステムで動作する場合もあります。

レプリケートを必要とするトランザクション・データが大量に生成される環境では、追加のパラレル処理を使用したスループットの向上が必要になることがあります。Oracle GoldenGate では、Oracle GoldenGate のインスタンス当たり、最大 5,000 の Extract および Replicat の同時プロセスがサポートされます。Extract および Replicat の各プロセスには、約 25 から 55MB のメモリーが必要で、トランザクションのサイズや同時トランザクションの数によっては、さらに必要になります。

Oracle GoldenGate プロセスが使用する実際の物理メモリー量は、Oracle GoldenGate プログラムではなく、オペレーティング・システムによって制御されます。Oracle GoldenGate キャッシュ・マネージャは、オペレーティング・システムのメモリー管理機能を使用して、Oracle GoldenGate プロセスが持続的かつ効率的に動作するようにします。Oracle GoldenGate のメモリー要件評価の詳細は、『*Oracle GoldenGate Windows and UNIX リファレンス・ガイド*』の CACHEMGR パラメータに関する項を参照してください。

1.4.2 ディスクの要件

Oracle GoldenGate レプリケーション・サーバーに推奨されるハードウェア構成は、次のとおりです。

- 300GB ディスク 4 個
- デュアルコア CPU4 個
- RAM 8GB

次のように空きディスク領域を割り当てます。

- データベースおよびプラットフォームに応じて 50 ~ 150MB を割り当てます。これには、圧縮されたダウンロード・ファイルのスペースと圧縮されていないファイルのスペースが含まれます。インストールの完了後にダウンロード・ファイルを削除できます。
- システムにインストールしている Oracle GoldenGate の各インスタンスの作業ディレクトリおよびバイナリ用に 40MB を割り当てます。たとえば、Oracle GoldenGate の 2 つのビルドを 2 つの別々のディレクトリにインストールするには、80MB の空き容量を割り当てます。
- Oracle GoldenGate の証跡 (作業データが含まれているファイル) をホストするシステムに追加の 1GB のディスク容量を割り当てます。証跡によって消費される領域は、処理されるデータ量に応じて変わるため、必要な量がこれを前後する場合もあります。証跡のサイズに関するガイドラインは、『Oracle GoldenGate Windows and UNIX 管理者ガイド』を参照してください。
- Oracle GoldenGate をクラスタ環境にインストールするには、すべてのクラスタ・ノードで利用できる共有ファイル・システムに Oracle GoldenGate のバイナリおよびファイルをインストールします。

1.4.3 Relay Services Gateway(RSG)vprocs

レプリケーション作業は、レプリケーション・サーバーとの接続のためにソース・データベース・サーバー上の RSG vprocs で実行されます。接続では、TCP/IP プロトコルを実装します。Teradata V12 からは、各システム・ノードに 1 つの RSG があります。

1.4.4 ネットワーク

Oracle GoldenGate をサポートするには、次のネットワーク・リソースを使用できる必要があります。

- Oracle GoldenGate に関連するシステムを、DNS を含む TCP/IP サービスを使用するように構成します。Oracle GoldenGate では IPv4 および IPv6 がサポートされており、これらのプロトコルのいずれか、または両方がサポートされているシステムで動作できます。
- Oracle GoldenGate のプロセスをホストし、Oracle GoldenGate の接続先となるすべてのシステムのホスト名または IP アドレスを使用してネットワークを構成します。ホスト名の方が便利です。
- Oracle GoldenGate には、予約されていない無制限の TCP/IP ポートが必要で、その数は、構成内のプロセスの数とタイプによって異なります。必要なポートに対応する Manager プロセスの構成方法の詳細は、『Oracle GoldenGate Windows and UNIX 管理者ガイド』を参照してください。
- Oracle GoldenGate に割り当てたポートを記録します。Manager プロセスの構成時にパラメータを使用してそれらのポートを指定します。
- Oracle GoldenGate のポートを経由した接続を受け入れるようにファイアウォールを構成します。

1.4.5 オペレーティング・システムの権限

Manager プロセスには、Oracle GoldenGate のプロセスを制御する権限と Oracle GoldenGate ディレクトリのファイルおよびサブディレクトリの読取り、書込み、削除を行う権限を持つオ

オペレーティング・システム・ユーザーが必要です。Extract プロセスおよび Replicat プロセスには、データベースにアクセスする特権が必要です。

1.4.6 コンソール

オペレーティング・システムとコマンド・コンソールのキャラクタ・セットは、同一にする必要があります。Microsoft Windows システムでは、オペレーティング・システムはあるキャラクタ・セットに設定されていますが、DOS コマンド・プロンプトでは、別の古い DOS キャラクタ・セットが使用されるため、不一致が発生します。Oracle GoldenGate では、オペレーティング・システムのキャラクタ・セットを使用して GGSCI コマンド出力に情報が送信されるため、一致しないコンソールのキャラクタ・セットが原因で、文字が正しく表示されません。GGSCI セッションを実行する前に、次の DOS コマンドを使用して、コンソールのキャラクタ・セットを設定できます。

```
chcp OS character set
```

コード・ページの設定後に文字が正しく表示されない場合は、コンソールのフォントを、拡張キャラクタ・セットがある Lucida Console に変更してください。

1.4.7 その他のプログラム

Oracle GoldenGate をサポートするための追加の考慮事項は、次のとおりです。

- Windows システムに Oracle GoldenGate をインストールする前に、Microsoft Visual C++ 2005 SP1 再頒布可能パッケージをインストールおよび構成します。それが、このパッケージの SP1 バージョンであることを確認し、サーバーに適したビットのバージョンを取得していることを確認してください。このパッケージによって、Visual C++ ライブラリのランタイム・コンポーネントがインストールされます。詳細についておよびこのパッケージをダウンロードするには、<http://www.microsoft.com> に移動してください。
- Oracle GoldenGate では、プラットフォーム上で仮想化ソフトウェアを使用して作成された仮想マシン環境を完全にサポートします。Oracle GoldenGate を仮想マシン環境にインストールするときに、データベースと一致するビルドおよび仮想マシンのオペレーティング・システム (ホスト・システムではない) を選択します。

1.5 データベースの要件

この項では、Teradata データベースに固有の Oracle GoldenGate の要件について説明します。

1.5.1 データベース構成

- 次の適切な ODBC (Open Database Connectivity) ドライバをインストールします。
 - TTU 12.0 ファミリ V12 の 12.00.00.01 またはそれ以降
 - TTU 13.0 ファミリ V13.1 の 13.00.00.00 またはそれ以降
- データ・ソース名 (DSN) の作成も含め、ソースとターゲットの各システムに ODBC を構成します。DSN には、そのデータベースに接続する方法に関する情報が保存されます。手順は、『*ODBC ドライバ for Teradata ユーザー・ガイド*』を参照してください。
- ソース表に Teradata レプリケーション・グループを作成します。手順については、『*Teradata Replication Services Using Oracle GoldenGate*』のドキュメントを参照してください。

1.5.2 Oracle GoldenGate プロセス用のデータベース・ユーザー

- Oracle GoldenGate 専用のデータベース・ユーザーを作成します。データベースに接続する必要がある、次の Oracle GoldenGate プロセスすべてに同じユーザーを使用できます。
 - Extract (ソース・データベース)
 - Replicat (ターゲット・データベース)

- DEFGEN ユーティリティ (ソース・データベースまたはターゲット・データベース)
- データのセキュリティを維持したり、Oracle GoldenGate の処理を的確に監視したりするには、他のユーザー、アプリケーションまたはプロセスに対して Oracle GoldenGate データベース・ユーザーでのログインまたは操作を許可しないでください。
- ソースの Teradata データベースからの取得に Oracle GoldenGate を使用する場合は、次のセキュリティ許可を Extract データベース・ユーザーに発行します。


```
GRANT SELECT ON DBC.REPGROUP TO user;
GRANT SELECT ON DBC.TVM TO user;
GRANT SELECT ON DBC.DBASE TO user;
GRANT SELECT ON DBC.ERRORMSGs TO user;
GRANT SELECT ON DBC.TVFIELDS TO user;
GRANT SELECT ON DBC.INDEXES TO user;
GRANT SELECT ON DBC.INDOUBTRESLOG TO user;
GRANT REPLCONTROL TO user;
GRANT ALL ON database TO user;
GRANT ALL ON SYSUDTLIB TO user WITH GRANT OPTION;
```
- ターゲットの Teradata データベースへのレプリケートに Oracle GoldenGate を使用する場合は、すべてのターゲット表に対する SELECT、INSERT、UPDATE および DELETE を、Replicat データベース・ユーザーに付与します。

1.6 サポートされている Teradata データ型

次の表は、Oracle GoldenGate でサポートされている Teradata データ型を示します。適用される制限事項および条件は、この表の後に記載されています。

表 1-1 Oracle GoldenGate でサポートされているデータ型 (Teradata バージョン別)

データ型	v12	v13	v13.1
BLOB	いいえ	はい	はい
BYTEINT	はい	はい	はい
VARBYTE	はい	はい	はい
BIGINT	はい	はい	はい
BYTEINT	はい	はい	はい
DATE	はい	はい	はい
DECIMAL (18 桁以下)	はい	はい	はい
DECIMAL (19 ~ 38 桁)	いいえ	はい	はい
DOUBLE PRECISION	はい	はい	はい
FLOAT	はい	はい	はい
INTEGER	はい	はい	はい
NUMERIC (18 桁以下)	はい	はい	はい
NUMERIC (19 ~ 38 桁)	いいえ	はい	はい
REAL	はい	はい	はい
SMALLINT	はい	はい	はい
TIME	はい	はい	はい
TIMESTAMP	はい	はい	はい
INTERVAL	はい	はい	はい
INTERVAL DAY	はい	はい	はい

表 1-1 Oracle GoldenGate でサポートされているデータ型 (Teradata バージョン別) (続き)

データ型	v12	v13	v13.1
INTERVAL DAY TO HOUR	はい	はい	はい
INTERVAL DAY TO MINUTE	はい	はい	はい
INTERVAL DAY TO SECOND	はい	はい	はい
INTERVAL HOUR	はい	はい	はい
INTERVAL HOUR TO MINUTE	はい	はい	はい
INTERVAL HOUR TO SECOND	はい	はい	はい
INTERVAL MINUTE	はい	はい	はい
INTERVAL MINUTE TO SECOND	はい	はい	はい
INTERVAL MONTH	はい	はい	はい
INTERVAL SECOND	はい	はい	はい
INTERVAL YEAR	はい	はい	はい
INTERVAL YEAR TO MONTH	はい	はい	はい
CHAR	はい	はい	はい
CLOB	いいえ	はい	はい
CHAR VARYING	はい	はい	はい
LONG VARCHAR	はい	はい	はい
VARCHAR	はい	はい	はい
GRAPHIC	はい	はい	はい
LONG VARGRAPHIC	はい	はい	はい
VARGRAPHIC	はい	はい	はい
PERIOD (DATE)	いいえ	はい	はい
PERIOD (TIME)	いいえ	はい	はい
PERIOD (TIMESTAMP)	いいえ	はい	はい
UDT	いいえ	はい	はい

1.6.1 数値データ型に関するサポートの制限事項

数値データ型は、Teradata のソース・データベースとターゲット・データベース間で完全にサポートされています。ソース・データベースで Teradata よりも多い桁数がサポートされている場合に、それらのデータ型を別のタイプのデータベースから Teradata にレプリケートすると、切捨てが発生する可能性があります。

サポートされる浮動小数点数の範囲と精度は、ホスト・マシンによって異なります。一般的に、精度は 16 桁まで正確ですが、データベースのドキュメントを確認して必要な近似値を決定する必要があります。Oracle GoldenGate では、サポートされている精度を超える値は丸められるか切り捨てられます。

1.6.2 シングルバイト文字データ型に関するサポートの制限事項

シングルバイト文字型は、Teradata のソースと Teradata のターゲット間と他のデータベースと Teradata 間に設定されているシングルバイトのラテン文字内で完全にサポートされています。VARCHAR 列または CHAR 列の長さは、32k-1 バイトを超えることはできません。UTF-16 を使用する場合、この長さは 16k-2 文字です。

1.6.3 マルチバイト文字データに関するサポートの条件および制限事項

- Windows レプリケーション・サーバーまたは Linux レプリケーション・サーバーに Oracle GoldenGate をインストールしてください。
- Teradata ODBC ドライバ 12.0.0.x またはそれ以降のバージョンを使用してください。
- マルチバイト・データ型にフィルタリング、マッピングおよび変換を使用しないでください。
- ソースの Teradata 表に使用できるのは、CHAR、VARCHAR、INTEGER、SMALLINT、DATE、TIME および TIMESTAMP 列のみです。マルチバイト・データ型をレプリケートしている間は、その他のデータ型をレプリケートすることはできません。
- CHAR 列または VARCHAR 列の長さは、32k-1 バイトを超えることはできません。UTF-16 を使用する場合、それらの列の長さは 16k-2 文字を超えることはできません。
- 初期化ファイルで ODBC ドライバおよび Teradata Access Module (TAM) を UTF-16 キャラクタ・セットに設定します (参照)。
- Replicat グループを作成するときに、ADD REPLICAT コマンドで NOBDCHECKPOINT オプションを使用してください。Replicat データベースのチェックポイント機能では、UTF-16 文字セットに設定されている ODBC ドライバはサポートされません。チェックポイントはディスク上のチェックポイント・ファイルに保持されます。
- Linux 上で Unicode をサポートするには、Oracle GoldenGate をパススルー・モードで構成し、ODBC Teradata Linux ドライバを 12.0.0.7 以降のバージョンあるいは 13.0.0.5 以降のバージョンにする必要があります。パススルー・モードの詳細は、『Oracle GoldenGate Windows and UNIX リファレンス・ガイド』の PASSTHRU パラメータに関する項を参照してください。

1.6.4 バイナリ・データ型に関するサポートの制限事項

制限事項はありません。これらのデータ型は、ソースの Teradata とターゲットの Teradata 間および他のソース・データベースとターゲットの Teradata 間でサポートされています。

1.6.5 ラージ・オブジェクト・データ型に関するサポートの制限事項

- UDT をレプリケートするには、ターゲット・データベースを Teradata Database 12.00.00.01 以降にする必要があります。
- ラージ・オブジェクトを他のデータベースから Teradata にレプリケートするには、ターゲット・システムで Teradata ODBC ドライバ 12.0 以降のバージョンを使用してください。ターゲットでは、ODBC によって配布されるラージ・オブジェクトをサポートする必要があります。
- ODBC 構成ファイルで UseNativeLOBSupport フラグを有効にしてください。Teradata ODBC ドキュメントを参照してください。

1.6.6 日付データ型に関するサポートの制限事項

- 日付型は、Teradata のソース・データベースとターゲット・データベース間で完全にサポートされています。さらに、ターゲット列のサイズがソースのサイズ以上である場合には、Teradata と Oracle 間で INTERVAL がサポートされます。
- DATE、TIME および TIMESTAMP は、別のタイプのソース・データベースから Teradata にレプリケートするときに完全にサポートされます。

- TIME with TIMESZONE、TIMESTAMP with TIMEZONE および INTERVAL は、別のタイプのソース・データベースから Teradata にレプリケートするときにサポートされません。
- Oracle GoldenGate では、マイナスの日付はサポートされていません。

1.6.7 IDENTITY データ型に関するサポートの制限事項

Replicat によって正しい値が挿入されるように、ターゲットで IDENTITY を GENERATED BY DEFAULT AS IDENTITY として構成する必要があります。双方向のレプリケーション構成で IDENTITY を含めるには、ソース・システムおよびターゲット・システムで定義されている値の範囲をばらばらにする必要があります(たとえば、一方で奇数、他方で偶数)。

1.7 Teradata でサポートされているオブジェクトおよび操作

この項では、Oracle GoldenGate でサポートされているデータ操作およびデータベース・オブジェクトをリストします。

1.7.1 DML

- Oracle GoldenGate では、データベースによってサポートされている表ごとに最大列数がサポートされています。

1.7.2 DDL

次のいずれか 1 つの条件を満たしている場合に、Teradata DDL ステートメントをレプリケートできます。

- DDL ステートメントがレプリケーション・グループのメンバーである表に作用すること。
- DDL ステートメントがユーザー定義のレプリケーション・ルールに一致すること。
- DDL ステートメントによって、レプリケーション・グループの特定のプロパティが変更されること。

Oracle GoldenGate では、次の Teradata DDL 操作の抽出およびレプリケーションがサポートされており、文の長さは最大 2MB です。少なくとも TAM 13.0 が必要です。ソースおよびターゲットの両方のデータベースを Teradata Database 13.0 またはそれ以降のバージョンにする必要があります。

表 1-2 サポートされている Teradata DDL

操作	オブジェクト
CREATE	TABLE <i>table name</i> ¹
	GLOBAL TEMPORARY TABLE <i>table name</i> ²
	[RECURSIVE] VIEW <i>view name</i>
	MACRO <i>macro name</i>
	HASH INDEX <i>index name</i>
	JOIN INDEX <i>index name</i>
	TRIGGER <i>trigger name</i>
ALTER	TABLE
DROP	TABLE <i>table name</i>
	VIEW <i>view name</i>
	MACRO <i>macro name</i>
	HASH INDEX <i>index name</i>
	JOIN INDEX <i>index name</i>
	TRIGGER <i>trigger name</i>

表 1-2 サポートされている Teradata DDL(続き)

操作	オブジェクト
RENAME	TABLE <i>table name</i> TO VIEW <i>view name</i> TO MACRO <i>macro name</i> TRIGGER <i>trigger name</i>
GRANT ... ON	TABLE <i>table name</i>
REVOKE ... ON	VIEW <i>view name</i> MACRO <i>macro name</i>
REPLACE	[RECURSIVE] VIEW <i>view name</i> TRIGGER <i>trigger name</i> MACRO <i>macro name</i>
COMMENT ON ³	TABLE <i>table name</i> COLUMN <i>table name.column name</i> VIEW <i>view name</i> COLUMN <i>view name.column name</i> MACRO <i>macro name</i> TRIGGER <i>trigger name</i>
COLLECT STATISTICS ON ⁴	<i>table name</i>
DROP STATISTICS ON	

¹ レプリケーション・グループのメンバーである表の DDL 操作が自動的にキャプチャされます。

² 表の一時的な実体化状態を参照する DDL ステートメントはレプリケートできません。

³ データ・ディクショナリにユーザー定義のオブジェクト記述を作成する COMMENT 文のみが取得されま
す。

⁴ 表のアクセスおよび結合計画を生成するためにオプティマイザで使用するオプティマイザ・フォーム
のみがキャプチャされます。

注意: DDL サポートの実際のサイズには、文のテキストに加え、オブジェクト名の長さ、DDL 型および内部での DDL レコード保持のその他の特性に応じて異なる Oracle GoldenGate のメンテナンス・オーバーヘッドも含まれるため、実際のサイズ制限は概算です。

レプリケーション・グループのプロパティを変更する次の DDL ステートメントは、自動的にレプリケートされます。

- ADD 句/DROP 句を含む ALTER REPLICATION GROUP
- CREATE REPLICATION RULESET
- REPLACE REPLICATION RULESET
- DROP REPLICATION RULESET

