Oracle® GoldenGate

Oracle 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

B70206-02 (原本部品番号: E35957-03)

2015年2月



Oracle GoldenGate Oracle 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

B70206-02 (原本部品番号: E35957-03)

Copyright © 2010, 2015, 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を、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する場合は、次の通知が適用されます。

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際、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起因して損害が発生しても、Oracle Corporation およびその関連会社は一切の責任を負いかねます。

Oracle はオラクルおよびその関連会社の登録商標です。その他の社名、商品名等は各社の商標または登録商標である場合があります。

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。Oracle Corporation 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。Oracle Corporation 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

じめに	
対象読者	
ドキュメントのアクセシビリティについて	
関連ドキュメント	
表記規則	
システム要件およびプレインストール手順	
サポートされているプラットフォーム	
オペレーティング・システムの要件	
メモリーの要件	
ディスクの要件	
一時ディスクの要件	
ネットワーク	
オペレーティング・システムの権限	
Itanium の要件	
コンソール	
他のプログラム	
データベース構成	
各キャプチャ・モードでサポートされる Oracle データ型とオブジェクトのサマリー	
サポートされる Oracle データ型の詳細	
数値データ型	
文字データ型	
マルチバイト文字型	
バイナリ・データ型	
日付およびタイムスタンプのデータ型	
ラージ・オブジェクトのデータ型	
XML データ型	
ユーザー定義または抽象型	
サポートされていない Oracle データ型	
DML での Oracle オブジェクトと操作のサポートの詳細	•••••
表、ビューおよびマテリアライズド・ビュー	
順序	
Oracle DML でサポートされていないオブジェクトおよび操作	•••••
Oracle DDL のオブジェクトと操作のサポートの詳細	
Oracle DDL でサポートされているオブジェクトおよび操作	
サポートされていない Oracle DDL のオブジェクトおよび操作	
サポートされているオブジェクト名とサポートされていないオブジェクト名	

2 Oracle GoldenGate のインストール

インストールの概要	
Oracle GoldenGate のダウンロード	
ORACLE_HOME および ORACLE_SID の設定	
UNIX および Linux システムでの Oracle 変数の指定	
Winsows システムでの Oracle 変数の指定	
UNIX における動的ビルド用のライブラリ・パスの設定	
クラスタ内での Oracle GoldenGate のインストールの準備	
Oracle ユーザーとしてのインストール	
サポートされる Oracle クラスタ記憶域	
Oracle GoldenGate バイナリとファイルをインストールするクラスタ内の場所の決定	
Linux および UNIX での Oracle GoldenGate のインストール	
Windows での Oracle GoldenGate のインストール	
Windows クラスタへの Oracle GoldenGate のインストール	
Oracle GoldenGate ファイルのインストール	
Manager のカスタム名の指定	
Windows サービスとしての Manager のインストール	
Oracle GoldenGate のクラスタへの統合	
クラスタでの標準の要件	
Windows クラスタ・リソースとしての Oracle GoldenGate の追加	2
Oracle 順序のサポートのインストール	2
Oracle データベース用 Oracle GoldenGate の DDL サポートのイン DDL オブジェクトの概要 DDL オブジェクトのインストール	
DDL オブジェクトの概要	
DDL オブジェクトの概要 DDL オブジェクトのインストール ソースおよびターゲットの Oracle データベースでの Oracle Golde	nGate の構
DDL オブジェクトの概要 DDL オブジェクトのインストール ソースおよびターゲットの Oracle データベースでの Oracle Golde この手順でできること	nGate の構
DDL オブジェクトの概要 DDL オブジェクトのインストール ソースおよびターゲットの Oracle データベースでの Oracle Golde この手順でできること パラメータ・ファイルの作成と編集	nGate の構
DDL オブジェクトの概要	nGate の構成
DDL オブジェクトの概要 DDL オブジェクトのインストール ソースおよびターゲットの Oracle データベースでの Oracle Golde この手順でできること パラメータ・ファイルの作成と編集 Oracle GoldenGate を構成する基本的な手順の概要 プロセスとファイルの名前の選択	nGate の構成
DDL オブジェクトの概要 DDL オブジェクトのインストール ソースおよびターゲットの Oracle データベースでの Oracle Golde この手順でできること パラメータ・ファイルの作成と編集 Oracle GoldenGate を構成する基本的な手順の概要 プロセスとファイルの名前の選択 グループ名の選択	nGate の構成
DDL オブジェクトの概要 DDL オブジェクトのインストール ソースおよびターゲットの Oracle データベースでの Oracle Golde この手順でできること パラメータ・ファイルの作成と編集 Oracle GoldenGate を構成する基本的な手順の概要 プロセスとファイルの名前の選択 グループ名の選択 ファイル名の選択	nGate の構成
DDL オブジェクトの概要 DDL オブジェクトのインストール ソースおよびターゲットの Oracle データベースでの Oracle Golde この手順でできること パラメータ・ファイルの作成と編集 Oracle GoldenGate を構成する基本的な手順の概要 プロセスとファイルの名前の選択 グループ名の選択 ファイル名の選択 使用するキャプチャ方法の決定	nGate の構成
DDL オブジェクトの概要 DDL オブジェクトのインストール ソースおよびターゲットの Oracle データベースでの Oracle Golde この手順でできること パラメータ・ファイルの作成と編集 Oracle GoldenGate を構成する基本的な手順の概要 プロセスとファイルの名前の選択 グループ名の選択 ファイル名の選択 ファイル名の選択 クラシック・キャプチャについて	nGate の構成
DDL オブジェクトの概要 DDL オブジェクトのインストール ソースおよびターゲットの Oracle データベースでの Oracle Golde この手順でできること パラメータ・ファイルの作成と編集 Oracle GoldenGate を構成する基本的な手順の概要 プロセスとファイルの名前の選択 グループ名の選択 ファイル名の選択 ファイル名の選択 を使用するキャプチャ方法の決定 クラシック・キャプチャについて 統合キャプチャについて	nGate の構成
DDL オブジェクトの概要 DDL オブジェクトのインストール ソースおよびターゲットの Oracle データベースでの Oracle Golde この手順でできること パラメータ・ファイルの作成と編集 Oracle GoldenGate を構成する基本的な手順の概要 プロセスとファイルの名前の選択 グループ名の選択 ファイル名の選択 ファイル名の選択 キアチャ方法の決定 クラシック・キャプチャについて 統合キャプチャについて キャプチャ・モードの組合せ	nGate の構成
DDL オブジェクトの概要 DDL オブジェクトのインストール ソースおよびターゲットの Oracle データベースでの Oracle Golde この手順でできること パラメータ・ファイルの作成と編集 Oracle GoldenGate を構成する基本的な手順の概要 プロセスとファイルの名前の選択 グループ名の選択 ファイル名の選択 ファイル名の選択 セカシック・キャプチャ方法の決定 クラシック・キャプチャについて 統合キャプチャについて キャプチャについて テャプチャについて のTacle GoldenGate のデータベース・ユーザーの割当て	nGate の構成
DDL オブジェクトの概要 ソースおよびターゲットの Oracle データベースでの Oracle Golde この手順でできること パラメータ・ファイルの作成と編集 Oracle GoldenGate を構成する基本的な手順の概要 プロセスとファイルの名前の選択 グループ名の選択 ファイル名の選択 クラシック・キャプチャ方法の決定 クラシック・キャプチャについて 統合キャプチャについて キャプチャ・モードの組合せ Oracle GoldenGate のデータベース・ユーザーの割当て Oracle GoldenGate インスタンスの作成	nGate の構成
DDL オブジェクトの概要 DDL オブジェクトのインストール ソースおよびターゲットの Oracle データベースでの Oracle Golde この手順でできること パラメータ・ファイルの作成と編集 Oracle GoldenGate を構成する基本的な手順の概要 プロセスとファイルの名前の選択 グループ名の選択 ファイル名の選択 ファイル名の選択 セカラシック・キャプチャについて 統合キャプチャについて キャプチャ・モードの組合せ Oracle GoldenGate のデータベース・ユーザーの割当て Oracle GoldenGate インスタンスの作成 変更のキャプチャ用の Extract の構成	nGate の構成
DDL オブジェクトの概要 DDL オブジェクトのインストール ソースおよびターゲットの Oracle データベースでの Oracle Golde この手順でできること パラメータ・ファイルの作成と編集 Oracle GoldenGate を構成する基本的な手順の概要 プロセスとファイルの名前の選択 グループ名の選択 ファイル名の選択 セ用するキャプチャ方法の決定 クラシック・キャプチャについて 統合キャプチャについて キャプチャ・モードの組合せ Oracle GoldenGate のデータベース・ユーザーの割当て Oracle GoldenGate インスタンスの作成 変更のキャプチャ用の Extract の構成 プライマリ Extract の構成	nGate の構成
DDL オブジェクトの概要 DDL オブジェクトのインストール ソースおよびターゲットの Oracle データベースでの Oracle Golde この手順でできること パラメータ・ファイルの作成と編集 Oracle GoldenGate を構成する基本的な手順の概要 プロセスとファイルの名前の選択 グループ名の選択 ファイル名の選択 ファイル名の選択 のラシック・キャプチャ方法の決定 クラシック・キャプチャについて 統合キャプチャドこのいて キャプチャ・モードの組合せ Oracle GoldenGate のデータベース・ユーザーの割当て Oracle GoldenGate インスタンスの作成 変更のキャプチャ用の Extract の構成 プライマリ Extract の構成 データ・ポンプの構成	nGate の構成
DDL オブジェクトの概要 DDL オブジェクトのインストール ソースおよびターゲットの Oracle データベースでの Oracle Golde この手順でできること パラメータ・ファイルの作成と編集 Oracle GoldenGate を構成する基本的な手順の概要 プロセスとファイルの名前の選択 グループ名の選択 ファイル名の選択 ファイル名の選択 セ用するキャプチャ方法の決定 クラシック・キャプチャについて 統合キャプチャについて キャプチャ・モードの組合せ Oracle GoldenGate のデータベース・ユーザーの割当て Oracle GoldenGate インスタンスの作成 変更のキャプチャ用の Extract の構成 プライマリ Extract の構成 アライマリ Extract の構成 変更の配信用の Replicat の構成	nGate の構成
DDL オブジェクトの概要 DDL オブジェクトのインストール ツースおよびターゲットの Oracle データベースでの Oracle Golde この手順でできること パラメータ・ファイルの作成と編集 Oracle GoldenGate を構成する基本的な手順の概要 プロセスとファイルの名前の選択 グループ名の選択 クループ名の選択 使用するキャプチャ方法の決定 クラシック・キャプチャについて 統合キャプチャについて キャプチャ・モードの組合せ Oracle GoldenGate のデータベース・ユーザーの割当て Oracle GoldenGate インスタンスの作成 変更のキャプチャ用の Extract の構成 プライマリ Extract の構成 データ・ポンプの構成 変更の配信用の Replicat の構成 チェックポイント表の作成	nGate の構成
DDL オブジェクトの概要 DDL オブジェクトのインストール ソースおよびターゲットの Oracle データベースでの Oracle Golde この手順でできること パラメータ・ファイルの作成と編集 Oracle GoldenGate を構成する基本的な手順の概要 プロセスとファイルの名前の選択 グループ名の選択 ファイル名の選択 ファイル名の選択 セ用するキャプチャ方法の決定 クラシック・キャプチャについて 統合キャプチャについて キャプチャ・モードの組合せ Oracle GoldenGate のデータベース・ユーザーの割当て Oracle GoldenGate インスタンスの作成 変更のキャプチャ用の Extract の構成 プライマリ Extract の構成 アライマリ Extract の構成 変更の配信用の Replicat の構成	nGate の構成

構成のテスト	4
Dracle GoldenGate のためのデータベースの準備	
ソース表とターゲット表の整合性制約の準備	
ターゲット表でのトリガーと参照カスケード制約の無効化	
ターゲット表での制約チェックの延期	
ソース表とターゲット表での行の一意性の保証	
ロギング・プロパティの構成	
データベースレベルのサプリメンタル・ロギングの有効化	
スキーマレベルのサプリメンタル・ロギングの有効化	
表レベルのサプリメンタル・ロギングの有効化	
一意の行識別子がない表での行変更の制限	
文字セットの変換のサポート	
SETENV を使用した NLS_LANG の設定	
グローバリゼーション・サポートのその他の情報	
フェッチ・オプションの設定	
特別なデータ型の処理	
マルチバイト・キャラクタ型	
Oracle Spatial オブジェクト	
TIMESTAMP	
ラージ・オブジェクト (LOB)	
XML	
ユーザー定義型	
他のデータベース・プロパティの処理	
Oracle GoldenGate と Oracle Exadata を組み合せた使用	
Oracle Exadata への移行	
EHCC が有効な Exadata へのレプリケート	
クラシック・キャプチャを使用する場合の追加構成手順 クラシック・キャプチャ・モードでの Oracle TDE データの構成	
クラシック・キャプチャでの TDE サポートの概要	
クラシック・キャプチャ・モードでの TDE のキャプチャの要件	
必要なデータベースのパッチ	
TDE サポートの構成	
復号化後のデータのセキュリティを維持するための推奨事項	
TDE キャプチャがアクティブな場合の DDL の実行	
パラメータ・ファイルでの Oracle 共通鍵の更新	
Oracle RAC 環境での Oracle GoldenGate の使用	
クラシック・キャプチャ・モードの堪合の Oracle ASM インスタンスからのキャプチャ	
クラシック・キャプチャ・モードの場合の Oracle ASM インスタンスからのキャプチャ ASM 内のトランザクション・ログへのアクセス	
ASM 内のトランザクション・ログへのアクセス	

保存するデータ量の判断	
ログ・アーカイブのパージ	
アーカイブ場所の指定	
その他のプラットフォームに保存されるログのマウント	
アーカイブ・ログのみを読み取るための Oracle GoldenGate の構成	
ALO モードの制限および要件	
ALO モードの Extract の構成	
ログ読取りのボトルネックの回避	
Dracle データベース用 DDL 同期の構成	
DDL 同期の概要	
Oracle GoldenGate DDL サポートの制限	
DDL 文の長さ	
サポートされているトポロジ	
フィルタリング、マッピングおよび変換	
名前変更	
表からのフェッチと DDL との相互作用	
SQL 内のコメント	
コンパイル・エラー	
時間隔パーティション化	
DDL サポートに関する構成のガイドライン	
データベースの権限	
パラレル処理	
データ・ポンプの DDL と DML	
オブジェクト名	
データ定義	
切捨て	
初期同期	
CREATE または RENAME の後のデータ継続性	
DDL スコープの理解	
マップされるスコープ	
マップされないスコープ	
他のスコープ	
DDL 内の修飾されていないオブジェクト名の正しい識別	
DDL サポートの有効化	
DDL レプリケーションのフィルタリング	
PL/SQL コードによるフィルタ	
フィルタ・ルールの追加および削除	
DDL パラメータを使用したフィルタリング	
DDL パラメータ・オプションの組合せ	
特別なフィルタのケース	
DDL EXCLUDE ALL	
暗黙的 DDL	
Oracle GoldenGate による導出オブジェクト名の処理方法	
ベース・オブジェクトに対する MAP はあるが、導出オブジェクトに対する MAP はない場合	
ベース・オブジェクトと導出オブジェクトに対する MAP がある場合	
道出オブジェカトに対する MAP けなるが、ベース・オブジェカトに対する MAP けたい担今	

導出オブジェクトとしての新規表	
導出オブジェクトのマッピングの無効化	
DDL 文字列置換の使用	
Replicat によって実行される DDL の伝播の制御	
- アクティブ / アクティブ (双方向) 構成での DDL の伝播	
カスケード構成での DDL の伝播	
サプリメンタル・ログ・グループの自動追加	•••••
レプリケートされた DDL からのコメントの削除	
IDENTIFIED BY パスワードのレプリケート	
処理に対する DDL の評価方法	
DDL 処理エラーの処理	
DDL トリガー・エラーの処理	
DDL レポート情報の表示	
Extract DDL レポート	
プロセス・レポートの統計	
DDL 履歴表のメタデータの表示	
DDL 処理のトレース	
DDL トリガーのトレース	
インスタンス化の前提条件	
変更のキャプチャと配信の構成	
競合処理の追加	
DDL 処理の無効化	
ターゲット表の準備	
インスタンス化の手順の効率化	
プロセス・グループ間でのパラメータの共有	
パラレル・プロセスの使用	
初期ロードの構成	
データベース・ユーティリティを使用したロード	
SQL*Loader へのダイレクト・バルク・ロード	
入力ファイルから SQL*Loader へのロード	
Extract のマイニング・データベースでの登録	
変更キャプチャ・プロセスおよび変更配信プロセスの追加 RMAN アーカイブ・ログ削除ポリシーの設定	
プライマリ Extract の追加	
ローカル証跡の追加	
データ・ポンプ Extract グループの追加	
リモート証跡の追加	
Replicat グループの追加	
データベース・ユーティリティを使用する場合のインスタンス化の実行手順	
SQL*Loader へのダイレクト・バルク・ロードを使用する場合のインスタンス化の実行手順	頁
入力ファイルから SQL*Loader の場合のインスタンス化の実行手順	
- · · · · · · · · · · · · · · · · · · ·	
インスタンス化後の処理の監視	

9	プロセスの制御	
	プロセスを起動するとき	9-1
	インスタンス化の完了後のプロセスの起動	9-1
10	Oracle DDL レプリケーション環境の管理	
	DDL トリガーの有効化と無効化	10-1
	DDL ドッカーの有効化と無効化	
	DDL マーカー表の削除	
	DDL 、	
	DDL 履歴表の削除	
	DDL トレース・ファイルのパージ	
	DDL サポートが有効である場合のデータベースのパッチおよびアップグレードの適用	
	DDL サポートが有効である場合の Oracle GoldenGate のパッチおよびアップグレードの適用	
	既存の DDL 環境のクリーン状態への復元	
	システムからの DDL オブジェクトの削除	
	0.00 als 0.01da.00da.0073.43.71	
11	Oracle GoldenGate のアンインストール	
	プロセスの停止	
	DDL 環境の削除	
	データベース・オブジェクトの削除	
	(Windows) Oracle GoldenGate Windows コンポーネントの削除	
	Oracle GoldenGate ファイルの削除	11-3
Α	統合キャプチャ用のダウンストリーム・マイニング・データベースの構成	
	ダウンストリーム・デプロイ用のキャプチャ・オプションの評価	A-1
	ダウンストリーム・デプロイ用のソース・データベースの準備	A-2
	ソース・ユーザー・アカウントの作成	A-2
	ソースからダウンストリーム・マイニング・データベースへの REDO 転送の構成	A-2
	ダウンストリーム・マイニング・データベースの準備	A-4
	ダウンストリーム・マイニング・ユーザー・アカウントの作成	
	ローカル REDO ログ・ファイルをアーカイブするためのマイニング・データベースの構成	
	リアルタイム・キャプチャ用のダウンストリーム・マイニング・データベースの準備	
В	統合キャプチャ用のダウンストリーム・マイニング・データベースの構成例	
	例1: リアルタイム・モードでの1つのソース・データベースからのキャプチャ	B-1
	ローカル REDO をアーカイブするためのマイニング・データベースの準備	B-1
	ソース・データベースから受信した REDO をスタンバイ REDO ログにアーカイブするためのマイ	'ニン
	グ・データベースの準備 B-2	
	REDO をマイニング・データベースに送信するためのソース・データベースの準備	B-2
	DBMSCAP での統合キャプチャ (ext1) の設定	B-2
	例 2: アーカイブログのみモードでの複数のソースからのキャプチャ	B-3
	ローカル REDO をアーカイブするためのマイニング・データベースの準備	В-З
	ソース・データベースからの REDO をアーカイブするためのマイニング・データベースの準備	B-4
	REDO をマイニング・データベースに送信するための1つ目のソース・データベースの準備	B-4
	REDO をマイニング・データベースに送信するための2つ目のソース・データベースの準備	
	ダウンストリーム・マイニング・データベースでの Extract の設定	
	, , . , , , , , , ,	

	例 3: リアルタイム・モードとアーカイブログのみモードが混在する複数ソースからのキャプチャ	B-6
	ローカル REDO をアーカイブするためのマイニング・データベースの準備	B-6
	ソース・データベースから REDO を受け入れるためのマイニング・データベースの準備	B-7
	REDO をマイニング・データベースに送信するための1つ目のソース・データベースの準備	B-7
	REDO をマイニング・データベースに送信するための2つ目のソース・データベースの準備	B-8
	REDO をマイニング・データベースに送信するための3つ目のソース・データベースの準備	B-8
	ダウンストリーム・マイニング・データベースでの Extract の設定	B-9
С	XML スキーマの変更のサポート	
	RegisterSchema のサポート	C-1
	DeleteSchema のサポート:	C-1
		C 1
	CopyEvolve のサポート	C-1
D	Oracle GoldenGate とのアクティブ / アクティブ伝播のための DBFS の準備	
D	Oracle GoldenGate とのアクティブ / アクティブ伝播のための DBFS の準備サポートされている操作および前提条件	D-1
D	Oracle GoldenGate とのアクティブ / アクティブ伝播のための DBFS の準備 サポートされている操作および前提条件	D-1 D-1
D	Oracle GoldenGate とのアクティブ / アクティブ伝播のための DBFS の準備 サポートされている操作および前提条件	D-1 D-1 D-2
D	Oracle GoldenGate とのアクティブ / アクティブ伝播のための DBFS の準備 サポートされている操作および前提条件 必要なパッチの適用 これらのプロシージャで使用されている例 DBFS 順序番号のパーティション化	D-1 D-1 D-2 D-2
D	Oracle GoldenGate とのアクティブ / アクティブ伝播のための DBFS の準備 サポートされている操作および前提条件	D-1 D-1 D-2 D-2 D-3
	Oracle GoldenGate とのアクティブ / アクティブ伝播のための DBFS の準備 サポートされている操作および前提条件 必要なパッチの適用 これらのプロシージャで使用されている例 DBFS 順序番号のパーティション化 DBFS ファイル・システムの構成	D-1 D-1 D-2 D-2 D-3
	Oracle GoldenGate とのアクティブ / アクティブ伝播のための DBFS の準備 サポートされている操作および前提条件 必要なパッチの適用 これらのプロシージャで使用されている例 DBFS 順序番号のパーティション化 DBFS ファイル・システムの構成 ローカル・ピアとリモート・ピアの正しいマップ	D-1 D-1 D-2 D-2 D-3
	Oracle GoldenGate とのアクティブ / アクティブ伝播のための DBFS の準備 サポートされている操作および前提条件 必要なパッチの適用 これらのプロシージャで使用されている例 DBFS 順序番号のパーティション化 DBFS ファイル・システムの構成 ローカル・ピアとリモート・ピアの正しいマップ インストールされる Oracle GoldenGate コンポーネント Oracle Goldengate プログラムとユーティリティ Oracle Goldengate サブディレクトリ	D-1 D-2 D-2 D-3 D-4
	Oracle GoldenGate とのアクティブ / アクティブ伝播のための DBFS の準備 サポートされている操作および前提条件 必要なパッチの適用 これらのプロシージャで使用されている例 DBFS 順序番号のパーティション化 DBFS ファイル・システムの構成 ローカル・ピアとリモート・ピアの正しいマップ インストールされる Oracle GoldenGate コンポーネント Oracle Goldengate プログラムとユーティリティ	D-11 D-2 D-2 D-3 D-4 E-1 E-2 E-4

はじめに

Oracle GoldenGate for Oracle を使用すると、次のことが可能です。

- サポートされているほぼ同じ Oracle バージョン間や異なる Oracle バージョン間、あるいは サポートされている Oracle のバージョンとサポートされているその他のデータベースとの 間のトランザクション・データ変更のマップ、フィルタおよび変換。
- 異種 Oracle データベース間の Oracle DDL 操作のレプリケートおよびフィルタ。
- Oracle またはその他のデータベースのターゲット表への初期ロードと同期されたレプリケーション環境のインスタンス化。

このガイドには、Oracle データベース・システムでの Oracle GoldenGate のインストール、構成および実行を始めるに際して有用な情報が記載されています。Oracle 環境に適した、基本的な Oracle GoldenGate 構成 (ソースからターゲットまで)の構築に役立ちます。必要に応じて、ニーズに合せて構成を拡張するための追加情報が記載された他のドキュメントを示します。

対象読者

このガイドは、Oracle GoldenGate をインストール、構成および実行するインストール実行者、データベース管理者および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

オラクル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 Support へのアクセス

Oracle のお客様は、My Oracle Support にアクセスして電子サポートを受けることができます。 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・セットには、次のコンポーネントが含まれます。

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、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 インストレーション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 Adapter for Flat Files 管理者ガイド』
- 『Oracle GoldenGate Adapter for Java 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

このマニュアルでは次の表記規則を使用します。

表記規則	意味
太字	太字は、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 ("「ファイル」メニューから「 保存」 を選択します"など)を示します。太字は、本文中で定義されている用語および用語集に記載されている用語も示します。
イタリック体	イタリックは、ユーザーが特定の値を指定するプレースホルダ変数(パラメータ文 TABLE table_name など)を示します。イタリックは、ドキュメントのタイトルおよび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・イグジットやスクリプトなどのコードのコンポーネント、ファイルおよびデータベース・オブジェクトの名前、URLのパス、および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、Oracle GoldenGateのパラメータ、コマンド、ユーザーが構成可能な関数および SQL コマンドとキーワードを表すために使用されます。
大文字	通常フォントの大文字は、特別な場合を除き、ユーティリティの名前を表 します。

表記規則	意味
{}	構文内の中カッコは、パイプ記号で区切ったオプションのセットを囲み、 その中の 1 つを選択する必要があることを表します。たとえば、 $\{option1 \mid option2 \mid option3\}$ です。
[]	構文内の大カッコは、オプションの要素を示します。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。オプション要素内の複数の要素は、パイプ記号で区切り ます。たとえば、[option1 option2]です。

システム要件およびプレインストール手順

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。

注意: Oracle GoldenGate では、Oracle ソース・データベースに対して2つ のキャプチャ・モードがサポートされます。一部のシステム要件は、選択さ れたキャプチャ・モードに応じて異なります。キャプチャ・モードの詳細は、 4-3ページの「使用するキャプチャ方法の決定」を参照してください。

1.1 サポートされているプラットフォーム

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。

1.2 オペレーティング・システムの要件

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 について概説します。

1.2.1 メモリーの要件

Oracle GoldenGate に必要なメモリー量は、実行する同時プロセス数によって異なります。 ソース・システムには少なくとも、ソース・データをキャプチャするプライマリ Extract プロセ ス、およびネットワーク経由で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・システムには、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つあります。場合によっては、必要な構成に応 じてこれらのプロセスがすべて同じシステム上で動作することがあります。

レプリケートが必要なトランザクション・データを大量に生成する環境の場合、追加のパラレ ル・プロセスを使用したスループットの改善が必要になる場合があります。Oracle GoldenGate では、Oracle GoldenGate のインスタンスごとに最大 5,000 個の 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 の各プロセスでは、約25~55MBのメ モリーが必要であり、トランザクションのサイズと同時トランザクション数によってはそれ以 上のメモリーが必要です。