注意: 新しいセキュリティ・トークンの生成に使用される ALTER REPLICATION GROUP 文は、レプリケートされません。

その他のサポートの制限事項および構成手順の詳細は、『Oracle GoldenGate Windows and UNIX 管理者ガイド』を参照してください。

Oracle GoldenGate のインストール

次の手順は、Oracle GoldenGate を初めてインストールする場合に実行します。Oracle GoldenGate をあるバージョンから別のバージョンにアップグレードするには、次にある手順に従ってください。

<http://www.oracle.com/technology/software/products/goldengate/index.html>

Oracle GoldenGate をインストールすると、処理の実行および管理に必要なすべてのコンポーネント (ドライバ、ライブラリなど、他のベンダーからリクエストされるコンポーネントを除く) がインストールされ、さらに Oracle GoldenGate のユーティリティがインストールされます。

インストール・プロセスは短時間で終了します。

2.1 インストールの概要

Oracle GoldenGate をインストールするには、次の手順が必要です。

- [2.2 項「Oracle GoldenGate のダウンロード」](#)
- [2.3 項「UNIX 上での動的ビルドのライブラリ・パスの設定」](#)
- [2.4 項「クラスタへの Oracle GoldenGate のインストール準備」](#)
- [2.5 項「Linux および UNIX 上での Oracle GoldenGate のインストール」](#)
- [2.6 項「Windows への Oracle GoldenGate のインストール」](#)
- [2.7 項「クラスタへの Oracle GoldenGate の統合」](#)

2.2 Oracle GoldenGate のダウンロード

Oracle GoldenGate 構成に含める各システムに、Oracle GoldenGate の適切なビルドをダウンロードします。

1. <http://edelivery.oracle.com> に移動します。
2. 「ようこそ」ページで次の操作を実行します。
 - 使用する言語を選択します。
 - 「**続行**」をクリックします。
3. 「輸出確認」ページで次の操作を実行します。
 - 識別情報を入力します。
 - 「トライアル・ライセンス契約」を受け入れます (永久ライセンスを持っている場合でも)。
 - 「輸出規制」を受け入れます。
 - 「**続行**」をクリックします。

4. 「メディア・パック検索」ページで次の操作を実行します。
 - 「Oracle Fusion Middleware」製品パックを選択します。
 - ソフトウェアをインストールするプラットフォームを選択します。
 - 「実行」をクリックします。
5. 「結果リスト」で次の操作を実行します。
 - ダウンロードするメディア・パックを選択します。
 - 「続行」をクリックします。
6. 「ダウンロード」ページで次の操作を実行します。
 - 必要なコンポーネントごとに「ダウンロード」をクリックします。自動ダウンロード・プロセスに従って mediapack.zip ファイルをシステムに転送します。

注意： ソフトウェアをインストールする前に、新機能、新しい要件または現在の構成に影響するバグ修正を確認します。readme ファイルで既知の問題を確認してください。

2.3 UNIX 上での動的ビルドのライブラリ・パスの設定

Oracle GoldenGate では、共有ライブラリを使用します。UNIX システムに Oracle GoldenGate をインストールするとき、GGSCI または他の Oracle GoldenGate プロセスを実行する前に次の手順を実行しておく必要があります。UNIX システム上の Oracle GoldenGate インストール・ディレクトリ外から Oracle GoldenGate プログラムを実行する場合、次の操作を実行します。

- (オプション) Oracle GoldenGate インストール・ディレクトリを PATH 環境変数に追加します。
- (必須) Oracle GoldenGate インストール・ディレクトリを共有ライブラリの環境変数に追加します。

たとえば、Oracle GoldenGate インストール・ディレクトリ /users/ogg の場合、次の例の 2 番目のコマンドにこれらの変数を設定する必要があります。

表 2-1 ライブラリ変数を必要とするコマンド

コマンド	環境変数に GG ライブラリが必要であるか
\$ users/ogg > ./ggsci	いいえ
\$ users > ./ogg/ggsci	はい

Korn シェルで環境変数を設定する場合

```
PATH=installation_directory:$PATH
export PATH
shared_libraries_variable=absolute_path_of_installation_directory:$shared_libraries_variable
export shared_libraries_variable
```

Bourne シェルで環境変数を設定する場合

```
export PATH=installation_directory:$PATH
export shared_libraries_variable=absolute_path_of_installation_directory:$shared_libraries_variable
```

C シェルで変数を設定する場合

```
setenv PATH installation_directory:$PATH
setenv shared_libraries_variable absolute_path_of_installation_directory:$shared_libraries_variable
```

ここで、*shared libraries variable* は、表 2-2 に示されている変数のいずれかです。

表 2-2 UNIX/Linux ライブラリ・パス変数 (プラットフォーム別)

プラットフォーム ¹	環境変数
IBM AIX	LIBPATH
IBM z/OS	
HP-UX	SHLIB_PATH
Sun Solaris	LD_LIBRARY_PATH ²
HP Tru64 (OSF/1)	
LINUX	

¹ 特定のプラットフォームは、Oracle GoldenGate で、ご使用のデータベース用にサポートされている場合とサポートされていない場合があります。

² 32 ビットの Oracle データベースが実行されている 64 ビット環境では、32 ビットの Oracle ライブラリを含めるために、Oracle GoldenGate に LD_LIBRARY_PATH が必要です。

例

```
export LD_LIBRARY_PATH=/ggs/10.0:$LD_LIBRARY_PATH
```

注意： Oracle GoldenGate プロセスに必要なライブラリを表示するには、そのプロセスを開始する前に `ldd goldengate_process` シェル・コマンドを使用します。欠落がある場合には、このコマンドによってエラー・メッセージも表示されます。

2.4 クラスタへの Oracle GoldenGate のインストール準備

このトピックでは、Oracle GoldenGate をクラスタ環境にインストールする際に適用されるインストール要件を説明します。Oracle GoldenGate は、Oracle で動作保証されている任意のクラスタ管理ソリューションと組み合わせて使用できます。

2.4.1 Oracle GoldenGate のバイナリおよびファイルをインストールするクラスタ内の場所の決定

少なくともいくつかの Oracle GoldenGate オブジェクトを共有記憶域にインストールする必要があります。クラスタのすべてのノードから独立していて、すべてのノードで使用可能な、クラスタ対応の共有記憶域を選択してください。ベスト・プラクティスは、Oracle GoldenGate 全体を共有記憶域にインストールすることです。この場合、パラメータ・ファイルを変更せずに、任意のノードから Oracle GoldenGate プロセスを開始できます。アクティブ・ノードが停止した場合でも、インストール・ディレクトリに保存されている処理チェックポイントを使用して、すぐに別のノードでプロセスを開始できます。

共有記憶域ではなく、各ノードに Oracle GoldenGate のバイナリおよびファイルをインストールする場合は、次に当てはまる必要があります。

- Oracle GoldenGate インストールの場所のパスが、すべてのノードで同一であることが必要です。
- Oracle GoldenGate のリカバリ要件をサポートするには、少なくとも、次のディレクトリを共有記憶域にインストールしてください。UNIX または Linux では、各ノードのインストール・ディレクトリから、それらに対してシンボリック・リンクを作成できます。
 - br
 - dirchk
 - dirdat
 - dirtmp

これらは、インストール中に CREATE SUBDIRS を発行した際に作成されたディレクトリです。

- dirprm ディレクトリのパラメータ・ファイルが共有ドライブに配置されていない場合は、すべてのノードで同一であることが必要です。ノード間で別々にする必要がある環境設定を解決するために、ローカルの Manager プロセスから継承するように環境設定を行うことや、ノード固有の Oracle GoldenGate マクロ・ファイルを参照することが可能です。このシナリオを強制するのが困難な可能性もあるため、共有ドライブにパラメータ・ファイルを格納しておくことで、継承の考慮事項を回避できます。

Oracle GoldenGate をインストールしたら、[2.7 項「クラスタへの Oracle GoldenGate の統合」](#)も参照してください。

2.5 Linux および UNIX 上での Oracle GoldenGate のインストール

Linux または UNIX システム、あるいはクラスタ内の適切な場所に Oracle GoldenGate for Oracle をインストールするには、次の手順に従います。詳細は、[2.4 項「クラスタへの Oracle GoldenGate のインストール準備」](#)を参照してください。

2.5.1 Oracle GoldenGate ファイルのインストール

1. Oracle GoldenGate の mediapack.zip ファイルを、Oracle GoldenGate をインストールするシステムおよびディレクトリに抽出します。

2. コマンド・シェルを実行します。

3. ディレクトリを、新しい Oracle GoldenGate ディレクトリに変更します。

4. Oracle GoldenGate ディレクトリから GGSCI プログラムを実行します。

```
GGSCI
```

5. GGSCI では、次のコマンドを発行して Oracle GoldenGate の作業ディレクトリを作成します。

```
CREATE SUBDIRS
```

6. 次のコマンドを発行して GGSCI を終了します。

```
EXIT
```

7. Teradata Access Module (TAM) ライブラリをレプリケーション・サーバーにある Oracle GoldenGate のルート・ディレクトリにインストールします。TAM は、*Vendor Access Module (VAM)* として知られている Oracle GoldenGate API と通信します。VAM によって、トランザクション・データの変更が Extract プロセスに渡されます。正しい TAM のバージョンと Teradata のバージョンを組み合わせる手順および Teradata データベースと Oracle GoldenGate を併用するために TAM を構成する手順については、『*Teradata Replication Services Using Oracle GoldenGate*』のドキュメントを参照してください。一般に、TAM のバージョンはデータベースのバージョンと一致させる必要があります。

2.6 Windows への Oracle GoldenGate のインストール

Windows システムまたはクラスタ内の適切な場所に Oracle GoldenGate for Oracle をインストールするには、次の手順に従います。詳細は、[2.4 項「クラスタへの Oracle GoldenGate のインストール準備」](#)を参照してください。

2.6.1 Windows クラスタへの Oracle GoldenGate のインストール

1. クラスタのノードの 1 つにログインします。

2. Oracle GoldenGate インストールの場所用のドライブを選択します。このドライブは、データベース・インスタンスが含まれているのと同じクラスタ・グループ内のリソースにする必要があります。
3. このクラスタ・グループがログインしているクラスタのノードによって所有されていることを確認します。
4. 次の手順に従って Oracle GoldenGate をインストールします。

2.6.2 Oracle GoldenGate ファイルのインストール

1. WinZip または同等の圧縮製品を使用して、ダウンロードしたファイルを解凍します。
2. Oracle GoldenGate をインストールするドライブ上のフォルダに、それらのファイルをバイナリ・モードで移動します。パスが引用符で囲まれている場合でも、名前に空白が含まれているフォルダに Oracle GoldenGate をインストールしないでください。次に例を示します。

C:\ "Oracle GoldenGate " は無効です。

C:\Oracle_GoldenGate は有効です。

3. Oracle GoldenGate フォルダから GGSCI プログラムを実行します。
4. GGSCI では、次のコマンドを発行して Oracle GoldenGate の作業ディレクトリを作成します。

```
CREATE SUBDIRS
```

5. 次のコマンドを発行して GGSCI を終了します。

```
EXIT
```

6. Teradata Access Module (TAM) ライブラリをレプリケーション・サーバーにある Oracle GoldenGate のルート・ディレクトリにインストールします。TAM は、*Vendor Access Module (VAM)* として知られている Oracle GoldenGate API と通信します。VAM によって、トランザクション・データの変更が Extract プロセスに渡されます。正しい TAM のバージョンと Teradata のバージョンを組み合わせる手順および Teradata データベースと Oracle GoldenGate を併用するために TAM を構成する手順については、『*Teradata Replication Services Using Oracle GoldenGate*』のドキュメントを参照してください。一般に、TAM のバージョンはデータベースのバージョンと一致させる必要があります。

2.6.3 Manager のカスタム名の指定

次のいずれかが当てはまる場合には、Manager プロセスのカスタム名を指定する必要があります。

- 既定の GGSMDR 以外に Manager の名前を使用する場合。
- このシステム上に Windows サービスとして実行している Manager プロセスが複数存在する場合。システム上の各 Manager の名前を一意にする必要があります。次の手順に進む前に、ローカルの Manager サービスの名前をメモしてください。

Manager のカスタム名を指定する手順

1. Manager プログラムが格納されているディレクトリから、GGSCI を実行します。
2. 次のコマンドを発行します。

```
EDIT PARAMS ./GLOBALS
```

注意：GLOBALS ファイルは Oracle GoldenGate インストール・ファイルのルートに存在する必要があるため、このコマンドでは ./ を使用する必要があります。

3. ファイルに次の行を追加します (`name` は、Manager サービスの 1 語で構成される名前です)。
MGRSERVNAME `name`
4. ファイルを保存します。ファイルは、GLOBALS の名前でファイル拡張子を付けずに自動的に保存されます。このファイルを移動しないでください。これは、Windows サービスのインストール時とデータ処理時に使用されます。

2.6.4 Windows サービスとしての Manager のインストール

既定では、Manager は、サービスとしてインストールせず、ローカル・アカウントまたはドメイン・アカウントによって実行可能です。ただし、このように実行すると、ユーザーがログアウトしたときに Manager が停止してしまいます。Manager をサービスとしてインストールすると、ユーザーの接続に関係なく Manager を操作でき、さらに手動で、あるいはシステムの起動時に Manager を開始するように Manager を構成できます。

Windows クラスタの場合に Manager をサービスとしてインストールする必要がありますが、それ以外の場合はオプションです。

Windows サービスとして Manager をインストールする手順

1. (推奨) システム管理者としてログオンします。
2. 「スタート」 → 「ファイル名を指定して実行」をクリックし、「ファイル名を指定して実行」ダイアログ・ボックスに「cmd」と入力します。
3. サービスとしてインストールしている Manager プログラムが格納されているディレクトリから、次の構文を使用して INSTALL ユーティリティを実行します。

```
install option [...]
```

ここで、option は次のいずれかです。

表 2-3 INSTALL ユーティリティのオプション

オプション	説明
ADDEVENTS	Oracle GoldenGate のイベントを Windows Event Manager に追加します。デフォルトでは、Oracle GoldenGate のエラーは一般的です。より具体的なエラー内容を生成するには、Oracle GoldenGate インストール・ディレクトリから SYSTEM32 ディレクトリに次のファイルをコピーします。 category.dll ggsmg.dll
ADDSERVICE	存在する場合は GLOBALS ファイルの MGRSERVNAME パラメータに指定されている名前、またはデフォルトの GGSMDR で、Manager をサービスとして追加します。ユーザーのログインとパスワードの変更に関係なくサービスを実行できるため、ADDSERVICE は、ローカル・システム・アカウント (ほとんどの Windows アプリケーションの基準) で実行するようにサービスを構成します。Manager を特定のアカウントで実行するには、USER オプションと PASSWORD オプションを使用します。 ¹ サービスがシステムの起動時に開始するようにインストールされます (「AUTOSTART」を参照)。インストール後に開始するには、システムを再起動するか、「コントロールパネル」の「サービス」アプレットから手動でサービスを開始します。
AUTOSTART	ADDSERVICE を使用して作成されたサービスをシステムの起動時に開始するよう設定します。MANUALSTART を使用しないかぎり、このオプションが既定です。
MANUALSTART	ADDSERVICE を使用して作成されたサービスを、GGSCI、スクリプトまたは「コントロールパネル」の「サービス」アプレットから手動で開始するよう設定します。デフォルトは AUTOSTART です。

表 2-3 INSTALL ユーティリティのオプション (続き)

オプション	説明
USER name	Manager を実行するドメイン・ユーザー・アカウントを指定します。name については、HEADQT\GGSMGR などのようにドメイン名、バックスラッシュおよびユーザー名を入力します。 既定では、Manager は、Local System アカウントを使用するようにインストールされます。
PASSWORD password	USER を使用して指定されたユーザーのパスワードを指定します。

¹ ユーザー・アカウントを変更するには、Windows の「Control Panel」の「Services」アプレットから「Properties」アクションを選択します。

4. (Windows Server 2008)Windows User Account Control(UAC)が有効である場合、コンピュータへのプログラム・アクセスを許可するか、または拒否するかを尋ねるプロンプトが表示されます。「許可」を選択して、INSTALL ユーティリティを実行できるようにします。

INSTALL ユーティリティによって、管理者権限があるローカル・システム・アカウントで Manager サービスがインストールされます。サービスとしてインストールすると、Manager の実行時に UAC によるプロンプトが表示されません。

注意: Manager をサービスとしてインストールしなかった場合には、GGSCI コマンド・プロンプトから開始するときに、Oracle GoldenGate ユーザーに対して Manager の権限レベルを確認するように UAC のプロンプトが表示されます。Oracle GoldenGate の他のプログラムの実行でもプロンプトが表示されます。

2.7 クラスタへの Oracle GoldenGate の統合

クラスタに Oracle GoldenGate をインストールした場合は、次の手順を実行して、クラスタ・ソリューションに Oracle GoldenGate を統合します。

2.7.1 クラスタの一般的な要件

1. Oracle GoldenGate Manager プロセス (および Manager のみ) を、その他のアプリケーションと同じように、クラスタ管理のリソースとして登録します。Manager はその他すべてのプロセスを管理する親プロセスであるため、クラスタ管理のソフトウェアが開始および停止する唯一の Oracle GoldenGate プロセスであることが必要です。
2. クラスタで仮想 IP アドレスが使用されている場合は、Manager プロセス用に使用可能な固定 IP アドレスを取得する必要があります。VIP はパブリック・サブネット上の使用可能な IP アドレスであることが必要で、DHCP で決定することはできません。Extract データ・ポンプのパラメータ・ファイルに、RMTHOST パラメータの入力値としてリモート Manager の VIP を指定します。Manager にアクセスするその他の Oracle GoldenGate 製品でも、その VIP を使用する必要があります。
3. クラスタのすべてのノードで、システム・クロックが同期していることを確認します。時計は、Extract が実行されるシステムの時計と同期している必要があります。Oracle GoldenGate では、ローカル・システムの時間が、重要な決定を行うコミット・タイムスタンプと比較されます。システム・クロックの同期の詳細は、www.ntp.org を参照するか、システム管理者に問い合わせてください。『Oracle GoldenGate Windows and UNIX リファレンス・ガイド』の THREADOPTIONS パラメータの IOLATENCY オプションに関する項も参照してください。
4. Manager を構成する際は、Manager がレプリケーション・プロセスを自動的に開始するように、AUTOSTART および AUTORESTART パラメータを追加します。必要な場合は、Oracle GoldenGate のユーザー・インタフェースから、Extract、Replicat およびその他の Oracle GoldenGate プロセスを制御できます。

5. 1つのノードのみで共有ドライブにマウントします。これにより、プロセスが別のノードで開始されるのを防ぐことができます。すべてのノードで同じマウント・ポイントを使用します。
6. このドキュメントに記載されているように Oracle GoldenGate を構成します。

2.7.2 Windows クラスタ・リソースとしての Oracle GoldenGate の追加

Oracle GoldenGate を Windows クラスタにインストールした場合、次の手順に従って Oracle GoldenGate をクラスタ・リソースとして設定し、すべてのノード上で Manager サービスを正しく構成します。

- クラスタ管理者で、Oracle GoldenGate の接続先となるデータベースが含まれているグループに Manager プロセスを追加します。
- Oracle GoldenGate を実行するすべてのノードが、リソースの可能性のある所有者として選択されていることを確認します。
- Manager の Windows サービスに、次の依存性（「コントロール パネル」の「サービス」から構成可能）があることを確認します。
 - データベース・リソース
 - Oracle GoldenGate ディレクトリを含むディスク・リソース
 - データベースのトランザクション・ログ・ファイルを含むディスク・リソース
 - データベースのトランザクション・ログ・バックアップ・ファイルを含むディスク・リソース