Oracle GoldenGate のプロセスによって使用され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・マネージャでは、オペレーティング・システムのメモリー管理機能を 活用して、Oracle GoldenGate プロセスが継続的かつ効率的な方法で動作できるようにします。 Oracle GoldenGate のメモリー要件を求めるには、『Oracle GoldenGate Windows and UNIX *Reference Guide*』の CACHEMGR パラメータに関する項を参照してください。

1.2.2 ディスクの要件

次の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・ファイルの領域と圧縮されていないファイルの領域が含まれます。 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。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の領域を割り当てます。
- 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 使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 Oracle ユーザーとしてインストールします。詳細は、2-5 ページの「クラスタ内での **Oracle GoldenGate** のインストールの準備」を参照してください。
- 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 のディスク領域。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 異なるため、これとは多少異なる容量が必要となる場合があります。『Oracle GoldenGate Administrator's Guide』で証跡のサイズ設定のガイドラインを参照してください。

1.2.3 一時ディスクの要件

デフォルトでは、Oracle GoldenGate によって Oracle GoldenGate インストール・ディレクト リの dirtmp サブディレクトリに、ディスクにスワップされるデータが保持されます。キャッ シュ・マネージャでは、システム上のすべての空き領域を使用できることを前提とします。ト ランザクション・ボリュームとトランザクション・サイズが大きい場合、このディレクトリは すぐに一杯になります。I/O の競合とディスクに関連する Extract の失敗が起こらないようにす るには、ディスクをこのディレクトリ専用にします。CACHEMGR パラメータの CACHEDIRECTORY オプションを使用して、このディレクトリに名前を割り当てることができます。

1.2.4 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・リソースが使用可能である必要 があります。

- 最適なパフォーマンスと信頼性を得る(特に、ターゲットでの待機時間を低く抑える)に は、可能なかぎり高速のネットワークを使用し、すべての障害点に冗長性をインストール
- DNS を含む TCP/IP サービスを使用するようにシステムを構成します。Oracle GoldenGate は IPv4と IPv6 をサポートし、これらのプロトコルのいずれか、または両方ともがサポー トされるシステムで稼働します。
- 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 では、予約されていない無制限の TCP/IP ポートが必要とされますが、 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。必要なポートを処理する よう Manager プロセスを構成する方法の詳細は、『Oracle GoldenGate Windows and UNIX *Administrator's Guide*』を参照してください。
- Oracle GoldenGate に割り当てたポートを記録しておきます。 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。

1.2.5 オペレーティング・システムの権限

Oracle GoldenGate をインストールしてプロセスを実行するために必要なオペレーティング・ システムの権限は次のとおりです。

- Windows 上にインストールするには、Oracle GoldenGate をインストールするユーザーが 管理者としてログインする必要があります。
- UNIX 上にインストールするには、Oracle GoldenGate をインストールするユーザーが Oracle GoldenGate インストール・ディレクトリに対する読込み / 書込み権限を持っている 必要があります。
- Oracle GoldenGate Extract、Replicat および Manager のプロセスは、Oracle GoldenGate ディレクトリ内のファイルおよびサブディレクトリの読取り、書込みおよび削除を行う権 限を持つオペレーティング・システム・ユーザーとして動作する必要があります。さらに、 Manager プロセスには、他の Oracle GoldenGate のプロセスを制御する権限が必要です。
- (クラシック・キャプチャ・モード) クラシック・キャプチャ・モードでは、Extract プロセ スで REDO ログを直接読み取り、オンラインとアーカイブの両方のログ・ファイルの読取 りアクセス権限を持つオペレーティング・システム・ユーザーとして操作する必要があり ます。UNIX システムの場合、ユーザーは Oracle インスタンスを所有するグループのメン バーであることが必要です。このドキュメントのインストール手順で Manager プロセスを Windows サービスとしてインストールするには、正しい権限が割り当てられるように管理 者としてインストールする必要があります。Manager をサービスとしてインストールでき ない場合、読取りアクセス権限を手動で Extract プロセスに割り当ててから、常に管理者と して Manager および Extract を実行します。
- Extract、Replicat および Manager のオペレーティング・システム・ユーザーを Oracle GoldenGate 専用とします。Oracle GoldenGate のプロセスを実行するユーザーは、データ ベース認証の構成方法に応じて機密情報を入手できる場合があります。

1.2.6 Itanium の要件

Microsoft Itanium システム上に Oracle GoldenGate をインストールするには、vcredist IA64.exe ランタイム・ライブラリ・パッケージをインストールする必要があります。このパッ ケージは Microsoft Web サイトからダウンロードできます。このパッケージには、Oracle GoldenGate が Itanium プラットフォーム上で動作するのに必要な VisualStudio DLL が含まれ ています。これらのライブラリをインストールしない場合、Oracle GoldenGate によって次の エラーが生成されます。

The application failed to initialize properly (0xc0150002). Click on Ok to terminate the application.

1.2.7 コンソール

オペレーティング・システムとコマンド・コンソールの文字セットが同じである必要がありま す。オペレーティング・システムがある文字セットに設定され、DOS コマンド・プロンプトで 別の古い DOS 文字セットを使用する Microsoft Windows システムでは不一致が起こります。 Oracle GoldenGate では、オペレテーィング・システムの文字セットを使用して GGSCI コマン ド出力に情報を送信します。その結果、コンソールの文字セットの不一致が原因で文字が正し く表示されません。次の DOS コマンドを使用して、GGSCI セッションを開く前にコンソール の文字セットを設定できます。

chcp OS character set

コード・ページの設定後、文字が正しく表示されない場合、拡張文字セットを持つ Lucida Console にコンソール・フォントを変更してみます。

1.2.8 他のプログラム

Oracle GoldenGate をサポートするために考慮すべきその他の事項を次に示します。

- 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。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ます。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。詳細およびこのパッケージのダウンロードについては、http://www.microsoft.comにアクセスしてください。
- 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。たとえば、RHAS 4.0 仮想マシンで Oracle 11g を実行している Windows システムに、実際の Linux マシンと同様に Oracle 11g の RHAS 4.0 ビルドをインストールします。

1.3 データベース構成

この項では、Oracle データベースに固有の Oracle GoldenGate 要件を示します。

- Windows システム上で複数の Oracle インスタンスに対して Oracle GoldenGate を実行するには、それぞれに対して Oracle GoldenGate のインスタンスをインストールする必要があります。
- 32 ビットの Oracle データベースが存在する 64 ビットの Sun Solaris、HP Tru64 (OSF/1) および LINUX マシンの場合、Oracle GoldenGate では 32 ビットの Oracle ライブラリが含まれるように LD_LIBRARY_PATH を設定する必要があります。 LD_LIBRARY_PATH の設定の詳細は、2-3 ページの「UNIX における動的ビルド用のライブラリ・パスの設定」を参照してください。
- Bequeath 接続を使用するようにデータベースが構成されている場合、sqlnet.ora ファイル に bequeath detach=true 設定が含まれている必要があります。
- (統合キャプチャ・モード)Oracle GoldenGate 11.2.1 では、統合 Extract 11.2.x 用 11.2.0.3 データベース固有のバンドル・パッチ (Doc ID 1411356.1) が適用された Oracle 11.2.0.3 に対する統合キャプチャのサポートが導入されました。このモードでは、ソース・システムまたはダウンストリーム Oracle データベースのログマイニング・サーバーを使用します。詳細は、4-3 ページの「使用するキャプチャ方法の決定」を参照してください。
- Oracle GoldenGate のプログラムが Oracle XDK ライブラリにアクセスできるように、完全な Oracle クライアントを Oracle GoldenGate と併用する必要があります。それらのライブラリがない Oracle Instant Client は使用しないでください。完全なクライアントは Oracle Web サイトからダウンロードできます。
- Oracle Real Application Cluster (RAC) 環境に Oracle GoldenGate をインストールするには、RAC ノードによってアクセスされる共有ドライブに Oracle GoldenGate をインストールします。詳細は、2-5 ページの「クラスタ内での Oracle GoldenGate のインストールの準備」を参照してください。
- 追加のデータベース・ユーザーの権限と構成要件については、このマニュアル内の他の箇所で説明しています。

1.4 各キャプチャ・モードでサポートされる Oracle データ型と オブジェクトのサマリー

この項では、選択したキャプチャ・モードに応じて Oracle GoldenGate でサポートされる Oracle データ型のサマリー情報と詳細情報を示します。キャプチャ・モードの詳細は、4-3 ページの「使用するキャプチャ方法の決定」を参照してください。

Oracle データ型、オブジェクトおよび操作のサポートの詳細は、1-7ページの「サポートされ る Oracle データ型の詳細」に記載されています。

表 1-1 キャプチャ・モードごとにサポートされているデータ型

データ型	クラシック・キャプチャ	統合キャプチャ
DATE 列および DATETIME 列を含むスカラー列	REDO からキャプチャされます。	REDO からキャプチャされます。
BASICFILE LOB 列	DML (INSERT/UPDATE/DELETE) を介して行われた LOB の変更は、REDO からキャプチャされます。	LOB の変更はすべて (DML を介したものも DBMS_LOB パッケージを介したものも) REDO からキャプチャされます。
	DBMS_LOB パッケージを介して行われた LOB の変更は、ベース表から値をフェッチして キャプチャされます。	
SECUREFILE LOB	SECUREFILE LOB は、LOB が変換(圧縮、暗号化など)されておらず、行外に格納される場合および変更が DML 文を介して行われている場合にのみ、REDO からキャプチャされます。 、次の場合、LOB はベース表からフェッチされます。 LOB は暗号化されています。 LOB は行内に格納されています。 LOB は DBMS_LOB パッケージを介して変更されています。 LOB は 直複除外されています。 LOB は フラグメント操作を介して変更されています。	SECUREFILE LOB は、次の場合を除き、REDOからキャプチャされます。 LOB は重複除外されています。 LOB はフラグメント操作を介して変更されています。 NOLOGGING LOB 次の場合、LOB はベース表からフェッチされます。 LOB は重複除外されています。 LOB は可ラグメント操作を介して変更されています。 NOLOGGING LOB ソース・データベースの COMPATIBLE 設定が11.2.0.0.0 以上である必要があります。
索引構成表 (IOT)	■ NOLOGGING LOB 次の制限付きで REDO からキャプチャされます。 ■ マッピング表を使用する IOT はサポートされません。	REDO からキャプチャされます。
	 IOT表への直接ロード挿入では、 SORTED 句は使用できません。 COMPRESS 句で指定された接頭辞圧縮を 使用する IOT はサポートされません。 	
CLOB として格納された XML	■ REDO からキャプチャされます。	REDO からキャプチャされます。 ソース・データベース互換性が 11.0.0.0.0 以 上に設定されている必要があります。
バイナリとして格納された XML	ベース表からフェッチされます。	REDO からキャプチャされます。 ソース・データベース互換性が 11.2.0.3.0 以上に設定されている必要があります。

表 1-1 キャプチャ・モードごとにサポートされているデータ型 (続き)

データ型	クラシック・キャプチャ	統合キャプチャ
オブジェクト・リレーショ	サポートされません。	REDO からキャプチャされます。
ナルとして格納された XML		ソース・データベース互換性が 11.2.0.3.0 以上に設定されている必要があります。
XMLTyp 表	サポートされません。	REDO からキャプチャされます。
ADT(抽象データ型)	ソース表からフェッチされます。	ソース表からフェッチされます。
		ソース・データベース互換性が 11.2.0.3.0 以上に設定されている必要があります。
コレクション (VARRAY)	ソース表からフェッチされます。	ソース表からフェッチされます。
		ソース・データベース互換性が 11.2.0.0.0 以上に設定されている必要があります。
コレクション (ネストされ	制限付きでソース表からフェッチされます。	制限付きでソース表からフェッチされます。
た表)	1-12 ページの「DML での Oracle オブジェクトと操作のサポートの詳細」を参照してください。	1-12 ページの「DML での Oracle オブジェクトと操作のサポートの詳細」を参照してください。
オブジェクト表	ソース表からフェッチされます。	ソース表からフェッチされます。
Transparent Data	REDO からキャプチャされます。	REDO からキャプチャされます。
Encryption(列の暗号化と表領域の暗号化)	追加手順が必要です。6-1ページの「クラ	追加設定が必要です。
M-M-7-10 /	シック・キャプチャ・モードでの Oracle TDE データの構成」を参照してください。	ソース・データベース互換性が 11.0.0.0.0 以 上に設定されている必要があります。
基本圧縮	サポートされません。	REDO からキャプチャされます。
		ソース・データベース互換性が 11.2.0.0.0 以 上に設定されている必要があります。
OLTP 圧縮	サポートされません。	REDO からキャプチャされます。
		ソース・データベース互換性が 11.2.0.0.0 以 上に設定されている必要があります。
Exadata Hybrid Columnar	サポートされません。	REDO からキャプチャされます。
圧縮		ソース・データベース互換性が 11.2.0.0.0 以 上に設定されている必要があります。
非 RAC データベース上の XA	REDO からキャプチャされます。	REDO からキャプチャされます。
RAC データベース上の XA	サポートされません。	REDO からキャプチャされます。
	サポートされるには、XA のすべてのブラン チが同一インスタンスに向けられているこ とを確認する必要があります。	ソース・データベース互換性が 11.2.0.0.0 以上に設定されている必要があります。
非 RAC データベース上の PDML	REDO からキャプチャされます。	REDO からキャプチャされます。
RAC データベース上の	サポートされません。	REDO からキャプチャされます。
PDML	サポートされるには、PDML トランザク ションから生成される子トランザクション が複数のインスタンスにまたがらないこと を確認する必要があります。	

1.5 サポートされる Oracle データ型の詳細

次に、Oracle GoldenGate でサポートされる Oracle データ型の詳細を概説します。特に断りの ないかぎり、クラシックと統合の両方のキャプチャ・モードでサポートされます。これらの モードの詳細は、4-1 ページの「ソースおよびターゲットの Oracle データベースでの Oracle GoldenGate の構成」を参照してください。

1.5.1 数値データ型

次の数値データ型がサポートされます。

- Oracle によって許可されている最大サイズまでの NUMBER
- BINARY FLOAT
- BINARY DOUBLE

1.5.1.1 サポートの制限

浮動小数点数の範囲および精度のサポートは、ホスト・マシンによって異なります。一般には、 精度は16桁が正確ですが、想定される近似値を判断するにはデータベースのドキュメントを確 認してください。Oracle GoldenGate では、サポートされる精度を超える値は丸められるか切り 捨てられます。

1.5.2 文字データ型

次の文字データ型がサポートされます。

- CHAR
- VARCHAR2
- LONG
- NCHAR
- NVARCHAR2

1.5.3 マルチバイト文字型

次のマルチバイト文字型がサポートされます。

- NCHAR および NVARCHAR2 のマルチバイト文字データ型
- CHAR 列および VARCHAR2 列に保存されるマルチバイト・データ

1.5.3.1 サポートの制限

- Oracle GoldenGate でマルチバイト文字データをサポートするには、ソースとターゲットの データベースが、レプリケートされる表と順序のスキーマ定義に関して論理上同一である 必要があります。変換、フィルタリングおよび他の操作はサポートされていません。2つ のデータベース間の文字セットは、次のいずれかである必要があります。
 - 同一。たとえば、ソースとターゲットで SHIFT-IIS などです。
 - 同等。同じ文字セットではありませんが、同じ文字のセットを含みます (SHIFT-JIS と EUC-JP など)。
 - ターゲットはソースのスーパーセットです。たとえば、UNICODE はすべての文字タ イプのスーパーセットです。したがって、他のすべての文字セットのスーパーセット
- 長さセマンティクスがバイト単位であるか、文字単位であるかに関係なく、マルチバイ ト・データがサポートされています。

その他の構成要件については、5-9ページの「特別なデータ型の処理」を参照してください。

1.5.4 バイナリ・データ型

次のバイナリ・データ型がサポートされます。

- RAW
- LONG RAW

1.5.5 日付およびタイムスタンプのデータ型

次の日時データ型がサポートされます。

- DATE
- TIMESTAMP (「サポートの制限」を参照)

1.5.5.1 サポートの制限

- Oracle GoldenGate では、負の日付はサポートされていません。
- 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のみ、INTERVAL DAY および INTERVAL YEAR がサポートされます。
- Oracle GoldenGate では、UTC オフセットで表された TIMESTAMP WITH TIME ZONE (TIMESTAMP '2011-01-01 8:00:00 -8:00') のキャプチャとレプリケーションがサポートさ れます。
- TZR (リージョン ID) で表された TIMESTAMP WITH TIME ZONE は、データ変更のレプリ ケーションについてはサポートされますが、初期ロード、SQLEXEC または列をデータベー スからフェッチする必要がある操作についてはサポートされません。これらの場合、列が 選択されると、データベースによってリージョン ID が時間オフセットに変換されます。 Replicat では、タイムスタンプは、時間オフセット値付きの日付と時間のデータとしてレ プリケートされます。

TZR と指定された TIMESTAMP WITH TIME ZONE を適切にサポートし、同様に TIMESTAMP WITH LOCAL TIMEZONE も適切に処理するには、5-9 ページの「特別なデータ型の処理」を参照してく ださい。

1.5.6 ラージ・オブジェクトのデータ型

次のラージ・オブジェクト型がサポートされます。

- CLOB
- NCLOB
- BT₁OB
- SECUREFILE と BASICFILE が両方ともサポートされています。

1.5.6.1 一般的なサポートの制限 - 統合およびクラシック・キャプチャ・ モード

ラージ・オブジェクトのサイズが 4K を超える場合、Oracle GoldenGate によってデータが Oracle GoldenGate の証跡内にセグメント単位で保存されます。最初の 4K はベース・セグメン トに保存され、残りは 2K の一連のセグメントに保存されます。Oracle Golden Gate では、この サイズのラージ・オブジェクトのフィルタリング、列マップまたは操作はサポートされていま せん。Oracle GoldenGate の全機能は、4K 以下のオブジェクトに使用できます。

1.5.6.2 サポートの制限 - クラシック・キャプチャ・モード

- BASICFILE LOB は、記憶域に関係なく、REDO ログからキャプチャされますが、次の場合 にはデータベースからフェッチされます。
 - Extract で LOB が無効と判断される場合。
 - LOB データが REDO ログ内にない場合。これは、BASICFILE LOB が no logging オプ ションを指定して作成されたときに起こります。
 - LOB が CACHE 属性を使用して作成されている場合。
 - LOB が一部のみ更新されている場合。Oracle GoldenGate では、部分的な列データは サポートされていません。LOBがLOBリセット・レコードから開始しない場合、あ るいは新しい LOB の長さに則した最初のバイトで開始せず、最後のバイトで終了しな い場合に、Extract によって LOB データは不完全であるとみなされます。部分更新は、 OCILOBWrite(), OCILobAppend(), OCiLobCopy(), OCILobLoadFromFile(), OCILobTrim() の各 OCI コールと、dbms_lob パッケージのプロシージャを使用して行 われた更新によって生成されます。
 - ページ番号がない、END MARKER がない、あるいはキャプチャされたサイズと想定され たサイズが一致しないなどの LOB データの異常が Extract によって検出される場合。
- SECUREFILE LOB は、更新が完了し、LOB の変換 (列の圧縮、暗号化または重複除外)が行 われず、行外に格納される場合にのみ、REDO ログからキャプチャされます。SECUREFILE LOBは、次の場合にデータベースからフェッチされます。
 - LOB が行内に格納される場合。
 - LOB が圧縮または暗号化のいずれかにより変換される場合。
 - LOBが CACHE 属性を使用して作成されている場合。
 - Extract によって LOB インスタンスが無効であることが決定される場合。
 - REDO ログに LOB データがない場合。これは、deduplicate、no logging、 filesystem like logging のいずれかのオプションを使用して LOB を作成したときに 起こる可能性があります。
 - LOB \$\psi^s\$ OCILOBWrite(), OCILobAppend(), OCiLobCopy(), OCILobLoadFromFile(), OCILobTrim() を使用するか、dbms lob パッケージのプロシージャを介して更新され
 - ページ番号がない、END MARKER がない、あるいはキャプチャされたサイズと想定され たサイズが一致しないなど、Extract によって検出されるその他の異常。
- SECUREFILE LOB をある記憶域から別の記憶域に(たとえば、ENCRYPT から DECRYPTに)変更 すると、Oracle によって表全体が更新され、Extract によってそれらの更新がログからキャ プチャされます。そのため、Oracle によって表に関連付けられているすべてのデータ・ブ ロックが更新されているように見えます。ALTER TABLE コマンドによって DEFAULT 値が NULL 値の列に設定される場合も、これが発生する可能性があります。

その他の設定要件については、5-9ページの「特別なデータ型の処理」を参照してください。

1.5.7 XML データ型

次の XML 型がサポートされます。

- 統合キャプチャ・モードの Oracle GoldenGate では、XML CLOB、XML オブジェクト・リレー ショナルおよび XML バイナリとして格納される XMLType 列および XMLType 表がサポートされます。
- クラシック・キャプチャ・モードの Oracle GoldenGate では、XML CLOB および XML バイナリ として格納される XMLType 列および XMLType 表がサポートされます。

1.5.7.1 サポートの制限 - 統合およびクラシック・キャプチャ・モード

- Oracle GoldenGate では、XMLType データを LOB として処理します。
- XML が含まれているソースおよびターゲットのオブジェクトは同一にする必要がありま す。フィルタリングと操作はサポートされていません。TABLE 文または MAP 文で COLMAP 句 を使用して、オブジェクトの XML 表現を文字列にマップできます。
- 次のものはサポートされません。
 - 階層が有効な表(これは、Oracle XML データベース・リポジトリで管理されます。)
 - 表への移入を行う CTAS (CREATE TABLE AS SELECT) 文で作成された XMLType 表。DDL サポートが有効な場合、Oracle GoldenGate でCTAS 文をレプリケートし、基となる ターゲット表からのデータの選択を可能にします。元の挿入はレプリケートされませ ん。XMLType 表の場合、行オブジェクト ID がソースとターゲットの間で一致する必 要がありますが、これは、Replicat で論理 SQL 文を使用すると保持されません。空の CTAS 文 (新規表にデータを挿入しない)で作成された XMLType 表では適切に保持さ れます。
 - 主キーベースのオブジェクト識別子 (OID) を持つ XMLType 表
 - 1つの XML 列を持つ非 XMLType 表
 - XML バイナリおよび XML オブジェクト・リレーショナルに対する SQL*Loader 直接パス挿入
- XMLSchema ベースの XMLType 表と列はサポートされますが、XMLSchema に対する変更は レプリケートされず、dbms xml パッケージを使用してソースとターゲットの両方のデータ ベースに登録する必要があります。
- サポートされている表が、スカラー列で構成された1つ以上の一意キー制約を持つか、す べてのスカラー列の組合せで一意性が保証される必要があります。Extract または Replicat では、行の識別用の XML 属性で構成された一意キー制約または主キー制約は使用できま せん。

1.5.7.2 サポートの制限 - 統合キャプチャ・モード

- Oracle データベースはリリース 11.2.0.3 以降であり、Extract は統合キャプチャ・モードで 構成されている必要があります。
- XMLType を更新する SET 値全体の長さは、新しいコンテンツと他の演算子、XQuery バイン ド値を含めて最大32Kです。

1.5.7.3 サポートの制限 - クラシック・キャプチャ・モード

- XML バイナリの場合、REDO ログに十分な情報が含まれていないため、Oracle GoldenGate によってソース・データベースから追加の行データがフェッチされます。フェッチされた データは元のトランザクションの一部ではないため、不整合が発生する可能性があります。
- XMLType 列が含まれている表には、主キー、一意制約がある列または一意の索引のいずれ かを設定する必要があります。

XML のレプリケートの詳細は、5-9ページの「特別なデータ型の処理」を参照してください。

1.5.8 ユーザー定義または抽象型

ソースおよびターゲットのオブジェクトの構造が同一である場合、Oracle GoldenGate では ユーザー定義型 (UDT) または抽象データ型 (ADT) がサポートされます。スキーマ名は別々にす ることができます。

1.5.8.1 一般的なサポートの制限 - 統合およびクラシック・キャプチャ・ チード

- UDT をフェッチする必要があるため、UDT が含まれている表には、主キー、一意な制約が ある列または一意な索引を設定する必要があります。
- Oracle GoldenGate では、CLOB、CFILE、BFILE または INTERVAL YM、INTERVAL DS および OPAQUE (XMLType はサポートされているため、除く) のスカラー型が組み込まれた UDT は サポートされません。
- キーに UDT が含まれている、または UDT が列のみであるオブジェクトまたはリレーショ ナル表はサポートされていません。
- RMTTASK パラメータでは、ユーザー定義型 (UDT) はサポートされていません。
- バイナリまたは印刷できない文字が含まれている CHAR 属性および VARCHAR 属性は、サポー トされていません。
- オブジェクトの列または行内に値が含まれている UDT は、TABLE 文または MAP 文のフィル タ基準で使用したり、Oracle GoldenGate の列変換関数の入力または出力、SQLEXEC または 他の組込みデータ操作ツールとして使用することはできません。Oracle の同様のソースお よびターゲットにのみ、サポートが提供されます。
- Oracle GoldenGate では、REF 型はサポートされていません。

その他の設定要件については、第5章の「特別なデータ型の処理」を参照してください。

1.5.8.2 コレクション型に対する制限 - 統合およびクラシック・キャブ チャ・モード

- ネストされた表のデータを更新する際には、ネストされた表が含まれている行を同時に更 新する必要があります。
- VARRAYS とネストされた表をフェッチすると、そのたびに変更だけではなく、列の内容全 体がフェッチされます。

1.5.8.3 オブジェクト表に対する制限 - 統合およびクラシック・キャプ チャ・モード

- Oracle GoldenGate では、一方向のアクティブ / アクティブ構成でオブジェクト表がサポー トされます。オブジェクト表は REDO ログからキャプチャされますが、LOB、コレクショ ン型などの標準のリレーショナル表の場合はデータベースからフェッチされる特定のデー タ型もオブジェクト表の場合はフェッチされます。同様に、標準の表の場合にコレクショ ン型に適用される現在の制限が、オブジェクト表の場合のそれらの型にも適用されます。
- Oracle のオブジェクト表は、サポートされているターゲット・データベースの Oracle 以外 のオブジェクト表にマップできます。
- 主キーは、オブジェクト表のルートレベルのオブジェクト属性で定義する必要があり、 リーフレベルの属性を含めることはできません。キーが定義されていない場合、Oracle GoldenGate では有効なすべての列が擬似キーとして使用されます。
- Oracle GoldenGate では、オブジェクト表の DDL 操作のレプリケーションはサポートされ ていません。この制限には、オブジェクト表の ALTER と関連付けられているデータベー ス・オブジェクトのバージョニングが含まれます。
- オブジェクト表またはオブジェクト表が含まれているリレーショナル表では、シノニムは サポートされていません。

1.5.8.4 Spatial 型に対する制限 - 統合およびクラシック・キャプチャ・

Oracle GoldenGate では、SDO GEOMETRY、SDO TOPO GEOMETRY および SDO GEORASTER (ラスター 表)がサポートされています。

その他の構成情報は、5-9ページの「特別なデータ型の処理」を参照してください。

1.5.9 サポートされていない Oracle データ型

Oracle GoldenGate では、次のデータ型はサポートされません。

- スカラー、LOB、VARRAYS、ネストされた表やREF を伴う抽象データ型 (ADT)
- ΔΝΥΠΑΤΑ
- ANYDATASET
- ANYTYPE
- BFILE
- ORDDICOM
- TIMEZONE_ABBR
- URITYPE
- UROWID

その他の除外については、1-5ページの「各キャプチャ・モードでサポートされる Oracle デー タ型とオブジェクトのサマリー」の「*サポートの制限*」を参照してください。

1.6 DML での Oracle オブジェクトと操作のサポートの詳細

この項では、DML 操作のキャプチャとレプリケーションについて Oracle GoldenGate でサポー トされる Oracle オブジェクトと操作について概説します。

1.6.1 表、ビューおよびマテリアライズド・ビュー

Oracle GoldenGate では、標準の表、索引構成表、クラスタリング表およびマテリアライズド・ ビューに対して行われる次の DML 操作がサポートされます。

- INSERT
- UPDATE
- DELETE
- 関連するトランザクション制御操作

1.6.1.1 標準の表のサポートの制限

次の制限は、統合キャプチャ・モードとクラシック・キャプチャ・モードに適用されます。

Oracle GoldenGate では、長さが 2MB までの行数が含まれている表がサポートされていま す。各文字 LOB/LONG 列の制限は最大 4KB で、各バイナリ LOB 列の制限は最大 8KB で す。この行サイズ制限は、主に行識別子として使用されている列の更新操作に影響を与え ます。この識別子は、主キーや一意キー、Oracle GoldenGate パラメータ・ファイル内で 定義されているキーにすることも、キーが定義されていない場合はすべての列にすること もできます。行識別子を更新した場合、2 MB の長さには後イメージだけでなく、更新のた めにターゲットで正しいキーを捜すために必要となる完全な前イメージも含める必要があ ります。

- 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。
- Oracle GoldenGate では、データベースによってサポートされている最大列サイズがサポー トされています。
- Oracle GoldenGate では、列に次のデータ型のいずれかが含まれていないかぎり、列が1つ だけ含まれている表がサポートされています。
 - LOB
 - LONG
 - ネストされた表
 - ユーザー定義データ型
 - VARRAY
 - XML
- Oracle GoldenGate では、未使用の列を含む表がサポートされていますが、デフォルトでは サポートが無効になっており、Extract はそれらで異常終了します。5-12 ページの「他の データベース・プロパティの処理」を参照してください。
- Oracle GoldenGate では、次のパーティション化属性を持つ表がサポートされます。
 - レンジ・パーティション化
 - ハッシュ・パーティション化
 - 時間隔パーティション化
 - システム・パーティション化
 - コンポジット・パーティション化
 - 仮想列ベースのパーティション化
 - 参照パーティション化
 - リスト・パーティション化

5-12 ページの「他のデータベース・プロパティの処理」を参照してください。

- Oracle GoldenGate では、仮想列を含む表がサポートされますが、これらの列に対する変更 データのキャプチャも適用も行われません。データベースでは仮想列がトランザクショ ン・ログに書き込まれず、Oracle データベースでは仮想列に対する DML が許可されませ ん。同じ理由から、初期ロール・データは仮想列に適用できません。仮想列から非仮想列 へのデータのマップは行えます。5-12ページの「他のデータベース・プロパティの処理」 を参照してください。
- Oracle GoldenGate では、一意キーまたは一意の索引の一部である仮想列は無視されます。 仮想列が表の唯一の一意の識別子の場合、Oracle GoldenGate では、残りのすべての列を 使用して行を識別するため、残りの列で一意性が保証されない場合、間違ったターゲット 行が削除または更新される可能性があります。
- Oracle GoldenGate では、Oracle Exadata との間のレプリケーションがサポートされます。 5-13 ページの「Oracle GoldenGate と Oracle Exadata を組み合せた使用」を参照してくだ さい。
- Oracle GoldenGate では、Transparent Data Encryption (TDE) がサポートされます。統合 キャプチャの場合、ソース・データベースは、11.0.0.0 以上の互換性設定の Oracle バー ジョン 11.1.0 である必要があります。列レベルの暗号化は、Oracle 10.2.0.5、11.1 および 11.2 のすべてのバージョンでサポートされます。表領域レベルの暗号化は、Oracle 10.2.0.5 および 11.2.0.1 のすべてのバージョンでサポートされます。TDE は、統合キャプチャ・ モードでは設定要件なしでサポートされます。クラシック・キャプチャ・モードの TDE に は設定が必要です。6-1 ページの「クラシック・キャプチャ・モードでの Oracle TDE デー タの構成」を参照してください。

- Oracle GoldenGate では、DDL レプリケーション・サポートの一環として、あるいは DDL サポートとは無関係のスタンドアロン機能として TRUNCATE 文がサポートされます。5-12 ページの「他のデータベース・プロパティの処理」を参照してください。
- Oracle GoldenGate では、直接ロードの INSERT のキャプチャがサポートされます。サプリ メンタル・ロギングが有効であることと、データベースがアーカイブ・ログ・モードであ ることが必要です。次の直接ロード方法がサポートされています。
 - /*+ APPEND */ヒント
 - /*+ PARALLEL */ ヒント (クラシック・キャプチャ・モードの RAC 以外の場合のみ)
 - DIRECT=TRUE を設定した SQLLDR
- Oracle GoldenGate では、統合キャプチャ・モードで基本表、拡張表および EHCC 圧縮表 が完全にサポートされます。クラシック・キャプチャ・モードでは、Oracle GoldenGate で Oracle Exadata 上の通常の表および EHCC 圧縮表への配信は行えますが、これらの表か らのキャプチャは行えません。5-12ページの「他のデータベース・プロパ ティの処理」を参照してください。
- Oracle GoldenGate では、統合キャプチャ・モードでの XA および PDML 分散トランザク ションがサポートされます。

1.6.1.2 索引構成表のサポートの制限

次の制限がクラシック・キャプチャ・モードに適用されます。

- マッピング表を使用するIOTは、クラシック・キャプチャ・モードでサポートされません。 マッピング表を使用する IOT に対する DDL は、適切にレプリケートされますが、後続の DML は失敗します。この制限は、統合キャプチャ・モードとクラシック・キャプチャ・ モードに適用されます。
- キー圧縮が有効 (key compression 句の COMPRESS キーワードで指定) な IOT は、クラシッ ク・キャプチャ・モードではサポートされませんが、統合キャプチャ・モードではサポー トされます。

1.6.1.3 ビューのサポートの制限

次の制限は、統合キャプチャ・モードとクラシック・キャプチャ・モードに適用されます。

- Extract が初期ロード・モードの場合、Oracle GoldenGate でビューからのキャプチャがサ ポートされます (REDO ログではなく、ソース・ビューからの直接のキャプチャ)。
- Oracle GoldenGate では、変更データはビューからキャプチャされませんが、ビューの基に なる表からのキャプチャはサポートされます。
- Oracle GoldenGate では、ビューが本質的に更新可能であるかぎり、ビューにレプリケート できます。ソース表とターゲット・ビューの構造は同一である必要があります。

ビューの構成要件については、5-12ページの「他のデータベース・プロパティの処理」を参照 してください。

1.6.1.4 マテリアライズド・ビューのサポートの制限

次の制限は、統合キャプチャ・モードとクラシック・キャプチャ・モードに適用されます。

- WITH ROWID を指定して作成されたマテリアライズド・ビューはサポートされていません。
- WITH ROWID を指定してマテリアライズド・ビューのログを作成できます。
- ソース表に主キーを設定しておく必要があります。
- マテリアライズド・ビューの切捨てはサポートされていません。DELETE FROM 文を使用で きます。
- Oracle GoldenGate のいくつかの初期ロード方法では、マテリアライズド・ビューの LOB が サポートされていません。

- Replicat の場合、マテリアライズド・ビューを更新可能にする必要があります。
- マテリアライズド・ビューの全体リフレッシュの DML(DDL ではない) はサポートされま す。この機能の DDL サポートが必要な場合は、Oracle GoldenGate のサポート事例を開い てください。

1.6.1.5 クラスタリング表のサポートの制限

索引およびハッシュ・クラスタは、統合とクラシックの両方のキャプチャ・モードでサポート されますが、クラシック・キャプチャ・モードでは次の制限が適用されます。

- 暗号化および圧縮されたクラスタリング表は、クラシック・キャプチャではサポートされ ていません。
- クラシック・キャプチャ・モードの Extract では、クラスタ・サイズが変わらない場合、索 引クラスタリング表への DML 変更がキャプチャされます。クラスタ・サイズの増減につ ながる DDL によって、Extract でその表に対する後続の DML が不適切にキャプチャされ る可能性があります。

1.6.2 順序

- Oracle GoldenGate では、一方向のアクティブ / パッシブ高可用性構成での順序値のレプリ ケーションがサポートされます。
- Oracle GoldenGate によって、ターゲットの順序値が常にソースの順序値よりも大きくなり ます(または、キャッシュが0の場合、等しくなります)。

1.6.2.1 順序のサポートの制限

次の制限は、統合キャプチャ・モードとクラシック・キャプチャ・モードに適用されます。

- Oracle GoldenGate では、アクティブ / アクティブ双方向構成での順序値のレプリケーショ ンはサポートされません。
- ソースおよびターゲットの順序のキャッシュ・サイズと増分間隔は同一にする必要があり ます。キャッシュは 0 (NOCACHE) を含む任意のサイズにできます。
- 順序はサイクルまたは非サイクルに設定できますが、ソースおよびターゲットのデータ ベースを同様に設定する必要があります。

構成要件については、5-12ページの「他のデータベース・プロパティの処理」を参照してくだ さい。

1.6.3 Oracle DML でサポートされていないオブジェクトおよび操作

次のものは、クラシック・キャプチャ・モードでも統合キャプチャ・モードでもサポートされ ません。

- REF
- アクティブ / アクティブ双方向構成での順序値
- データベース・リプレイ
- EXTERNAL として作成された表

次のものは、クラシック・キャプチャ・モードでのみサポートされません。

- Exadata Hybrid Columnar 圧縮
- OLTP 表圧縮された表からのキャプチャ
- COMPRESS を使用して作成または変更される表領域および表からのキャプチャ
- 暗号化および圧縮されたクラスタリング表からのキャプチャ
- シノニム

分散トランザクション。Oracle バージョン 11.1.0.6 以降では、次のコマンドを使用して非分 散にする(データベースの再起動が必要)と、これらのトランザクションをキャプチャでき ます。

alter system set CLUSTERWIDE GLOBAL TRANSACTIONS=FALSE;

- XA および PDML 分散トランザクション
- WITH ROWIDを指定して作成されたマテリアライズド・ビュー
- マテリアライズド・ビューの切捨て

1.7 Oracle DDL のオブジェクトと操作のサポートの詳細

この項では、DDL 操作のキャプチャとレプリケーションについて Oracle GoldenGate でサポー トされる Oracle オブジェクトと操作タイプについて概説します。

1.7.1 Oracle DDL でサポートされているオブジェクトおよび操作

次の文は、統合キャプチャ・モードとクラシック・キャプチャ・モードに適用されます。

- Oracle GoldenGate のすべてのトポロジ構成は、Oracle DDL レプリケーションでサポート されています。
- Oracle DDL のアクティブ / アクティブ (双方向)レプリケーションは、同一のメタデータが 含まれている2つ(および2つだけ)のデータベース間でサポートされています。
- Oracle GoldenGate では、次のオブジェクトでサイズが最大 2MB までの DDL 操作がサポー トされています。
 - クラスタ
 - 関数
 - 索引
 - パッケージ
 - プロシージャ
 - 表
 - 表領域
 - ロール
 - 順序
 - シノニム
 - トリガー
 - 型
 - ビュー
 - マテリアライズド・ビュー
 - ユーザー

2MB のサイズ制限にはパッケージ、プロシージャおよび関数が含まれます。

注意: DDL サポートのサイズには、文のテキストに加え、オブジェクト名の 長さ、DDL タイプおよび内部での 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。

特定のサポート・ガイドライン、サポート制限および構成手順は、『Oracle GoldenGate Windows and UNIX Administrator's Guide』を参照してください。

1.7.2 サポートされていない Oracle DDL のオブジェクトおよび操作

次の文は、統合キャプチャ・モードとクラシック・キャプチャ・モードに適用されます。

1.7.2.1 Oracle 予約済スキーマ

次のスキーマ名は Oracle 予約済スキーマと見なされるため、Oracle GoldenGate DDL 構成から 除外する必要があります。Oracle GoldenGate では、これらのスキーマは無視されます。

ANONYMOUS

AURORA

\$JIS

\$UTILITY

\$AURORA

\$ORB

\$UNAUTHENTICATED

CTXSYS

DBSNMP

DMSYS

DSSYS

EXFSYS

MDSYS

ODM MTR

OLAPSYS

ORDPLUGINS

ORDSYS

OSE\$HTTP\$ADMIN

OUTLN

PERFSTAT

PUBLIC

REPADMIN

SYS

SYSMAN

SYSTEM

TRACESVR

WKPROXY

WKSYS WMSYS

XDB

1.7.2.2 サポートされていないその他の DDL

Oracle GoldenGate では、次のものはサポートされていません。

- ALTER TABLE ...MOVE TABLESPACE
- ネストされた表に対する DDL
- ALTER DATABASE および ALTER SYSTEM (これらは DDL と見なされません)
- スタンバイ・データベースに対する DDL

また、クラシック・キャプチャ・モードでは、次のようなパスワードベースの列の暗号化を含 む DDL はサポートされません。

- CREATE TABLE t1 (a number, b varchar2(32) ENCRYPT IDENTIFIED BY my_password);
- ALTER TABLE t1 ADD COLUMN c varchar2(64) ENCRYPT IDENTIFIED BY my_password;

1.8 サポートされているオブジェクト名とサポートされていな いオブジェクト名

Oracle GoldenGate でのオブジェクト名と大文字小文字のサポートの詳細は、『Oracle GoldenGate Windows and UNIX Administrator's Guide』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、次の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、ライブラリなどの他のベンダーから必要と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。

- Oracle GoldenGate のダウンロード
- ORACLE_HOME および ORACLE_SID の設定
- UNIX における動的ビルド用のライブラリ・パスの設定
- クラスタ内での Oracle GoldenGate のインストールの準備
- Linux および UNIX での Oracle GoldenGate のインストール
- Windows での Oracle GoldenGate のインストール
- Oracle GoldenGate のクラスタへの統合
- Oracle 順序のサポートのインストール

2.2 Oracle GoldenGate のダウンロード

Oracle GoldenGate 構成に含まれる各システムに適した Oracle GoldenGate の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」を受諾します(永久ライセンスを所有している場合も 受諾)。

- 「輸出規制」を受諾します。
- **「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。
- 「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 確認します。

2.3 ORACLE_HOME および ORACLE_SID の設定

システム環境変数の ORACLE HOME および ORACLE SID が正しい Oracle インスタンスに設定さ れていることを確認します。Oracle GoldenGate プロセスでは、データベースへの接続時にそ れらを参照します。

2.3.1 UNIX および Linux システムでの Oracle 変数の指定

システム上に1つの Oracle インスタンスが存在する場合には、システム・レベルで ORACLE HOME と ORACLE SID を設定します。これらをそのように設定できない場合、インスタンスに接 続されるすべての Extract および Replicat のグループのパラメータ・ファイルで次の SETENV 文 を使用します。SETENV パラメータはシステムの設定より優先され、これらのパラメータによっ て、Oracle GoldenGate プロセスがデータベースに接続する際にセッション・レベルでの変数 の設定が可能になります。

SETENV (ORACLE_HOME = "path to Oracle home location")

SETENV (ORACLE SID = "SID")

システムに複数の Oracle インスタンスがあり、Extract および Replicat のプロセスがそれらに 接続する場合、各プロセス・グループのパラメータ・ファイルで SETENV 文を使用してそれを正 しいインスタンスに指し示す必要があります。たとえば、別の Oracle インスタンスからキャプ チャする、2つの Extract グループのパラメータ・ファイルを次に示します。

グループ 1:

EXTRACT ora9a

SETENV (ORACLE HOME = "/home/oracle/ora/product")

SETENV (ORACLE SID = "oraa")

USERID ggsa, PASSWORD ggsa

RMTHOST sysb

RMTTRAIL /home/ggs/dirdat/rt

TABLE hr.emp;

TABLE hr.salary;

グループ 2:

EXTRACT orab

SETENV (ORACLE HOME = "/home/oracle/ora/product")

SETENV (ORACLE SID = "orab")

USERID qqsb, PASSWORD qqsb

RMTHOST sysb

RMTTRAIL /home/ggs/dirdat/st

TABLE fin.sales;

TABLE fin.cust;

2.3.2 Winsows システムでの Oracle 変数の指定

システム上に1つの Oracle インスタンスが存在する場合、Oracle GoldenGate については ORACLE HOME と ORACLE SID のレジストリ設定で十分です。それらの設定がレジストリで不正確 であり、それらの設定を変更できない場合、次のようにオーバーライドを設定できます。

- 1. デスクトップまたは「スタート」メニュー (Windows のバージョンに応じて) で、「マイコ ンピュータ」を右クリックして「プロパティ」を選択します。
- 「プロパティ」で「詳細設定」タブをクリックします。
- 「環境変数」をクリックします。
- 4. 「システム環境変数」の下にある「新規」をクリックします。
- 「変数名」にORACLE HOME と入力します。
- 6. 「変数値」に Oracle バイナリのパスを入力します。
- 7. 「OK」をクリックします。
- **8. 「新規」**を再度クリックします。
- 9. 「変数名」に ORACLE_SID と入力します。
- 10.「変数値」にインスタンス名を入力します。
- **11.** 「**OK**」をクリックします。

システムに複数の Oracle インスタンスがあり、Extract および Replicat のプロセスがそれらに 接続する場合、次の手順を実行します。

- 1. 前の手順(システムに1つの Oracle インスタンス)を使用して、システム変数の ORACLE HOME および ORACLE SID を最初の Oracle インスタンスに設定します。
- そのインスタンスに接続する Oracle GoldenGate プロセスをすべて起動します。
- 次のOracle インスタンスに対して手順を繰り返しますが、まず既存のORACLE HOME 変数お よびORACLE SID変数を編集して新しい情報を指定します。
- **4.** そのインスタンスに接続する Oracle GoldenGate プロセスを起動します。
- 5. 残りの Oracle インスタンスに対して編集と起動の手順を繰り返します。

2.4 UNIX における動的ビルド用のライブラリ・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。

1. データベースのライブラリがシステムの共有ライブラリ環境変数に追加されていることを 確認します。通常、この手順はデータベースのインストール時に実行します。質問がある 場合には、データベース管理者に問い合せてください。

Oracle GoldenGate がデータベースと同じサーバー上で実行されている場合は、次のすべ てを 32 ビット、64 ビットまたは IA64 のいずれか 1 つのビット・タイプに統一する必要が あります。

- Oracle ライブラリのバージョン
- Oracle GoldenGate のバージョン
- データベースのバージョン

Oracle GoldenGate が SQL*Net 経由でリモートからデータベース・サーバーに接続する場 合は、次の条件を満たす必要があります。

- Replicat: Oracle クライアント・ライブラリと Oracle GoldenGate ビルドで Oracle の バージョン、ビット・タイプ (32 ビット、64 ビット、IA64) およびオペレーティング・ システムのバージョンを同一にしておく必要があります。
- Extract: Oracle クライアント・ライブラリと Oracle GoldenGate ビルドで Oracle のバー ジョン、ビット・タイプ (32 ビット、64 ビット、IA64) およびオペレーティング・シ ステムのバージョンを同一にしておく必要があります。さらに、両方のオペレーティ ング・システムは同じエンディアンであることが必要です。
- 2. UNIX システム上の Oracle GoldenGate インストール・ディレクトリの外部から 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・ディレクトリが /users/ogg の場合、次の例 の2番目のコマンドには、これらの環境変数が設定されている必要があります。

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

説明: shared libraries variable は表 2-1 に示す変数のいずれかです。

表 2-1 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	環境変数
IBM AIX	LIBPATH
HP-UX	SHLIB_PATH
Sun Solaris	LD_LIBRARY_PATH ¹
HP Tru64(OSF/1)	
LINUX	

³² ビットの Oracle データベースが存在する 64 ビット環境の場合、Oracle GoldenGate では 32 ビットの Oracle ライブラリが含まれるように LD_LIBRARY_PATH を設定する必要があります。

export LD LIBRARY PATH=/ggs/11.0:\$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 のプロセスを起動する前に 1dd 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。

2.5 クラスタ内での Oracle GoldenGate のインストールの準備

この項では、Oracle GoldenGate をクラスタ環境にインストールする場合のインストール要件 について説明します。Oracle GoldenGate は、Oracle で動作保証されているクラスタ管理ソ リューションと組み合せて使用できます。Oracle Clusterware ソリューションによって、 Oracle RAC データベースと組み合せても組み合せなくても使用できるという利点が得られ、 Oracle GoldenGate が実行される非データベース・サーバーを含めることができます。

2.5.1 Oracle ユーザーとしてのインストール

Oracle クラスタにインストールするには、Oracle オペレーティング・システム・ユーザーとし てインストールします。

2.5.2 サポートされる Oracle クラスタ記憶域

最低でも、なんらかの 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。クラスタのどのノードからも独立していて、すべてのノードから使用可能なクラ スタ対応の共有記憶域を選択します。次のものを使用できます。

- Oracle クラスタ・ファイルシステム (OCFS)。Oracle GoldenGate をローカル・ディレクト リではなく、OCFS ボリュームにマウントされたディレクトリにインストールします。構 成の詳細は、OCFS2のドキュメントを参照してください。
- Oracle 自動ストレージ管理システム (AFS)。クラシック・キャプチャを使用する場合、6-7 ページの「クラシック・キャプチャ・モードの場合の Oracle ASM インスタンスからの キャプチャ」の設定要件を参照してください。
- Oracle Database ファイル・システム (DBFS)。Oracle GoldenGate を DBFS にインストール することはできませんが、一度に1つのサーバーにのみマウントされる DBFS クラスタに サブディレクトリ (インストール時に CREATE SUBDIRS で作成)を格納できます。高可用性 の要件については、D-1 ページの「Oracle GoldenGate とのアクティブ / アクティブ伝播の ための DBFS の準備」を参照してください。

2.5.3 Oracle GoldenGate バイナリとファイルをインストールするクラスタ 内の場所の決定

ベスト・プラクティスは、Oracle GoldenGate 全部を共有記憶域にインストールすることです。 これによって、パラメータ・ファイルを変更せずにどのノードからも Oracle GoldenGate プロ セスを起動できます。アクティブ・ノードに障害が発生した場合、インストール・ディレクト リに保持されている処理のチェックポイントを使用して、別のノードでプロセスをただちに起 動できます。

共有記憶域ではなく、各ノードに Oracle GoldenGate のバイナリとファイルをインストールする場合、次の点を満たす必要があります。

- Oracle GoldenGate インストールの場所が、すべてのノードで同じパスである必要があります。
- 最低でも次のディレクトリを共有記憶域にインストールし、Oracle GoldenGate のリカバリ 要件をサポートします。UNIX または Linux では、各ノードのインストール・ディレクト リからのシンボリック・リンクを作成できます。
 - br
 - dirchk
 - dirdat
 - dirtmp

これらのディレクトリは、インストール時に CREATE SUBDIRS を発行して作成されるディレクトリの一部です。

■ dirprmディレクトリのパラメータ・ファイルは、共有ドライブに配置されていない場合、 すべてのノードで同一である必要があります。ノードごとに異なる環境設定を解決するために、ローカル Manager プロセスから継承するか、ノード固有の Oracle Golden Gate マクロ・ファイルを参照するよう環境設定を設定できます。このシナリオを実現するのは難しいため、パラメータ・ファイルを共有ドライブに格納することで内在する問題を回避できます。

Oracle GoldenGate のインストール後、2-9 ページの「Oracle GoldenGate のクラスタへの統合」も参照してください。

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

次の手順に従って、Oracle GoldenGate for Oracle を Linux システム、UNIX システムまたはクラスタ内の適切な場所にインストールします。詳細は、2.5 項「クラスタ内での Oracle GoldenGate のインストールの準備」を参照してください。

- **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、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

2.7 Windows での Oracle GoldenGate のインストール

次の手順に従って、Oracle GoldenGate for Oracle を Windows システムまたはクラスタ内の適切な場所にインストールします。詳細は、2.5 項「クラスタ内での Oracle GoldenGate のインストールの準備」を参照してください。

2.7.1 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 のインストール場所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る同じクラスタ・グループ内のリソースであることが必要です。
- 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.** 「Oracle GoldenGate ファイルのインストール」に従って、Oracle GoldenGate をインストールします。

2.7.2 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. それらのファイルをバイナリ・モードで、Oracle GoldenGate をインストールするドライ バ上のフォルダに移動します。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。例を次に示します。
 - C:\"Oracle GoldenGate" は有効ではありません。
 - 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、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. 次のファイルを Oracle GoldenGate インストール・ディレクトリから SYSTEM32 ディレクトリにコピーします。

category.dll ggsmsg.dll

2.7.3 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。

- デフォルトの GGSMGR 以外の名前を Manager に使用します。
- 複数の Manager プロセスが Windows サービスとしてこのシステムで実行されています。 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 サービスの名前を確認してください。

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 ルートにあるため、このコマンドの./の部分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。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。

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

デフォルトでは、Manager は、サービスとしてインストールされず、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」の順にクリックし、「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 プログラムを実行します。

install option [...]