Oracle GoldenGate システムの準備

この章では、Oracle GoldenGate をサポートするためのデータベースおよびシステムの準備について説明します。

3.1 処理に向けた表の準備

次の表属性は、Oracle GoldenGate 環境で処理する必要があります。

3.1.1 トリガーおよびカスケード制約を無効にする

ターゲットの Sybase 表でトリガー、カスケード削除制約およびカスケード更新制約を無効にしたり、Oracle GoldenGate データベースのユーザーによる変更を無視するためにそれらの制約を変更したりします。Oracle GoldenGate では、トリガーまたはカスケード制約によって生じる DML をレプリケートします。ターゲット表で同じトリガーまたは制約を有効にした場合、レプリケートされたバージョンのためにそのトリガーまたは制約が冗長になり、データベースによってエラーが返されます。ソース表が emp_src および salary_src、ターゲット表が emp_targ および salary_targ である場合の次の例を検討してください。

1. emp_src に削除が発行されます。
2. これによって、削除が salary_src にカスケードされます。
3. Oracle GoldenGate によって、両方の削除がターゲットに送信されます。
4. 親削除が最初に到達し、emp_targ に適用されます。
5. 親削除によって、削除が salary_targ にカスケードされます。
6. salary_src からカスケードされた削除が salary_targ に適用されます。
7. 行はすでに手順 5 で削除されたため、その行を見つけることはできません。

3.1.2 行識別子の割当て

Oracle GoldenGate では、レプリケートされた更新および削除の正しいターゲット行を検索するために、ソース表とターゲット表にいくつかの一意の行識別子が必要です。

3.1.2.1 使用する行識別子の種類を決定する Oracle GoldenGate の仕組み

TABLE または MAP 文に KEYCOLS 句が使用されている場合を除き、Oracle GoldenGate では、次の優先順位に従って使用する行識別子が選択されます。

1. 主キー
2. 英数字の順で最初の一意のキー (タイムスタンプもマテリアライズされていない計算結果列も含まれません)。

3. 前述のキー・タイプが存在しない場合 (他のキー・タイプが表に定義されている可能性があります)、Oracle GoldenGate では、Oracle GoldenGate でキーにサポートされていないもの、または Oracle GoldenGate 構成から除外されているものを除き、データベースで一意的キーに使用可能なすべての列の疑似列が作成されます。

注意: 使用できないその他のキーが表に存在する場合、または表にキーが 1 つも存在しない場合、Oracle GoldenGate では、レポート・ファイルに適切なメッセージが記録されます。すべての列からキーを作成すると、ソース・システムでの Oracle GoldenGate のパフォーマンスが低下します。ターゲットでは、このキーは Replicat であまり効率的でないより大きい WHERE 句が使用される原因となります。

3.1.2.2 KEYCOLS を使用したカスタム・キーの指定

表に前述のいずれのタイプの行識別子も存在しない場合、または前述の識別子を使用しない場合、表に常に一意の値が含まれる列が存在するときには、代替キーを定義できます。この代替キーは、Extract の TABLE パラメータおよび Replicat の MAP パラメータに KEYCOLS 句を指定して定義します。指定されたキーが、Oracle GoldenGate により検出される既存の主キーまたは一意のキーをオーバーライドします。詳細は、『Oracle GoldenGate Windows and UNIX リファレンス・ガイド』を参照してください。

3.2 Tearadata のレプリケーション・グループの作成

レプリケーション・グループを作成するには、CreateGroupStmtFile (Create Group 文) ファイルを使用します。CreateGroupStmtFile を使用することで、レプリケーション・グループの正しい識別子情報が自動的に tam.ini ファイルに書き込まれます。詳細は、『Teradata Replication Services Using Oracle GoldenGate』のドキュメントを参照してください。

相互に依存関係があるすべてのオブジェクトは、同じレプリケーション・グループで指定する必要があります。同じレプリケーション・グループ内にトランザクション全体を含める必要があります。

Create Group ステートメントのファイルを作成する手順

1. テキスト・エディタを使用してテキスト・ファイルを作成します。
2. 次の行を入力します。
 - Teradata のコマンド create replication group
 - Teradata レプリケーション・グループの名前
 - レプリケーション・グループに関連付けられる表リスト。表は 1 つのレプリケーション・グループにのみ関連付けることができます。Oracle GoldenGate の Extract グループごとに使用可能なレプリケーション・グループは 1 つのみです。
3. そのファイルを接尾辞 .sql を付けて、Oracle GoldenGate のインストール・ディレクトリ内のディレクトリ (ルート・レベルまたは dirtam などの作成するサブディレクトリ) に保存します。
4. TAM 初期化ファイルで CreateGroupStmtFile パラメータを使用して、このファイル名を指定します。詳細は、3.5 項「TAM 初期化ファイルの構成」を参照してください。

例 3-1 Create Group ステートメントのサンプル・ファイル

```
Create Replication Group HRTRG1 (HR.EMPLOYEE, HR.EMP_INFO, HR.EMP_DEPT, HR.EMP_REVIEWS);
```

3.3 Teradata TAM による DDL 取得のアクティブ化

Teradata TAM から Oracle GoldenGate に渡す DDL を指定するには、レプリケーション・グループにレプリケーションのルールセット文を作成します。ルールセット・ステートメントによって、1 つ以上の DDL キャプチャ・ルールのセットが作成され、指定したレプリケーション・グループにそれらのルールが関連付けられます。DDL 操作が実行されると、それらのルールが DDL ステートメントのターゲット・オブジェクトの名前および型に適用され、ただちにそれらのオブジェクトがレプリケーションで使用可能になります。レプリケーション・グループのメンバーである表の DDL 操作が自動的にキャプチャされます。

注意： Teradata の RSG は、適切に構成する必要があります。詳細は、『*Teradata Replication Services Using Oracle GoldenGate*』のドキュメントを参照してください。

Teradata TAM による DDL キャプチャをアクティブにする手順

1. REPLCONTROL 権限を持つユーザーとしてログインします。
2. レプリケーションのルールセットを作成します。

```
[CREATE | REPLACE] REPLICATION RULESET rule set name
[, DEFAULT]
FOR replication group name
AS rule specification [, rule specification]
```

条件：

- rule specification は次のようになります。

```
object kind LIKE string literal [ESCAPE character literal]
[AND NOT LIKE string literal [ESCAPE character literal] ]
```
- object kind は次のようになります。

```
TABLE | TEMPORARY TABLE | VIEW | MACRO | TRIGGER | INDEX
```

使用に関する要件

- レプリケーション・グループに表リストで明示的に定義されたメンバーを含めるか、レプリケーション・グループをレプリケーションのルールセットに関連付けられている空のグループにする必要があります。
- 文の CREATE フォームを使用し、指定したレプリケーション・グループに同じルール・セット名のルール・セットがすでに存在する場合には、CREATE 文が失敗します。
- 文の REPLACE フォームを使用し、指定したレプリケーション・グループに同じルール・セット名のルール・セットがすでに存在する場合には、既存のルールセットが新しいルール・セットに置き換えられます。
- DEFAULT オプションを使用した場合、ルール・セットにあるすべてのルールがデフォルトのルールとみなされます。他のルールがオブジェクトと一致しない場合、既定のルールが適用されます。
 - 既定以外のルールは、別のレプリケーション・グループに関連付けられている既定以外のルールと同じオブジェクトと一致してはなりません。
 - 既定のルールは、別のレプリケーション・グループに関連付けられている既定のルールと同じオブジェクトと一致してはなりません。
 - LIKE 句および NOT LIKE 句を使用してパターン文字列を指定し、SQL 文のオブジェクトの完全修飾名と照合します。パターン文字列には、ワイルドカード文字を含めることができます。パターンとオプションの ESCAPE 文字は、LIKE 述語演算子と同じように一緒に使用します。

レプリケーション・グループおよびルールセットの作成の詳細は、『*Teradata Replication Services Using Oracle GoldenGate*』のドキュメントを参照してください。

例

この例では、MyRepGroup という名前のレプリケーション・グループに Sales1 という名前のルール・セットを作成して、SalesDB という名前のデータベースに作成されている表を取得し、さらにビュー名に接尾辞 `_s` が付いていない SalesDB のビューに影響を与える DDL も取得します。

```
CREATE REPLICATION GROUP MyRepGroup
CREATE REPLICATION RULESET Sales1 FOR MyRepGroup AS
TABLE LIKE 'SalesDB.%',
VIEW LIKE 'SalesDB.%' AND NOT LIKE '%z_s' ESCAPE 'z'
```

アンダースコア (`_`) がワイルドカードとして処理されないようにエスケープ文字を使用することに留意してください。

Oracle GoldenGate の DDL レプリケーションを構成するには、『*Oracle GoldenGate Windows and UNIX 管理者ガイド*』を参照してください。

3.4 DDL の取得の無効化

この項では、レプリケーション・グループを保持する場合または削除する場合のいずれかのケースで、Teradata DDL の取得およびレプリケーションを無効化する手順を説明します。

どちらの場合にも REPLCONTROL 権限が必要です。

DDL の取得を無効化し、レプリケーション・グループを保持する場合

```
DROP REPLICATION RULESET rule set name FOR replication group name
```

例：

```
DROP REPLICATION RULESET Sales1 FOR MyRepGroup;
```

DDL の取得を無効化し、レプリケーション・グループを削除する場合

```
DROP REPLICATION GROUP group name [ ( table name [, ... ] )]
```

例：

```
DROP REPLICATION GROUP MyRepGroup
```

3.5 TAM 初期化ファイルの構成

Teradata Access Module(TAM) によって、Extract プロセスに Teradata 環境へのリンクが提供されます。TAM を構成するには、初期化ファイルを作成します。

1. テキスト・エディタを使用してテキスト・ファイルを作成します。
2. 次の必須パラメータをファイルに追加します。必要に応じて、一覧に記載されているオプション・パラメータも使用できます。例については、[例 3-2](#) を参照してください。

表 3-1 TAM 初期化ファイルの必須パラメータ

パラメータ	説明
Mode={Replication}	必須 レプリケーション・モードをチェンジ・データ・キャプチャにすることを指定します。チェンジ・データ・キャプチャに Extract を構成するには、『 <i>Oracle GoldenGate Windows and UNIX 管理者ガイド</i> 』を参照してください。

表 3-1 TAM 初期化ファイルの必須パラメータ (続き)

パラメータ	説明
DictOdbcConnString=ODBC connection string for metadata	必須 辞書表へのアクセス権限を持つユーザーのログオン文字列。
MgmtOdbcConnString= ODBC connection string for management functions	必須 管理機能を実行する権限を持つユーザーのログオン文字列 (CREATE REPLICATION GROUP など)。このログオンには、REPLCONTROL 権限が必要です。
CreateGroupStmtFile= name	TAM12 およびそれ以降のバージョンに必須 新しいグループの CREATE REPLICATION GROUP 文が含まれている Create Group 文のファイル名。Create Group 文のファイルを使用してレプリケーション・グループを作成しなかった場合には、このパラメータを省略またはコメントアウトし、GroupID パラメータと SecurityToken パラメータを使用します。
GroupID= ID	SecurityToken を使用する場合に必須 TAM に関連付けられるレプリケーション・グループの ID。GroupID を使用する場合、SecurityToken を使用する必要があります。 このコマンドを使用して、レプリケーション・グループの ID を表示できます。 HELP REPLICATION GROUP rep group name; 例: help replication group g1; ID は、グループの名の横にある Identifier 列にある数値 ID です。
SecurityToken= token	Create Group ステートメントを使用してレプリケーション・グループを作成しなかった場合に必須 TAM に関連付けられるセキュリティ・トークン。Create Group 文のファイルを使用してグループを作成し、CreateGroupStmtFile パラメータを使用してそのファイルを指定した場合、SecurityToken パラメータおよび GroupID パラメータはランタイムで自動的に生成されるため、それらのパラメータを省略できます。
AltControlRSG= IP または name	オプション プライマリ・サーバーが失敗した場合に中央の RSG として継承できるサーバーの IP アドレスまたは名前を指定します。
CharacterSet= {ASCII UTF16}	必須 このレプリケーション・グループの文字セット。UTF16 に関連するオーバーヘッドが多くなるため、必須の場合にのみ UTF16 を使用します。
ControlRSG=address[:port] DataRSG1=address[:port] [DataRSG2=address[:port]] [...]	必須 RSG のノード・アドレス。これはノード名または IP アドレスにすることができます。これにはオプションのポート番号を付加できます。中央の RSG はシステムの RSG で最上位の番号にする必要があります。
Encryption= {None Control Data All}	必須 暗号化するメッセージのタイプ。データ・メッセージだけの暗号化とデータ・メッセージと制御メッセージの両方の暗号化では、オーバーヘッドに少し違いがあります。All を使用して暗号化を指定するか、None を使用して暗号化なしを指定します。

表 3-1 TAM 初期化ファイルの必須パラメータ (続き)

パラメータ	説明
RsgTimeoutSec=number of seconds	オプション データの RSG ポーリング時のタイムアウト (秒)。デフォルトは 1 です。有効な範囲は 0 から 60 秒です。
RsgTimeoutMSec=number of milliseconds	オプション データの RSG ポーリング時のタイムアウト (ミリ秒)。デフォルトは 0 です。有効な範囲は 0 から 60000000 ミリ秒です。
Tracing= { Debug Performance All None}	オプション デバッグ・トレースのレベル。 デフォルトは None です。
MaxProtTransCompleteThresh=number of transactions	オプション、最大保護モードに有効 保持できる未処理のトランザクション数で、これを超えると、Oracle GoldenGate のチェックポイントをリクエストする必要があります。これは、最大保護モードで動作する際に重要となります。高速でトランザクションを適用するセッションが大量にある場合には、このパラメータを 10 などの大きい値に設定できます。ただし、セッション数が少ない場合やセッション速度が遅い場合、このパラメータを 1 から 4 などの小さい値に設定し、待ち時間を最小限に抑え、スループットを最大化することができます。デフォルトは 0 です。有効な範囲は 0 から 24 トランザクションです。
Bidirectional {TRUE FALSE}	オプション データの前イメージを Oracle GoldenGate に送信するかどうかを指定します。Extract パラメータ GETUPDATEBEFORES を使用する場合には、TRUE にします。デフォルトは FALSE で、CDC オーバーヘッドおよび使用される通信帯域幅を減らすためにデータの後イメージのみが送信されます。

- そのファイルを tam.ini という名前の ASCII ファイルとして、Oracle GoldenGate のインストール・ディレクトリ内 (ルート・レベルまたは dirtam などの作成するサブディレクトリ) に保存します。
- Extract パラメータ・ファイルで VAM パラメータを使用して、このファイル名を指定します。

例 3-2 EXASample TAM 初期化ファイル。

```

Mode=Replication
DictOdbcConnString=DSN=myDsn;uid=myUser;pwd=myPass
MgmtOdbcConnString=DSN=myDsn;uid=myUser;pwd=myPass
CreateGroupStmtFile=c:\GGS\Teradata\dirtam\hrtrg1.sql
CharacterSet=ASCII
ControlRSG=10.10.10.49:1152
DataRSG1=10.10.10.50:1153
DataRSG2=node3
DataRSG3=node4:1155
Encryption=All

```


Oracle GoldenGate の構成

Oracle GoldenGate は、ソースの Teradata データベースの Change Data Capture (CDC) コンポーネントと連携して、**最大保護モード**または**最大パフォーマンス・モード**のいずれかのモードで動作します。モードによって、使用されるコミット・プロトコルと、Oracle GoldenGate が Teradata のアプリケーションに影響を与えるかどうかが決まります。

- [最大保護モードでの Oracle GoldenGate の構成](#)
- [最大パフォーマンス・モードでの Oracle GoldenGate の構成](#)

4.1 最大保護モードでの Oracle GoldenGate の構成

最大保護モードは、Teradata データベースに推奨される Oracle GoldenGate の構成です。最大保護モードでは、ソース・サーバー上の CDC とレプリケーション・サーバー上のプライマリ Extract プロセス間 (TAM 経由) で 2 フェーズ・コミット・プロトコルを使用することで、Oracle GoldenGate を本番システムに組み込みます。通信が中断したり、コンポーネントが失敗したりした場合にトランザクションが送信中に失われたり、重複したりしないように、2 フェーズ・コミットでは、ソース・データベースだけでなく、Oracle GoldenGate にコミットされるソース・トランザクションも必要です。

このモードでは、すべてのデータを受信し、ディスク上の Oracle GoldenGate VAM 証跡に保存したことをプライマリ Extract で認識されるまで、トランザクションは未決定です (コミットされません)。

- 一定のタイムアウト期間内に CDC で確認を受信した場合、アプリケーションへのコミット、データベースへのコミットおよび Oracle GoldenGate による伝播にトランザクションが CDC によって解放されます。
- 一定のタイムアウト期間内に CDC で確認を受信しなかった場合、CDC によってトランザクションがロールバックされ、エラー・メッセージがアプリケーションのユーザーに戻されます。

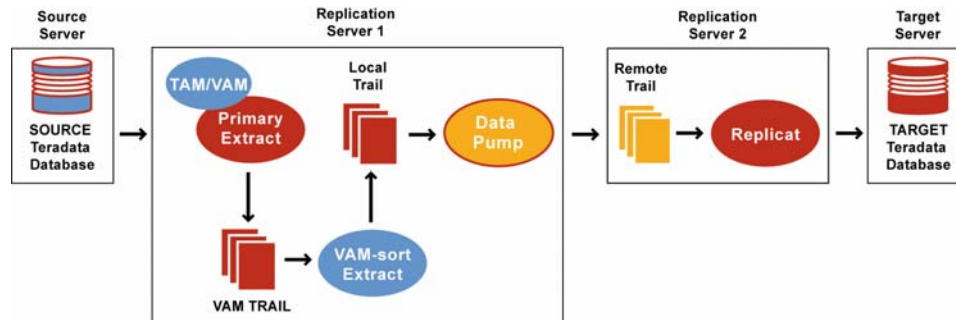
VAM 証跡は、トランザクション・ログと同様の一連のファイルです。これには、受信データが受信順に保存されますが、必ずしもトランザクション順であるとはかぎりません。VAM ソートによる Extract として知られているセカンダリ Extract プロセスによって、データがトランザクション順にソートされ、ロールバックを受信した場合 (2 フェーズ・コミットが失敗したため) にはトランザクションが削除されるか、さらに処理するためにトランザクションが定期的な証跡に解放されます。

4.1.1 推奨される最大保護の構成

別々のレプリケーション・サーバーに Extract プロセスと Replicat プロセスをインストールし、Extract プロセスがインストールされているサーバー上でデータ・ポンプとローカル証跡を併用します。この構成では、プライマリ Extract グループによってデータが取得され、VAM ソートによる Extract によってソートされたデータがローカル・ディスクでの定期的な Oracle GoldenGate 証跡まで保持されます。データ・ポンプ Extract によってこの証跡が読み取られ、そのデータが TCP/IP 経由で Replicat レプリケーション・サーバー上の証跡に送信されますが、そのデータが Replicat プロセスによって再度読み取られ、ターゲットに適用されます。

Extract サーバーと Replicat サーバー間の通信が失敗した場合に影響を受けるのは、データ・ポンプだけです。機能停止が続いても、他の 2 つの Extract プロセスでは、メモリーが不足することなく作業を続行できます。