説明: option は次のいずれかです。

表 2-2 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 のイベント・マネージャに追加します。			
ADDSERVICE	GLOBALS ファイルの MGRSERVNAME パラメータで指定された名前がある場合はその名前で、そうでない場合はデフォルトの GGSMGR で、Manager をサービスとして追加します。ADDSERVICE により、大部分の Windows アプリケーションの標準であるローカル・システム・アカウントとして実行されるようサービスは構成されますが、これはユーザー・ログインやパスワードの変更の影響を受けずにサービスを実行できるためです。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)。インストール後にサービスを開始するには、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 開始するように設定します。MANUALSTART を使用しないかぎり、こ れがデフォルトです。			
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に設定します。デフォルトは AUTOSTART です。			

表 2-2 INSTALL ユーティリティのオプション(続き)	表 2-2	INSTALL	ユーティ	リティ	のオプシ	ノョン	(続き)
---------------------------------	-------	---------	------	-----	------	-----	------

オプション	説明
USER name	Manager の実行に使用するドメイン・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。
	デフォルトでは、Manager は、ローカル・システム・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 に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「コントロール パネル」の「サービス」アプレット から「プロパティ」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 ユーティリティの実行を許可します。

INSTALL ユーティリティよって、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 がサービスとしてインストールされていない場合、Manager を GGSCI コマンド・プロンプトから起動する際、Manager の権限の強化を 確認する UAC プロンプトが Oracle GoldenGate ユーザーに表示されます。 他の Oracle GoldenGate プログラムの実行でもプロンプトが表示されます。

2.8 Oracle GoldenGate のクラスタへの統合

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

2.8.1 クラスタでの標準の要件

- 1. Oracle GoldenGate Manager プロセス (Manager のみ) を、他のアプリケーションと同様に クラスタ管理対象リソースとして登録します。Manager は、他のすべてのプロセスを管理 する親プロセスであるため、クラスタ管理ソフトウェアが起動および停止する唯一の Oracle GoldenGate プロセスです。
- 2. クラスタで仮想 IP アドレス (Oracle Clusterware など) を使用する場合、Manager 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 アドレスである必要があり、DHCP を介して決定されません。Extract データ・ポンプのパラメータ・ファイルで、リモート Manager の VIP を RMTHOST パラ メータの入力値として指定します。Manager にアクセスする他の Oracle GoldenGate 製品 も VIP を使用する必要があります。
- 3. (バージョン 10.2 より前の Oracle データベース) クラスタ内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。クロックは Extract が実行されるシステム のクロックと同期している必要があります。Oracle GoldenGate では、重大な決定を下すた めにローカル・システムの時間とコミットのタイムスタンプが比較されます。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。これは、統合キャプ チャの場合要件ではありません。
- 4. クラスタ内のすべてのノードで COMPATIBLE パラメータ設定が同じであることを確認しま す。

- 5. Manager を構成する際、Manager でレプリケーション・プロセスを自動的に起動するよう に AUTOSTART および AUTORESTART パラメータを追加します (4-9 ページの「Oracle GoldenGate インスタンスの作成」を参照してください)。必要に応じて、Oracle GoldenGate ユーザー・インタフェース内から Extract、Replicat および他の Oracle GoldenGate プロセスを制御できます。
- 6. 1つのノードのみで共有ドライブをマウントします。これによって、他のノードでプロセ スが起動されなくなります。すべてのノードで同じマウント・ポイントを使用します。
- 7. このドキュメントに記載のとおり、Oracle GoldenGate を構成します。

2.8.2 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・アドミニストレータで、Oracle GoldenGate の接続先のデータベース・インスタ ンスを含むグループに Manager プロセスを追加します。
- Oracle GoldenGate が稼働するすべてのノードが、リソースの使用可能な所有者として選択 されていることを確認します。
- Manager Windows サービスに次の依存関係 (「サービス」コントロール・パネルから構成 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

2.9 Oracle 順序のサポートのインストール

Oracle 順序をサポートするには、いくつかのデータベース・プロシージャをインストールする 必要があります。これらのプロシージャは、Oracle GoldenGate プロセスを初めて起動した直 後(通常、初期データ同期手順を実行するとき)に発行する Oracle GoldenGate FLUSH SEQUENCE コマンドをサポートします。

Oracle 順序オブジェクトのインストール手順

ソース・システムおよびターゲット・システムで、手順を実行します。

- 1. SQL*Plus で、SYSDBA としてソースおよびターゲットの Oracle システムに接続します。
- 2. Oracle GoldenGate DDL レプリケーション機能をサポートするデータベース・ユーザーが すでに割り当てられている場合は、この手順をスキップできます。そうでない場合は、両 方のシステムの SQL*Plus で、DDL ユーザーでもあるデータベース・ユーザーを作成しま す。

CREATE USER DDLuser IDENTIFIED BY password;

GRANT CONNECT, RESOURCE, DBA TO DDLuser;

- 3. 各システムの Oracle GoldenGate インストール・ディレクトリから GGSCI を実行します。
- **4.** GGSCIで、次のコマンドを各システムで発行します。

EDIT PARAMS ./GLOBALS

5. 各 GLOBALS ファイルに GGSCHEMA パラメータを入力し、この手順のこれまでに作成した DDL ユーザーのスキーマを指定します。

GGSCHEMA schema

- 6. ファイルを保存し、閉じます。
- 7. 両方のシステムの SQL*Plus で、sequence.sql スクリプトを Oracle GoldenGate インストール・ディレクトリのルートから実行します。このスクリプトによって、Oracle GoldenGate プロセスで使用されるいくつかのプロシージャが作成されます。(これらを手動で実行しないでください。)最初の手順で作成したユーザー情報の入力が求められます。@sequence.sql
- 8. ソース・システムの SQL*Plus で、DBLOGIN コマンドの発行に使用されるデータベース・ユーザーに updateSequence プロシージャに対する EXECUTE 権限を付与します。このユーザーについて記録しておいてください。プロシージャをコールする FLUSH SEQUENCE コマンドを発行する前に、DBLOGIN を使用してデータベースにログインします。

GRANT EXECUTE on DDLuser.updateSequence TO DBLOGINuser;

- 9. ターゲット・システムの SQL*Plus で、Replicat データベース・ユーザーに replicateSequence プロシージャに対する EXECUTE 権限を付与します。
 GRANT EXECUTE on DDLuser.replicateSequence TO Replicatuser;
- **10.** ソース・システムの SQL*Plus で、次の文を発行します。
 ALTER TABLE sys.seg\$ ADD SUPPLEMENTAL LOG DATA (PRIMARY KEY) COLUMNS;

Oracle データベース用 Oracle GoldenGate の DDL サポートのインストール

この章では、DDL レプリケーションをサポートするオブジェクトのインストール手順について 説明します。DDL をキャプチャしてレプリケートするように Oracle GoldenGate を構成する場 合、第7章を参照してください。

注意:順序の DDL サポート (CREATE, ALTER、DROP、RENAME) は順序値自体の レプリケートと互換性がありますが、順序値をレプリケートするために必須 ではありません。順序値のレプリケートのみを行う場合、Oracle GoldenGate の DDL サポート環境をインストールする必要はありません。Extract 構成で SEQUENCE パラメータを使用するのみです。

3.1 DDL オブジェクトの概要

Oracle GoldenGate の DDL 環境をインストールするには、表 3-1 に示すデータベース・オブ ジェクトをインストールします。

表 3-1 DDL 同期オブジェクト

オブジェクト	目的	デフォルト名
DDL マーカー表	DDL 情報を保存します。この表では挿入のみを受信します。	GGS_MARKER
マーカー表での順序	マーカー表の列に使用します。	GGS_DDL_SEQ
DDL 履歴表	オブジェクト・メタデータ履歴を 保存します。この表では挿入、更 新、削除を受信します。	GGS_DDL_HIST
オブジェクト ID 履歴表	構成されたオブジェクトのオブ ジェクト ID が含まれます。	GGS_DDL_HIST_ALT
DDL トリガー	DDL 操作を実行します。操作に関する情報をマーカー表および履歴表に書き込みます。トリガーとともにいくつかのパッケージがインストールされます。	GGS_DDL_TRIGGER_BEFORE
DDL スキーマ	DDL 同期オブジェクトが含まれます。	なし。インストール時および GLOBALS ファイルで指定する必要 があります。
ユーザー・ロール	DDL 操作の実行に必要なロールを確立します。	GGS_GGSUSER_ROLE

表 3-1 DDI	- 同期オブジェ	クト	(続き)
-----------	----------	----	------

オブジェクト	目的	デフォルト名
内部設定表	内部でのみ使用できるデータベー ス表。	GGS_SETUP
ddl_pin	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 トリガーをピン留めし ます。	ddl_pin
ddl_cleartrace.sql	DDL トレース・ファイルを削除します。	ddl_cleartrace.sql
ddl_status.sql	Oracle GoldenGate の DDL オブ ジェクトがインストールされてい ることを確認します。	ddl_status.sql
marker_status.sql	マーカー表がインストールされて いることを確認します。	marker_status.sql
ddl_tracelevel.sql	DDL トレースのレベルを設定しま す。	ddl_tracelevel.sql

3.2 DDL オブジェクトのインストール

次の手順に従って、Oracle GoldenGate DDL キャプチャをサポートするデータベース・オブ ジェクトをインストールします。

- 1. Oracle GoldenGate DDL オブジェクトを含めるスキーマを選択します。このスキーマは大 文字と小文字を区別しません。
- **2.** Oracle GoldenGate スキーマに次の権限を付与します。

GRANT EXECUTE ON utl_file TO schema;

- 3. Oracle GoldenGate DDL に対するデフォルト表領域を作成します。この表領域は、DDL ス キーマ専用である必要があります。他のスキーマと共有できません。
- 4. この DDL 表領域に対して AUTOEXTEND を ON に設定し、表領域のサイズを GGS DDL HIST 表 および GGS MARKER 表の拡張に対応できるサイズに設定します。特に、GGS DDL HIST 表は DDL アクティビティ全体に比例して増加します。
- 5. (オプション) DDL 表領域が一杯になった場合にユーザー DDL アクティビティが失敗する ようにするには、params.sql スクリプトを編集して ddl fire error in trigger パラ メータを TRUE に設定します。表領域が一杯になると、Extract は DDL をキャプチャできな いため、DDL を停止することで表領域のサイズを拡張する時間ができ、DDL キャプチャ が失われることを防ぎます。ただし、表領域のサイズをこのように管理する場合、ビジネ スの混乱を防ぐためにビジネス・アプリケーションと Extract を絶えず監視する必要があり ます。ベスト・プラクティスとして、最初の段階で表領域を適切なサイズに設定し、表領 域が一杯にならないように AUTOEXTEND を ON に設定します。

警告: params.sql スクリプトは、元の状態を保持するため、バックアップを 作成してから編集してください。

6. GLOBALS ファイルを作成します (あるいは、既存のものを編集します)。

EDIT PARAMS ./GLOBALS

注意: EDIT PARAMS により、単純なテキスト・ファイルが作成されます。 EDIT PARAMS 後にファイルを保存すると、Oracle GoldenGate ディレクトリの ルートにファイル拡張子なしのGLOBALS(大文字)という名前で保存されま す。ファイルの名前や場所を変更しないでください。

7. GLOBALS ファイルで、次のパラメータを GLOBALS ファイルに追加することで、DDL スキー マ名を指定します。

GGSCHEMA schema name

- 8. (オプション)表 3-1 に示すその他のオブジェクトの名前を変更するには、残りのインス トールを進める前にここで変更する必要があります。そうしない場合、Oracle GoldenGate の DDL 処理を停止して DDL オブジェクトを再インストールする必要があります。データ ベース・オブジェクトのデフォルト名を受け入れることをお薦めします。表 3-1 に示す名 前(スキーマを除く)を変更するには、次のいずれか、または両方を実行します。
 - params.sql スクリプトにすべての名前の変更を記録します。このスクリプトを編集し て適切なパラメータを変更します。このスクリプトを実行しないでください。
 - 表 3-2 に示されている名前を GLOBALS ファイルにリストします。使用する正しいパラ メータは、表の「パラメータ」列に示されます。

表 3-2 DDL オブジェクト名を変更するための GLOBALS パラメータ

オブジェクト	パラメータ		
マーカー表 MARKERTABLE new_table_name ¹			
履歴表	DDLTABLE new_table_name		

これらの表の名前は修飾しないでください。これらの表のスキーマ名は、GGSCHEMA で指定されたスキー マ、または GLOBALS で GGSCHEMA を指定していない場合は現行ユーザーのスキーマです。

 (オプション)パーティションが数百個ある表からのキャプチャのパフォーマンスを改善す るには、次の例に示す params.sql スクリプトの SQL に no unnest ヒントを追加します。 ヒントは太字フォントで示します。

SELECT KEY.KEY NAME INDEX NAME, KEY.COLUMN NAME, KEY.DESCEND

(SELECT /*+ NO UNNEST */ C.CONSTRAINT NAME KEY NAME, C.COLUMN NAME COLUMN NAME, C.POSITION

POSITION, 'ASC' DESCEND FROM DBA CONS COLUMNS C WHERE C.OWNER = :B2 AND C.TABLE NAME = :B1 AND C.CONSTRAINT NAME IN (SELECT /*+ NO UNNEST */ CON1.NAME

SYS.USER\$ USER1, SYS.USER\$ USER2, SYS.CDEF\$ CDEF, SYS.CON\$ CON1, SYS.CON\$ CON2, SYS.OBJ\$ OBJ1, SYS.OBJ\$ OBJ2 WHERE USER1.NAME = :B2 AND OBJ1.NAME = :B1 AND CDEF.TYPE# = 3 AND BITAND(CDEF.DEFER, 36) = 4 AND (CDEF.TYPE# = 5 OR CDEF.ENABLED IS NOT NULL) AND CON2.OWNER# = USER2.USER#(+) AND CDEF.ROBJ# = OBJ2.OBJ#(+) AND CDEF.RCON# = CON2.CON#(+) AND OBJ1.OWNER# = USER1.USER# AND CDEF.CON# = CON1.CON# AND CDEF.OBJ# = OBJ1.OBJ#) UNION SELECT /*+ NO UNNEST */ I.INDEX NAME KEY NAME, C.COLUMN NAME COLUMN NAME, C.COLUMN

POSITION, C.DESCEND DESCEND FROM DBA INDEXES I, DBA IND COLUMNS C WHERE I.TABLE OWNER = :B2 AND I.TABLE NAME = :B1 AND I.UNIQUENESS = 'UNIQUE' AND I.OWNER = C.INDEX OWNER AND I.INDEX NAME = C.INDEX NAME AND :B1 = C.TABLE NAME AND :B2 = C.TABLE OWNER AND I.INDEX NAME IN (SELECT /*+ NO UNNEST */INDEX NAME FROM DBA INDEXES WHERE TABLE OWNER = :B2 AND TABLE NAME = :B1 AND UNIQUENESS = 'UNIQUE') AND I.INDEX NAME NOT IN (SELECT /*+ NO UNNEST */C.CONSTRAINT NAME FROM DBA CONS COLUMNS C WHERE C.OWNER = :B2 AND

C.TABLE_NAME = :B1 AND C.CONSTRAINT_NAME IN (SELECT /*+ NO_UNNEST */ C1.NAME FROM SYS.USER\$

U1, SYS.USER\$ U2, SYS.CDEF\$ D, SYS.CON\$ C1, SYS.CON\$ C2, SYS.OBJ\$ O1, SYS.OBJ\$ O2 WHERE U1.NAME = :B2 AND O1.NAME = :B1 AND D.TYPE# IN (2, 3) AND (D.DEFER IS NULL OR D.DEFER = 0 OR BITAND(D.DEFER, 36) = 4) AND (D.TYPE# = 5 OR D.ENABLED IS NOT NULL) AND C2.OWNER# = U2.USER#(+) AND D.ROBJ# = O2.OBJ#(+) AND D.RCON# = C2.CON#(+) AND O1.OWNER# = U1.USER# AND D.CON# = C1.CON# AND D.OBJ# = O1.OBJ#) AND EXISTS (SELECT /*+ NO_UNNEST */ 'X' FROM DBA_TAB COLUMNS

T WHERE T.OWNER = C.OWNER AND T.TABLE_NAME = C.TABLE_NAME AND T.COLUMN_NAME = C.COLUMN NAME))) KEY ORDER BY KEY.KEY NAME, KEY.POSITION

警告:記載された手順に従ってオブジェクト名を変更する必要がある場合以外、params.sql の他のパラメータを編集しないでください。

- 10. GLOBALS ファイルおよび params.sql ファイルを保存し、閉じます。
- 11. 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。
- **12.** SQL*Plus のセッション、ビジネス・アプリケーションのセッション、Oracle GoldenGate のプロセスのセッション、Oracle を使用するその他のソフトウェアのセッションを含め、すべての Oracle セッションを終了します。新しいセッションの開始を回避します。
- **13.** SQL*Plus を実行し、SYSDBA 権限を持つユーザーでログインします。この権限は、Oracle に必要な SYS スキーマへの DDL トリガーのインストールに必要です。他のすべての DDL オブジェクトはステップ 1 で作成したスキーマにインストールされます。
- **14.** marker_setup.sql スクリプトを実行します。プロンプトが表示されたら Oracle GoldenGate スキーマの名前を入力し、**[Enter]** を押してスクリプトを実行します。このスクリプトによって、Oracle GoldenGate DDL マーカー・システムのサポートがインストールされます。

@marker_setup.sql

15. ddl_setup.sql スクリプトを実行します。ステップ 1 で構成した DDL スキーマの名前を指定するよう求められます。

このスキーマの表領域が他のユーザーと共有されていると、ddl_setup.sql は失敗します。 ただし、デフォルト表領域の AUTOEXTEND が ON に設定(推奨設定)されていなければ、失 敗しません。

注意: データベースが Oracle 10g の場合、Oracle のごみ箱を無効にする必要があります。

@ddl setup.sql

16. role_setup.sql スクリプトを実行します。プロンプトが表示されたら、DDL スキーマ名を入力します。このスクリプトによって、DDL 同期に必要なロールが削除および作成され、Oracle GoldenGate DDL オブジェクトに対する DML 権限が付与されます。

@role_setup.sql

17. 作成されたロール (デフォルト名は GGS_GGSUSER_ROLE) を Oracle GoldenGate Extract ユーザーに付与します。プロセスに異なるユーザー名がある場合、複数の付与が必要になることがあります。

GRANT role TO user;

18. ddl_enable.sql スクリプトを実行して DDL トリガーを有効にします。 @ddl_enable.sql

オプションのパフォーマンス・ツールをインストールして使用する手順

DDL トリガーのパフォーマンスを改善するには、ddl pin スクリプトをデータベースの起動の 一部にします。次のように、Oracle GoldenGate の DDL ユーザー名を指定してスクリプトを起 動する必要があります。

@ddl_pin DDL_user

このスクリプトによって、トリガーで使用される PL/SQL パッケージがメモリーにピン留めさ れます。SQL*Plus からこのスクリプトを実行する場合、Oracle GoldenGate インストール・ ディレクトリから SYSDBA として接続します。このスクリプトは Oracle の dmbs shared pool システム・パッケージに依存するため、ddl pin を使用する前にそのパッケージをインストー ルしてください。

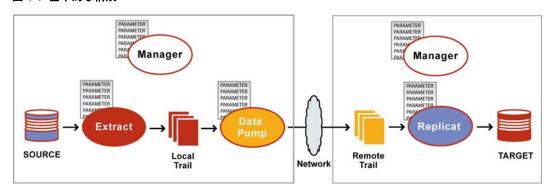
ソースおよびターゲットの Oracle データ ベースでの Oracle GoldenGate の構成

この章では、ソース Oracle データをキャプチャし、ターゲット Oracle データベースに適用す るための Oracle GoldenGate の構成について説明します。

4.1 この手順でできること

ここでは、トランザクション・データ変更を Oracle ソースから同一の Oracle ターゲットにレ プリケートするプロセスごとに Oracle GoldenGate パラメータ (構成)ファイルの基本的なセッ トを構成する方法について説明します。ビジネス要件によってはより複雑なトポロジが必要で すが、この手順がその他の構成手順の基礎となります。

図 4-1 基本的な構成



手順を実行することで、次のことが可能です。

- 基本的な構成ファイルが作成されます。
- 後で環境に適用する機能や要件について決定を下してパラメータを追加し、それらを拡張 します。
- コピーを使用して、一から作成するよりも短時間で追加パラメータ・ファイルを作成しま

4.2 パラメータ・ファイルの作成と編集

デプロイ・プロセス全体を通して Oracle GoldenGate パラメータ・ファイルを使用します。パ ラメータ・ファイルを作成および編集するには、Manager の起動後、GGSCI で EDIT PARAMS コマンドを使用します。

EDIT PARAMS name of parameter file

Oracle GoldenGate パラメータ・ファイルの使用方法の詳細は、『Oracle GoldenGate Windows and UNIX 管理者ガイド』を参照してください。

4.3 Oracle GoldenGate を構成する基本的な手順の概要

次の決定を行います。

- プロセスとファイルの名前の選択
- 使用するキャプチャ方法の決定
- Oracle GoldenGate のデータベース・ユーザーの割当て

次に示す、オブジェクトの基本的なセットを作成します。

両方のシステムで:

Oracle GoldenGate インスタンスの作成

ソース・システムで:

変更のキャプチャ用の Extract の構成

ターゲット・システムで:

■ 変更の配信用の Replicat の構成

この章では、次の点に関する推奨事項も示します。

- 統合キャプチャのチューニングの推奨事項
- 最良のパフォーマンスを得るための追加プロセス・グループの構成
- デプロイの次の手順
- トランザクション変更のレプリケートを開始するタイミング
- 構成のテスト

4.4 プロセスとファイルの名前の選択

構成手順を開始する前に、Oracle GoldenGate のプロセスとファイルの命名規則を作成すると 便利です。意味のある名前を選択すると、ディスプレイ、エラー・ログおよび外部監視プログ ラムで複数のプロセスやファイルを区別するのに役立ちます。また、環境の変更や拡張に応じ て後でプロセスやファイルを追加する際の命名にも対応します。

この項では、次の点について説明します。

- グループ名の選択
- ファイル名の選択

4.4.1 グループ名の選択

Oracle GoldenGate プロセスの名前は、パラメータ・ファイル内および Oracle GoldenGate 環 境のインスタンス化(第8章)時にプロセスを作成する際に指定します。少なくとも、次の処理 コンポーネントの名前を選択する必要があります。

- プライマリ Extract: プライマリ Extract プロセスでは、データベース・ソースからデータと DDL をキャプチャします。
- **データ・ポンプ**: データ・ポンプは、キャプチャされたデータをソース・システムのローカ ル証跡から読み取り、ネットワーク経由でターゲットに送信するセカンダリ Extract プロセ スです。データ・ポンプによって記憶域の柔軟性が増し、プライマリ Extract プロセスを

TCP/IP アクティビティから分離します。どのようにデータ・ポンプをキャプチャ構成に組 み入れるかについては、図4-1を参照してください。

Replicat: Replicat プロセスでは、リモート証跡を読み取り、トランザクション変更をター ゲット・データベースに適用します。

キャプチャ方法、パフォーマンス・チューニング、ロード・バランシングなどの決定に応じて、 複数の Replicat グループ、複数のプライマリ Extract グループとデータ・ポンプを作成する必 要がある場合があります。プロセスの命名規則については、『Oracle GoldenGate Windows and UNIX リファレンス・ガイド』の ADD EXTRACT コマンドおよび ADD REPLICAT コマンドに関す る項を参照してください。

4.4.2 ファイル名の選択

キャプチャされたデータを処理して、次の Oracle Golden Gate プロセス・ダウンストリームに よって処理されるための格納場所である、証跡と呼ばれる一連のファイルを作成する必要があ ります。基本的な構成は次のとおりです。

- ソース・システムのローカル証跡
- ターゲット・システムのリモート証跡

名前は、相対パス名または完全修飾パス名です。実際の証跡名は、./dirdat/tr のように 2 文 字のみで構成できます。Oracle GoldenGate により、新規ファイルが作成されるたび、この名前 に 6 桁の順序番号が付加され、./dirdat/aa000002 のようになります。

証跡ファイル、その格納方法および管理方法の詳細は、『Oracle GoldenGate Windows and UNIX *管理者ガイド*』を参照してください。

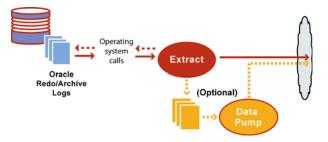
4.5 使用するキャプチャ方法の決定

Oracle ソース・データベースの場合、クラシック・キャプチャ・モードまたは統合キャプチャ モードのいずれかを使用できます。次に、これらのモードと各モードでサポートされるデータ ベースのバージョンについて説明します。

4.5.1 クラシック・キャプチャについて

クラシック・キャプチャ・モードでは、Oracle GoldenGate Extract プロセスは、ソース・シス テムの Oracle REDO またはアーカイブ・ログ・ファイルから、あるいはスタンバイ・システム の転送されたアーカイブ・ログからデータ変更をキャプチャします。

図 4-2 クラシック・キャプチャ



クラシック・キャプチャでは、ほとんどの Oracle データ型が完全にサポートされ、複雑なデー タ型は制限付きでサポートされます。クラシック・キャプチャは、最も一般的に使用されてい る広範な Oracle データ型と機能をサポートするオリジナルで高速かつ実績のある Oracle GoldenGate のキャプチャ方法です。クラシック・キャプチャは、Oracle GoldenGate でサポー トされるどのソース Oracle RDBMS に対しても使用できます。

次のもののサポートにクラシック・キャプチャを使用できます。

ソース・データベース互換性が 11.2.0.0.0 未満に設定された ADT、VARRAY、NOLOGGING LOB_o

- ソース・データベース互換性が 11.0.0.0.0 未満に設定された Transparent Data Encryption サポート。
- ソース・データベース互換性が 11.2.0.0.0 未満に設定された SECUREFILE LOB サポート。
- ソース・データベース互換性が 11.2.0.0.0 未満に設定された NOLOGGING LOB サポート。

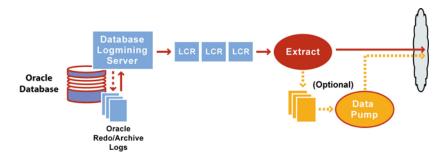
詳細は、1-5ページの「各キャプチャ・モードでサポートされる Oracle データ型とオブジェクトのサマリー」を参照してください。

クラシック・キャプチャを使用する場合、Oracle GoldenGate パラメータ・ファイルの基本的なセットを生成するための構成手順を完了した後、追加で手順を実行する必要があります。6-1ページの「クラシック・キャプチャを使用する場合の追加構成手順」を参照してください。

4.5.2 統合キャプチャについて

統合キャプチャ・モードでは、Oracle GoldenGate Extract プロセスは、データベース・ログマイニング・サーバーと直接やり取りし、論理変更レコード (LCR) 形式でデータの変更を受け取ります。統合キャプチャでは、クラシック・キャプチャより多くのデータと記憶域のタイプがサポートされ、サポートがより透過的です。詳細は、1-5ページの「各キャプチャ・モードでサポートされる Oracle データ型とオブジェクトのサマリー」を参照してください。

図 4-3 統合キャプチャ



統合キャプチャのその他の利点は、次のとおりです。

- 統合キャプチャはデータベースと完全に統合されているため、Oracle RAC、ASM および TDE と組み合せる場合に追加設定は必要ありません。
- 統合キャプチャでは、データベース・ログマイニング・サーバーを使用して Oracle REDO ストリームにアクセスします。アーカイブ・ログの異なるコピー間またはオンライン・ログの異なるミラー・バージョン間での切替えは自動的に行われます。したがって、統合キャプチャでは、ディスクの破損、ハードウェア障害またはオペレータの誤操作によってログ・ファイルが使用できない場合、アーカイブ・ログやオンライン・ログの他のコピーが使用できれば、この状態が透過的に処理されます。
- 統合キャプチャによって、表のフィルタリングが高速になります。
- 統合キャプチャでは、Point-in-Time リカバリおよび RAC 統合がより効率的に処理されます。
- 統合キャプチャには、統合ログ管理の機能が備わっています。Oracle Recovery Manager(RMAN) で、Extract で必要なアーカイブ・ログが自動的に保持されます。