図 4-1 推奨される最大保護の構成



4.1.2 最大保護モードでの Extract の構成

ソースのレプリケーション・サーバー上で次の手順を実行します。

1. 『Oracle GoldenGate Windows and UNIX 管理者ガイド』の手順に従って Manager プロセスを構成します。

2. Manager パラメータ・ファイルでは、PURGEOLDEXTRACTS パラメータを使用してローカル証跡からのファイルの削除を制御します。

3. GGSCI を実行します。

4. プライマリ Extract グループを作成します。ドキュメントで使用する目的で、このグループを ext と表します。

```
ADD EXTRACT ext, VAM
```

5. VAM 証跡にするローカル証跡を作成します。

```
ADD EXTTRAIL VAM_trail, EXTRACT ext
```

EXTRACT 引数を使用して、この証跡をプライマリ Extract グループにリンクします。その Extract グループによってこの証跡が VAM 証跡として作成されます。

6. EDIT PARAMS コマンドを使用して、プライマリ Extract グループにパラメータ・ファイルを作成します。データベース環境に適用される他のパラメータに加え、例 4-1 に示されているパラメータを含めてください。

注意： データベース・パスワード、証跡内のデータおよび TCP/IP 経由で送信されるデータの暗号化には、暗号化オプションがあります。セキュリティ機能の詳細は、『Oracle GoldenGate セキュリティ・ガイド』を参照してください。

例 4-1 プライマリ Extract グループのパラメータ

```
-- Identify the Extract group:
EXTRACT ext
-- Specify database login information as needed for the database:
[SOURCEDB dsn1,] [USERID user[, PASSWORD pw]]
-- Specify the VAM trail:
EXTTRAIL VAM_trail
-- Specify that this Extract creates and writes to a VAM trail:
DSOPTIONS CREATETRANLOG
-- Specify library, TAM initialization file, and other parameters:
VAM library, PARAMS ("init file", "...>")
```

```
-- Specify tables to be captured:
TABLE owner.table;
```

7. VAM ソートによる Extract グループを作成して、VAM 証跡を読み取ります。ドキュメントで使用する目的で、このグループを `extsort` と表します。

```
ADD EXTRACT extsort, VAMTRAILSOURCE VAM_trail
```

8. ローカル証跡を追加して、ソートされたデータを受信します。

```
ADD EXTTRAIL local_trail, EXTRACT extsort
```

EXTRACT 引数を使用して、この証跡を VAM ソートによる Extract グループにリンクします。データ・ポンプ・グループによって、この証跡が読み取られます。

9. EDIT PARAMS コマンドを使用して、VAM ソートによる Extract グループにパラメータ・ファイルを作成します。データベース環境に適用される他のパラメータに加え、例 4-2 に示されているパラメータを含めてください。

例 4-2 VAM ソートによる Extract グループのパラメータ

```
-- Identify the Extract group:
EXTRACT extsort
-- Specify database login information as needed for the database:
[SOURCEDB dsn1,][USERID user[, PASSWORD pw]]
-- Specify that this Extract reads a VAM trail and sorts the data:
DSOPTIONS SORTTRANLOG
-- Specify the local trail to receive the sorted data:
EXTTRAIL local_trail
-- Specify tables to be captured:
TABLE owner.table;
```

10. データ・ポンプ・グループを作成して、ローカル証跡を読み取り、そのデータを次のいずれかにあるリモート証跡に送信します。

- ターゲットの Teradata データベースに対して Replicat を実行しているレプリケーション・サーバー。
- Oracle GoldenGate でサポートされている別のデータベース・プラットフォームに対して Replicat を実行しているターゲット・サーバー。

```
ADD EXTRACT pump, EXTTRAILSOURCE local_trail
```

ドキュメントで使用する目的で、このグループを `pump` と表します。

11. リモート証跡を追加します。

```
ADD RMTTRAIL remote_trail, EXTRACT pump
```

EXTRACT 引数を使用して、リモート証跡をデータ・ポンプ・グループにリンクします。

12. データ・ポンプにパラメータ・ファイルを作成します。データベース環境に適用される他のパラメータに加え、例 4-3 に示されているパラメータを含めてください。

例 4-3 データ・ポンプ Extract グループのパラメータ

```
-- Identify the data pump group:
EXTRACT pump
-- Specify database login information as needed for the database:
[SOURCEDB dsn1,][USERID user[, PASSWORD pw]]
-- Specify the name or IP address of the remote system:
RMTTHOST target, MGRPORT portnumber
-- Specify the remote trail:
RMTTRAIL remote_trail
-- Allow mapping, filtering, conversion or pass data through as-is:
[PASSTHRU | NOPASSTHRU]
```

```
-- Specify tables to be captured:
TABLE owner.table;
```

注意： PASSTHRU モードを使用するには、ソース・オブジェクトおよびターゲット・オブジェクトの名前を同じにする必要があります。パラメータ・ファイルでは、列マッピング、フィルタリング、SQLEXEC 機能、変換、またはデータ操作が必要なその他の機能を指定することはできません。PASSTHRU および NOPASSTHRU と別の TABLE 文を組み合わせることで、通常の処理とパススルー処理を組み合わせることができます。

4.1.3 最大保護モードをサポートするための Replicat の構成

最大保護モードに関して、Replicat には特別な構成は必要ありません。この項では、大多数のターゲット・データベース・タイプで必要とされる基本的な Replicat パラメータについて説明します。追加のパラメータが必要になる場合があります。ご使用のターゲット・データベース向けの Oracle GoldenGate のドキュメント、および Oracle GoldenGate のリファレンス・ドキュメントも参照してください。

ターゲットのレプリケーション・サーバーまたはデータベース・システム上で次の手順を実行します。

1. 『Oracle GoldenGate Windows and UNIX 管理者ガイド』の手順に従って Manager プロセスを構成します。
2. Manager パラメータ・ファイルでは、PURGEOLDEXTRACTS パラメータを使用してローカル証跡からのファイルの削除を制御します。
3. Replicat チェックポイント表を作成します。この目的に複数のオプションがあります。手順は、『Oracle GoldenGate Windows and UNIX 管理者ガイド』を参照してください。
4. Replicat グループを作成します。ドキュメントで使用する目的で、このグループを rep と表します。

```
ADD REPLICAT rep, EXTTRAIL remote_trail
```

EXTTRAIL 引数を使用して、Replicat グループをソース・サーバー上でデータ・ポンプに指定されているリモート証跡にリンクします。

5. EDIT PARAMS コマンドを使用して、Replicat グループにパラメータ・ファイルを作成します。データベース環境に適用される他のパラメータに加え、例 4-4 に示されているパラメータを含めてください。

例 4-4 Replicat グループのパラメータ

```
-- Identify the Replicat group:
REPLICAT rep
-- State whether or not source and target definitions are identical:
SOURCEDEFS {full_pathname | ASSUMETARGETDEFS}
-- Specify database login information as needed for the database:
[TARGETDB dsn2,] [USERID user id[, PASSWORD pw]]
-- Specify error handling rules (See the NOTE following parameter file):
REPERROR (error, response)
-- Specify tables for delivery:
MAP owner.table, TARGET owner.table[, DEF template name];
```

注意： Teradata の Multiload は、最大保護モードの完全な 2 フェーズ・コミット・プロトコルに関与しません。復元の状況では、Replicat で一部の更新が 2 回適用された可能性があります。multiset 表に影響する場合には、この結果、重複する行が作成される可能性があります。Replicat パラメータで REPERROR パラメータを使用して、Replicat で重複する行が無視されるようにします。

4.2 最大パフォーマンス・モードでの Oracle GoldenGate の構成

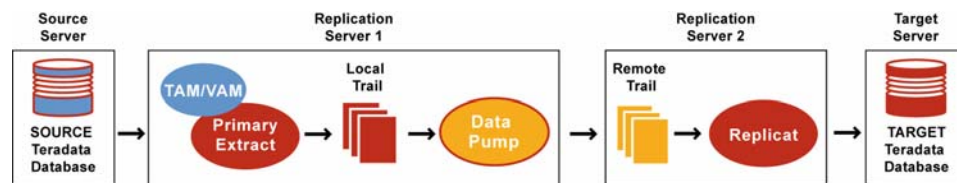
最大パフォーマンス・モードは、最大保護モードよりも速く、かつ容易ですが、フォルト・トレランスが低下します。ソース・アプリケーションによってコミットが発行されると、CDC によってレプリケーション・サーバーへのデータ送信が開始され、そこでそのデータがバッファに格納され、Extract によってソートされます。データ転送が終了すると、CDC によって Extract がコミットに送信され、コミットのトランザクションがアプリケーションおよびデータベースに解放されます。

最大パフォーマンスの構成では、受信データがディスクに保存されず、CDC と Extract 間にデータ損失を防止する確認システムもありません。プライマリ Extract と Teradata のソース間の通信が中断した場合、あるいはコンポーネントが失敗した場合には、ソース・データとターゲット・データの同期が維持されません。

4.2.1 推奨される最大パフォーマンスの構成

別々のレプリケーション・サーバーに Extract プロセスと Replicat プロセスをインストールし、Extract サーバー上でデータ・ポンプとローカル証跡を併用します。この構成では、プライマリ Extract によってトランザクションがローカルの Oracle GoldenGate 証跡まで保持されます。データ・ポンプの Extract によってこの証跡が読み取られ、そのデータが TCP/IP 経由で Replicat サーバー上の証跡に送信され、そのデータが Replicat によって再度読み取られてターゲットに適用されます。Extract サーバーと Replicat レプリケーション・サーバー間の通信が失敗した場合に影響を受けるのは、データ・ポンプだけです。プライマリ Extract では、受信データをメモリーに保持するのではなく、そのデータをディスクに書き込み続けます。そうしないと、長期間の機能停止でメモリーを使い果たす可能性があります。

図 4-2 推奨される最大パフォーマンスの構成



4.2.2 最大パフォーマンス・モードでの Extract の構成

ソースのレプリケーション・サーバー上で次の手順を実行します。

- 『Oracle GoldenGate Windows and UNIX 管理者ガイド』の手順に従って Manager プロセスを構成します。
- Manager パラメータ・ファイルでは、PURGEOLDEXTRACTS パラメータを使用してローカル証跡からのファイルの削除を制御します。
- GGSCI を実行します。
- プライマリ Extract グループを作成します。ドキュメントで使用する目的で、このグループを ext と表します。

```
ADD EXTRACT ext, VAM
```

5. ローカル証跡を追加します。

```
ADD EXTTRAIL local_trail, EXTRACT ext
```

EXTRACT 引数を使用して、この証跡をプライマリ Extract グループにリンクします。

6. EDIT PARAMS コマンドを使用して、プライマリ Extract グループにパラメータ・ファイルを作成します。データベース環境に適用される他のパラメータに加え、例 4-5 のパラメータを含めてください。

注意： データベース・パスワード、証跡内のデータおよび TCP/IP 経由で送信されるデータの暗号化には、暗号化オプションがあります。セキュリティ機能の詳細は、『Oracle GoldenGate Windows and UNIX 管理者ガイド』を参照してください。

例 4-5 プライマリ Extract グループのパラメータ

```
EXTRACT ext
-- Identify the Extract group:
-- Specify database login information as needed for the database:
[SOURCEDB dsn1,][USERID user[, PASSWORD pw]]
-- Specify the local trail that this Extract writes to:
EXTTRAIL local_trail
-- Specify that this Extract is in maximum performance mode:
DSOPTIONS COMMITTEDTRANLOG, RESTARTAPPEND
-- Specify library, TAM initialization file, and other parameters:
VAM library, PARAMS ("init file", "...")
-- Specify tables to be captured:
TABLE owner.table;
```

7. データ・ポンプ・グループを作成して、ローカル証跡を読み取り、そのデータを次のいずれかにあるリモート証跡に送信します。
- ターゲットの Teradata データベースに対して Replicat を実行しているレプリケーション・サーバー。
 - Oracle GoldenGate でサポートされている別のデータベース・プラットフォームに対して Replicat を実行しているターゲット・サーバー。

```
ADD EXTRACT pump, EXTTRAILSOURCE local_trail
```

ドキュメントで使用する目的で、このグループを pump と表します。

8. リモート証跡を追加します。

```
ADD RMTTRAIL remote_trail, EXTRACT pump
```

EXTRACT 引数を使用して、リモート証跡をデータ・ポンプ・グループにリンクします。

9. データ・ポンプにパラメータ・ファイルを作成します。データベース環境に適用される他のパラメータに加え、例 4-6 に示されているパラメータを含めてください。

例 4-6 Extract データ・ポンプのパラメータ

```
-- Identify the data pump group:
EXTRACT pump
-- Specify database login information as needed for the database:
[SOURCEDB dsn1,][USERID user[, PASSWORD pw]]
-- Specify the name or IP address of the remote system:
RMTTHOST target, MGRPORT portnumber
-- Specify the remote trail:
RMTTRAIL remote_trail
```

```
-- Allow mapping, filtering, conversion or pass data through as-is:
[PASSTHRU | NOPASSTHRU]
-- Specify tables to be captured:
TABLE owner.table
```

注意： PASSTHRU モードを使用するには、ソース・オブジェクトおよびターゲット・オブジェクトの名前を同じにする必要があります。パラメータ・ファイルでは、列マッピング、フィルタリング、SQLEXEC 機能、変換、またはデータ操作が必要なその他の機能を指定することはできません。PASSTHRU および NOPASSTHRU と別の TABLE 文を組み合わせることで、通常の処理とパススルー処理を組み合わせることができます。

4.2.3 最大パフォーマンス・モードをサポートするための Replicat の構成

最大保護モードに関して、Replicat には特別な構成は必要ありません。この項では、大多数のターゲット・データベース・タイプで必要とされる基本的な Replicat パラメータについて説明します。追加のパラメータが必要になる場合があります。ご使用のターゲット・データベース向けの Oracle GoldenGate のドキュメント、および Oracle GoldenGate のリファレンス・ドキュメントも参照してください。

ターゲットのレプリケーション・サーバーまたはデータベース・システム上で次の手順を実行します。

1. 『Oracle GoldenGate Windows and UNIX 管理者ガイド』の手順に従って Manager プロセスを構成します。
2. Manager パラメータ・ファイルでは、PURGEOLDEXTRACTS パラメータを使用してローカル証跡からのファイルの削除を制御します。
3. Replicat チェックポイント表を作成します。この目的に複数のオプションがあります。手順は、『Oracle GoldenGate Windows and UNIX 管理者ガイド』を参照してください。
4. Replicat グループを作成します。ドキュメントで使用する目的で、このグループを rep と表します。

```
ADD REPLICAT rep, EXTTRAIL remote_trail
```

EXTTRAIL 引数を使用して、Replicat グループをソース・サーバー上でデータ・ポンプに指定されているリモート証跡にリンクします。

5. EDIT PARAMS コマンドを使用して、Replicat グループにパラメータ・ファイルを作成します。データベース環境に適用される他のパラメータに加え、例 4-7 に示されているパラメータを含めてください。

例 4-7 Replicat グループのパラメータ

```
-- Identify the Replicat group:
REPLICAT rep
-- State whether or not source and target definitions are identical:
SOURCEDEFS full_pathname | ASSUMETARGETDEFS
-- Specify database login information as needed for the database:
[TARGETDB dsn2,] [USERID user id[, PASSWORD pw]]
-- Specify error handling rules:
REPEROR (error, response)
-- Specify tables for delivery:
MAP owner.table, TARGET owner.table[, DEF template name];
;
```

4.3 Oracle GoldenGate その他の構成ガイドライン

Oracle GoldenGate 環境をインストールして構成したら、次の内容を考慮してください。

4.3.1 DDL サポートの構成

Oracle GoldenGate では、Teradata DDL のレプリケーションをサポートしています。詳細は、5-1 ページの「[Teradata データベースへの DDL 同期の構成](#)」を参照してください。

4.3.2 大量の更新操作および削除操作の処理

大量の行数を更新または削除する操作によって、サブスクライバ・データベースの行ごとに個別の更新および削除が生成されます。これによって、Teradata のサブスクライバ・システムでロック・マネージャのオーバーフローが発生し、その結果、Replicat プロセスが終了する可能性があります。

それらのエラーを回避するには、次のいずれかの手順を実行します。

- それらの操作のレプリケーションを一時停止してから、ソース・システムおよびターゲット・システムでそれらの操作を手動で実行します。レプリケーションを一時停止するには、そのセッションにのみレプリケーションを一時停止する次のコマンドを使用します。その表の他のセッションの操作は正常にレプリケートされます。

```
set session override replication on;
commit;
```
- Replicat パラメータ MAXTRANSOPS を 1000 未満の値に設定します。このパラメータによって、大きいトランザクションが小さいトランザクションに分割されます。

4.3.3 複数の接続の回避

既定では、Extract プロセスと Replicat プロセスによって、カタログのクエリに新しい接続が作成されます。DBOPTIONS パラメータと NOCATALOGCONNECT オプションを併用することで、この余分な接続を回避できます。

4.3.4 ODBC 経由での Replicat パフォーマンスの改善

Replicat プロセスのスループットを改善するには、複数の証跡および並列 Replicat プロセスの使用を検討します。各 Replicat では、単一の ODBC セッションが使用されるため、かなり速い配信速度を実現するには、複数のセッションが必要になる場合があります。構成手順は、『[Oracle GoldenGate Windows and UNIX 管理者ガイド](#)』を参照してください。

4.3.5 初期同期の実行

Oracle GoldenGate を使用してトランザクションの変更を初めて送信する前に、ソース・データおよびターゲット・データの初期同期を実行します。2 つの Teradata データベースの同期に望ましい方法は、Teradata データ・ローダー・ユーティリティのいずれかを使用することです。推奨されるユーティリティは MultiLoad です。初期ロードを構成するには、『[Oracle GoldenGate Windows and UNIX 管理者ガイド](#)』を参照してください。

Teradata データベースへの DDL 同期の構成

この章では、Oracle GoldenGate における DDL サポートの理解とその構成方法について説明します。

5.1 このドキュメントについて

このドキュメントには、Teradata 環境内の Oracle GoldenGate ソリューションのセットアップ固有の情報が記載されています。このドキュメントは、Teradata データベースおよび Teradata Replication Solutions の基本的な知識を持つ読者を対象にしています。さらに、次の内容を正しく構成していることを前提とします。

- Relay Services Gateway (RSG)
- Change Data Capture (CDC)
- Teradata Access Module (TAM)
- レプリケーション・グループ

Teradata データベースにレプリケーションを構成する方法の詳細は、『Teradata Replication Solutions』のドキュメントを参照してください。

5.2 DDL 同期の概要

Oracle GoldenGate では、あるデータベースから別のデータベースへの DDL 操作の同期がサポートされています。DDL 同期は、次のような場合にアクティブになります。

- ビジネス・アプリケーションが、ソースとターゲットのオブジェクトにアクティブにアクセスし、更新している場合。
- Oracle GoldenGate のトランザクション・データの同期がアクティブな場合。

DDL のレプリケーションおよびトランザクション・データ変更 (DML) のレプリケーションをサポートするコンポーネント同士は、相互に独立しています。そのため、次のような同期が可能です。

- DDL の変更のみ
- DML の変更のみ
- DDL と DML の両方

5.3 Oracle GoldenGate における DDL サポートの制限事項

このトピックでは、DDL 機能の制限事項について説明します。このドキュメントの発行後に明らかになったその他の制限事項については、ソフトウェアに同梱されている Oracle GoldenGate のリリース・ノートまたは readme ファイルを参照してください。

5.3.1 DDL 文の長さ

Oracle GoldenGate では、文字数ではなく、バイト数で DDL 文の長さが計測されます。サポートされる長さは約 2MB で、いくつかある特性の中でも、影響を受けるオブジェクトの名前やその DDL タイプによってサイズに幅がある内部オーバーヘッドが考慮されています。DDL がサポートされているサイズより長い場合は、Extract により警告が発行され、DDL 操作が無視されます。

5.3.2 サポートされているトポロジ

Oracle GoldenGate では、同じ構成においてのみ DDL 同期がサポートされます。ソースとターゲットのオブジェクト定義は同一であることが必要です。

Oracle GoldenGate は、スタンバイ・データベースでは DDL をサポートしていません。

Oracle GoldenGate では、サポートされているすべての一方向構成、および双方向構成 (2 つのシステム間のみ) で DDL レプリケーションがサポートされています。

5.3.3 フィルタリング、マッピングおよび変換

DDL 操作を Oracle GoldenGate プロセスで変換することはできません。ただし、プライマリ Extract プロセスまたは Replicat プロセスで、ソース DDL を別のターゲット・オブジェクトにマップしてフィルタすることは可能です。データ・ポンプ Extract または VAM ソートによる Extract で DDL をマッピングすることやフィルタリングすることはできないため、DDL はプライマリ Extract から受信したものととして渡されます。これは PASSTHRU モードと呼ばれます。

たとえば、ALTER TABLE TableA がデータ・ポンプ Extract または VAM ソートによる Extract で、ALTER TABLE TableA として処理されるとします。別の内容を指定する TABLE 文に関係なく、これをそのプロセスで ALTER TABLE TableB としてマップすることはできません。

5.3.4 DDL レスポンス時間

Teradata データベースおよび Oracle GoldenGate コンポーネントを含むレプリケーション・システム間の同期プロトコルに内在する遅延が原因で、レプリケーション用に取得される DDL 文のレスポンス時間が長くなる場合があります。レスポンス時間のオーバーヘッドは、ほとんどの条件下で 1 秒を超えないようにする必要があります。取得されない DDL のレスポンス時間には、大きな影響はありません。UDT または LOB を含む表の変更データを取得するパフォーマンス上のコストは、それらのデータ型を含まない表と比較して、データをエクスポートするその他の方法と比較できる必要があります。

5.4 DDL サポートの構成ガイドライン

次に、Oracle GoldenGate プロセスが DDL レプリケーションをサポートするよう構成する際に考慮するガイドラインを示します。

5.4.1 データベース権限

DDL の取得とレプリケーションをサポートするために Oracle GoldenGate に必要なデータベース権限については、1-4 ページの「[Oracle GoldenGate プロセス用のデータベース・ユーザー](#)」を参照してください。

5.4.2 並列処理

Extract/Replicat の並列処理を使用している場合は、関連する DDL と DML を同じプロセス・ストリームに保持し、データの整合性を確保します。プロセスを次のように構成します。

- あるオブジェクトに対するすべての DDL および DML が、同じ Extract グループおよび同じ Replicat グループで処理されるようにします。
- 相互に関連するすべてのオブジェクトが、同じプロセス・グループで処理されるようにします。

たとえば、ReplicatA が Table1 の DML を処理する場合は、Table1 の DDL も処理する必要があります。Table2 に Table1 に対する外部キーがある場合は、その DML と DDL の操作も ReplicatA によって処理される必要があります。

ある Extract グループが、様々な Replicat グループによって読み取られる複数の証跡に書き込みを行う場合、Extract はすべての証跡にすべての DDL を送信します。Replicat パラメータ・ファイルにある DDL パラメータのフィルタ・オプションを使用して DDL をフィルタするには、各 Replicat グループを使用します。

5.4.3 データ・ポンプ Extract および VAM ソートによる Extract の DDL と DML

データ・ポンプ Extract または VAM ソートによる Extract を使用している場合、オブジェクトで DDL サポートが使用されているときには、DML を PASSTHRU モード用に構成します。DDL は、PASSTHRU モードのデータ・ポンプ Extract または VAM ソートによる Extract を介して渡され、DML も同様にする必要があります。DML のフィルタリング、マッピングまたは変換は、プライマリ Extract または Replicat によって行われる必要があります。ただし、DDL サポートを使用しない表は NOPASSTHRU モードに構成でき、データのフィルタリングやデータ・ポンプによる操作を実行できます。

表を PASSTHRU、NOPASSTHRU またはその両方に構成する手順

1. データ・ポンプ Extract または VAM ソートによる Extract のパラメータ・ファイルで、DDL サポートを使用する表を含むすべての TABLE 文の前に、PASSTHRU パラメータを配置します。
2. データのフィルタリング、マッピングまたは変換を実行する場合は、同じパラメータ・ファイルで、DDL サポートを使用しない表を含む TABLE 文の前に、NOPASSTHRU パラメータを配置します。
3. データ・ポンプ Extract または VAM ソートによる Extract の DDL 構成パラメータ (DDL、DDOPTIONS、DDL SUBST、DDL ERROR)、または DDL オプションが指定された Oracle GoldenGate のトレース・パラメータは使用しないでください。

PASSTHRU および NOPASSTHRU の詳細は、『Oracle GoldenGate Windows and UNIX リファレンス・ガイド』を参照してください。

5.4.4 オブジェクト名

Oracle GoldenGate では、データベース定義のオブジェクト名、ケースおよびキャラクタ・セットが保持されます。このサポートにより、データベース階層のすべてのレベルにおいて、シングルバイト名とマルチバイト名、記号およびアクセント文字が保持されます。オブジェクト名のサポートの詳細は、『Oracle GoldenGate Windows and UNIX 管理者ガイド』を参照してください。

疑問符 (?) やアスタリスク (*) のワイルドカードを使用して、DDL 同期をサポートする構成パラメータにオブジェクト名を指定できます。ワイルドカードのサポートの詳細は、『Oracle GoldenGate Windows and UNIX 管理者ガイド』を参照してください。ワイルドカードを正しく処理するため、WILDCARDRESOLVE パラメータはデフォルトで DYNAMIC に設定されています。WILDCARDRESOLVE がそれ以外に設定されている場合、DDL 操作を処理している Oracle GoldenGate プロセスが異常終了し、プロセス・レポートにエラーが書き込まれます。

5.4.5 データ定義

DDL サポートには同一の構成が必要であるため、Replicat パラメータ・ファイルに ASSUMETARGETDEFS パラメータを使用する必要があります。オブジェクトが DDL サポート向けに構成されている場合に、SOURCEDEFS パラメータが使用されていると、Replicat が異常終了します。ASSUMETARGETDEFS の詳細は、『Oracle GoldenGate Windows and UNIX リファレンス・ガイド』を参照してください。

5.4.6 初期同期

DDL レプリケーションを構成するには、ソース・データベースと同期しているターゲット・データベースから開始します。DDL サポートは Replicat の初期ロード・メソッドと互換性があります。

また、DDL サポートには、Teradata FastLoad の初期同期メソッドとの互換性もあります。この機能の詳細は、Teradata のドキュメントを参照してください。

初期ロードの実行前に、DDL の抽出とレプリケーションを無効化します。DDL 処理は、Extract および Replicat パラメータ・ファイルの DDL パラメータで制御されます。

5.5 DDL スコープについて

データベース・オブジェクトはスコープに分類されます。スコープは、オブジェクトに対する DDL 操作が Oracle GoldenGate でどのように処理されるかを定義するカテゴリです。次のスコープがあります。

- MAPPED
- UNMAPPED
- OTHER

スコープを使用すると、DDL 操作、文字列の置換およびエラー処理のフィルタリングをより詳細に制御できます。

5.5.1 MAPPED スコープ

TABLE および MAP 文に指定されるオブジェクトは、MAPPED スコープです。これらの文の抽出およびレプリケーションの手順は、オーバーライド・ルールが適用される場合を除き、指定されたオブジェクトのデータ (DML) と DDL の両方に適用されます。TABLE 文および MAP 文のオブジェクトでは、表 5-1 にリストされている DDL 操作がサポートされます。

表 5-1 MAP 文および TABLE 文にマップ可能なオブジェクト

操作	オブジェクト
CREATE	TABLE (AS SELECT を含む)
ALTER	INDEX
DROP	TRIGGER
RENAME	VIEW
COMMENT ON	FUNCTION
	PROCEDURE
	MACRO
GRANT	TABLE
REVOKE	

次に、MAPPED スコープに適用される制限事項を示します。

- TABLE および MAP では、オブジェクト名に使用されている可能性がある一部の特殊文字がサポートされていません。それらの文字のリストは、『Oracle GoldenGate Windows and UNIX 管理者ガイド』を参照してください。サポートされていない特殊文字を含むオブジェクトは、UNMAPPED および OTHER のスコープでサポートされます。
- COMMENT ON のサポートは、COMMENT ON TABLE および COMMENT ON COLUMN に適用されます。
- HASH 操作および JOIN 操作では、索引に対する DDL はサポートされていません。

Extract では、TABLE 文の指示に従って、MAPPED スコープにより DDL 取得用のオブジェクトがマークされます。Replicat の場合は、MAPPED スコープにより、レプリケーション用の DDL がマークされ、MAP 文の TARGET 句の所有者と名前指定されたオブジェクトにマップされます。

次の TABLE 文および MAP 文があるとします。

Extract (ソース)	Replicat (ターゲット)
TABLE fin.expen;	MAP fin.expen, TARGET fin2.expen2;
TABLE hr.tab*;	MAP hr.tab*, TARGET hrBackup.bak_*;

また、次のようなソース DDL 文があるとします。

```
ALTER TABLE fin.expen ADD notes varchar2(100);
```

この例では、ソース表 fin.expen が、異なる所有者と表名にマップする TARGET 句が指定された MAP 文に存在するため、ターゲット DDL 文は次のようになります。

```
ALTER TABLE fin2.expen2 ADD notes varchar2(100);
```

同様に、次のソースおよびターゲット DDL 文は、例の TABLE 文および MAP 文の 2 つ目のセットに使用できます。

```
ソース: CREATE TABLE hr.tabPayables ... ;
```

```
ターゲット: CREATE TABLE hrBackup.bak_tabPayables ... ;
```

オブジェクトが MAPPED スコープの場合は、DDL サポートをさらに絞り込む場合を除き、DDL 構成パラメータでそのオブジェクト名を省略できます。TABLE 文および MAP 文でオブジェクト名を変更する必要がある場合、変更内容はそれらのオブジェクトの DDL に自動的に適用されます。

TABLE 文にはオブジェクトを指定し、MAP 文には指定しない場合、そのオブジェクトの DDL は、ソースではスコープが MAPPED になり、ターゲットでは UNMAPPED になります。

5.5.1.1 ALTER INDEX のマッピング

ALTER INDEX...RENAME コマンドを別のターゲット索引名にマップすることはできませんが、別のターゲット所有者にマップすることは可能です。次の ALTER INDEX...RENAME で考察します。

```
ALTER INDEX src.ind RENAME TO indnew;
```

この DDL は、ワイルドカードを使用して、次のようにマップできます。

```
MAP src.* TARGET tgt.*;
```

または次のように、ソースとターゲットの指定で元の索引名を使用することを確認し、明示的にマップすることも可能です。

```
MAP src.ind TARGET tgt.ind;
```

前述のケースのいずれの場合も、ターゲット DDL は次のようになります。

```
ALTER INDEX tgt.ind RENAME TO indnew;
```

次のような MAP 文は無効です。

```
MAP src.ind TARGET tgt.indnew;
```

この文は古い名前を新しい名前にマップしており、ターゲット DDL は次のようになります。

```
ALTER INDEX tgt.indnew RENAME TO indnew;
```

5.5.2 UNMAPPED スコープ

TABLE 文または MAP 文で DDL 操作の使用がサポートされているが、ベース・オブジェクト名がそれらのパラメータに含まれていない場合は、UNMAPPED スコープになります。

オブジェクト名を、ソースでは UNMAPPED スコープ (Extract の TABLE 文には含まれない) にし、ターゲットでは MAPPED スコープ (Replicat の MAP 文に含まれる) にすることも、その逆にすることも可能です。Replicat 構成で Teradata の DDL が UNMAPPED スコープの場合、次のいずれかの方法でターゲットに適用されます。

- 必要な Replicat 接続パラメータ TARGETDB に DSN (tdtarg など) のみが含まれ、データベース名が含まれない場合は、ソース DDL と同じ所有者 (データベース名) とオブジェクト名のターゲット・オブジェクトに適用されます。
- TARGETDB に特定のデータベース名 (db@tdtarg など) が使用されている場合は、TARGETDB の所有者が指定されたターゲットにすべての DDL 操作が適用されます。

5.5.3 OTHER スコープ

マップできない DDL 操作は、OTHER スコープになります。Replicat 構成で DDL が OTHER スコープの場合、ソースの DDL と所有者およびオブジェクト名が同一のターゲットに適用されます。OTHER スコープの例は、データ・ファイル名に対して操作を実行する DDL など、システム固有の参照を行う DDL 操作です。

5.6 DDL サポートの有効化

デフォルトでは、DDL レプリケーション・サポートのステータスは次のようになります。

- Teradata TAM により Oracle GoldenGate VAM にすべての DDL が送信されますが、ソースでは、Oracle GoldenGate の DDL サポートはデフォルトで無効化されています。DDL を取得するには、DDL パラメータを使用して Extract を構成する必要があります。
- レプリケートされるトランザクション・データの一貫性を維持するため、ターゲットでは、DDL サポートはデフォルトで有効化されています。デフォルトで、証跡に含まれるすべての DDL 操作が Replicat により処理されます。必要な場合は、DDL パラメータを使用し、Replicat を構成して DDL 操作を無視するかフィルタすることが可能です。

5.7 DDL レプリケーションのフィルタリング

特定 (またはすべて) の DDL が要件に応じてターゲット・データベースに適用されるようにするには、DDL パラメータを使用して DDL 操作をフィルタします。

オプションを指定せずに使用すると、DDL パラメータによるフィルタリングが実行されず、すべての DDL 操作が次のように伝播されます。

- Extract パラメータとして、サポートされているすべてのデータベース・オブジェクトで生成された、サポートされている全 DDL 操作が取得されて証跡に送信されます。
- Replicat パラメータとして、Oracle GoldenGate の証跡からすべての DDL 操作がレプリケートされ、ターゲットに適用されます。これは、このパラメータを使用しない場合のデフォルトの動作と同じです。

オプションを指定して使用すると、DDL パラメータがフィルタリング・エージェントとして機能し、次に基ついで DDL 操作が含まれるか除外されます。

- スコープ
- オブジェクト・タイプ
- 操作タイプ
- オブジェクト名
- DDL コマンド構文またはコメント、あるいはその両方の文字列

パラメータ・ファイルに使用できる DDL パラメータは 1 つのみですが、包含オプションおよび除外オプションを複数組み合わせて、DDL を必要なレベルにフィルタできます。

- DDL フィルタリング・オプションは、トランザクション・ソースから取得するプライマリ Extract には有効ですが、データ・ポンプ Extract には無効です。
- 複数のフィルタ・オプションの指定を組み合わせた場合、AND 文として論理的にリンクされます。
- レプリケートされる DDL 文では、複数のオプションが指定されたすべてのフィルタ基準を満たしている必要があります。
- 複雑なフィルタ基準を使用する場合は、本番で使用する前に、テスト環境で構成をテストすることをお勧めします。

注意: DDL パラメータ文を作成する前に、5.10 項「処理で DDL が評価される仕組み」を確認することをお勧めします。

例 5-1 DDL レプリケーションをフィルタリングする構文

```
DDL [
{INCLUDE | EXCLUDE}
[, MAPPED | UNMAPPED | OTHER | ALL]
[, OPTYPE type]
[, OBJTYPE 'type']
[, OBJNAME name]
[, INSTR 'string']
[, EVENTACTIONS (action specification)
]
[...]
```

表 5-2 DDL の包含オプションおよび除外オプション

オプション	説明
INCLUDE EXCLUDE	<p>包含句または除外句の開始を識別するには、INCLUDE および EXCLUDE を使用します。</p> <ul style="list-style-type: none"> ■ 包含句には、このパラメータの影響を受ける DDL を識別するフィルタ基準が含まれます。 ■ 除外句には、このパラメータから特定の DDL を除外するフィルタ基準が含まれます。 <p>包含句または除外句では、INCLUDE または EXCLUDE キーワードの後に、適用するパラメータのその他のオプションの有効な組合せが続いている必要があります。</p> <p>EXCLUDE を使用する場合は、対応する INCLUDE 句を作成する必要があります。たとえば、次は無効です。</p> <pre>DDL EXCLUDE OBJNAME "hr.*"</pre> <p>ただし、次のいずれかを使用できます。</p> <pre>DDL INCLUDE ALL, EXCLUDE OBJNAME "hr.*" DDL INCLUDE OBJNAME "fin.*" EXCLUDE "fin.ss"</pre> <p>同じ条件を含む INCLUDE より、EXCLUDE が優先されます。包含句および除外句は複数使用できます。</p>

表 5-2 DDL の包含オプションおよび除外オプション (続き)

オプション	説明
MAPPED UNMAPPED OTHER ALL	DDL の操作スコープに基づいて INCLUDE または EXCLUDE を適用するには、MAPPED、UNMAPPED、OTHER および ALL を使用します。 <ul style="list-style-type: none"> ■ MAPPED では、MAPPED スコープの DDL 操作に INCLUDE または EXCLUDE が適用されます。MAPPED のフィルタリングは、その他の DDL パラメータ・オプションで指定されているフィルタリングの前に実行されます。 ■ UNMAPPED では、UNMAPPED スコープの DDL 操作に INCLUDE または EXCLUDE が適用されます。 ■ OTHER では、OTHER スコープの DDL 操作に INCLUDE または EXCLUDE が適用されます。 ■ ALL では、すべてのスコープの DDL 操作に INCLUDE または EXCLUDE が適用されます。
OPTYPE <i>type</i>	CREATE、ALTER および RENAME など、特定タイプの DDL 操作に INCLUDE または EXCLUDE を適用するには、OPTYPE を使用します。 <i>type</i> には、データベースに対して有効な任意の DDL コマンドを使用します。たとえば、ALTER 操作を含める場合の正しい構文は、次のとおりです。 DDL INCLUDE OPTYPE ALTER
OBJTYPE ' <i>type</i> '	特定タイプのデータベース・オブジェクトに INCLUDE または EXCLUDE を適用するには、OBJTYPE を使用します。 <i>type</i> には、データベースに対して有効な任意のオブジェクト・タイプ (TABLE、INDEX、TRIGGER など) を使用します。オブジェクト・タイプの名前は、一重引用符で囲みます。次に例を示します。 DDL INCLUDE OBJTYPE 'INDEX'
OBJNAME <i>name</i>	<i>owner.table_name</i> など、オブジェクトの完全修飾名に INCLUDE または EXCLUDE を適用するには、OBJNAME を使用します。オブジェクト名にのみ、ワイルドカードを使用できます。 例： DDL INCLUDE OBJNAME <i>accounts.*</i> Replicat パラメータ・ファイルに MAPPED を指定した OBJNAME を使用する場合、OBJNAME の値は、MAP 文の TARGET 句で指定された名前に参照する必要があります。たとえば、次の MAP 文の場合、正しい値は OBJNAME <i>fin2.*</i> です。 MAP <i>fin.exp_*</i> , TARGET <i>fin2.*</i> ; 次の例の CREATE TABLE 文は、ソースで次のように実行されます。 CREATE TABLE <i>fin.exp_phone</i> ; ターゲットでは次のようになります。 CREATE TABLE <i>fin2.exp_phone</i> ; MAP 文にターゲット所有者が指定されていない場合は、Replicat により、USERID パラメータで指定されているデータベース・ユーザーにマップされます。 トリガーなど、導出されたオブジェクトを作成する DDL の場合、OBJNAME の値は、導出されたオブジェクトの名前ではなく、ベース・オブジェクトの名前にする必要があります。 たとえば、次の DDL 文を含める場合、正しい値は <i>hr.insert_trig</i> ではなく、 <i>hr.accounts</i> です。 CREATE TRIGGER <i>hr.insert_trig</i> ON <i>hr.accounts</i> ; RENAME 操作の場合、OBJNAME の値は新しい表名にする必要があります。たとえば、次の DDL 文を含める場合、正しい値は <i>hr.acct</i> です。 ALTER TABLE <i>hr.accounts</i> RENAME TO <i>acct</i> ;