4.5.2.1 統合キャプチャでサポートされるデータベースのバージョン

データベースのバージョンによって、統合キャプチャで使用可能なデータ型サポートが決まります。

■ **フル・サポート**: すべての Oracle データ型と記憶域のタイプをサポートするには、ソース・データベースの互換性設定を統合 Extract 11.2.x 用 11.2.0.3 データベース固有のバンドル・パッチ (Doc ID 1411356.1) が適用された 11.2.0.3.0 以上にする必要があります。このパッチを Oracle サポートから入手するには、次へ移動します。

https://support.oracle.com/oip/faces/secure/km/DocumentDisplay.jspx ?id=1411356.1

制限付きサポート: 統合キャプチャは、互換性が 11.2.0.3 未満のソース・データベースに対 する 11.2.0.3.0 ダウンストリーム・マイニング・データベースで使用できますが、このモー ドでは、SECUREFILE LOB、XML 列、Transparent Data Encryption および UDT はサポート されません。ダウンストリーム・マイニング・データベースは、統合 Extract 11.2.x 用 11.2.0.3 データベース固有のバンドル・パッチ (Doc ID 1411356.1) が適用されている必要が あります。4-5ページの「統合キャプチャ・デプロイ・オプション」を参照してください。

様々なバージョンの RDBMS でサポートされるデータ型の違いについては、1-5ページの「各 キャプチャ・モードでサポートされる Oracle データ型とオブジェクトのサマリー」を参照して ください。

4.5.2.2 統合キャプチャ・デプロイ・オプション

統合キャプチャには、マイニング・データベースのデプロイ場所に応じて使用されるデプロ イ・オプションが2つあります。マイニング・データベースは、ログマイニング・サーバーが デプロイされているデータベースです。

- **ローカル・デプロイ**: ローカル・デプロイの場合、ソース・データベースとマイニング・ データベースは同じです。ソース・データベースは、REDO ストリームをマイニングして 変更をキャプチャする対象のデータベースで、ログマイニング・サーバーのデプロイ先で もあります。統合キャプチャはデータベースと完全に統合されているため、このモードで は、特別なデータベース設定は不要です。
- **ダウンストリーム・デプロイ**: ダウンストリーム・デプロイでは、ソース・データベースと マイニング・データベースは、異なるデータベースです。ログマイニング・サーバーは、 ダウンストリーム・データベースに作成します。REDO ログをダウンストリーム・マイニ ング・データベースに転送してそこでキャプチャを行うようにソース・データベースの REDO 転送を構成します。キャプチャのオーバーヘッドや変換によるその他のオーバー ヘッドおよび本番サーバーの他の処理を軽減するには、キャプチャにダウンストリーム・ マイニング・サーバーを使用することをお薦めしますが、これにはログの転送や他の構成 が必要です。

ダウンストリーム・マイニング構成を使用する場合、ソース・データベースとマイニング・ データベースは同じプラットフォームのものである必要があります。たとえば、ソース・デー タベースが Windows 64 ビットで稼働している場合、ダウンストリーム・データベースのプ ラットフォームも Windows 64 ビットである必要があります。ダウンストリーム・マイニン グ・データベースを構成するには、A-1ページの「統合キャプチャ用のダウンストリーム・マ イニング・データベースの構成」を参照してください。

4.5.3 キャプチャ・モードの組合せ

統合 Extract 11.2.x 用 11.2.0.3 データベース固有のバンドル・パッチ (Doc ID 1411356.1) が適用 された Oracle 11.2.0.3 の場合、統合キャプチャ、クラシック・キャプチャまたは両モードの組 合せを使用できます。表の属性とデータ型に応じて、異なるキャプチャ・モードの複数の Extract 間に表を分けることができます。Oracle GoldenGate のパラメータ・ファイル、証跡、 変換機能、マッピング・オプションおよびレプリケーション・メカニズムは、基本的には両方 のモードで同じです。

4.6 Oracle GoldenGate のデータベース・ユーザーの割当て

この手順では、Oracle GoldenGate プロセスごとにデータベース・ユーザーを 1 つ作成し、適 切なデータベース権限を割り当てます。

- 1. ソース・システムとターゲット・システムの Oracle GoldenGate にそれぞれ専用のソー ス・データベース・ユーザーとターゲット・データベース・ユーザーを作成します。ソー スまたはターゲットの Oracle データベースに接続する Oracle GoldenGate プロセスすべて に同じユーザーを割り当てることができます。
 - Extract(ソース・データベース): ユーザーは、ソース・データベースでメタデータ問合 せを実行し、REDO ストリームで直接サポートされないデータ型についてソース表か

らデータをフェッチします。これらのデータ型のリストは、1-5ページの「各キャプ チャ・モードでサポートされる Oracle データ型とオブジェクトのサマリー」を参照し てください。統合キャプチャのローカル・マイニング・デプロイでは、このユーザー は、作成、変更、ログマイニング・サーバーへの接続およびログマイニング・サー バーからの論理変更レコード(LCR)の受信も行います。

- Replicat(ターゲット・データベース): このユーザーは、Replicat チェックポイント表 の作成と、DML、DDL および初期ロード操作の適用に使用されます。
- Manager (DDL サポートを使用する場合、ソース・データベース): このユーザーは、 DDL サポートが使用される場合、Oracle GoldenGate データベース・オブジェクトに 対してメンテナンスを実行します。
- DEFGEN (ソースまたはターゲット・データベース): このユーザーは、ローカル・メタ データの問合せを実行し、リモート Oracle GoldenGate インスタンスにメタデータを 提供するデータ定義ファイルを作成します。
- 2. Extract を統合キャプチャ・モードで使用し、かつダウンストリーム・マイニング・データ ベースを使用する場合、ダウンストリーム・データベースでマイニング・ユーザーを作成 します。このユーザーは、マイニング・データベースで作成、変更およびログマイニン グ・サーバーへの接続を行い、ログマイニング・サーバーから論理変更レコード (LCR) を 受信します。このユーザーはソース Extract ユーザーと同じでも別でもかまいません。

注意:マイニング・ユーザーの名前は慎重に選択してください。このユー ザーによる作成後、データベース・ログマイニング・サーバーは、他のユー ザーが変更したり、使用できません。

3. Oracle GoldenGate ユーザーに正しい権限を割り当てるには、表 4-1 を参照してください。

表 4-1 Oracle GoldenGate プロセス別のデータベース権限

ユーザーの権限	Extract (クラシック・ キャプチャ)	Extract (統合 キャプチャ)	Replicat	Manager	目的
CREATE SESSION, ALTER SESSION	X	X	X		
ALTER SYSTEM	X	Х			
RESOURCE	Х	X	X		RESOURCE を Replicat に付与できない場合、次の文を使用します (tablespace は、ターゲット・オブジェクトが含まれるすべての表領域です)。 ALTER USER user QUOTA {size
					UNLIMITED ON tablespace;
CONNECT	X	X	X		Replicat がターゲットのオブジェクトまたは PL/SQL プロシージャを所有する場合にのみ必要です。Replicat に CONNECT を付与できない場合、Replicat で作成する必要があるオブジェクトに対する CREATE object を付与します。
SELECT ANY DICTIONARY	X	X	X		
FLASHBACK ANY TABLE	X	X			
または					
owner.tableに対 する FLASHBACK					
SELECT ANY TABLE	X	X	X		
または					
owner.tableに対 する SELECT					
$\begin{array}{c} {\rm dba_clusters} \mathcal{O} \\ {\rm SELECT} \end{array}$	X	Х			
ターゲット表に対 する INSERT、 UPDATE、DELETE			X		
CREATE TABLE			Х		データベースのチェックポイント機能 を有効にするために ADD CHECKPOINTTABLE を使用する場合に必 要です。
ターゲット表に DDL 操作を発行 するために必要な 権限 (DDL サポー トのみ)			X		
DBMS_FLASHBACK パッケージの EXECUTE	Х	Х			Oracle GoldenGate によって DBMS_ FLASHBACK.GET_SYSTEM_CHANGE_NUMBER がコールされます。

表 4-1 Oracle GoldenGate プロセス別のデータベース権限(続き)

	Extract (クラシック・	Extract (統合	Danillant	Managara	
ユーザーの権限 GGS_GGSUSER_ ROLE	キャプチャ) X	キャプチャ) X	Replicat	Manager	目的 Oracle GoldenGate が所有する DDL オブジェクトに対する DML 権限のロール (DDL サポートを使用する場合)。ロールは DDL オブジェクトのインストール時に作成されます。このロールをインストールするユーザーは、SYSDBA 権限が必要です。
Oracle GoldenGate DDL オブジェクトに対す る DELETE				Х	Oracle GoldenGate DDL データベース・オブジェクトを保守するパラメータを使用する場合にのみ必要です。
LOCK ANY TABLE			X		ダイレクト・バルク・ロードを使用してデータを SQL*Loader に挿入する Oracle GoldenGate の初期ロード方法 を使用する場合にのみ必要です。
sys.dbms_ internal_clkm	Х				クラシック・キャプチャ・モードでの Oracle Transparent Data Encryption(TDE) のレプリケートに必 要です。
SELECT ANY TRANSACTION	X				Extract が最新の Oracle ASM API を使用する場合に必要です。詳細は、Oracle GoldenGate リファレンス・ドキュメントの TRANLOGOPTIONS のDBLOGREADER オプションに関する項を参照してください。
Oracle 11.2.0.3 以降: dbms_goldengate_auth.grant_admin_privilegeを使用して付与される権限		Х			統合キャプチャ・モードでデータベース・ログマイニング・サーバーと対話する場合に必要です。4-3ページの「使用するキャプチャ方法の決定」を参照してください。
11.2.1.3 より前の Oracle リリース: dbms_streams_ auth.grant _admin_ privilege を使用 して付与される権 限。					
V_\$DATABASE ビューに対する SELECT		X			ダウンストリーム・キャプチャ構成で のダウンストリーム Extract マイニング の場合にのみ必要です。
dbms_logmrn_d パッケージに対する EXECUTE および sys.logmnr_ buildlogからの SELECT		Х			Oracle ソース・データベースのバージョンが 11.1.0.5 以上かつ 11.2.0.1 以下の場合にソース Extract ユーザーが REGISTER EXTRACT コマンドを発行するために必要です。

- 4. 各データベース・ユーザーの記録を保持します。Oracle GoldenGate のパラメータ・ファ イルでそれらを指定する必要があります。
- 5. データのセキュリティを維持したり、Oracle GoldenGate の処理を正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。
- 6. Extract がクラシック・キャプチャ・モードで実行される場合、次の機能を使用するには追 加のユーザーまたは権限が必要な場合があります。
 - RMAN ログ保存 (6-9 ページの「ログ保存のオプション」を参照)
 - TDE サポート (6-1 ページの「クラシック・キャプチャ・モードでの Oracle TDE データ の構成」を参照)
 - ASM (6-7 ページの「クラシック・キャプチャ・モードの場合の Oracle ASM インスタ ンスからのキャプチャ」を参照)

4.7 Oracle GoldenGate インスタンスの作成

各 Oracle GoldenGate インストールは、Manager プロセスをベースにしています。これは、 Oracle GoldenGate プロセスのインスタンス化、ポート番号の割当ておよびファイル・メンテ ナンスを実行するコントローラ・プロセスです。Manager プロセスとその子プロセス、および それに関連するプログラムとファイルが合さって1つの Oracle GoldenGate インスタンスを構 成します。

Oracle GoldenGate を実行するには、Oracle GoldenGate 環境に含まれるすべてのシステムで Manager プロセスが実行されている必要があります。Manager を実行するには、まず、そのパ ラメータ・ファイルを作成します。

Manager パラメータ・ファイルを作成する手順

- 1. Oracle GoldenGate ディレクトリから ggsci プログラムを実行して、Oracle GoldenGate ソ フトウェア・コマンド・インタフェース (GGSCI) を開きます。
- 2. GGSCI で Manager パラメータ・ファイルを編集します。

EDIT PARAMS MGR

- 1行に1つずつ Manager パラメータを追加します。パラメータ文が複数行にわたる場合、 各改行の前にアンパサンド(&)を使用します。
 - 必須 Manager パラメータは PORT のみですが、DYNAMICPORTLIST をお薦めします。
 - クラスタ環境では、AUTOSTART および AUTORESTART パラメータを使用して Manager を 構成し、Manager の起動またはフェイルオーバー時に Oracle GoldenGate プロセスが 自動的に起動または再起動されるようにします。
 - PURGEOLDEXTRACTS を使用して、証跡ファイルの累積を管理します。
 - Manager パラメータの詳細は、『Oracle GoldenGate Windows and UNIX リファレンス・ ガイド』を参照してください。

Manager とそのネットワーク接続の構成の詳細は、『Oracle GoldenGate Windows and UNIX 管理者ガイド』を参照してください。

4. ファイルを保存し、閉じます。

例 4-1

次の例では、Manager パラメータ・ファイルは UNIX システム上にあり、必須パラメータと推 奨パラメータを使用しています。

PORT 7809 DYNAMICPORTLIST 7810-7820, 7830 AUTOSTART ER t* AUTORESTART ER t*, RETRIES 4, WAITMINUTES 4 STARTUPVALIDATIONDELAY 5 PURGEOLDEXTRACTS /ogg/dirdat/tt*, USECHECKPOINTS, MINKEEPHOURS 2

4.8 変更のキャプチャ用の Extract の構成

次の手順をソース・システムで実行し、変更のキャプチャおよびネットワーク経由の転送をサ ポートするプライマリおよびデータ・ポンプ Extract プロセスを構成します。

4.8.1 プライマリ Extract の構成 (クラシックまたは統合モード)

次の手順では、トランザクション・データをキャプチャする Extract を構成します。

1. ソース・システムの GGSCI で、Extract パラメータ・ファイルを作成します。

EDIT PARAMS name

説明: name は、プライマリ Extract の名前です。

2. 次に示す順序でExtract パラメータを入力します。パラメータ文ごとに新しい行を開始しま す。クラシック・キャプチャおよび統合キャプチャの例を示します。入力変数は異なりま す。詳細は、表 4-2 を参照してください。

クラシック・キャプチャ・モードでのプライマリ Extract グループの基本的なパラメータ:

EXTRACT finance

USERID ogg,

PASSWORD AACAAAAAAAAAAJAUEUGODSCVGJEEIUGKJDJTFNDKEJFFFTC &

AES128, ENCRYPTKEY securekev1

ENCRYPTTRAIL AES192, KEYNAME mykey1

EXTTRAIL /ggs/dirdat/lt

SEQUENCE hr.employees seq;

TABLE hr.*;

ソース・データベースがマイニング・データベースの場合の統合キャプチャ・モードでの プライマリ Extract グループの基本的なパラメータ:

EXTRACT financep

USERID ogg, &

PASSWORD AACAAAAAAAAAAJAUEUGODSCVGJEEIUGKJDJTFNDKEJFFFTC &

AES128, ENCRYPTKEY securekey1

ENCRYPTTRAIL AES192, KEYNAME mykey1

EXTTRAIL /ggs/dirdat/lt

SEQUENCE hr.employees seq;

TABLE hr. *:

マイニング・データベースがダウンストリーム・データベースの場合の統合キャプチャ・ モードでのプライマリ Extract グループの基本的なパラメータ:

EXTRACT financep

USERID ogg, PASSWORD AACAAAAAAAAAAJAUEUGODSCVGJEEIUGKJDJTFNDKEJFFFTC &

AES128, ENCRYPTKEY securekey1

TRANLOGOPTIONS [MININGUSER oggm, &

MININGPASSWORD AACAAAAAAAAAAJAUEUGODSCVGJEEIUGKJDJTFNDKEJFFFTC &

AES128, ENCRYPTKEY securekey1]

TRANLOGOPTIONS INTEGRATEDPARAMS (MAX SGA SIZE 164, &

DOWNSTREAM REAL TIME MINE y)

ENCRYPTTRAIL AES192, KEYNAME mykey1

EXTTRAIL /qqs/dirdat/lt

SEQUENCE hr.employees_seq; TABLE hr.*;

表 4-2 プライマリ Extract の基本的なパラメータ (クラシックまたは統合モード)

パラメータ	説明				
EXTRACT group name	group name は、Extract グループの名前です。				
USERID user id, PASSWORD pw [encryption options]	4-6 ページの「Oracle GoldenGate のデータベース・ユーザーの割当て」で作成したデータベース・ユーザーのデータベース接続情報を指定します。				
	■ USERID では、Extract データベース・ユーザーを指定します。				
	■ PASSWORD では、ユーザーのパスワードを指定します。				
	encryption optionsでは、パスワードを暗号化する方法の1つを指定します。				
	その他のログイン・オプションと暗号化の詳細は、『Oracle GoldenGate Windows and UNIX リファレンス・ガイド』を参照してください。				
	使用する Oracle GoldenGate キャプチャ・モードに必要な権限が、このユーザーにあることを確認します。				
	■ 必須ソース・データベース・ログイン (4-6 ページの「Oracle GoldenGate の データベース・ユーザーの割当て」)				
	■ RMAN ログ保存 (6-9 ページの「ログ保存のオプション」)				
	■ TDE サポート (6-1 ページの「クラシック・キャプチャ・モードでの Oracle TDE データの構成」)				
	■ ASM (6-7 ページの「クラシック・キャプチャ・モードの場合の Oracle ASM インスタンスからのキャプチャ」)				
TRANLOGOPTIONS MININGUSER user id, MININGPASSWORD pw [,encryption options]	(統合キャプチャ・モード)ダウンストリーム・マイニング・データベース(使用する場合)で、ログマイニング・サーバーの接続情報を指定します。				
	■ MININGUSER では、ダウンストリーム・マイニング・データベースの Extract ユーザーを指定します。これは、A-1 ページの「統合キャプチャ用のダウンストリーム・マイニング・データベースの構成」で作成したユーザーです。				
	■ MININGPASSWORD では、ユーザーのパスワードを指定します。				
	■ encryption optionsでは、パスワードを暗号化する方法の1つを指定します。				
	データベース・ログマイニング・サーバーがソース・データベースと異なるデータベースにある場合にのみ使用します。そうでない場合は USERID のみを使用します。MININGUSER を使用する場合、両方のデータベースへのログインが必要なため、USERID とともに使用します。その他のログイン・オプションと暗号化の詳細は、『Oracle GoldenGate Windows and UNIX リファレンス・ガイド』を参照してください。				
TRANLOGOPTIONS [INTEGRATEDPARAMS (parameter[,])]	(統合キャプチャ・モード)データベース・ログマイニング・サーバーを含む Oracle データベースにパラメータを渡します。ダウンストリーム・マイニング・データベースが使用される場合に、ログマイニング・サーバーで使用される SGA メモリーの量、ログマイニング・サーバーをサポートするプロセスの数および統合キャプチャがリアルタイム・モードで実行されるか、アーカイブログ・モードで実行されるかの指定に使用します。4-17ページの「統合キャプチャのチューニングの推奨事項」を参照してください。				
ENCRYPITRAIL encryption options	ローカル証跡を暗号化します。暗号化のオプションは、『Windows and UNIX リファレンス・ガイド』を参照してください。				

表 4-2 プライマリ Extrac	t の基本的なパラメータ	(クラシック	または統合モー	ド)	(続き)
4X 7-2 / / Y / LAUGU	ハリスタイトリング・ソファーブ	1 1 1 1 1 1	^ ~/_ I ◆ N L 口 L	1 · /	(M) C

パラメータ	説明
EXTTRAIL pathname	プライマリ Extract によってキャプチャされたデータが書き込まれるローカル証 跡のパス名を指定します。
SEQUENCE owner.sequence;	キャプチャする Oracle 順序を指定します。
	■ ownerは、スキーマ名です。
	sequence は、順序の名前です。
	文はセミコロンで終えます。
TABLE owner.table;	データの変更を抽出する表を指定します。
	■ ownerは、スキーマ名です。
	■ table は、表の名前またはワイルドカードで定義される表のグループです。
	スキーマ名はワイルドカード指定できません。複数のスキーマの表からデータを抽出するには、スキーマごとに別々の TABLE 文を使用します。例を次に示します。
	TABLE fin.*;
	TABLE hr.*;
	TABLE 文はセミコロンで終えます。
	表および順序をワイルドカード指定から除外するには、TABLE 文の後で TABLEEXCLUDE owner.table パラメータを使用します。
	データのフィルタリング、マッピングおよび操作を制御するその他のオプションとその詳細は、『Oracle GoldenGate Windows and UNIX リファレンス・ガイド』の TABLE に関する項を参照してください。

- 3. このマニュアルの他の箇所で推奨されているオプションの Extract のパラメータや『Oracle GoldenGate Windows and UNIX リファレンス・ガイド』に示されているその他のパラメー タを入力します。
- 4. ファイルを保存し、閉じます。

4.8.2 データ・ポンプの構成

次の手順では、ローカル証跡を読み取り、データをネットワーク経由でリモート証跡に送信す るデータ・ポンプを構成します。

1. ソース・システムの GGSCI で、データ・ポンプ・パラメータ・ファイルを作成します。

EDIT PARAMS name

説明: name は、データ・ポンプ Extract の名前です。

2. 次に示す順序でデータ・ポンプ・パラメータを入力します。パラメータ文ごとに新しい行 を開始します。入力変数は異なります。詳細は、表 4-3 を参照してください。

データ・ポンプ Extract グループの基本的なパラメータ:

EXTRACT extpump

USERID ogg, PASSWORD AACAAAAAAAAAAJAUEUGODSCVGJEEIUGKJDJTFNDKEJFFFTC &

AES128, ENCRYPTKEY securekey1 DECRYPTTRAIL AES192, KEYNAME mykey1

RMTHOST fin1, MGRPORT 7809 ENCRYPT AES192, KEYNAME securekey2

ENCRYPTTRAIL AES192, KEYNAME mykey1

RMTTRAIL /ggs/dirdat/rt

SEQUENCE hr.employees_seq;

TABLE hr.*;

表 4-3 データ・ポンプ Extract の基本的なパラメータ

パラメータ	説明
EXTRACT group name	group name は、データ・ポンプ Extract の名前です。
USERID user id, PASSWORD pw [encryption options]]	データベース接続情報を指定します。
	■ USERID では、Extract データベース・ユーザーを指定します。
	■ PASSWORD では、ユーザーのパスワードを指定します。
	■ encryption optionsでは、パスワードを暗号化する方法の1つを指定します。
RMIHOST hostname, MGRPORT portnumber, [, ENCRYPT algorithm KEYNAME keyname]	■ RMTHOST では、ターゲット・システムの名前または IP アドレスを指定します。
	■ MGRPORT では、ターゲットで Manager が実行されるポートの番号を指定します。
	■ ENCRYPT では、TCP/IP でのデータの暗号化 (オプション)を指定します。
	その他のオプションと暗号化の詳細は、『Oracle GoldenGate Windows and UNIX リファレンス・ガイド』を参照してください。
ENCRYPTTRAIL encryption options	このデータ・ポンプの書込み先のターゲットのリモート証跡を暗号化します。暗号化のオプションは、『Windows and UNIX <i>リファレンス・ガイド</i> 』を参照してください。
DECRYPTTRAIL encryption options	データ・ポンプが読み取るローカル入力証跡を復号化します。encryption options は、 プライマリ Extract グループに使用される ENCRYPTTRAIL 文のオプションと一致する必要 があります。
RMTTRAIL pathname	リモート証跡のパス名を指定します。
SEQUENCE owner.sequence;	Oracle 順序を指定します。ほとんどの場合、このリストは、プライマリ Extract パラメータ・ファイルのリストと同じです。
	■ owner は、スキーマ名です。
	■ sequence は、順序の名前です。
	文はセミコロンで終えます。
TABLE owner.table;	表を指定します。ほとんどの場合、このリストは、プライマリ Extract パラメータ・ファイルのリストと同じです。
	■ owner は、スキーマ名です。
	■ table は、表の名前またはワイルドカードで定義される表のグループです。
	スキーマ名はワイルドカード指定できません。複数のスキーマの表からデータを抽出する には、スキーマごとに別々の TABLE 文を使用します。例を次に示します。
	TABLE fin.*; TABLE hr.*;
	TABLE 文はセミコロンで終えます。
	表および順序をワイルドカード指定から除外するには、TABLE 文の後で TABLEEXCLUDE owner.table パラメータを使用します。
	データのフィルタリング、マッピングおよび操作を制御するその他のオプションとその詳細は、 \mathbb{F} Oracle GoldenGate $\mathit{Windows}$ and UNIX UZFUZX ・ガイド』の TABLE に関する項を参照してください。

- 3. このマニュアルの他の箇所で推奨されているオプションの Extract のパラメータや『Oracle GoldenGate Windows and UNIX リファレンス・ガイド』に示されているその他のパラメー タを入力します。
- 4. ファイルを保存し、閉じます。

4.9 変更の配信用の Replicat の構成

ターゲット・システムで次の手順を実行し、ターゲット・システムへの変更の配信をサポート するオブジェクトを構成します。

4.9.1 チェックポイント表の作成

Replicat は、ターゲット・データベースのチェックポイント表にチェックポイントを保持しま す。各チェックポイントは、Replicat トランザクション内でチェックポイント表に書き込まれ ます。チェックポイントはトランザクションに対して成功または失敗のいずれかであるため、 プロセスまたはデータベースに障害が発生した場合でも、Replicat によってトランザクション が一度のみ適用されることが保証されます。

注意: この手順では、デフォルトのチェックポイント表をインストールしま す。ほとんどの場合、デフォルトで十分です。複数のチェックポイント表を 使用 (Replicat グループごとに異なる表を使用するなど) することができま す。デフォルト以外のチェックポイント表 (デフォルトの表をオーバーライ ドする)を使用するには、8-1ページの「Oracle GoldenGate レプリケーショ ンのインスタンス化と開始」の手順で Replicat プロセスを作成する際、ADD REPLICAT の CHECKPOINTTABLE オプションを使用します。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 11

4.9.1.1 ターゲット・データベースへのチェックポイント表の追加

1. ターゲットの Oracle GoldenGate ディレクトリから GGSCI を実行し、DBLOGIN コマンドを 発行してターゲット・データベースにログインします。

DBLOGIN, USERID db_user [, PASSWORD pw [encryption options]]

説明:

- USERID db user、PASSWORD pw および encryption options では、CREATE TABLE 権限を 持つユーザーのデータベース資格証明とオプションのパスワード暗号化を指定します。 コマンド・オプションの詳細は、『Oracle GoldenGate Windows and UNIX リファレン ス・ガイド』を参照してください。
- 2. GGSCI で、任意のスキーマ (Oracle GoldenGate 専用であることが推奨されます) にチェッ クポイント表を作成します。

ADD CHECKPOINTTABLE owner.table

4.9.1.2 Oracle GoldenGate 構成でのチェックポイント表の指定

1. GLOBALS ファイルを作成します (あるいは、既存のものを編集します)。

EDIT PARAMS ./GLOBALS

注意: EDIT PARAMS により、単純なテキスト・ファイルが作成されます。 EDIT PARAMS 後にファイルを保存すると、ファイル拡張子なしの GLOBALS (大 文字)という名前で保存されます。これはそのままにし、ファイルはルート の Oracle Golden Gate ディレクトリに置いておきます。

2. GLOBALS ファイルで、CHECKPOINTTABLE パラメータを入力します。

CHECKPOINTTABLE owner.table

説明: owner.table は、データベースでサポートされる所有者と名前です。

3. GLOBALS ファイルを保存し、閉じます。

4.9.1.3 チェックポイント表へのデフォルトの非同期 COMMIT の無効化

Replicat がチェックポイント表を使用する際、NOWAIT オプションを指定して非同期の COMMIT を使用し、パフォーマンスを向上させます。Replicat はこの COMMIT の適用直後に処理を続行で き、同時にデータベースでのトランザクションの記録はバックグラウンドで行われます。 Replicat パラメータ・ファイルで DISABLECOMMITNOWAIT オプションを指定して DBOPTIONS パラ メータを使用すると、NOWAIT を指定した非同期の COMMIT を無効にできます。

注意: Replicat グループに対してチェックポイント表が使用されない場合、 チェックポイントはディスク上のファイルに保持されます。この場合、 Replicat は WAIT を指定して COMMIT を使用し、データベースの障害が発生し た場合に、チェックポイント・ファイルに書き込まれたトランザクションの 状態がリカバリ後の状態と異なるような矛盾が生じないようにします。

4.9.2 Replicat の構成

次の手順では、特別なマッピングやデータの変換をしない基本的な方法で Replicat プロセスを 構成します。高度なマッピングのオプションについては、『Oracle GoldenGate Windows and UNIX 管理者ガイト』を参照してください。

1. ターゲット・システムの GGSCI で、Replicat パラメータ・ファイルを作成します。

EDIT PARAMS name

説明: name は、Replicat グループの名前です。

2. 次に示す順序でReplicat パラメータを入力します。パラメータ文ごとに新しい行を開始し ます。詳細は、表 4-4 を参照してください。

REPLICAT financer

USERID ogg, PASSWORD AACAAAAAAAAAAAJAUEUGODSCVGJEEIUGKJDJTFNDKEJFFFTC & AES128, ENCRYPTKEY securekey1

- -- SUPPRESSTRIGGERS is for Oracle 10.2.0.5 & later patches, and
- -- for Oracle 11.2.0.2 and later 11gR2 versions
- -- See following "Note" for how SUPPRESSTRIGGERS works.