表 5-2 DDL の包含オプションおよび除外オプション (続き)

オプション	説明
INSTR 'string'	<p>コマンド構文内に特定の文字列が使用されている DDL 文に INCLUDE または EXCLUDE を適用するには、INSTR を使用します。たとえば、次では索引を作成する DDL が除外されます。</p> <pre>DDL INCLUDE ALL EXCLUDE INSTR 'CREATE INDEX'</pre> <p>文字列を一重引用符で囲みます。文字列の検索では、大 / 小文字は区別されません。</p> <p>INSTR では、文字列の中に一重引用符 (') を使用することも、NULL 値を使用することもできません。</p>
INSTRWORDS 'word list'	<p>指定した語を含む DDL 文に INCLUDE または EXCLUDE を適用するには、INSTRWORDS を使用します。</p> <p>word list に、任意の順序で語を指定し、一重引用符で囲みます。空白を含めるには、その空白 (および該当する場合は空白と語) を二重引用符で囲みます。二重引用符は、文を囲む場合にも使用できます。</p> <p>指定するすべての語が、有効化される INSTRWORDS の DDL に存在する必要があります。</p> <p>例：</p> <pre>ALTER TABLE INCLUDE INSTRWORDS 'ALTER CONSTRAINT " xyz"</pre> <p>これらの例は次に一致します。</p> <pre>ALTER TABLE ADD CONSTRAINT xyz CHECK</pre> <p>および</p> <pre>ALTER TABLE DROP CONSTRAINT xyz</pre> <p>INSTRWORDS では、文字列の中に一重引用符 (') を使用することも、NULL 値を使用することもできません。</p>
EVENTACTIONS (action specification)	<p>イベント・レコードと呼ばれるトランザクション・ログまたは証跡の DDL レコードに基づき、Extract または Replicat プロセスが定義されたアクションを実行します。DDL イベントは、DDL パラメータのその他のフィルタリング・オプションによって決められたとおりに、DDL レコードを Extract またはデータ・ポンプで証跡に書き込める場合、または Replicat で実行可能な場合にトリガーされます。このシステムを使用し、データベース・イベントに基づいて処理をカスタマイズできます。</p> <p>action specification については、MAP および TABLE パラメータの下の EVENTACTIONS を参照してください。</p> <p>DDL レコードで EVENTACTIONS を使用するためのガイドラインは、次のとおりです。</p> <ul style="list-style-type: none"> ■ CHECKPOINTBEFORE: 各 DDL レコードは自律型であるため、DDL レコードは必ずトランザクションの開始になります。そのため、CHECKPOINT BEFORE イベント・アクションは、DDL レコードに対して暗黙的です。 ■ IGNORE: このオプションは、DDL レコードに対して無効です。DDL 操作は自律型であるため、レコードを無視することは、トランザクション全体を無視することと同じです。 <p>EVENTACTIONS は導出されたオブジェクトであるため、次の DDL オブジェクトがサポートされていません。</p> <ul style="list-style-type: none"> ■ 索引 ■ トリガー ■ シノニム ■ 表の RENAME および ALTER TABLE RENAME

5.7.1 DDL パラメータ・オプションの組合せ

次に、DDL パラメータ・オプションの組合せ方の例を示します。

例 5-2 DDL パラメータ・オプションの組合せ

```
DDL &
INCLUDE UNMAPPED &
  OPTYPE alter &
  OBJTYPE 'table' &
  OBJNAME users.tab* &
INCLUDE MAPPED OBJNAME * &
EXCLUDE MAPPED OBJNAME temporary.tab
```

この文で組み合わせたフィルタ基準は、次を指定します。

- TABLE または MAP 文でマップされていない表 (UNMAPPED スコープ) のすべての ALTER TABLE 文が INCLUDE されますが、それはそれらの表の所有者が users である場合およびその名前が tab で開始される場合のみです
- また、TABLE または MAP 文でマップされているすべての表 (MAPPED スコープ) のすべての DDL 操作タイプが INCLUDE されます
- さらに、スコープが MAPPED のすべての表のすべての DDL 操作タイプが EXCLUDE されますが、それはそれらの表の所有者が temporary である場合およびその名前が tab で開始される場合のみです。

5.7.2 DDL EXCLUDE ALL

DDL EXCLUDE ALL は特別な処理オプションで、DDL 操作自体のレプリケーションをブロックしながら、Oracle GoldenGate の最新のオブジェクト・メタデータを管理します。DDL EXCLUDE ALL は、Oracle GoldenGate 以外の方法でターゲットに DDL を適用しているが、Oracle GoldenGate でターゲット・オブジェクトにデータ変更をレプリケートする必要がある場合に使用できます。現在のメタデータがオブジェクト変更として Oracle GoldenGate に提供されるため、Oracle GoldenGate プロセスの停止と開始が不要になります。DDL EXCLUDE ALL には、次の特別な条件が適用されます。

- DDL EXCLUDE ALL では、INCLUDE 句を使用する必要がありません。
- DDL EXCLUDE ALL を使用する際には、すぐに DML を解決できるように、WILDCARDRESOLVE パラメータを IMMEDIATE に設定することが必要な場合があります。

すべての DDL メタデータおよび操作がレプリケートされないようにするには、DDL パラメータ全体を省略します。

5.8 Oracle GoldenGate における導出オブジェクト名の処理方法

DDL 操作には、ベース・オブジェクト名と導出オブジェクト名を含めることができます。ベース・オブジェクトは、データを含むオブジェクトです。導出オブジェクトは、ベース・オブジェクトの一部の属性を継承し、そのオブジェクトに関連する関数を実行するオブジェクトです。ベース・オブジェクトと導出オブジェクトの両方を含む DDL 文は、次のとおりです。

- RENAME
- 索引またはトリガーの CREATE および DROP

次の DDL 文があるとします。

```
CREATE INDEX hr.indexPayrollDate ON TABLE hr.tabPayroll (payDate);
```

この場合は、表がベース・オブジェクトです。その名前 (hr.tabPayroll) がベース名で、MAPPED スコープの TABLE または MAP とのマッピング対象です。導出オブジェクトは索引で、その名前 (hr.indexPayrollDate) が導出名です。

5.8.1 導出オブジェクトのマッピング

導出名は、ベース・オブジェクトとは別に、それ自体の TABLE または MAP 文でマップできます。または、1 つの MAP 文で両方を処理できます。MAP の場合、ターゲットへの導出オブジェクト名の交換は、後続の項で示すようにして行われます。

5.8.1.1 項「MAP がベース・オブジェクトに存在して導出オブジェクトに存在しない場合」

5.8.1.2 項「MAP がベース・オブジェクトと導出オブジェクトに存在する場合」

5.8.1.3 項「MAP が導出オブジェクトに存在してベース・オブジェクトに存在しない場合」

5.8.1.1 MAP がベース・オブジェクトに存在して導出オブジェクトに存在しない場合

MAP 文がベース・オブジェクトに存在して、導出オブジェクトに存在しない場合、結果は導出オブジェクトの暗黙的マッピングになります。DDL パラメータに MAPPED が指定されているとみなされ、Replicat により、ベース・オブジェクトと同じターゲット所有者が導出オブジェクトに指定されます。導出オブジェクトの名前は、ソース文のままです。たとえば、次のような内容があるとします。

Extract (ソース)	Replicat (ターゲット)
TABLE hr.tab*;	MAP hr.tab*, TARGET hrBackup.*;

次のソース DDL 文があるとします。

```
CREATE INDEX hr.indexPayrollDate ON TABLE hr.tabPayroll (payDate);
```

CREATE INDEX 文は、Replicat により、ターゲットで次のように実行されます。

```
CREATE INDEX hrBackup.indexPayrollDate ON TABLE hrBackup.tabPayroll (payDate);
```

暗黙的マッピングのルールは、導出オブジェクトにベース・オブジェクトと同じ所有者を指定するという一般的な業界のプラクティスに基づいています。また、ベース・オブジェクトと同じターゲット所有者が索引を所有している場合、暗黙的マッピングでは、導出オブジェクト名を明示的にマップする必要がありません。

5.8.1.2 MAP がベース・オブジェクトと導出オブジェクトに存在する場合

MAP 文がベース・オブジェクトと導出オブジェクトにも 1 つ存在する場合、結果は明示的マッピングになります。DDL パラメータに MAPPED が指定されているとみなされ、それ自体の TARGET 句に従って、所有者と各オブジェクトの名前が Replicat により変換されます。たとえば、次のような内容があるとします。

Extract (ソース)	Replicat (ターゲット)
TABLE hr.tab*;	MAP hr.tab*, TARGET hrBackup.*;
TABLE hr.index*;	MAP hr.index*, TARGET hrIndex.*;

次のソース DDL 文があるとします。

```
CREATE INDEX hr.indexPayrollDate ON TABLE hr.tabPayroll (payDate);
```

CREATE INDEX 文は、Replicat により、ターゲットで次のように実行されます。

```
CREATE INDEX hrIndex.indexPayrollDate ON TABLE hrBackup.tabPayroll (payDate);
```

ベース・オブジェクトとは異なる所有者がターゲットの索引を所有している必要がある場合、またはターゲットの名前がソースの名前と異なる必要がある場合は、明示的マッピングを使用します。

5.8.1.3 MAP が導出オブジェクトに存在してベース・オブジェクトに存在しない場合

MAP 文が導出オブジェクトに存在して、ベース・オブジェクトに存在しない場合、Replicat では、どちらのオブジェクトに対しても名前の変換が行われません。ターゲット DDL 文はソースと同じになります。導出オブジェクトをマップするには、次のいずれかを選択します。

- ベース・オブジェクトに明示的 MAP 文を使用します。
- 名前的问题がない場合は、ワイルドカードを使用して、ベース・オブジェクトと導出オブジェクトの両方を同じ MAP 文にマップします。
- 名前をどのように変換するかに応じて、各オブジェクトの MAP 文を作成します。

5.8.2 導出オブジェクトとしての新しい表

ここでは、次から作成される新しい表が Oracle GoldenGate でどのように処理されるかを説明します。

5.8.2.1 項「RENAME」

5.8.2.2 項「CREATE TABLE AS SELECT」

5.8.2.1 RENAME

RENAME 操作では、ベース・オブジェクトは常に新しい表名です。次の例において、ベース・オブジェクト名は `index_paydate` です。

```
RENAME hr.indexPayrollDate TO index_paydate;
```

導出オブジェクト名は `hr.indexPayrollDate` です。

5.8.2.2 CREATE TABLE AS SELECT

CREATE TABLE AS SELECT 文には、基礎となる任意の数のオブジェクトに影響を与える SELECT 文と INSERT 文が含まれます。ターゲットでは、Oracle GoldenGate により、ターゲット・データベースから AS SELECT 句のデータが取得されます。

ターゲット・データベースには AS SELECT 句のオブジェクトが存在し、その名前がソースのものと同じであることが必要です。

MAP 文では、Oracle GoldenGate により、新しい表の名前 (CREATE TABLE name) のみが TARGET の指定にマップされますが、AS SELECT 句の基礎となるオブジェクトの名前はマップされません。名前が TARGET の指定に変換されると、データ不整合の原因となるオブジェクトに対する依存性が存在する可能性があります。

次に、ソースの CREATE TABLE AS SELECT 文と、この文が Oracle GoldenGate によってターゲットにどのようにレプリケートされるかの例を示します。

```
CREATE TABLE a.tab1 AS SELECT * FROM a.tab2;
```

Replicat の MAP 文は次のようになります。

```
MAP a.tab*, TARGET a.x*;
```

Replicat が適用するターゲット DDL 文は次のようになります。

```
CREATE TABLE a.xtab1 AS SELECT * FROM a.tab2;
```

AS SELECT * FROM 句の表の名前は、ソースでの名前 (tab2) のままになります。

ソースとターゲットで、基礎となるオブジェクトのデータの一貫性を保つには、Oracle GoldenGate によるデータ・レプリケーションに備えて構成します。前述の例では、次の文を使用してこの要件に対応できます。

ソース	ターゲット
TABLE a.tab*;	MAPEXCLUDE a.tab2
	MAP a.tab*, TARGET a.x*;
	MAP a.tab2, TARGET a.tab2;

5.8.3 導出オブジェクトのマッピングの無効化

MAP 文の TARGET 句に従った、導出オブジェクト名の変換が行われなくするには、その文に含まれる DDLOPTIONS パラメータに NOMAPDERIVED オプションを指定して使用します。NOMAPDERIVED は、ベース・オブジェクトまたは導出オブジェクトの名前を含む明示的な MAP 文をオーバーライドします。導出オブジェクトが含まれるソース DDL は、所有者とオブジェクト名がソースと同一のターゲットにレプリケートされます。

表 5-3 に、MAP 文がベース・オブジェクトのみ、導出オブジェクトのみ、またはその両方に存在するかどうかに基づいて、NOMAPDERIVED と比較した MAPDERIVED の結果を示します。

表 5-3 マッピング構成に基づいた、ターゲットにおける [NO]MAPDERIVED の結果

ベース・オブジェクト	導出オブジェクト	MAP/NOMAP DERIVED	MAP ごとの変換	導出オブジェクトにおけるベース・オブジェクトの所有者の取得
マップされている ¹	マップされている	MAPDERIVED	はい	いいえ
マップされている	マップされていない	MAPDERIVED	いいえ	はい
マップされていない	マップされている	MAPDERIVED	いいえ	いいえ
マップされていない	マップされていない	MAPDERIVED	いいえ	いいえ
マップされている	マップされている	NOMAPDERIVED	いいえ	いいえ
マップされている	マップされていない	NOMAPDERIVED	いいえ	いいえ
マップされていない	マップされている	NOMAPDERIVED	いいえ	いいえ
マップされていない	マップされていない	NOMAPDERIVED	いいえ	いいえ

¹ 「マップされている」とは、MAP 文に含まれていることを示しています。

次の例では、NOMAPDERIVED と比較した MAPDERIVED の結果を示します。

表 5-4 に示されている 1 つ目の例では、ベース名と導出名の両方が MAPDERIVED で変換されるため、トリガーと表のどちらもターゲットの rpt に所有されていることを示しています。

表 5-4 導出オブジェクト名のデフォルトのマッピング (MAPDERIVED)

MAP 文	Extract に取得されたソース DDL 文	Replicat により適用されたターゲット DDL 文
MAP fin.*, TARGET rpt.*;	CREATE TRIGGER fin.act_trig ON fin.acct;	CREATE TRIGGER rpt.act_trig ON rpt.acct;

表 5-5 に示されている 2 つ目の例では、NOMAPDERIVED で変換ができなくなっているため、トリガーは fin に所有されています。

表 5-5 NOMAPDERIVED を使用している場合の導出オブジェクト名のマッピング

MAP 文	Extract に取得されたソース DDL 文	Replicat により適用されたターゲット DDL 文
MAP fin.*, TARGET rpt.*;	CREATE TRIGGER fin.act_trig ON fin.acct;	CREATE TRIGGER fin.act_trig ON rpt.acct;

注意： RENAME 文の場合、新しい表名はベース表の名前とみなされ、古い表名は導出表の名前とみなされます。

5.9 DDL 文字列置換の使用

Oracle GoldenGate での処理中に、DDL 操作内で文字列を置換できます。この機能により、ディレクトリ名の変更やマッピング、およびデータ構造に直接関連しない他のことも行いやすくなります。文字列置換は、DDL SUBST パラメータで制御します。

詳細は、『Oracle GoldenGate Windows and UNIX リファレンス・ガイド』を参照してください。

注意： DDL SUBST パラメータ文を作成する前に、次の「[処理で DDL が評価される仕組み](#)」の項を確認することをお勧めします。

5.10 処理で DDL が評価される仕組み

ここでは、ソースおよびターゲットのシステムで、DDL 文が Oracle GoldenGate でどのように処理されるかを説明します。Oracle GoldenGate パラメータの異なる基準が処理される順序を示し、Extract と Replicat がそれぞれ DDL を処理する方法の違いを説明します。

5.10.1 Extract が DDL を評価する仕組み

1. Extract により DDL 操作が取得されます。
2. Extract により、DDL パラメータが検索されます。(この例では、それが存在すると仮定します。)
3. Extract により、ベース・オブジェクト名と導出オブジェクト名 (存在する場合) が取得されます。
4. Extract により、DDL スコープ (MAPPED、UNMAPPED または OTHER) が判断されます。次の場合に MAPPED になります。
 - マッピングで操作タイプおよびオブジェクト・タイプがサポートされる場合。
および
 - ベース・オブジェクト名 / 導出オブジェクト名 (RENAME の場合) が、TABLE パラメータに存在する場合。

次の場合に UNMAPPED になります。

- マッピングで操作タイプおよびオブジェクト・タイプがサポートされない場合。
および
- ベース・オブジェクト名 / 導出オブジェクト名 (RENAME の場合) が、TABLE パラメータに存在しない場合。

それ以外の場合、操作は OTHER と識別されます。

5. Extract により、INCLUDE および EXCLUDE 句の DDL パラメータが確認され、それらの句の DDL パラメータ基準が評価されます。INCLUDE または EXCLUDE が TRUE に評価されるには、すべての操作が TRUE に評価される必要があります。次のようになります。
 - EXCLUDE 句が TRUE に評価されると、Extract により DDL 操作が破棄され、別の DDL 操作が評価されます。この場合は、処理手順が新たに始まります。
 - INCLUDE 句が TRUE に評価されるか、DDL パラメータに INCLUDE 句または EXCLUDE 句が存在しない場合は、Extract に DDL 操作が含まれ、処理ロジックが続行されます。

6. Extract により DDLSUBST パラメータが検索され、INCLUDE および EXCLUDE 句が評価されます。これらの句の基準が TRUE になると、Extract により、文字列置換が実行されます。Extract により、パラメータ・ファイルの各 DDLSUBST 文に対して DDL 操作が評価されます。true の DDLSUBST 文すべてで、Extract により、DDLSUBST パラメータがファイルにリストされている順序で文字列置換が実行されます。
7. Extract により、DDL 文が証跡に書き込まれます。

5.10.2 Replicat が DDL を評価する仕組み

1. Replicat により、DDL 操作が証跡から読み取られます。
2. Replicat により、DDL の同期スコープが評価され、DDL が名前マッピングに適切であるかどうか判断されます。その他は OTHER スコープになります。
3. Replicat により、パラメータ・ファイルの MAP 文が評価されます。(証跡から読み取る際に) この DDL のソースのベース・オブジェクト名がいずれかの MAP 文に含まれている場合、その操作のスコープは MAPPED とマークされます。それ以外の場合、スコープは UNMAPPED とマークされます。
4. Replicat により、ソースのベース・オブジェクト名が、MAP 文の TARGET 句に指定されたベース・オブジェクト名で置換されます。
5. 導出オブジェクトがある場合、Replicat は DDLOPTIONS MAPDERIVED を検索します。存在する場合は、Replicat により、ソースの導出名が MAP 文のターゲットの導出名で置換されます。
6. Replicat により、INCLUDE および EXCLUDE 句の DDL パラメータが確認され、それらに含まれる DDL パラメータ基準が評価されます。INCLUDE または EXCLUDE が TRUE に評価されるには、すべての操作が TRUE に評価される必要があります。次のようになります。
 - いずれかの EXCLUDE 句が TRUE に評価されると、Replicat により DDL 操作が破棄され、別の DDL 操作の評価が開始されます。この場合は、処理手順が新たに始まります。
 - いずれかの INCLUDE 句が TRUE に評価されるか、DDL パラメータに INCLUDE 句または EXCLUDE 句が存在しない場合は、Replicat に DDL 操作が含まれ、処理ロジックが続行されます。
7. Replicat により DDLSUBST パラメータが検索され、INCLUDE および EXCLUDE 句が評価されます。これらの句のオプションが TRUE になると、Replicat により、文字列置換が実行されます。Replicat により、パラメータ・ファイルの各 DDLSUBST 文に対して DDL 操作が評価されます。true の DDLSUBST 文すべてで、Replicat により、DDLSUBST パラメータがファイルにリストされている順序で文字列置換が実行されます。
8. Replicat により、DDL 操作がターゲット・データベースで実行されます。
9. エラーがない場合は、Replicat により次の DDL 文が処理されます。エラーがある場合は、Replicat により次の手順が実行されます。
10. Replicat により、Replicat の DDLERROR パラメータ文の INCLUDE および EXCLUDE ルールが、パラメータ・ファイルに表示される順序で分析されます。Replicat によりエラー・コードのルールが検出された場合は、指定のエラー処理が適用されますが、それ以外の場合は DEFAULT の処理が適用されます。
11. エラー処理を行っても DDL 操作が成功しない場合は、ルールに指定されているとおりに、異常終了、操作の無視または破棄のいずれかが Replicat により実行されます。

注意： MAP 文に同じソースに対する複数のターゲットが存在する場合は、それぞれに対して処理ロジックが実行されます。

5.11 DDL 処理エラーの処理

Extract により検出された、メタデータのないオブジェクトのエラー、および DDL がターゲット・データベースに適用される際に発生する Replicat のエラーを処理するには、DDLERROR パラメータを使用します。DDLERROR オプションを使用すると、処理の停止や特定の方法によるその他のエラーの処理など、デフォルトの方法で大部分のエラーを処理できます。同じパラメータ・ファイルに DDLERROR のインスタンスを複数使用して、予想されるすべてのエラーを処理できます。オプションと使用方法については、『Oracle GoldenGate Windows and UNIX リファレンス・ガイド』を参照してください。

5.12 DDL レポート情報の表示

Oracle GoldenGate では、デフォルトで、Extract および Replicat のレポートの最後に DDL 操作に関する基本統計が表示されます。拡張 DDL レポートを有効化するには、DDLOPTIONS パラメータに REPORT オプションを指定して使用します。拡張レポートには、DDL 処理に関する次の情報が含まれます。

- Oracle GoldenGate により処理された DDL 操作に関する順を追った履歴。
- 使用されている DDL のフィルタリング・パラメータおよび処理パラメータ。

拡張 DDL レポートの情報により、レポート・ファイルのサイズが増大しますが、トラブルシューティングなどの特定の状況で役に立ちます。

レポートを表示するには、GGSCI で VIEW REPORT コマンドを使用します。

```
VIEW REPORT group
```

5.12.1 Extract DDL レポート

Extract レポートには、次がリストされます。

- 取得された各 DDL 操作の構文全体、その Oracle GoldenGate CSN 番号、Teradata の順序番号およびバイト単位の操作サイズ。
- 文字列置換や INCLUDE と EXCLUDE のフィルタリングなど、操作に処理基準がどのように適用されたかを示す後続のエントリ。
- 操作が証跡に書き込まれたか除外されたかを示す別のエントリ。

次に、Extract レポート・ファイルから取得された例を示します。

例 5-3 Extract レポートのサンプル

```
2011-01-21 18:41:40 GGS INFO 2100 DDL found, operation [DROP TABLE "SMIJATOVDBS"."src13_tabtable_9" ;
(size 59)], start CSN [2500FF3F0200363A], DDL seqno [000000250000000000000381500000021].
2011-01-21 18:41:40 GGS INFO 2100 DDL operation included [include mapped objname "*"], optype [DROP],
objtype [TABLE], objowner [SMIJATOVDBS], objname [SRC13_TABTABLE_9].
2011-01-21 18:41:40 GGS INFO 2100 DDL operation written to extract trail file.
```

5.12.2 Replicat DDL レポート

Replicat レポートには、次がリストされます。

- Replicat が証跡から処理した各 DDL 操作の構文全体。
- 操作のスコープ (MAPPED、UNMAPPED、OTHER)、およびオブジェクト名がターゲットの DDL 文でどのようにマップされたか (該当する場合) を示す後続のエントリ。
- 処理基準がどのように適用されたかを示す別のエントリ。
- 操作が成功したか失敗したか、および Replicat によりエラー処理ルールが適用されたかどうかを示す追加のエントリ。

次に、Replicat パラメータ・ファイルから取得された例を示します。

例 5-4 Replicat レポートのサンプル

```
2011-01-21 18:41:44 GGS INFO 2104 DDL found, operation [DROP TABLE "SMIJATOVDBS"."src13_tabtable_9" ;
(size 59)].
2011-01-21 18:41:44 GGS INFO 2100 DDL is of mapped scope, after mapping new operation [DROP TABLE
"SMIJATOVDBT"."SRC13_TABTABLE_9" ; (size 59)].
2011-01-21 18:41:44 GGS INFO 2100 Executing DDL operation.
2011-01-21 18:41:44 GGS INFO 2105 DDL operation successful.
```

5.12.3 プロセス・レポートの統計

GGSCI で SEND コマンドを使用すると、Extract や Replicat のレポートに DDL 処理に関する現在の統計を送信できます。

```
SEND {EXTRACT | REPLICAT} group REPORT
```

統計には、次の総数が表示されます。

- すべての DDL 操作
- スコープが MAPPED の操作
- スコープが UNMAPPED の操作
- スコープが OTHER の操作
- 除外された操作 (操作数から含まれる操作を引いた数)
- エラー (Replicat のみ)
- 再試行されたエラー (Replicat のみ)
- 破棄されたエラー (Replicat のみ)
- 無視された操作 (Replicat のみ)

5.13 DDL 処理のトレース

Oracle GoldenGate のテクニカル・サポートでサポート・ケースを開くと、トレースの有効化を依頼される場合があります。次のパラメータが DDL のトレースを制御します。

- TLTRACE は Extract トレースを制御します
- TRACE および TRACE2 は Replicat トレースを制御します。

これらのパラメータには、DDL のトレースを DML のトレースから分離するオプションがあります。詳細は、『Oracle GoldenGate Windows and UNIX リファレンス・ガイド』を参照してください。

Oracle GoldenGate 構成でのオブジェクトの変更

この章では、Oracle GoldenGate レプリケーション・ソリューションの使用時にいくつかの一般的なメンテナンス作業を実行する手順について説明します。

6.1 Extract グループの削除

Oracle GoldenGate の Extract グループを削除するには、Teradata レプリケーション・グループから Extract プロセスを切り離す必要があります。

1. GGSCI を起動します。
2. Extract の実行中に、次のコマンドを発行します。

```
SEND EXTRACT group, VAMMESSAGE "control:terminate"
```
3. Extract を停止します。

```
STOP EXTRACT group
```
4. Extract グループを強制的に削除します。

```
DELETE EXTRACT group !
```
5. 任意の Teradata クライアントから次のコマンドを発行します。

```
DROP REPLICATION GROUP repgroup name
```

6.2 既存の Extract グループへの表の追加

1. Oracle GoldenGate にリンクされているソース表でアクティビティを一時停止します。
2. GGSCI を起動します。
3. GGSCI で次のコマンドを発行します。

```
INFO EXTRACT group
```
4. Checkpoint Lag 行で、Extract の遅れがあるかどうかを確認します。必要に応じて、遅れがゼロになるまで INFO EXTRACT を発行し続けますが、ゼロは現時点ですべてのトランザクション・データが処理されていることを示します。
5. Extract の実行中に、次のコマンドを発行します。

```
SEND EXTRACT group, VAMMESSAGE "control:terminate"
```
6. Extract グループを停止します。

```
STOP EXTRACT group
```

7. 任意の Teradata クライアントから次のコマンドを新しい表に追加します。
ALTER REPLICATION GROUP *group* ADD *database.table*
8. 任意の Teradata クライアントから次のコマンドを発行して、セキュリティ・トークンを生成します。
ALTER REPLICATION GROUP *group*
9. TAM 初期化ファイルを編集し、SecurityToken パラメータを使用してセキュリティ・トークンを指定します。
10. Extract パラメータ・ファイルを編集して、新しい表を指定する TABLE パラメータを追加します。
EDIT PARAMS *group*
11. そのファイルを保存して閉じます。
12. GGSCI で次のコマンドを発行し、Extract を開始します。
START EXTRACT *group*
13. Oracle GoldenGate にリンクされているソース表でアクティビティを許可します。

6.3 新しい Extract グループへの表の追加

1. Oracle GoldenGate にリンクされているすべての表のソース・データベースでアクティビティを一時停止します。
2. 現在の Teradata の Create Group 文のファイルを編集し、CREATE REPLICATION GROUP 文から表を削除します。
3. GGSCI を起動します。
4. GGSCI で、現在の Extract グループに対して次のコマンドを発行します。
INFO EXTRACT *group*
5. Checkpoint Lag 行で、Extract の遅れがあるかどうかを確認します。必要に応じて、遅れがゼロになるまで INFO EXTRACT を発行し続けますが、ゼロは現時点ですべてのトランザクション・データが処理されていることを示します。
6. GGSCI で次のコマンドを発行します。
SEND EXTRACT *group*, VAMMESSAGE "control:terminate"
7. 現在の Extract グループを停止します。
STOP EXTRACT *group*
8. 現在の Extract パラメータ・ファイルを編集します。
EDIT PARAMS *group*
9. 表が含まれている TABLE パラメータを削除します。
10. 任意の Teradata クライアントから次のコマンドを発行して、削除対象の表が含まれているレプリケーション・グループを削除します。
ALTER REPLICATION GROUP *group* DROP *table*
11. GGSCI で次のコマンドを発行して現在の Extract グループを開始し、その割り当てられた表 (移動した表がない) の処理を続行できるようにします。
START EXTRACT *group*

12. 移動した表の TABLE 文が含まれている新しい Extract グループを追加してから、使用している取得方法に適切な他のプロセス、証跡およびパラメータ・ファイルを追加します。手順については、4-1 ページの「Oracle GoldenGate の構成」を参照してください。
13. 新しい tam.ini ファイルと、表が含まれている新しい Teradata の Create Group 文のファイルを作成します。3-4 ページの「TAM 初期化ファイルの構成」および 3-2 ページの「Teradata のレプリケーション・グループの作成」を参照してください。
14. 新しい Extract グループと関連プロセスを開始します。
START EXTRACT new_group
15. Oracle GoldenGate にリンクされているすべてのソース表で再開するユーザー・アクティビティを許可します。

6.4 表列の変更

1. Oracle GoldenGate にリンクされているすべての表のソース・データベースでアクティビティを一時停止します。
2. GGSCI を起動します。
3. GGSCI で、Extract グループに対して次のコマンドを発行します。
INFO EXTRACT group
4. Checkpoint Lag 行で、Extract の遅れがあるかどうかを確認します。必要に応じて、遅れがゼロになるまで INFO EXTRACT を発行し続けますが、ゼロは現時点ですべてのトランザクション・データが処理されていることを示します。
5. Extract の実行中に、次のコマンドを発行します。
SEND EXTRACT group, VAMMESSAGE "control:terminate"
6. Extract グループを停止します。
STOP EXTRACT group
7. GGSCI で、Replicat グループに対して次のコマンドを発行します。
INFO REPLICAT group
8. Checkpoint Lag 行で、Replicat の遅れがあるかどうかを確認します。必要に応じて、遅れがゼロになるまで INFO REPLICAT を発行し続けますが、ゼロは証跡のすべてのデータが処理されていることを示します。
9. Replicat グループを停止します。
STOP REPLICAT group
10. ソース・データベースとターゲット・データベースで表変更を実行します。
11. Extract プロセスと Replicat プロセスを開始します。
START EXTRACT group
START REPLICAT group
12. Oracle GoldenGate にリンクされているすべてのソース表で再開するユーザー・アクティビティを許可します。

Oracle GoldenGate のアンインストール

この手順では、Oracle GoldenGate の証跡のデータが不要であり、現在の Oracle GoldenGate 環境を保持する必要がないことを前提とします。現在の環境およびデータを保持するには、この手順を開始する前に Oracle GoldenGate ディレクトリとすべてのサブディレクトリのバックアップを作成します。

7.1 Linux または UNIX からの Oracle GoldenGate のアンインストール

1. コマンド・シェルを実行します。
2. (推奨) システム管理者として、または Oracle GoldenGate コマンドを発行する権限とオペレーティング・システムからファイルおよびディレクトリを削除する権限を持つユーザーとしてログオンします。
3. GGSCI を実行します。
4. Extract の実行中に、次のコマンドを発行します。

```
SEND EXTRACT group, VAMMESSAGE "control:terminate"
```
5. Extract を停止します。

```
STOP EXTRACT group
```
6. Extract グループを強制的に削除します。

```
DELETE EXTRACT group !
```
7. Manager プロセスを停止します (対話型プロンプトの回避に次のコマンドの ! を使用できる場合)。

```
Stop Manager [!]
```
8. GGSCI を終了します。
9. 任意の Teradata クライアントから次のコマンドを発行します。

```
DROP REPLICATION GROUP repgroup name
```
10. インストール・ディレクトリを削除することで、Oracle GoldenGate のファイルを削除します。
11. 必要に応じて、データベースから Oracle GoldenGate 関連のオブジェクトを削除します。

7.2 Windows (非クラスタ) からの Oracle GoldenGate のアンインストール

1. (推奨) システム管理者として、または Oracle GoldenGate コマンドを発行する権限とオペレーティング・システムからファイルおよびディレクトリを削除する権限を持つユーザーとしてログオンします。
2. GGSCI を実行します。
3. Extract の実行中に、次のコマンドを発行します。

```
SEND EXTRACT group, VAMMESSAGE "control:terminate"
```
4. Extract を停止します。

```
STOP EXTRACT group
```
5. Extract グループを強制的に削除します。

```
DELETE EXTRACT group !
```
6. Manager プロセスを停止します (対話型プロンプトの回避に次のコマンドの ! を使用できる場合)。

```
STOP MANAGER [!]
```
7. GGSCI を終了します。
8. 任意の Teradata クライアントから次のコマンドを発行します。

```
DROP REPLICATION GROUP repgroup name
```
9. 「スタート」 → 「ファイル名を指定して実行」 をクリックし、「ファイル名を指定して実行」ダイアログ・ボックスに「cmd」と入力します。
10. ディレクトリを Oracle GoldenGate インストール・ディレクトリに変更します。
11. 次の構文を使用して INSTALL ユーティリティを実行します。

```
install deleteevents deleteservice
```

このコマンドによって、Windows Event Manager への Oracle GoldenGate イベントのレポートが停止され、Manager サービスが削除されます。
12. Windows の SYSTEM32 フォルダから CATEGORY.DLL ファイルと GGMSG.DLL ファイルを削除します。
13. Oracle GoldenGate インストール・フォルダを削除します。
14. 必要に応じて、データベースから Oracle GoldenGate 関連のオブジェクトを削除します。

7.3 Windows クラスタからの Oracle GoldenGate のアンインストール

1. Manager リソースが含まれているクラスタ・グループを所有するクラスタのノードにログインします。システム管理者として、または Oracle GoldenGate コマンドを発行する権限とオペレーティング・システムからファイルおよびディレクトリを削除する権限を持つユーザーとしてログオンします。
2. GGSCI を実行します。
3. Extract の実行中に、次のコマンドを発行します。

```
SEND EXTRACT group, VAMMESSAGE "control:terminate"
```


4. Extract を停止します。
`STOP EXTRACT group`
5. Extract グループを強制的に削除します。
`DELETE EXTRACT group !`
6. Manager プロセスを停止します (対話型プロンプトの回避に次のコマンドの!を使用できる場合)。
`STOP MANAGER [!]`
7. GGSCI を終了します。
8. クラスタ・アドミニストレータ・ツールを使用して Manager リソースをオフラインにします。
9. そのリソースを右クリックし、「削除」を選択して削除します。
10. 次の構文を使用して INSTALL ユーティリティを実行します。
`INSTALL DELETEEVENTS DELETESERVICE`

このコマンドによって、Windows Event Manager への Oracle GoldenGate イベントのレポートが停止され、Manager サービスが削除されます。
11. Windows の SYSTEM32 フォルダから CATEGORY.DLL ファイルと GGSMMSG.DLL ファイルを削除します。
12. クラスタ・グループをクラスタの次のノードに移動し、手順 10 から繰り返します。
13. 任意の Teradata クライアントから次のコマンドを発行します。
`DROP REPLICATION GROUP repgroup name`
14. Oracle GoldenGate インストール・フォルダを削除します。
15. 必要に応じて、データベースから Oracle GoldenGate 関連のオブジェクトを削除します。

インストールされる Oracle GoldenGate のコンポーネント

この付録では、Oracle GoldenGate のインストール・ディレクトリにある Oracle GoldenGate ソフトウェアによって作成または使用されるプログラム、ディレクトリおよびその他のコンポーネントについて説明します。特定のプラットフォーム上にインストールされる可能性がある追加ファイルは、ここに記載されていません。ここに記載されているファイルは、すべてのプラットフォーム上にインストールされるとはかぎりません。

A.1 Oracle GoldenGate のプログラムおよびユーティリティ

この項では、ルートの Oracle GoldenGate インストール・ディレクトリにインストールされるプログラムについて説明します。

注意：一部のプログラムはインストールに含まれない場合があります。たとえば、ご使用のプラットフォームで取得または配信のみが Oracle GoldenGate によってサポートされる場合には、`extract` プログラムまたは `replicat` プログラムはそれぞれインストールされません。同様に、特定のデータベースをサポートするために特有のファイルがインストールされることがあります。