DBOPTIONS SUPPRESSTRIGGERS

DECRYPTTRAIL AES192, KEYNAME mykey1

ASSUMETARGETDEFS

DISCARDFILE /users/ogg/disc

MAP hr.*, TARGET hr2.*;

注意: SUPPRESSTRIGGERS によって、トリガー本体が実行されなくなります。 ただし、トリガーの WHEN 部分はコンパイルして適切に実行し、データベー ス・エラーが起きないようにする必要があります。

表 4-4 Replicat の基本的なパラメータ

パラメータ	説明
REPLICAT group name	group name は、Replicat グループの名前です。
USERID user id, PASSWORD pw [encryption options]]	データベース接続情報を指定します。
	■ USERID では、Replicat データベース・ユーザーを指定します。
	■ PASSWORD では、ユーザーのパスワードを指定します。
	■ encryption optionsでは、パスワードを暗号化する方法の1つを指定します。
	その他のログイン・オプションと暗号化の詳細は、『Oracle GoldenGate <i>Windows and UNIX リファレンス・ガイド</i> 』を参照してください。
DBOPTIONS SUPPRESSTRIGGERS, DEFERREFCONST	■ SUPPRESSTRIGGERS では、Oracle GoldenGate でのレプリケーション用に構成された ターゲット・オブジェクトでトリガーが発生しないようにします。
	■ DEFERREFCONST では、DEFERRABLE に制約を設定し、ターゲット・データベースによる カスケード制約の適用を Replicat トランザクションのコミットまで遅延させます。
	これらの DBOPTIONS のオプションに関するその他の重要な情報は、 \mathbb{F} Oracle GoldenGate Windows and UNIX リファレンヌ・ガイド』を参照してください。
ASSUMETARGETDEFS	データ定義の解釈方法を指定します。ASSUMETARGETDEFSでは、ソース表とターゲット表で、セマンティクスも含めて定義が同一であるとみなされます。(この手順では、定義は同一であるものとします。)
	ソース表とターゲット表の定義が異なる場合、かわりに SOURCEDEFS を使用し、DEFGEN でソース・データ定義ファイルを作成します。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。
DISCARDFILE full_ pathname	Replicat が、データベース・エラーになったレコードなどの拒否されたレコード・データを書き込むファイルのフルパス名を指定します。破棄ファイルはオプションですが、推奨されます。その他のオプションについては、『Oracle GoldenGate <i>Windows and UNIX リファレンス・ガイド</i> 』を参照してください。
DECRYPTTRAIL encryption options	この Replicat が読み取る入力証跡を復号化します。encryption options は、この入力証跡に対する ENCRYPTTRAIL 文のオプションと一致する必要があります。
MAP owner.table,	ソース表とターゲット表の関係を指定します。
TARGET owner.table;	■ owner は、スキーマ名です。
	■ table は、表の名前または複数の表に対するワイルドカード定義です。
	スキーマ名はワイルドカード指定できません。複数のスキーマの表をマップするには、スキーマごとに別々の MAP 文を使用します。例を次に示します。
	MAP fin.*, TARGET fin.*; MAP hr.*, TARGET hr.*;
	MAP 文はセミコロンで終えます。
	ワイルドカード指定から表を除外するには、MAPEXCLUDE パラメータを使用します。
	データのフィルタリング、マッピングおよび操作を制御するその他のオプションとその詳細は、『Oracle GoldenGate <i>Windows and UNIX リファレンス・ガイド</i> 』の MAP に関する項を参照してください。

- 3. このマニュアルの他の箇所で推奨されているオプションの Extract のパラメータや『Oracle GoldenGate Windows and UNIX リファレンス・ガイド』に示されているその他のパラメー タを入力します。
- 4. ファイルを保存し、閉じます。

4.10 統合キャプチャのチューニングの推奨事項

統合キャプチャでは、ソース・データベースの REDO ストリームのマイニングに、マイニン グ・データベースのデータベース・ログマイニング・サーバーを使用します。ログマイニン グ・サーバーでは、特定の数のプロセスと特定の量の共有メモリーという2種類のデータベー ス・リソースが使用されます。これらのリソースは、Extract の TRANLOGOPTIONS パラメータの INTEGRATEDPARAMS リストに次のパラメータを指定して制御できます。

- ログマイニング・サーバーで使用される共有メモリーの量を制御するには、値を MB で指 定して max sqa size パラメータを指定します。
- ログマイニング・サーバーで使用されるプロセスの数を制御するには、parallelism パラ メータを指定します。

たとえば、次のように指定すると、ログマイニング・サーバーでメモリーを 200MB、パラレル 度3を使用するよう指定できます。

TRANLOGOPTIONS INTEGRATEDPARAMS (max sqa size 200, parallelism 3)

ログマイニング・サーバーで使用される共有メモリーは、データベースのシステム・グローバ ル領域 (SGA) の Streams プール部分です。したがって、特定のデータベース・インスタンスに 対して統合キャプチャ・モードでの実行が予測される Extract プロセスの数に対して使用可能な メモリーが十分あるようにするには、streams pool sizeを十分に高い値に設定する必要があ ります。Streams プールは、データベースの他のコンポーネント (AO など) でも使用されるた め、Oracle GoldenGate の Streams プールのサイズを設定する際、このことを必ず考慮に入れ てください。

デフォルトでは、1つの統合キャプチャ Extract で、max sga_size を 1GB、parallelism を 2 にしてログマイニング・サーバーが実行されるよう要求します。したがって、同一データベー ス・インスタンスで3つの Extract を統合キャプチャ・モードで実行する場合、3GB 以上のメ モリーをStreamsプールに割り当てる必要があります。ベスト・プラクティスとしては、 Streams プールの 25 パーセントが使用可能であるようにします。たとえば、3 つの Extract を統 合キャプチャ・モードで実行する場合、次のように streams_pool_size を設定します。

3 GB + (3 GB * 0.25) = 3.75 GB

REDO の量が多いシステムでソース・データベースが稼働している場合、_log_buffer_size を 128 (デフォルトは8)に設定すると、ログマイニング・サーバーのパフォーマンスがよくなるこ ともパフォーマンス・テストで確認されています。このパラメータを設定するには、次のコマ ンドを使用します。

SQL> alter system set log buffer size=128;

4.11 最良のパフォーマンスを得るための追加プロセス・グルー プの構成

ソース・アプリケーションによるトランザクションの生成とそれらの変更のターゲット・デー タベースへの適用とのタイム・ラグについて、許容できる値を指定するビジネス・ルールを作 成します。これらのルールによって、Oracle GoldenGate が最良の状態で実行されるために必 要な Extract および Replicat のパラレル・プロセスの数が決まります。

Oracle GoldenGate でレプリケートする予定のすべての表に関するサイズおよびアクティビ ティ率を収集し、次のようにします。

- アクティビティ率の低い(複数の)表に、1つの Extract グループを割り当てます。
- アクティビティ率の高い表に、専用の Extract グループを割り当てます。

データ・ポンプと Replicat の専用のグループと連携するようにこれらの Extract グループを構 成します。1つの Extract および Replicat プロセスに対するパラメータ・ファイルの基本的な セットが整ったら、次のことが可能です。

- それらを新規プロセス・グループの名前にコピーします。
- コピーを編集してその他のプロセス・グループ用のパラメータ・ファイルを作成します。

最良のパフォーマンスを得るための Oracle GoldenGate の構成の詳細は、『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』を参照してくださ

4.12 デプロイの次の手順

その柔軟性から、Oracle GoldenGate には、プロセスを起動する前に検討する必要のある機能 やオプションが数多くあります。ビジネス・ニーズに合せて Oracle GoldenGate をさらに構成 するには、次のものを参照してください。

- データベースと REDO ログを準備する場合、および必要なその他のパラメータを確認する 場合は、このマニュアルの以降の章を参照してください。
- DDL 操作をキャプチャして適用するように Oracle GoldenGate を構成する場合、第7章を 参照してください。
- ソースまたはターゲットのいずれかのデータベースが Oracle 以外の場合、そのデータベー ス用の Oracle GoldenGate インストレーションおよびセットアップ・ガイドのインストー ルと構成の手順に従います。その他の情報については、Oracle GoldenGate の管理とリ ファレンスのドキュメントを参照してください。
- 特定のレプリケーション・トポロジを実現するためのその他の構成のガイドラインについ ては、『Oracle Golden Gate Windows and UNIX 管理者ガイド』を参照してください。このガ イドには、次の構成に関する手順が含まれています。
 - ライブ・レポートでの Oracle GoldenGate の使用
 - リアルタイム・データ分散での Oracle GoldenGate の使用
 - リアルタイム・データ・ウェアハウスのための Oracle GoldenGate の構成
 - ライブ・スタンバイ・データベース管理のための Oracle GoldenGate の使用
 - アクティブ / アクティブ型高可用性のための Oracle GoldenGate の使用

ガイドには、次の点に関する情報も含まれています。

- Oracle GoldenGate のアーキテクチャ
- Oracle GoldenGate のコマンド
- Oracle GoldenGate の初期ロード方法
- セキュリティの構成
- カスタマイズ機能の使用
- データ・フィルタリングと操作の構成
- このガイドで示された Oracle GoldenGate GGSCI コマンドおよび Oracle GoldenGate パラ メータの構文オプションとその説明については、『Oracle GoldenGate Windows and UNIX リファレンス・ガイド』を参照してください。

4.13 トランザクション変更のレプリケートを開始するタイミング

ソース・データとターゲット・データが同期している状態、つまり、ソース表とターゲット表 の対応する行に同一のデータ値が含まれる場合にレプリケーションを開始する必要があります。 現在のユーザー・アクティビティがない完全に新規のソース・データベースとターゲット・ データベースから始める場合以外、変更の同期を有効にし、初期ロードをターゲットに適用す る間に進行中のトランザクション変更をキャプチャする必要があります。初期ロードでは、 ソース・データのポイントインタイム・スナップショットがキャプチャされてターゲットに適 用されると同時に、その時点以降に行われた変更が Oracle GoldenGate によって保持されます。

データベースと Oracle GoldenGate プロセス (必要に応じて DDL サポートも)の構成要件を満 たしたら、8-1 ページの「Oracle GoldenGate レプリケーションのインスタンス化と開始」を参 照して、初期ロード、変更のレプリケートの開始およびロード後の競合の処理を行います。

4.14 構成のテスト

本番マシンにデプロイする前に、テスト構成で構成をテストすることが重要です。これは、信 頼できるソースのデータがレプリケーション・プロセスからアクセスされる可能性のあるアク ティブ / アクティブまたは高可用性の構成で特に重要です。テストによって、ターゲットでの 再ロードや他のトラブルシューティング・アクティビティのためにユーザー・アクティビティ を中断することなく、構成の誤りやデータの問題を検出して解決できます。

Oracle GoldenGate のためのデータベースの

この章では、キャプチャおよびレプリケーションがサポートされるように Oracle GoldenGate がやり取りするソースおよびターゲットの Oracle データベースを適切に構成する手順について 説明します。ソース・システムのみに適用される手順、ターゲット・システムのみに適用され る手順、およびその両方に適用される手順があります。4-1 ページの「ソースおよびターゲット の Oracle データベースでの Oracle GoldenGate の構成! で作成した基本のパラメータ・ファイ ルに新規パラメータを追加します。

データベースを準備する手順は次のとおりです。

- ソース表とターゲット表の整合性制約の準備
- ロギング・プロパティの構成
- 一意の行識別子がない表での行変更の制限
- 文字セットの変換のサポート
- フェッチ・オプションの設定
- 特別なデータ型の処理
- 他のデータベース・プロパティの処理
- Oracle GoldenGate と Oracle Exadata を組み合せた使用

5.1 ソース表とターゲット表の整合性制約の準備

トリガー、カスケード制約および一意の識別子は、Oracle GoldenGate 環境で適切に構成する 必要があります。Extract または Replicat を起動してデータを処理する前に、次のガイドライン を確認して、構成に関係のあるものを実装します。

5.1.1 ターゲット表でのトリガーと参照カスケード制約の無効化

トリガーとカスケード制約は、Oracle ターゲット表で無効にする必要があります。Oracle GoldenGate には、Oracle のバージョンに応じてトリガーやカスケード制約を自動的に処理す るオプションが用意されています。

SUPPRESSTRIGGERS を指定した DBOPTIONS: Oracle 10.2.0.5 と 10.2.0.5 に対するそれ以降の パッチおよび Oracle 11.2.0.2 以降の 11gR2 のバージョンに対してサポートされます。この パラメータによって、Replicat は、セッション中、トリガーによって行われる処理を無効 にします。トリガーは無効になりませんが、かわりに、トリガー本体が実行されなくなり ます。ただし、トリガーの WHEN 部分はコンパイルして適切に実行し、データベース・ エラーが起きないようにする必要があります。

- DEFERREFCONST を指定した DBOPTIONS: カスケード更新制約とカスケード削除制約のチェッ クと適用を Replicat トランザクションのコミットまで遅延させます。
- 他の Oracle のバージョンの場合、トリガーと整合性制約を無効にするか、それらを手動で 変更して、Replicat データベース・ユーザーを無視します。

Oracle GoldenGate では、トリガーまたはカスケード制約によって生じた DML がレプリケート されるため、制約を無効にする必要があります。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、それがレプリケートされたバージョンのために冗長になり、データベースに よってエラーが返されます。次の例を考えてみます。この場合、ソース表は emp_src および salary src、ターゲット表は emp targ および salary targ です。

- 1. emp src に削除が発行されます。
- 2. それによって、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. カスケードされた salary src の削除が 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、それを見つけることはできません。

5.1.2 ターゲット表での制約チェックの延期

ターゲットで制約チェックを延期する必要がある場合があります。

- 1. ソースで制約が DEFERRABLE である場合、ターゲットの制約も DEFERRABLE にする必要があ ります。次のパラメータ文のいずれかを使用して、Replicat のトランザクションがコミッ トされるまで制約チェックを延期できます。
 - Replicat セッション全体で制約を延期するには、Replicat パラメータ・ファイルのルー ト・レベルで SQLEXEC を使用します。

SQLEXEC ("alter session set constraint deferred")

- DEFERREFCONST オプションを指定して Replicat パラメータ DBOPTIONS を使用し、各 Replicat トランザクションに対する制約チェックを遅延させます。
- 2. 一時的な主キーの重複によって生じた整合性エラーに対応するように Replicat を構成する 必要がある場合があります。一時的な主キーの重複は、トランザクションの実行中に一時 的に生じ、トランザクションのコミット時に解決する重複です。この種の操作では、通常、 SET x = x+n 式または新しい値が既存の値と等しくなるように値を変更するその他の操作 を使用します。

次に、制約が延期されない場合に一時的な主キーの重複が発生する可能性がある一連の値 変更を示します。例では、主キー列は CODE であり、現在のキー値(更新前)は1、2 およ び3です。

```
update item set code = 2 where code = 1;
update item set code = 3 where code = 2;
update item set code = 4 where code = 3;
```

この例では、キー値2がすでに表に存在するため、Replicatによって最初の更新がターゲット に適用されると、ORA-00001(一意制約)エラーになります。Replicat トランザクションによっ て制約違反エラーが返されます。デフォルトでは、Replicat によってそれらの違反および異常 終了は処理されません。

5.1.2.1 11.2.0.2 より前のバージョンでの一時的な主キーの重複の処理

11.2.0.2 より前のバージョンで一時的な主キーの重複を処理するには、Replicat パラメータ HANDLETPKUPDATE を使用します。この構成では、Replicat は、制約を一時的に延期して一時的 な主キーの更新を処理します。この機能をサポートするには、ターゲット表でDEFERRABLE INITIALLY IMMEDIATE として制約を作成または変更する必要があります。制約が DEFERRABLE で ない場合、Replicat は、HANDLECOLLISIONS および REPERROR パラメータを使用して指定されて いるルール(存在する場合)に従ってエラーを処理するか、異常終了します。

5.1.2.2 11.2.0.2 以降のバージョンでの一時的な主キーの重複の処理

11.2.0.2 より後のバージョンの場合、Replicat は、デフォルトでは Oracle Workspace Manager の作業領域を使用して一時的な主キーの重複を自動的に解決しようとします。この構成では、 Replicat は、コミット時まで制約チェックを延期します。制約は、明示的に延期可能として定 義されている必要はありません。

一時的な主キーの重複の自動処理の要件は、次のとおりです。

- Replicat データベース・ユーザーに次の Oracle Streams 関数へのアクセス権を付与します。 DBMS XSTREAM GG.ENABLE TDUP WORKSPACE()
- ターゲット表は延期可能な制約を持つことはできません。延期可能な制約がある場合、 Replicat はエラーを返し、異常終了します。

延期可能な制約を持つ表を処理するには、制約が DEFERRABLE INITIALLY IMMEDIATE であるこ とを確認し、その表をマップする MAP 文で HANDLETPKUPDATE パラメータを使用します。 HANDLETPKUPDATE パラメータでは、デフォルトである重複の自動処理がオーバーライドされま す。

作業領域の使用によって、次の Oracle GoldenGate のエラー処理パラメータが影響を受けます。

- HANDLECOLLISIONS
- REPERROR

Replicat によって Oracle Workspace Manager の作業領域が有効になると、HANDLECOLLISIONS、 REPERROR などの Oracle GoldenGate パラメータで指定されたエラー処理が無視されます。かわ りに、グループ化されたトランザクションが Replicat によって異常終了され (BATCHSQL が有効 な場合)、アクティブな作業領域を使用して通常のモードで更新が再試行されます。再度 ORA-00001 が発生する場合、Replicat でトランザクションがロールバックされ、有効なエラー 処理ルールを使用してトランザクションが再試行されます。

注意: 非更新レコードに対して ORA-00001 が発生する場合、 HANDLECOLLISIONS、REPERROR などのエラー処理のパラメータによってエラー が処理されます。

一時的な主キーの重複を含む操作に、LOB や XMLType などの行外列の更新も含まれる場合、 作業領域は使用できません。したがって、そのようなケースはサポートされず、ターゲットで 未検出のデータの破損が発生する可能性があります。この例は、次のとおりです。

update T set PK = PK + 1, C LOB = "ABC";

5.1.3 ソース表とターゲット表での行の一意性の保証

Oracle GoldenGate では、レプリケートされた更新および削除に対して正しいターゲット行を 見つけるために、ソース表とターゲット表に一意の行識別子が必要です。TABLE 文または MAP 文で KEYCOLS 句を使用しない場合には、Oracle GoldenGate により、使用される行識別子が次 の優先順位で選択されます。

- 1. 主キー
- 2. 仮想列、UDT、ファンクション・ベース列および NULL 値を許可する列がない英数字順で 最初の一意キー。キーに表示されない索引の一部である列を含めることはできません。
- 3. 仮想列、UDT、ファンクション・ベース列はないが、NULL 値を許可する列がある英数字 順で最初の一意キー。キーに表示されない索引の一部である列を含めることはできません。
- 4. 前述のどのキー・タイプも存在しない場合(表に定義されているその他のキー・タイプが 存在する可能性があっても)、Oracle GoldenGate によって、一意なキーでデータベースを 使用できるすべての列(仮想列、UDT、ファンクション・ベースの列およびOracle GoldenGate の構成から明示的に除外されている列を除く)で擬似キーが作成されます。

注意:表に使用不可のキーが他に存在する場合、または表にキーが存在しな い場合、Oracle GoldenGate によって適切なメッセージがレポート・ファイ ルに記録されます。すべての列からキーを作成すると、ソース・システムで 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 キーはより大きく効率的ではない WHERE 句が Replicat で使用される原因とな ります。

表に適切なキーがない場合、あるいは既存のキーを使用しない場合は、表に一意の値が常に含 まれる列があれば、代替キーを定義できます。Extract の TABLE パラメータと Replicat の MAP パ ラメータ内に KEYCOLS 句を含めることで、この代替キーを定義します。指定されたキーが Oracle GoldenGate で検出される既存の主キーまたは一意キーよりも優先されます。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。

5.2 ロギング・プロパティの構成

Oracle GoldenGate は、REDO ログを使用して、データおよびソース・トランザクションのレ プリケートに必要なメタデータをキャプチャします。Oracle GoldenGate の処理を開始する前 に、ソース・システムで Oracle REDO ログを適切に構成する必要があります。この必要なロギ ングによって REDO の量が増えるため、Oracle GoldenGate の処理を開始する直前までログの 有効化を待つ場合があります。

この項では、次のロギング・レベルについて説明します。

- データベースレベルのサプリメンタル・ロギングの有効化
- スキーマレベルのサプリメンタル・ロギングの有効化
- 表レベルのサプリメンタル・ロギングの有効化

5.2.1 データベースレベルのサプリメンタル・ロギングの有効化

Oracle ソース・データベースを強制ロギング・モードにすることをお薦めします。強制ロギン グ・モードにより、ユーザーまたは記憶域の設定がその反対にオーバーライドされ、すべての トランザクションおよびロードのロギングが強制されます。これにより、Extract 構成内のソー ス・データが失われることはありません。

さらに、Oracle GoldenGate を使用する際には、データベースレベルのオプションである最小 サプリメンタル・ロギングが Oracle ソース・データベースに必要です。これにより、行連鎖情 報が存在する場合は、その情報が更新操作の REDO ログに追加されます。

注意: データベースレベルの主キー (PK) および一意の索引 (UI) のロギング は、レプリケーション以外で表に対してオーバーヘッドが必要以上に追加で 作成されるため、使用しないでください。そのようなロギング・オプション がビジネス目的で必要な場合以外は、Oracle GoldenGate ついてデーター ベース・レベルで最小サプリメンタル・ロギングを有効にさえすれば十分で す。

次の手順を実行して最小サプリメンタル・ロギングおよび強制ロギングを確認し、必要に応じ て有効にします。

- 1. ALTER SYSTEM 権限を持つユーザーとして SQL*Plus にログインします。
- 次のコマンドを発行してデータベースがサプリメンタル・ロギング・モードおよび強制ロ ギング・モードにあるかどうかを確認します。いずれの問合せについても結果が YES の場 合、データベースは Oracle GoldenGate の要件を満たしています。

SELECT supplemental_log_data_min, force_logging FROM v\$database;

3. 一方または両方のプロパティについて結果が NO の場合、必要に応じて次の手順を続行して 有効にします。

SQL> ALTER DATABASE ADD SUPPLEMENTAL LOG DATA; SQL> ALTER DATABASE FORCE LOGGING;

4. 次のコマンドを発行して、これらのプロパティが有効になっていることを確認します。 SELECT supplemental_log_data_min, force_logging FROM v\$database;

問合せの出力は、両方のプロパティについて YES である必要があります。

5. ログ・ファイルを切り替えます。

SQL> ALTER SYSTEM SWITCH LOGFILE;

5.2.2 スキーマレベルのサプリメンタル・ロギングの有効化

Oracle GoldenGate では、スキーマレベルのサプリメンタル・ロギングがサポートされていま す。スキーマレベルのロギングは、Oracle GoldenGate DDL レプリケーション機能を使用する 場合、Oracle ソース・データベースに必要ですが、それ以外の場合は任意です。

スキーマレベルのロギングでは、スキーマ内の既存の各表と、その後の DDL 変更によって追加 される新規表すべてについて、主キーおよびすべての有効な一意の索引の値が自動的に記録さ れます。スキーマレベルのロギングのその他のロギング効果と、使用方法に関するその他の考 慮事項については、『Oracle GoldenGate Windows and UNIX リファレンス・ガイド』の ADD SCHEMATRANDATA コマンドに関する項を参照してください。

次の手順を実行して、スキーマレベルのサプリメンタル・ロギングを有効にします。

- 1. (Solaris システム)11.2.0.2 より前のバージョンの場合、ソース Oracle データベースに Oracle Patch 10423000 を適用します。
- **2.** ソース・システムで GGSCI を実行します。
- スキーマレベルのサプリメンタル・ロギングを有効にする権限を持つユーザーとして DBLOGIN コマンドを発行します。

DBLOGIN USERID user, PASSWORD password [encryption options]

DBLOGIN のパスワード暗号化オプションについては、『Oracle GoldenGate Windows and UNIX リファレンス・ガイド』を参照してください。

4. Oracle GoldenGate を使用してデータの変更をキャプチャする対象の各スキーマに対し、 ADD SCHEMATRANDATA コマンドを発行します。

ADD SCHEMATRANDATA schema

たとえば、次のコマンドでは、finance スキーマと hr スキーマのサプリメンタル・ロギン グを有効にします。

ADD SCHEMATRANDATA finance

ADD SCHEMATRANDATA hr

5.2.3 表レベルのサプリメンタル・ロギングの有効化

スキーマレベルのロギングを使用しない場合、主キー値または一意の索引値の表レベルのロギ ングが Oracle GoldenGate では必要です。

注意:表レベルのロギングよりもスキーマレベルのロギングを選択する前に、 スキーマレベルのロギングが DDL サポートに必要ない場合は、ログの量に対 するその影響について、『Windows and UNIX リファレンス・ガイド』の ADD SCHEMATRANDATAに関する項を参照してください。

スキーマレベルのロギングを使用する場合でも、表レベルのロギングのオプションを使用して 主キーのロギングを回避し、かわりに、Oracle GoldenGate で使用する代替キーを指定するこ ともできます。さらに、特定の Oracle GoldenGate 機能をサポートするために、表レベルで キー以外の列値を記録することが必要な場合があります。

ADD TRANDATA コマンドを使用して表レベルのサプリメンタル・ロギングのオプションを有効に します。

- 1. ソース・システムで GGSCI を実行します。
- 2. 表レベルのサプリメンタル・ロギングを有効にする権限を持つユーザーとして DBLOGIN コ マンドを発行します。

DBLOGIN USERID user, PASSWORD password [encryption options]

DBLOGIN のパスワード暗号化オプションについては、『Oracle GoldenGate Windows and UNIX リファレンス・ガイド』を参照してください。

3. ADD TRANDATA コマンドを発行します。

ADD TRANDATA table [, COLS (columns)] [, NOKEY]

説明:

- table は表の所有者と名前です。表名にワイルドカードを使用できますが、所有者名 には使用できません。
- オプション指定なしの ADD TRANDATA は、表に定義されている一意制約のタイプ、また は一意制約がない場合はすべての列に適した ADD SUPPLEMENTAL LOG DATA ALWAYS 句 を使用する ALTER TABLE コマンドを発行します。このコマンドは、スキーマレベルの ロギングを使用しない場合、Oracle GoldenGate の基本的な表レベルのロギング要件 を満たしています。
- COLS columns は、KEYCOLS 句またはフィルタリングおよび操作に必要なキー以外の列 を記録します。丸カッコは必須です。NOKEY オプションも指定されている場合を除き、 主キーに加えてこれらの列が記録されます。
- NOKEY によって、主キーまたは一意キーのロギングが回避されます。代替 KEYCOLS 列を 記録するには、TABLE パラメータおよび MAP パラメータで KEYCOLS 句を、ADD TRANDATA コマンドで COLS 句を使用する必要があります。

4. ADD TRANDATA で COLS オプションを使用する場合、ターゲットでそれらの列に一意の索引 を作成して行の取得を最適化します。KEYCOLS 句の代替キーとしてこれらの列を記録する 場合は、Oracle GoldenGate プロセスを構成するときに KEYCOLS 句を TABLE 文および MAP 文に追加する必要があります。

使用方法に関する考慮事項については、『Oracle GoldenGate Windows and UNIX リファレンス・ ガイド』の ADD TRANDATA コマンドに関する項を参照してください。

5.3 一意の行識別子がない表での行変更の制限

ターゲットの Oracle 表に主キーまたは一意なキーがない場合、重複する行が存在する可能性が あります。この場合、Oracle GoldenGate で、必要以上に多くのターゲット行が更新または削 除され、ソース・データとターゲット・データが非同期となる可能性がありますが、注意を喚 起するエラー・メッセージは表示されません。

更新される行数を制限するには、Replicat パラメータ・ファイルで DBOPTIONS パラメータに LIMITROWS オプションを使用します。LIMITROWS を使用すると、処理されるのは1行のみのた め、ターゲット・システムでの Oracle GoldenGate のパフォーマンスが向上することがありま

5.4 文字セットの変換のサポート

Windows ベースまたは UNIX ベースのソース・データベースから、別の文字セットを持つター ゲットの Oracle データベースにレプリケートする場合、Replicat は Oracle による文字セットの 変換を許可します。z/OS システムからレプリケートする場合、Replicat で変換を行います。

Oracle ターゲット・データベースによる文字セットの変換をサポートするには、ターゲット・ システムで NLS LANG 環境変数をソース・データベースの文字セットに設定する必要がありま す。システム・レベルで設定するかわりに、Replicat パラメータ SETENV を使用して、Replicat クライアント・セッションの NLS LANG を設定できます。パラメータ・ファイルから設定する場 合、システム・レベルで設定する場合より変更される可能性が低くなります。(注意: Oracle GoldenGate は、ソースでプログラムによって NLS LANG をソース・データベースの文字セット に設定します。)

5.4.1 SETENV を使用した NLS LANG の設定

次の手順では、Replicat パラメータ・ファイルから MLS LANG を設定します。

1. 次の構文を使用して、SETENVパラメータでNLS LANGを設定します。

SETENV (NLS_LANG = NLS_LANGUAGE_NLS_TERRITORY.NLS_CHARACTERSET)

UNIX プラットフォームでの例を次に示します。

SETENV (NLS LANG = "AMERICAN AMERICA.AL32UTF8")

2. Oracle GoldenGate の Manager プロセスを停止してから起動し、新しい変数が認識される ようにします。

5.4.2 グローバリゼーション設定の表示

データベースのグローバリゼーション設定とバイトまたは文字のセマンティクスの使用の有無 を決定するには、SQL*Plusで次のコマンドを使用します。

SHOW PARAMETER NLS LANGUAGE

SHOW PARAMETER NLS TERRITORY

SELECT name, value\$ from SYS.PROPS\$ WHERE name = 'NLS_CHARACTERSET';

SHOW PARAMETER NLS_LENGTH_SEMANTICS

GGSCIのVIEW REPORT コマンドによって、現在のデータベース言語と文字設定が表示され、 NLS LANGが設定されているかどうかが示されます。

5.4.3 グローバリゼーション・サポートのその他の情報

Oracle GoldenGate による文字セット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。

5.5 フェッチ・オプションの設定

特定の更新レコードを処理するために、Extract によってソース・データベースから追加の行 データがフェッチされます。Oracle GoldenGate でフェッチされるデータは次のとおりです。

- ユーザー定義型
- ネストされた表
- XMLType オブジェクト

デフォルトでは、Oracle GoldenGate でフラッシュバック問合せを使用して UNDO(ロール バック)表領域から値をフェッチします。そのように、Oracle GoldenGate では特定の時点また はSCN で読取り一貫性行イメージを再構築し、REDO レコードと一致させることができます。

最適なフェッチ結果を得るには、ソース・データベースを次のように構成します。

1. 次のように Oracle 初期化パラメータ UNDO MANAGEMENT および UNDO RETENTION を (秒単位 で)設定することで、REDO保存に十分な時間を設定します。

UNDO MANAGEMENT=AUTO UNDO RETENTION=86400

UNDO_RETENTION は、大規模な環境では上方に調整できます。

2. 次の式を使用して、UNDO表領域で必要な領域を計算します。

undo space = UNDO RETENTION * UPS + overhead

説明:

- undo spaceは、UNDOブロック数です。
- UNDO RETENTION は、UNDO RETENTION パラメータの値(秒単位)です。
- UPS は、1秒当たりの UNDO ブロック数です。
- overhead は、メタデータ (トランザクション表など) の最小オーバーヘッドです。 システム・ビュー V\$UNDOSTAT を使用して、UPS および overhead を見積ります。
- **3.** LOB が含まれている表には、次のいずれかを実行します。
 - LOB 記憶域句を RETENTION に設定します。これは、UNDO_MANAGEMENT を AUTO に設定し たときに作成される表のデフォルトです。

- RETENTION のかわりに PCTVERSION を使用する場合、PCTVERSION を初期値の 25 に設定 します。これは、STATS EXTRACT コマンドでレポートされるフェッチ統計に基づいて 調整できます(表 5-1 を参照)。 これらの統計の STAT OPER ROWFETCH CURRENTBYROWID フィールドまたはSTAT OPER ROWFETCH CURRENTBYKEYフィールドの値が高い場合、統 計で低い値を示すまで PCTVERSION を 10 ずつ増やします。
- **4.** 次の権限を Oracle GoldenGate の Extract ユーザーに付与します。

GRANT FLASHBACK ANY TABLE TO db_user

または

GRANT FLASHBACK ON owner.table TO db user

Oracle GoldenGate には、フェッチを管理するために次のパラメータが用意されています。

表 5-1 フェッチの管理用の Oracle GoldenGate パラメータとコマンド

パラメータまたはコマンド	説明
REPORTFETCH オプションを使用 した STATS EXTRACT	必要に応じて Extract フェッチ統計を表示します。
コマンド	
REPORTFETCH オプションを使用 した STATOPTIONS	常にフェッチ統計が表示されるように STATS EXTRACT コマンドを設定します。
パラメータ	
MAXFETCHSTATEMENTS パラメータ	Extract によってソース・データベースに保持される準備済の問合せおよび SQLEXEC 操作のオープン・カーソル数を制御します。
MAXFETCHSTATEMENTS パラメータ	Extract のデフォルトのフェッチ動作 (Extract でフラッシュバック問合せを実行するか、現在のイメージを表からフェッチするか)を制御します。
USELATESTVERSION または NOUSELATESTVERSION オプション を使用した FETCHOPTIONS	UNDO の保存期限が切れている場合や、表の構造が変更された場合など、Extract のフラッシュバック問合せの失敗を処理します。Extract は、現在のイメージを表からフェッチするか、失敗を無視するように設定できます。
パラメータ	
REPFETCHEDCOLOPTIONS パラメータ	フェッチされたデータが含まれる証跡レコード、または列が見つからない証跡レコードの処理時に、Replicat で応答を制御します。

5.6 特別なデータ型の処理

この項は、特に断りのないかぎり、Extract がクラシック・キャプチャ・モードで動作する場合 も統合キャプチャ・モードで動作する場合も該当します。次の Oracle データ型に対する特別な 構成要件について説明します。

- マルチバイト・キャラクタ型
- Oracle Spatial オブジェクト
- TIMESTAMP
- ラージ・オブジェクト(LOB)
- XML
- ユーザー定義型

5.6.1 マルチバイト・キャラクタ型

マルチバイト・キャラクタは、サポートされている文字セットの一部としてサポートされます。 Oracle ソース・データベースのセマンティクス設定が BYTE で、Oracle ターゲットの設定が CHAR の場合、構成で Replicat パラメータ SOURCEDEFS を使用し、DEFGEN ユーティリティに よって生成された定義ファイルをターゲットに配置します。これらの手順は、ソースおよび ターゲットのデータ定義が同一であるかどうかにかかわらず、セマンティクス内の差異のサ ポートに必要です。Replicatでは、定義ファイルを参照して固定サイズの文字列のサイズ上限 を決定します。

文字セット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。

SOURCEDEFS および DEFGEN ユーティリティの詳細は、『Oracle GoldenGate Windows and UNIX 管理者ガイド』を参照してください。

5.6.2 Oracle Spatial オブジェクト

SDO GEORASTER オブジェクト型の列が 1 つ以上含まれている表を Oracle ソースから Oracle ターゲットにレプリケートするには、それらが正しく処理されるよう、次の手順に従って Oracle GoldenGate を構成します。

- 1. GeoRaster 表とそれに関連するラスター・データ表に対する TABLE 文と MAP 文を作成しま
- 2. SDO GEORASTER データ型の METADATA 属性の値が 1MB を超える場合、XMLBUFSIZE オプショ ンを指定して DBOPTIONS パラメータを使用し、SDO GEORASTER データ型の埋込み SYS.XMLTYPE 属性を格納するメモリー・バッファのサイズを大きくします。バッファが小 さすぎる場合、Extract は異常終了します。XMLBUFSIZE の詳細は、『Oracle GoldenGate Windows and UNIX リファレンス・ガイド』の DBOPTIONS に関する項を参照してください。
- 3. ターゲットの GeoRaster 表と Spatial データの整合性を確保するには、トリガーをソースと ターゲットの両方で有効にしておきます。MAP パラメータの REPERROR オプションを使用し て、トリガーをターゲットで有効にしているために発生する「ORA-01403 データが見つか りません」エラーを処理します。ソースの GeoRaster 表の行を削除すると、トリガーに よって削除がラスター・データ表にカスケードされます。両方の削除がレプリケートされ ます。レプリケートされた親削除によって、ターゲット上でカスケードされた(子)削除が トリガーされます。レプリケートされた子削除が行われると、冗長になりエラーが発生し ます。REPERROR を使用するには、次のようにします。
 - ラスター・データ表が含まれている各 MAP 文で REPERROR 文を使用します。
 - Oracle エラー 1403 を SQL エラーとして使用します。
 - いずれかの応答オプションをエラー処理に使用します。

ターゲットの GeoRaster 表のアクティブなトリガーによって生じるラスター表のエラーを処理 する適切な方法は、DISCARD を指定して REPERROR を使用し、エラーをトリガーした、カスケー ドされた削除を破棄します。ターゲットの GeoRaster 表のトリガーによって、ラスター・デー タ表に対して削除が実行されるため、レプリケートされた削除は不要です。

MAP geo.st rdt, TARGET geo.st rdt, REPERROR (-1403, DISCARD);

エラー処理の監査証跡を保持する必要がある場合は、EXCEPTION を指定して REPERROR を使用 し、例外処理を呼び出します。この場合、例外表を作成して、次のようにソースのラスター・ ターゲット表に2回マップします。

- 1回目は実際のターゲットのラスター・データ表 (1403 エラーを処理する REPERROR を使用)。
- 2回目は例外表で、COLMAP 句を使用して 1403 エラーとその他の関連する情報をキャプチャ します。

例外表の使用の詳細は、『Oracle GoldenGate Windows and UNIX 管理者ガイド』を参照してく ださい。

REPERROR オプションの詳細は、『Oracle GoldenGate Windows and UNIX リファレンス・ガイ ド』を参照してください。

5.6.3 TIMESTAMP

タイムスタンプ・データをレプリケートするには、次のガイドラインに従います。

- 1. TZR で表された TIMESTAMP WITH TIME ZONE で Oracle GoldenGate が異常終了しないように するには、次のいずれかを指定して Extract パラメータ TRANLOGOPTIONS を使用します。
 - INCLUDEREGIONID: TZR で表された TIMESTAMP WITH TIME ZONE を、Oracle ソースから同 じバージョン以降の Oracle ターゲットにレプリケートします。
 - INCLUDEREGIONIDWITHOFFSET: TZR で表された TIMESTAMP WITH TIMEZONE を、バージョ ン 10g 以降の Oracle ソースから、それより前の Oracle ターゲットに、または Oracle ソースから Oracle 以外のターゲットにレプリケートします。

これらのオプションにより、TZR で表された TIMESTAMP WITH TIME ZONE をサポートしない Oracle バージョン、および UTC オフセットで表されたタイムゾーンのみをサポートする データベース・システムにレプリケートすることができます。詳細は、『Windows and *UNIX リファレンス・ガイド*』の TRANLOGOPTIONS に関する項を参照してください。

2. Oracle データベースで TIMESTAMP WITH LOCAL TIME ZONE データを受信先のデータベース のローカ・タイムゾーンに正規化するため、異なるタイムゾーンに存在するデータベース 間でタイムスタンプが正しく転送されません。たとえば、ニューヨーク市にあるソース・ サーバーの 5:00 a.m. EST を反映したデータは、サンフランシスコのターゲット・サーバー で 5:00 a.m. PST になります。本来は、タイムゾーン間の 3 時間の時差を反映して、サンフ ランシスコ時間で2:00 a.m. になるはずです。

Replicat によって適用されるすべてのレコードについて、タイムスタンプをターゲット・ タイムゾーンに適合させるには、Replicat パラメータ・ファイルで次のパラメータ文を USERID パラメータの後、最初の MAP 文の前に追加します。

SQLEXEC "ALTER SESSION SET TIME ZONE = 'value of source timezone'"

説明:value of source_timezoneは、ソース・タイムゾーンからのオフセットです。たと えば、前述の例を使用すると、値は'-03:00'です。ターゲットがソースの時間より進んでい るか遅れているかに応じて、プラス (+) またはマイナス (-) の記号を使用します。

5.6.4 ラージ・オブジェクト (LOB)

クラシックと統合の両方のキャプチャ・モードでの LOB に対する構成ガイドラインは、次のと おりです。

- 1. 可能な場合、行外にラージ・オブジェクトを保存します。
- 2. Replicat では、LOB データは断片的にターゲット・データベースに書き込まれます。シス テムでこの I/O の影響を最小限に抑えるために、Replicat では Oracle の LOB キャッシュ・ メカニズムが有効に設定され、バッファにフラグメントがキャッシュされて、バッファが いっぱいになると書込みのみが実行されます。たとえば、バッファのサイズが 25,000 バイ トの場合、LOB が 100,000 バイトであれば、Replicat では I/O が 4 回のみ実行されます。

- LOB データのサイズに合せてバッファ・サイズを最適化するには、LOBWRITESIZE オプ ションを指定して DBOPTIONS パラメータを使用します。値が大きいほど、1 つの LOB を書き込むための Replicat による I/O コールは少なくなります。
- Oracle の LOB キャッシュを無効にするには、DISABLELOBCACHING オプションを指定し て DBOPTIONS パラメータを使用します。 LOB キャッシュを無効にすると、Replicat に よって1つの I/O コールで Oracle に送信されるすべての内容がデータベースのメディ アに直接書き込まれます。
- 3. CLOB 列でバイナリ・データを保存できる場合、NLS LANG システム環境変数と NLS LANGUAGE データベース・パラメータを同じ値に設定します。
- 4. (統合キャプチャのみ該当)統合キャプチャでは、REDO ログから LOB がキャプチャされ ます。LOBドキュメントに対する UPDATE 操作では、LOB の変更された部分のみが記録さ れます。変更された部分のみが記録される場合に LOB ドキュメント全体の証跡への書込み を強制するには、Extract パラメータ・ファイルで FETCHPARTIALLOB オプションを指定して TRANLOGOPTIONS パラメータを使用します。 Extract は、ログマイニング・サーバーから部 分的な LOB コンテンツを受信すると、部分的な LOB を処理するのではなく、LOB イメー ジ全体をフェッチします。Oracle 以外のターゲットにレプリケートする場合、または LOB イメージ全体が必要なその他の場合にこのオプションを使用します。TRANLOGOPTIONS 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

5.6.5 XML

Oracle GoldenGate 制約内で XML を使用するためのツールは、次のとおりです。

- クラシックと統合の両キャプチャ・モードで、XML スキーマへの変更のキャプチャはサ ポートされませんが、スキーマを展開して、再同期をしなくてもそのレプリケーションを 再開できます。C-1ページの「XMLスキーマの変更のサポート」を参照してください。
- (統合キャプチャのみ該当)統合キャプチャでは、REDO ログから XML がキャプチャされま す。XML ドキュメントの UPDATE 操作では、OBJECT RELATIONAL または BINARY として格納 される場合、XMLの変更された部分のみが記録されます。変更された部分のみが記録され る場合に XML ドキュメント全体の証跡への書込みを強制するには、Extract パラメータ・ ファイルで FETCHPARTIALXML オプションを指定して TRANLOGOPTIONS パラメータを使用し ます。Extract は、ログマイニング・サーバーから部分的な XML コンテンツを受信すると、 部分的な XML を処理するのではなく、XML ドキュメント全体をフェッチします。Oracle 以外のターゲットにレプリケートする場合、または XML イメージ全体が必要なその他の 場合にこのオプションを使用します。TRANLOGOPTIONS 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。

5.6.6 ユーザー定義型

Extract は、UDT(オブジェクト表以外)をデータベースからフェッチします。詳細は、5-8ペー ジの「フェッチ・オプションの設定」を参照してください。

5.7 他のデータベース・プロパティの処理

次の表に、Oracle GoldenGate に影響する可能性のあるデータベース・プロパティと、その状 態の解決または回避に使用できるパラメータを示します。

表 5-2 他のデータベース・プロパティの処理

<i>デ</i> ータベース・プロパティ	問題点 / 解決策
未使用の列のある表	デフォルトでは、未使用の列はサポートされません。これをサポートするには、ALLOWUNUSEDCOLUMN オプションを指定して DBOPTIONS パラメータを使用し、警告を生成して処理を続行することを Extract に強制します。同じ未使用の列がターゲット表に存在するか、DEFGEN ユーティリティを使用して Replicat 用にソース定義ファイルを作成する必要があります。DDL レプリケーション構成に適切な ALTER TABLESET UNUSED 文を含めることができます。
時間隔パーティション化が ある表	時間隔パーティション化がある表をサポートするには、WILDCARDRESOLVE パラメータがデフォルトの DYNAMIC のままであることを確認します。
仮想列のある表	仮想列は記録されません。また、Oracle で仮想列に対する DML は許可されません。ただし、次のようにして、このデータをキャプチャし、仮想列ではないターゲット列にマップできます。
	Extract の TABLE 文に表を含め、TABLE の FETCHCOLS オプションを使用してデータベースの仮想列から値をフェッチします。
	Replicat の MAP 文で、ソースの仮想列を仮想ではないターゲット列にマップします。
本質的に更新可能なビュー を持つ表	本質的に更新可能なビューにレプリケートするには、関連付けの対象となるソース表とターゲット表をマップする同じ MAP 文に KEYCOLS 句を使用して、更新可能なビューの一意の列にキーを定義します。
異なる場所にある REDO ロ グまたはアーカイブ	TRANLOGOPTIONS パラメータには、データベースのデフォルトとは異なる場所または Extract が実行されているのとは異なるプラットフォームに REDO ログまたはアーカイブ が格納される環境を処理するためのオプションがあります。
TRUNCATE 操作	TRUNCATE 操作をレプリケートするには、次の2つのオプションのいずれかを選択します。
	GETTRUNCATES パラメータを使用したスタンドアロンの TRUNCATE サポートでは、TRUNCATE TABLE はレプリケートされますが、他の TRUNCATE オプションはレプリケートされません。
	完全 DDL サポートでは、TRUNCATE TABLE、ALTER TABLE TRUNCATE PARTITION および他の DDL がレプリケートされます。このサポートをインストールするには、3-1 ページの「Oracle データベース用 Oracle GoldenGate の DDL サポートのインストール」を参照してください。
順序	順序の DDL(CREATE、ALTER、DROP、RENAME) をレプリケートするには、Oracle GoldenGate DDL サポートを使用します。
	順序値のみをレプリケートするには、Extract パラメータ・ファイルで SEQUENCE パラメータを使用します。これには、Oracle GoldenGate DDL サポート環境は <i>不要</i> です。その他の要件については、『Oracle GoldenGate Windows and UNIX リファレンス・ガイド』のSEQUENCE に関する項を参照してください。

5.8 Oracle GoldenGate と Oracle Exadata を組み合せた使用

Oracle GoldenGate では、次のようにして Oracle Exadata Database Machine をサポートしま す。

- Oracle GoldenGate は、クラシック・キャプチャ・モードでも統合キャプチャ・モードでも Exadata からキャプチャできますが、EHCC 圧縮が有効な Exadata からキャプチャするに は、Extract を統合キャプチャ・モードにする必要があります。
- Oracle GoldenGate は、サポートされているどのデータベースからでもデータを Exadata に レプリケートできます。

一般に、Exadata とともに動作する Oracle GoldenGate の構成は、他の Oracle GoldenGate の 構成と同じです。次の説明では、標準の Oracle Golden Gate インストールとは異なる構成要件 を主に扱います。

5.8.1 Oracle Exadata への移行

Oracle GoldenGate では、SQL を介してアンロードおよびリロードが行われる Exadata への データの論理移行がサポートされます。他のデータベースから Exadata Database Machine に移 行するには、次のいずれかを行います。

- 初期ロードを使用した Exadata への移行
- アクティブ / パッシブ構成を使用した Exadata への移行

5.8.1.1 初期ロードを使用した Exadata への移行

元のデータベースから Exadata に移行する方法の1つは、一方のシステムから他方のシステム への単純な初期ロードを行うことです。これは、ソース表とターゲット表の一方向の初期同期 である一方、Oracle GoldenGate で、アクティブである必要のある進行中のトランザクション がキャプチャされます。Oracle GoldenGate を両方のシステムにインストールし、このガイド の構成、データベースの設定および初期ロードの手順に従います。初期ロード・オプションの 詳細は、『Oracle GoldenGate Windows and UNIX 管理者ガイド』を参照してください。

5.8.1.2 アクティブ / パッシブ構成を使用した Exadata への移行

アクティブ / パッシブの双方向構成を使用して Exadata に移行できます。アクティブ・データ ベースは、データの移行元のプライマリ・データベースで、パッシブ・データベースは Exadata マシンです。この構成では、テストが完了してオペレーションを Exadata マシンに切 り替えるまで、Exadata マシンは元のシステムと並行して動作することができます。

統合キャプチャを使用する場合、この構成をデプロイするために特別な設定は必要ありません。 ただし、クラシック・キャプチャ・モードを使用するには、Exadata マシンの Extract パラメー タ・ファイルに次の TRANLOGOPTIONS 文を含めます。

- DBLOGREADER を使用した TRANLOGOPTIONS: Oracle ASM インスタンスに接続するかわり に、Oracle データベース・サーバーを使用してログ・ファイルにアクセスし、読取りパ フォーマンスを向上させます。このオプションは、Oracle 10.2.0.5 および Oracle 11.1.0.7 以 降の 11g R2 の各バージョンについてサポートされます (Oracle 11g R1 の各バージョンにつ いてはサポートされません)。
- ASMUSER を使用した TRANLOGOPTIONS: ASM にログインしてログ・ファイルにアクセスしま す。このオプションでは、ソース・データベースが使用できなくなった場合にバックアッ プが提供され、Extract は ASM でのログ・ファイルの直接読取りを続けることができます。

これらは各々別の TRANLOGOPTIONS パラメータ文にします。ASMUSER は、データベース・サー バーに障害が発生して必要になるまでコメント・アウトします。これらのパラメータの構文と ASM の双方向性をチューニングするための他のオプションについては、『Oracle GoldenGate Windows and UNIX リファレンス・ガイド』の TRANLOGOPTIONS に関する項を参照してくださ

アクティブ / パッシブ構成を使用して移行を Exadata にデプロイするには、次のようにします。

- Oracle GoldenGate を両方のシステムにインストールし、このガイドの構成、データベース の設定および初期ロードの手順に従います。
- アクティブ / パッシブ・トポロジの構成の詳細は、『Oracle GoldenGate Windows and UNIX *管理者ガイド*』を参照してください。