表 A-1 インストールされる Oracle GoldenGate のプログラムおよびユーティリティ

プログラム	説明
convchk	チェックポイント・ファイルを新しいバージョンに変換します。
ddlgen	ソース・データベースの DDL に基づいて、ターゲット・データベースの表定義を生成します。主に、NonStop プラットフォームで使用されます。
defgen	データ定義が生成され、この定義は、ソース表とターゲット表の定義が異なる場合に Oracle GoldenGate のプロセスによって参照されます。
emscInt	Windows システムまたは UNIX システム上の Collector および Replicat によって作成されるイベント・メッセージを NonStop システム上の EMS に送信します。
extract	データベース表またはトランザクション・ログからキャプチャを実行するか、ベンダーのアクセス・モジュールからトランザクション・データを受信します。
ggmxinstall	SQL/MX データベース用の Oracle GoldenGate インストール・スクリプト。
ggsci	コマンドの発行やパラメータ・ファイルの管理に使用する Oracle GoldenGate へのユーザー・インターフェース。

表 A-1 インストールされる Oracle GoldenGate のプログラムおよびユーティリティ (続き)

プログラム	説明
ggsmgr.jcl ggsmgr.proc ggsmgrst.jcl ggsmgrst.proc	バッチ・ジョブまたは z/OS システム上のオペレータ・コンソールから Oracle GoldenGate の Manager プロセスを開始します。これは、DB2 z/OS データベースをサポートするためにインストールされます。
install	Oracle GoldenGate を Windows サービスとしてインストールし、Windows ベースのその他のサービス・オプションを提供します。
keygen	データ暗号化キーを生成します。
logdump	抽出証跡または抽出ファイルに格納される情報を表示および保存するためのユーティリティ。
mgr	リソース管理、Oracle GoldenGate のプロセスの制御と監視、レポートおよび GGSCI インタフェース経由のリクエストのルーティングに使用する (Manager) 制御プロセス。
replicat	データをターゲット・データベースの表に適用します。
reverse	トランザクション操作の順序を保持するユーティリティ。これによって、Replicat を使用してターゲット表の変更を取り消し、それらの表を前の状態に復元できるようになります。
server	Collector プロセス。これは、データをリモート証跡に書き込む Extract TCP/IP サーバー・コントローラです。
triggergen	トリガーベースの抽出方法をサポートする Oracle GoldenGate のログ表およびロギング・トリガーを作成するスクリプトを生成します。
vamserv	TMF 対応アプリケーションによって生成される TMF 監査証跡を読み取るために Extract によって開始されます。これは、SQL/MX データベースをサポートするためにインストールされます。

A.2 Oracle GoldenGate のサブディレクトリ

この項では、Oracle GoldenGate インストール・ディレクトリのサブディレクトリとその内容について説明します。

注意：一部のディレクトリはインストールに含まれない場合があります。

表 A-2 インストールされる Oracle GoldenGate のサブディレクトリ

ディレクトリ	説明
br	制限付きのリカバリ機能用のチェックポイント・ファイルが含まれます。
cfig	Oracle GoldenGate Monitor の構成に使用されるプロパティ・ファイルおよび XML ファイルが含まれます。
dirdb	Oracle GoldenGate Monitor アプリケーションまたは Oracle Enterprise Manager 内で使用される Oracle GoldenGate インスタンスから収集された情報の保持に使用するデータストアが含まれます。

表 A-2 インストールされる Oracle GoldenGate のサブディレクトリ (続き)

ディレクトリ	説明
dirchk	<p>Extract プロセスおよび Replicat プロセスによって作成されるチェックポイント・ファイルを格納します。これらのファイルには、データの正確さとフォルト・トレランスを確保するために現在の読み書き位置が保存されます。Oracle GoldenGate の内部形式で書き込まれます。</p> <p>ファイル名形式は <code>group_name+sequence_number.ext</code> で、<code>sequence_number</code> は古いファイルに追加される順序番号、<code>ext</code> は <code>cpe</code> (Extract チェックポイント・ファイル) または <code>cpr</code> (Replicat チェックポイント・ファイル) のいずれかです。</p> <p>これらのファイルを編集しないでください。</p> <p>例:</p> <p><code>ext1.cpe</code> <code>repl.cpr</code></p>
dirdat	<p>Replicat プロセスまたは別のアプリケーションやユーティリティによる追加処理に抽出されたデータを保存するために Extract プロセスによって作成される、Oracle GoldenGate の証跡ファイルおよび抽出ファイルのデフォルトの場所です。Oracle GoldenGate の内部形式で書き込まれます。</p> <p>ファイル名形式は、ユーザー定義の 2 文字の接頭辞とそれに続く 6 桁の順序番号 (証跡ファイル) または関連付けられた Extract プロセス・グループ (抽出ファイル) です。</p> <p>これらのファイルを編集しないでください。</p> <p>例:</p> <p><code>rt000001</code> <code>finance</code></p>
dirdef	<p>異機種間の同期化環境で使用されるソース・データまたはターゲット・データの定義を保存するために DEFGEN ユーティリティによって作成されるデータ定義ファイルのデフォルトの場所です。外部 ASCII で書き込まれます。ファイル名形式は、DEFGEN パラメータ・ファイルで指定されるユーザー定義の名前です。</p> <p>これらのファイルを編集して、新たに作成された表の定義を追加できます。定義ファイルの編集方法がわからない場合には、Oracle GoldenGate のテクニカル・サポートに問い合せてください。</p> <p>例:</p> <p><code>defs.dat</code></p>
dirjar	Oracle GoldenGate Monitor をサポートする Java 実行可能ファイルが含まれます。
dirout	このディレクトリは使用されなくなりました。
dirpcs	<p>ステータス・ファイルの既定の場所です。ファイル名形式は <code>group.extension</code> で、<code>group</code> はグループ名、<code>extension</code> は <code>pce</code> (Extract)、<code>pcr</code> (Replicat) または <code>pcm</code> (Manager) のいずれかです。</p> <p>これらのファイルが作成されるのは、プロセスを実行している間だけです。ファイルの内容はプログラム名、プロセス名、ポート番号およびプロセス ID です。</p> <p>これらのファイルを編集しないでください。</p> <p>例:</p> <p><code>mgr.pcm</code> <code>ext.pce</code></p>

表 A-2 インストールされる Oracle GoldenGate のサブディレクトリ (続き)

ディレクトリ	説明
dirprm	<p>Oracle GoldenGate プロセスのグループまたはユーティリティのランタイム・パラメータを保存するために Oracle GoldenGate ユーザーによって作成される、Oracle GoldenGate パラメータ・ファイルのデフォルトの場所です。外部 ASCII 形式で書き込まれます。ファイル名形式は、group name/user-defined name.prm または mgr.prm です。</p> <p>プロセスの停止後に Oracle GoldenGate パラメータの値を変更するために、これらのファイルを編集できます。これらのファイルは、テキスト・エディタで直接編集するか、GGSCI で EDIT PARAMS コマンドを使用して編集できます。</p> <p>例：</p> <p>defgen.prm finance.prm</p>
dirrec	Oracle GoldenGate では使用しません。
dirrpt	<p>処理実行に関連する統計情報をレポートするために Extract、Replicat および Manager の各プロセスによって作成されるプロセス・レポート・ファイルの既定の場所です。外部 ASCII 形式で書き込まれます。</p> <p>ファイル名形式は、group name+sequence number.rpt で、sequence number は古いファイルに追加される順序番号です。</p> <p>これらのファイルを編集しないでください。</p> <p>例：</p> <p>fin2.rpt mgr4.rpt</p>
dirsql	<p>triggen が非推奨になる前は、SQL スクリプトを格納するために triggen ユーティリティで使用されていました。現在は、Oracle GoldenGate をサポートするトレーニング・スクリプトや、ユーザー作成の SQL スクリプトの格納に使用されます。</p>
dirtmp	<p>キャッシュ・マネージャに割り当てられたメモリー・サイズを上回る場合に、トランザクション・データを保存するデフォルトの場所です。これらのファイルを編集しないでください。</p>
dirwlt	<p>Oracle GoldenGate Monitor をサポートする Oracle Wallet が含まれます。このディレクトリは、ウォレットを作成するユーティリティを実行するまでインストールされません。</p>
UserExitExamples	<p>ユーザー・イグジットの作成を支援するサンプル・ファイルが含まれます。</p>

A.3 その他の Oracle GoldenGate ファイル

この項では、ルート of Oracle GoldenGate インストール・ディレクトリに作成またはインストールされる、その他のファイル、テンプレートおよびオブジェクトについて説明します。

注意：一部のファイルは、データベースおよび OS プラットフォームに応じてご使用の環境にインストールされない場合があります。

表 A-3 インストールされるその他の Oracle GoldenGate ファイル

コンポーネント	説明
bcpfmt.tpl	Microsoft BCP/DTS 一括ロード・ユーティリティに実行ファイルを作成する際に Replicat で使用するテンプレート。
bcrypt.txt	Blowfish 暗号化ソフトウェアのライセンス契約。
cagent.dll	Oracle GoldenGate Monitor C サブエージェント用の Windows ダイナミック・リンク・ライブラリが含まれます。
category.dll	INSTALL ユーティリティによって使用される Windows ダイナミック・リンク・ライブラリ。
chkpt_db_create.sql	ローカル・データベースにチェックポイント表を作成するスクリプト。データベース・タイプごとに異なるスクリプトがインストールされます。
db2cnt1.tpl	IBM LOADUTIL 一括ロード・ユーティリティに制御ファイルを作成する際に Replicat で使用するテンプレート。
ddl_access.tpl	ソースの DDL を Microsoft Access の DDL に変換する DDLGEN ユーティリティで使用されるテンプレート。
ddl_cleartrace.sql	DDL トレース・ファイルを削除するスクリプト。(Oracle のインストール)
ddl_db2.tpl	ソースの DDL を DB2 の DDL (Linux、UNIX、Windows) に変換する DDLGEN ユーティリティで使用されるテンプレート。
ddl_db2_os390.tpl	ソースの DDL を DB2 の DDL (z/OS システム) に変換する DDLGEN ユーティリティで使用されるテンプレート。
ddl_ddl2file.sql	マーカー表からファイルに DDL を保存するスクリプト。
ddl_disable.sql	Oracle GoldenGate の DDL トリガーを無効にするスクリプト。(Oracle のインストール)
ddl_enable.sql	Oracle GoldenGate の DDL トリガーを有効にするスクリプト。(Oracle のインストール)
ddl_filter.sql	Oracle GoldenGate による DDL のフィルタリングをサポートするスクリプト。このスクリプトはプログラムで実行されるため、手動で実行しないでください。
ddl_informix.tpl	ソースの DDL を Informix の DDL に変換する DDLGEN ユーティリティで使用されるテンプレート。
ddl_mss.tpl	ソースの DDL を SQL Server の DDL に変換する DDLGEN ユーティリティで使用されるテンプレート。
ddl_mysql.tpl	ソースの DDL を MySQL の DDL に変換する DDLGEN ユーティリティで使用されるテンプレート。
ddl_nopurgeRecyclebin.sql	Oracle GoldenGate のサポート・スタッフが使用する空のスクリプト・ファイル。
ddl_nssql.tpl	ソースの DDL を NonStop SQL の DDL に変換する DDLGEN ユーティリティで使用されるテンプレート。
ddl_ora9.sql	Oracle GoldenGate の DDL サポートの一環としてプログラムで実行されるスクリプトのため、これらのスクリプトは実行しないでください。
ddl_ora10.sql	
ddl_ora11.sql	
ddl_ora10upCommon.sql	
ddl_oracle.tpl	ソースの DDL を Oracle の DDL に変換する DDLGEN ユーティリティで使用されるテンプレート。

表 A-3 インストールされるその他の Oracle GoldenGate ファイル (続き)

コンポーネント	説明
ddl_pin.sql	パフォーマンスを改善するために DDL トレース、DDL パッケージおよび DDL トリガーを固定するスクリプト。(Oracle のインストール)
ddl_purgeRecyclebin.sql	DDL のレプリケーション機能のサポートで Oracle のごみ箱をバージするスクリプト。
ddl_remove.sql	DDL 抽出トリガーおよびパッケージを削除するスクリプト。(Oracle のインストール)
ddl_session.sql ddl_session1.sql	Oracle の DDL オブジェクトのインストールをサポートします。このスクリプトはプログラムで実行されるため、手動で実行しないでください。
ddl_setup.sql	Oracle GoldenGate の DDL 抽出オブジェクトおよびレプリケーション・オブジェクトをインストールするスクリプト。(Oracle のインストール)
ddl_sqlmx.tpl	Tandem Enscribe の DDL を NonStop SQL/MX の DDL に変換する DDLGEN ユーティリティで使用されるテンプレート。
ddl_status.sql	Oracle GoldenGate の DDL サポート機能によって作成される各オブジェクトが存在し、正しく機能するかどうかを確認するスクリプト。(Oracle のインストール)
ddl_staymetadata_off.sql ddl_staymetadata_on.sql	Oracle の DDL トリガーでメタデータを収集するかどうかを制御するスクリプト。このスクリプトはプログラムで実行されるため、手動で実行しないでください。
ddl_sybase.tpl	ソースの DDL を Sybase の DDL に変換する DDLGEN ユーティリティで使用されるテンプレート。
ddl_tandem.tpl	ソースの DDL を NonStop SQL の DDL に変換する DDLGEN ユーティリティで使用されるテンプレート。
ddl_trace_off.sql ddl_trace_on.sql	DDL のトレースをオンにするかオフにするかを制御するスクリプト。
ddl_tracelevel.sql	DDL サポート機能のトレース・レベルを設定するスクリプト。(Oracle のインストール)
debug files	トレースがオンになった場合に存在する可能性があるデバッグ・テキスト・ファイル。
demo_db_scriptname.sql demo_more_db_scriptname.sql	チュートリアルや基本のテストで使用するデモンストレーション表を作成および移入するスクリプト。
.dmp files	Oracle GoldenGate プロセスにより、トレース目的で作成されるダンプ・ファイル。
ENCKEYS	暗号化キーを保存する、ユーザーが作成したファイル。外部 ASCII 形式で書き込まれます。
exitdemo.c	ユーザー・イグジット例。
exitdemo_utf16.c	ユーザー・イグジットとプロセスの間で交換される情報用のコールバック構造で、UTF16 エンコード・データを使用する方法を示すユーザー・イグジットの例。
freeBSD.txt	FreeBSD のライセンス契約。
ggmessage.dat	Oracle GoldenGate のプロセスによって返されるエラー、情報および警告のメッセージが含まれるデータ・ファイル。このファイルのバージョンはプロセスの開始時に確認されます。このバージョンは、プロセスが動作するようにそのプロセスのバージョンと一致していなければなりません。

表 A-3 インストールされるその他の Oracle GoldenGate ファイル (続き)

コンポーネント	説明
ggserr.log	Oracle GoldenGate によって生成される処理のイベント、メッセージ、エラーおよび警告を記録するファイル。
ggsmsg.dll	install プログラムによって使用される Windows のダイナミック・リンク・ライブラリ。
GLOBALS	Oracle GoldenGate のインスタンス全体に適用するパラメータを保存する、ユーザーが作成したファイル。
help.txt	GGSCI コマンド・インタフェースのヘルプ・ファイル。
icudt38.dll	International Components for Unicode 用の Windows 共有ライブラリ。
icuin38.dll	
icuc38.dll	
jagent.bat	Oracle GoldenGate Monitor の Java エージェント用の Windows バッチ・ファイル。
jagent.log	Oracle GoldenGate Monitor Agent のログ・ファイル。
jagentjni.log	
jagent.sh	Oracle GoldenGate Monitor の Java エージェント用の UNIX シェル・スクリプト。
LGPL.txt	Lesser General Public License ステートメント。これは、Free Software Foundation のフリー・ライブラリに適用されます。
libodbc.so	UNIX 上の Ingres 2.6 用の ODBC ファイル。
libodbc.txt	libodbc.so のライセンス契約。
libxml2.dll	Oracle GoldenGate XML プロシージャの XML ライブラリを含む Windows のダイナミック・リンク・ライブラリ。
libxml2.txt	libxml2.dll のライセンス契約。
marker.hist	NonStop ソース・システムからマーカーが渡された場合に Replicat によって作成されるファイル。
marker_remove.sql	DDL マーカー・ファイルを削除するスクリプト。(Oracle のインストール)
marker_setup.sql	Oracle GoldenGate の DDL マーカー表をインストールするスクリプト。(Oracle のインストール)
marker_status.sql	DDL マーカー表が正常にインストールされたことを確認するスクリプト。(Oracle のインストール)
notices.txt	サード・パーティ・ソフトウェアのライセンス・ファイル。
odbcinst.ini	UNIX 上の Ingres 2.6 用の ODBC 構成ファイル。
params.sql	DDL サポートの構成可能なパラメータを含むスクリプト。(Oracle のインストール)
pthread-win32.txt	pthread-VC.dll のライセンス契約。
pthread-VC.dll	Microsoft Windows の POSIX スレッド・ライブラリ。
prvtckm.plb	Oracle 暗号化データのレプリケーションをサポートします。
pw_agent_util.bat	Oracle GoldenGate Monitor Agent をサポートするスクリプト・ファイル。
pw_agent_util.sh	
role_setup.sql	Oracle GoldenGate の DDL サポートに必要なデータベース・ロールを作成するスクリプト。(Oracle のインストール)

表 A-3 インストールされるその他の Oracle GoldenGate ファイル (続き)

コンポーネント	説明
sampleodbc.ini	UNIX 上の Ingres 2.6 用のサンプル ODBC ファイル。
sqlldr.tpl	Oracle SQL*Loader 一括ロード・ユーティリティに制御ファイルを作成する際に Replicat で使用するテンプレート。
start.prm stop.prm	Manager プロセスを開始および停止する z/OS paramlib のメンバー。
startmgr stopmgr	GGSSCI から Manager プロセスを開始する z/OS Unix System Services のスクリプト。
startmgrcom stopmgrcom	Manager プロセス用の z/OS システム入力コマンド。
tcperrs	TCP/IP エラーに対応するユーザー定義の指示を含むファイル。
usrdecs.h	ユーザー・イグジット API。
xerces-c_2_8.dll	Apache XML のパーサー・ライブラリ。
zlib.txt	zlib 圧縮ライブラリのライセンス契約。

A.4 Oracle GoldenGate のチェックポイント表

データベースのチェックポイントを使用している場合、チェックポイント表は、ADD CHECKPOINTTABLE コマンドの実行時に、Oracle GoldenGate によってユーザー定義名でデータベースに作成されますが、chkpt_db_create.sql スクリプト (db はスクリプトがサポートするデータベース・タイプの略語) を使用してユーザーが作成することも可能です。

この表の列の名前または属性を変更しないでください。必要に応じて表保存属性を変更できません。

表 A-4 チェックポイント表の定義

列	説明
GROUP_NAME (主キー)	チェックポイント用にこの表を使用する Replicat グループ名。複数の Replicat グループで同じ表を使用できます。
GROUP_KEY (主キー)	GROUPNAME に加え、同じ表に書き込みを行っている Replicat のグループ数に関係なく、チェックポイントを一意に識別する一意の識別子。
SEQNO	チェックポイント・ファイルの順序番号。
RBA	チェックポイント・ファイルのチェックポイントの相対バイト・アドレス。
AUDIT_TS	チェックポイント・ファイルのチェックポイント位置のタイムスタンプ。
CREATE_TS	チェックポイント表が作成された日時。
LAST_UPDATE_TS	チェックポイント表が最後に更新された日時。
CURRENT_DIR	現在の Oracle GoldenGate のホーム・ディレクトリまたはフォルダ。