5.8.2 EHCC が有効な Exadata へのレプリケート

Oracle Exadata with Hybrid Columnar Compression (EHCC) への挿入操作を確実に行うには、 Replicat パラメータ・ファイルで INSERTAPPEND パラメータを使用します。INSERTAPPEND を使 用すると、Replicat では挿入に APPEND ヒントが使用されるため、圧縮されたままになります。 このヒントがない場合、記録は圧縮されずに挿入されます。INSERTAPPEND 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。

クラシック・キャプチャを使用する場合の 追加構成手順

この章では、クラシック・キャプチャ・モードで動作する場合の Extract に固有の追加構成と準 備の要件について説明します。その他の準備処理も必要です。次の項を参照してください。

1-1 ページの「システム要件およびプレインストール手順」

2-1 ページの「Oracle GoldenGate のインストール」

3-1 ページの「Oracle データベース用 Oracle GoldenGate の DDL サポートのインストール」

5-1 ページの「Oracle GoldenGate のためのデータベースの準備」

4-1 ページの「ソースおよびターゲットの Oracle データベースでの Oracle GoldenGate の構成」

7-1 ページの「Oracle データベース用 DDL 同期の構成」

Oracle GoldenGate の処理をインスタンス化する場合、次の項を参照してください。

8-1 ページの「Oracle GoldenGate レプリケーションのインスタンス化と開始」

9-1 ページの「プロセスの制御」

次の項も該当する場合があります。

C-1 ページの「XML スキーマの変更のサポート」

D-1 ページの「Oracle GoldenGate とのアクティブ / アクティブ伝播のための DBFS の準備」 クラシック・キャプチャの詳細は、4-3ページの「使用するキャプチャ方法の決定」を参照して ください。

6.1 クラシック・キャプチャ・モードでの Oracle TDE データの 構成

クラシック・キャプチャ・モードでの Extract の場合、TDE をサポートするには、次の特別な 構成手順が必要です。

6.1.1 クラシック・キャプチャでの TDE サポートの概要

クラシック・キャプチャ・モードでの Extract の場合、TDE サポートには 2 種類の鍵の交換が 必要です。

- 暗号化鍵は、表キー(列レベルの暗号化)、暗号化された REDO ログ・キー(表領域レベル の暗号化)、またはその両方のいずれにすることも可能です。この鍵は Oracle データベー スと Extract 間で共有されます。
- 復号化鍵は、ORACLEGG という名前で、そのパスワードは共通鍵と呼ばれます。この鍵は、 Oracle および Oracle GoldenGate のドメインに安全に格納されます。共通鍵を所有する ユーザーのみが、表キーおよび REDO ログ・キーを復号化できます。

復号化鍵は、組込み PL/SQL コードによって Extract プロセスに配信されます。Extract では共 通鍵を使用してデータが復号化されます。Extract では、ウォレットのマスター鍵自体は処理さ れず、またマスター鍵のパスワードも認識されません。これらは、Oracle データベース・セ キュリティ・フレームワークの範囲内です。

Extract では、復号化されたデータが破棄ファイル (DISCARDFILE パラメータで指定)を含む証 跡ファイル以外のファイルに書き込まれることはありません。"ENCRYPTED"という語が使用中の 破棄ファイルに書き込まれます。

Oracle GoldenGate のパフォーマンスに対するこの機能の影響には、データベース・パフォー マンスに対する復号化の影響が反映されます。Extract の起動時間がわずかに長くなる以外は、 TDE データのレプリケートによるパフォーマンスへの影響は最小限です。

6.1.2 クラシック・キャプチャ・モードでの TDE のキャプチャの要件

Extract で TDE キャプチャをサポートするための要件は次のとおりです。

- 高いセキュリティ基準を維持するため、Oracle GoldenGate Extract プロセスは oracle ユー ザー(Oracle データベースを実行するユーザー)の一部として実行する必要があります。こ れにより、鍵は oracle ユーザーと同じ権限でメモリー内で保護されます。
- Extract プロセスは、データベースのインストールと同じマシン上で実行される必要があり ます。
- TDE をハードウェア・セキュリティ・モジュールとともに使用する場合は、ソフトウェ ア・ウォレットを使用する必要があります。詳細は、6.1.4.3 項で HSM のみの構成から HSM とウォレットの構成に移行し、sqlnet.ora ファイルを正しく構成するための構成手 順を参照してください。

6.1.3 必要なデータベースのパッチ

Oracle 10.2.0.5 または 11.2.0.2 で TDE をサポートするには、Oracle Patch 10395645 をダウン ロードし、ソース・データベースに適用します。Oracle 11.2.0.3 パッチ・セットには、このパッ チが含まれています。My Oracle Support の Web サイト (https://support.oracle.com) でこのパッチが見つからない場合、サービス・リクエスト (SR) を送信してバックポートをリク エストします。

6.1.4 TDE サポートの構成

次に、Oracle サーバーと Extract プロセス間の通信を確立するために Oracle セキュリティ担当 者および Oracle GoldenGate 管理者が実行する手順を説明します。

6.1.4.1 Oracle の基準を満たす共通鍵の共有

Oracle パスワードの基準以上の共通鍵(パスワード)を共有します。このパスワードは他の ユーザーに知られないようにする必要があります。セキュアなパスワードの作成のガイドライ ンについては、『Oracle Database セキュリティ・ガイド』を参照してください。

6.1.4.2 Oracle DBA のタスク

1. SYSDBA システム権限を持つユーザーとして SQL*Plus にログインします。例を次に示しま す。

sqlplus sys/as sysdba

Connected.

Enter password: password

2. Oracle admin ディレクトリにインストールされている prvtclkm.plb ファイルを実行しま す。prvtclkm.plbファイルによって DBMS INTERNAL CLKM PL/SQL パッケージが作成され、 暗号化されたデータについて Oracle GoldenGate による Oracle データベースからの抽出が 可能になります。

@?/app/oracle/product/orcl111/rdbms/admin/prvtclkm.plb

3. DBMS INTERNAL CLKM PL/SQL パッケージに対する EXEC 権限を Extract のデータベース・ ユーザーに付与します。

GRANT EXECUTE ON DBMS INTERNAL CLKM TO psmith;

4. SQL*Plus を終了します。

6.1.4.3 Oracle セキュリティ担当者のタスク

- 1. Oracle GoldenGate では、ソフトウェア・ウォレットを使用する必要があります。HSM の みのモードを現在使用している場合は、次の手順を実行して、HSM とウォレットのモード に移行します。
 - a. 次の例に示すように sqlnet.ora ファイル構成を変更します (ウォレット・ディレクト リは、Oracle データベースの所有者がアクセスできる (rwx) ディスク上のどの場所で もかまいません)。この例では、ベスト・プラクティスの場所を示します (my db は \$ORACLE SID です)。

ENCRYPTION WALLET LOCATION= (SOURCE= (METHOD=HSM) (METHOD DATA= (DIRECTORY=/etc/oracle/wallets/my db)))

b. Oracle データベースの所有者として orapki (Wallet Manager) にログインし、 sqlnet.ora ファイルで指定した場所にウォレットを作成します。ウォレットのパス ワードの入力を求められたら、HSM パスワードと同じパスワード (または HSM 接続 **文字列)**を指定します。この 2 つのパスワードは同一である必要があります。

cd /etc/oracle/wallets/my db orapki wallet create -wallet /etc/oracle/wallets/my db

注意: Oracle データベース所有者には、ウォレットに対する全オペレーティ ング・システム権限が必要です。

2. ORACLEGG という名前のエントリをウォレットに作成します。ORACLEGG をこの鍵の名前に する必要があります。この鍵のパスワードは、共有している共通鍵である必要があります が、このパスワードをコマンドラインに入力しないでください。かわりに、入力を求めら れるまで待ちます。

mkstore -wrl ./ -createEntry ORACLE.SECURITY.CL.ENCRYPTION.ORACLEGG Oracle Secret Store Tool: Version 11.2.0.3.0 - Production Copyright (c) 2004, 2011, Oracle and/or its affiliates.All rights reserved. Your secret/Password is missing in the command line Enter your secret/Password: sharedsecret Re-enter your secret/Password: sharedsecret Enter wallet password: hsm/wallet_password

3. ORACLEGG エントリを確認します。

mkstore -wrl .-list Oracle Secret Store Tool: Version 11.2.0.3.0 - Production Copyright (c) 2004, 2011, Oracle and/or its affiliates. All rights reserved. Enter wallet password: hsm/wallet password Oracle Secret Store entries: ORACLE.SECURITY.CL.ENCRYPTION.ORACLEGG

4. SYSDBA システム権限を持つユーザーとして SQL*Plus にログインします。

5. ウォレットを閉じ、再度開きます。

SQL> alter system set encryption wallet close identified by "hsm/wallet password"; System altered.

SQL> alter system set encryption wallet open identified by "hsm/wallet_password"; System altered.

6. ログ・ファイルを切り替えます。

alter system switch logfile; System altered.

7. Oracle RAC 環境で、各ノードでウォレットのコピーを使用している場合、ここでコピーを 作成し、各ウォレットを再度開きます。

注意: 共有の場所にある1つのウォレットを、すべてのOracle RACノード 間でアクセスが同期化された状態で使用することをお薦めします。

6.1.4.4 Oracle GoldenGate 管理者のタスク

- 1. GGSCI を実行します。
- 2. ENCRYPT PASSWORD コマンドを発行して、Extract パラメータ・ファイル内で不明瞭化され るように共通鍵を暗号化します。 これはセキュリティ要件です。

ENCRYPT PASSWORD sharedsecret {AES128 | AES192 | AES256} ENCRYPTKEY keyname

説明:

- sharedsecret は、クリアテキストの共通鍵です。この値では大文字 / 小文字が区別さ れます。
- {AES128 | AES192 | AES256} では、Advanced Encryption Standard (AES) 暗号化を指 定します。使用する鍵の長さを表す、いずれかの値を指定します。
- keyname は、ENCKEYS 参照ファイル内の暗号化鍵の論理名です。 Oracle GoldenGate で は、この鍵を使用して、ENCKEYS ファイルで実際の鍵を検索します。鍵および ENCKEYS ファイルを作成する場合、『Oracle GoldenGate Windows and UNIX 管理者ガイド』を 参照してください。

例:

ENCRYPT PASSWORD sharedsecret AES256 ENCRYPTKEY mvkev1

3. Extract パラメータ・ファイルで、DECRYPTPASSWORD オプションを指定して DBOPTIONS パ ラメータを使用します。入力値として、暗号化された共通鍵と復号化鍵を指定します。

DBOPTIONS DECRYPTPASSWORD sharedsecret {AES128 | AES192 | AES256} ENCRYPTKEY keyname

説明:

- sharedsecret は、暗号化された共通鍵です。
- {AES128 | AES192 | AES256} は、ENCRYPT PASSWORD に使用したものと同じ値である 必要があります。
- keyname は、ENCKEYS 参照ファイル内の暗号化鍵の論理名です。

DBOPTIONS DECRYPTPASSWORD AACAAAAAAAAAAAIALCKDZIRHOJBHOJUH AES256 ENCRYPTKEY mykey1

- 4. SYSDBA システム権限を持つユーザーとして SQL*Plus にログインします。
- 5. ウォレットを閉じ、再度開きます。

SQL> alter system set encryption wallet close identified by "hsm/wallet password"; System altered.

SQL> alter system set encryption wallet open identified by "hsm/wallet password"; System altered.

6.1.5 復号化後のデータのセキュリティを維持するための推奨事項

Extract では、TDE データは復号化され、クリア・テキストとして証跡内に書き込まれます。 ターゲット・データベースへの経路全体でデータのセキュリティを維持するには、次の Oracle GoldenGate のセキュリティ機能もデプロイすることをお薦めします。

- 証跡内のデータの暗号化
- TCP/IP 間の遷移でのデータの暗号化

詳細は、『Windows and UNIX 管理者ガイド』のセキュリティに関する章を参照してください。

6.1.6 TDE キャプチャがアクティブな場合の DDL の実行

列レベルの暗号化が存在する表で DDL が実行される場合や表キーが更新される場合には、 DDL の実行中に表を停止するか、Oracle GoldenGate DDL サポートを有効にする必要がありま す。通常、キーの更新はセキュリティ違反に伴って行われ、すぐに実行される必要があるため、 いつでも対応できるよう、DDL 環境をアクティブにしておくとより実用的です。Oracle GoldenGate DDL 環境をインストールするには、このガイド内の手順を参照してください。 Oracle GoldenGate DDL サポートを構成するには、『Windows and UNIX 管理者ガイト』を参照 してください。表領域レベルの暗号化の場合は、Oracle GoldenGate DDL サポートは必要あり ません。

6.1.7 パラメータ・ファイルでの Oracle 共通鍵の更新

次の手順を使用して、Extract パラメータ・ファイル内で TDE 共通鍵を更新し、暗号化します。

- 1. GGSCI を実行します。
- 2. Extract プロセスを停止します。

STOP EXTRACT group

- 3. Oracle ウォレット内の ORACLEGG エントリを変更します。鍵の名前は、ORACLEGG のままと する必要があります。詳細は、『Oracle Database Advanced Security 管理者ガイド』を参照し てください。
- 4. ENCRYPT PASSWORD コマンドを発行して新しい共通鍵を暗号化します。

ENCRYPT PASSWORD sharedsecret {AES128 | AES192 | AES256} ENCRYPTKEY keyname

説明:

- sharedsecret は、クリアテキストの共通鍵です。この値では大文字 / 小文字が区別さ れます。
- {AES128 | AES192 | AES256} では、Advanced Encryption Standard (AES) 暗号化を指 定します。使用する鍵の長さを表す、いずれかの値を指定します。

keyname は、ENCKEYS 参照ファイル内の暗号化鍵の論理名です。 例:

ENCRYPT PASSWORD sharedsecret AES256 ENCRYPTKEY mykey1

5. Extract パラメータ・ファイルで、DECRYPTPASSWORD オプションを指定して DBOPTIONS パ ラメータを使用します。入力値として、暗号化された共通鍵と、Oracle GoldenGate に よって生成される復号化鍵またはユーザー定義の復号化鍵を指定します。

DBOPTIONS DECRYPTPASSWORD sharedsecret {AES128 | AES192 | AES256} ENCRYPTKEY keyname

説明:

- sharedsecret は、暗号化された共通鍵です。
- {AES128 | AES192 | AES256} は、ENCRYPT PASSWORD に使用したものと同じ値である 必要があります。
- keyname は、ENCKEYS 参照ファイル内の暗号化鍵の論理名です。

DBOPTIONS DECRYPTPASSWORD AACAAAAAAAAAAAIALCKDZIRHOJBHOJUH AES256 ENCRYPTKEY mykey1

- 6. SYSDBA システム権限を持つユーザーとして SQL*Plus にログインします。
- 7. ウォレットを閉じ、再度開きます。

SQL> alter system set encryption wallet close identified by "hsm/wallet password"; System altered.

SQL> alter system set encryption wallet open identified by "hsm/wallet password"; System altered.

8. Extract を開始します。

START EXTRACT group

6.2 Oracle RAC 環境での Oracle GoldenGate の使用

次の一般的なガイドラインは、クラシック・キャプチャ・モードで Extract が実行される場合の Oracle RAC に適用されます。

- 操作中になんらかの理由で Oracle GoldenGate を実行しているプライマリ・データベース・ インスタンスが停止または失敗すると、Extract は異常終了します。処理を再開するには、 インスタンスを再起動するか、Oracle GoldenGate のバイナリをデータベースを実行して いる別のノードにマウントして Oracle GoldenGate プロセスを再開します。別のノードか ら Oracle GoldenGate プロセスを開始する前に、元のノードで Manager プロセスを停止し
- REDO スレッド数は常に変更されるため、Extract グループをドロップおよび再作成する必 要があります。推奨される手順については、『Oracle GoldenGate Windows and UNIX 管理 者ガイド』を参照してください。
- トランザクションの発生元の RAC インスタンスに関係なく、Extract は必ずトランザク ションをコミット順に証跡ファイルに書き込みます。1 つ以上の RAC インスタンスがアイ ドルの可能性がある状態で Extract がアーカイブ・ログのみ (ALO) モードでキャプチャす る場合、アイドル・ノードでアーカイブ・ログ・スイッチを実行して、アクティブ・イン スタンスの操作が証跡ファイルにタイムリーに記録されるようにする必要があります。 archive_lag_target パラメータを設定して、事前に設定された間隔でこのログ・アーカイ ブを自動的に行うよう Oracle RDBMS に指定することができます。たとえば、ログが 15 分

間隔でアーカイブされるようにするには、アクティビティに関係なく、RAC システムのす べてのインスタンスで次のコマンドを発行します。

SQL> alter system set archive lag target 900

Extract を停止する前に RAC クラスタで最後のトランザクションを処理するには、ダミー・ レコードを Oracle Golden Gate によってレプリケートされるソース表に挿入してから、す べてのノード上でログ・ファイルを切り替えます。これにより、Extract のチェックポイン トが更新され、使用可能なすべてのアーカイブ・ログの読取りが可能であることが確認さ れます。さらに、それによってそれらのアーカイブ・ログのすべてのトランザクションが キャプチャされ、正しい順序で証跡に書き込まれることが確認されます。

次の表に、Oracle GoldenGate のパラメータの中で、Oracle RAC で特に便利なパラメータを示 します。

表 6-1 Oracle RAC 向けの Oracle GoldenGate パラメータ

パラメータ	説明
INQUEUESIZE オプションおよび OUTQUEUESIZE オプションを指定した THREADOPTIONS パラメータ	ターゲット・システムに送信する前に Extract によってメモリーのキューに入れられるデータの量を設定します。これらのパラメータを調整することによって、Oracle RAC で Extract のパフォーマンスを向上できる場合があります。
PURGEORPHANEDTRANSACTIONS NOPURGEORPHANEDTRANSACTIONS オプションおよび TRANSCLEANUPFREQUENCY オプションを指定した TRANLOGOPTIONS パラメータ	トランザクション時にノードが失敗して Extract でロールバックをキャプチャできない場合に発生する可能性がある親のないトランザクションを、Extractが処理する方法を制御します。データベースによってフェイルオーバー・ノードでロールバックが実行されますが、それ以外の場合、トランザクションはExtractトランザクション・リストに無限に残ったままになり、トランザクションを処理していた Extract スレッドに追加のチェックポイントが実行されません。デフォルトでは、このようなトランザクションが親なしであると確認されると、Oracle GoldenGate によってリストからパージされます。この機能は、GGSCIでの SEND EXTRACT コマンドを使用して必要に応じて制御することもできます。

6.3 クラシック・キャプチャ・モードの場合の Oracle ASM イン スタンスからのキャプチャ

この項では、Oracle 自動ストレージ管理 (ASM) に格納されているトランザクション・ログに対 して Oracle GoldenGate で操作する場合に適用される追加構成要件について説明します。 Oracle GoldenGate では、ASM インスタンスへの接続でログを読み取れる必要があります。

6.3.1 ASM 内のトランザクション・ログへのアクセス

Extract は、ASM に格納されているログを読み取るよう構成されている必要があります。ASM のバージョンに応じて、次のオプションを使用できます。

6.3.1.1 データベース・サーバーからの最適な ASM 接続

ASM インスタンスが次のバージョンのいずれかである場合、Extract パラメータ・ファイルで DBLOGREADER オプションを指定して TRANLOGOPTIONS パラメータを使用します。

- Oracle 10.2.0.5 以降の 10g R2 バージョン
- Oracle 11.2.0.2 以降の 11g R2 バージョン

これらのリリース (ただし Oracle 11g R1 バージョンではない場合)には、データベース・サー バーを使用して REDO およびアーカイブ・ログにアクセスする新しい ASM API が存在します。 使用した場合、この API によって Extract で最大 4MB の読込みバッファ・サイズが使用できる ようになります。バッファを大きくすると、REDO 率が高い場合に Extract のパフォーマンス

が向上する可能性があります。TRANLOGOPTIONS の DBLOGREADERBUFSIZE オプションを使用して バッファ・サイズを指定できます。

6.3.1.2 ASM 直接接続

ASM のバージョンが 6.3.1.1 項に示されているものではない場合、次のようにします。

1. ASM インスタンスに直接アクセスする Extract プロセスのユーザーを作成します。ASM イ ンスタンスでこのユーザーに SYS 権限または SYSDBA 権限を割り当てます。Oracle GoldenGate では、ASM ユーザーに対するオペレーティング・システム認証の使用はサ ポートされません。詳細は、表 6-2 を参照してください。

表 6-2 Extract データベースの権限: ASM インスタンス

ASM パスワードの構成 ¹	許可されたユーザー
ASM インスタンスとデータベースでパスワード・ファイルを共有する	Oracle GoldenGate ソース・データベース・ユーザーに SYSDBA を付与した場合はそのユーザーを使用することもできますが、SYSDBA 権限を持つその他のデータベース・ユーザーを使用することもできます。
ASM インスタンスとソース・データベースで 別々のパスワード・ファイルを設定する	ASM パスワード・ファイルをソース・データベースのパスワード・ファイルで上書きできます。この手順によって、ASM インスタンスのSYS パスワードがデータベースのパスワード・ファイルに含まれている値に変わり、ASM アクセス権限がデータベースのパスワード・ファイルの他のユーザーに付与されます。それを上書きする前に ASM ファイルのコピーを保存します。

現在の ASM パスワード・ファイルの構成方法を表示するには、ASM インスタンスにログオンして SQL*Plus で次のコマンドを発行します。 SQL> SELECT name, value FROM v\$parameter WHERE name = 'remote login passwordfile';

2. TRANLOGOPTIONS パラメータで ASMUSER オプションを使用して ASM ユーザーを指定しま

構文、詳細および関連パラメータについては、『Oracle GoldenGate Windows and UNIX リファ レンス・ガイド』の TRANLOGOPTIONS に関する項を参照してください。

6.3.2 ASM 接続の確認

Oracle GoldenGate Extract プロセスが確実に ASM インスタンスに接続できるようにするには、 tnsnames.ora ファイルに ASM インスタンスをリストします。Oracle GoldenGate がデータ ベースのホスト・マシンで実行されている場合には、ASM インスタンスへの接続に、Bequeath (BEQ) プロトコルを使用することをお薦めします。BEQ プロトコルにはリスナーは必要ありま せん。TCP/IP プロトコルを使用する場合は、Oracle リスナーが ASM インスタンスへの新しい 接続をリスニングしていることを確認します。listener.ora ファイルに、次のようなエントリ が含まれている必要があります。

```
SID LIST LISTENER ASM =
  (SID LIST =
    (SID DESC =
      (GLOBAL DBNAME = ASM)
      (ORACLE HOME = /u01/app/grid)
    (SID NAME = +ASM1)
 )
```

注意: リモート Extract 構成を使用する場合、BEQ 接続は機能しません。 TNSNAMES と TCP/IP プロトコルを使用します。

6.4 データの可用性の確認

Extract がクラシック・キャプチャ・モードで動作する場合、キャプチャ処理の継続性および整 合性を保証するには、アーカイブ・ロギングを有効にします。Extract が処理を終了する前にオ ンライン・ログが再利用される場合、アーカイブ・ログがセカンダリ・データ・ソースになり ます。オープン・トランザクションのアーカイブ・ログは、Extract でリカバリを実行するため にこれらからのデータの再キャプチャが必要となる場合に備え、システム上に保存する必要が あります。

警告: アーカイブ・ロギングを有効にできない場合、トランザクションがま だアクティブな間に Extract の停止につながる障害が発生した場合、ソースと ターゲットのオブジェクトを完全に再同期し、レプリケーションを再インス タンス化する必要があるというリスクを伴います。この方法で操作する必要 がある場合、次のガイドラインに従ってオンライン・ログを構成し、オンラ イン・ログが再利用される前に Extract でキャプチャする必要のあるデータが 保持されるようにします。長時間のトランザクションに加え、ネットワーク の機能停止とその他の外部要因によって発生する Extract のバックログも考慮 してください。

RAC 構成では、Oracle GoldenGate がインストールされているノードを含む、クラスタ内のす べてのノードのオンラインおよびアーカイブのログに Extract がアクセスできるようにしておく 必要があります。

6.4.1 Extract リカバリ・モード別のログ保存要件

Extract で使用される可能性がある様々なリカバリ・モード、およびそれぞれのログ保存要件に ついて次に要約します。

- デフォルトでは制限付きリカバリ・モードが有効で、Extract では BR パラメータを使用して 設定されている制限付きリカバリ期間の2倍までさかのぼってログにアクセスする必要が あります。この期間は、CHECKPOINTSECS パラメータによって制御されている、Extract の 標準のチェックポイント間隔の整数倍です。これらの2つのパラメータによって Oracle GoldenGate の制限付きリカバリ機能が制御され、障害が発生したときにオープンしていた 最も古いトランザクションがどんなに古くても、Extract では障害後にメモリー内のキャプ チャされたデータのリカバリが可能です。この要件の詳細は、『Oracle GoldenGate Windows and UNIX リファレンス・ガイド』の BR パラメータに関する項を参照してくださ
- Extract でリカバリを試みたときに制限付きリカバリ・メカニズムが失敗した場合、Extract は通常のリカバリ・モードに戻ります。この場合、失敗時にオープンしていたメモリー内 で最も古いトランザクションの始まりが含まれているアーカイブ・ログと、それ以降のす べてのログへのアクセス権を Extract に設定しておく必要があります。

6.4.2 ログ保存のオプション

必要なログがシステム上に保持されるようにするためのオプションは、Oracle のバージョンに よって異なります。

6.4.2.1 Oracle Enterprise Edition 10.2 以降

これらのバージョンの場合、Oracle Recovery Manager (RMAN) と連動して、Extract でリカバ リに必要となるログが保存されるよう Extract を構成できます。REGISTER EXTRACT コマンドを 発行する場合、Extract プロセスを作成する前にこの機能を有効にします (4-10 ページの「変更

のキャプチャ用の Extract の構成」を参照してください)。

この機能を使用するには、Extract データベース・ユーザーに、4-5 ページの「Oracle GoldenGate のデータベース・ユーザーの割当て」に示す基本的な権限以外に次の権限が必要で す。

表 6-3 Extract データベースの権限: Oracle EE 10.2 以降のログ保存

Oracle EE のパージョン	限
10.2	. パッケージを実行して Oracle Streams の管理権限を付与します。
	<pre>exec dbms_streams_auth.grant_admin_privilege('user')</pre>
	. logmnr_restart_ckpt\$ への INSERT を付与します。
	grant insert on system.logmnr_restart_ckpt\$ to user;
	. streams\$_capture_process に対する UPDATE を付与します。
	grant update on sys.streams\$_capture_process to user;
	. 'become user'権限を付与します。
	grant become user to user;
11.1 および 11.2.0.1	. パッケージを実行して Oracle Streams の管理権限を付与します。
	exec dbms_streams_auth.grant_admin_privilege('user')
	. 'become user'権限を付与します。
	grant become user to user;
11.2.0.3 以降	ペッケージを実行して Oracle Streams の管理権限を付与します。
	xec dbms_goldengate_auth.grant_admin_privilege('user')

ログ保存が有効な場合、Extract は制限付きリカバリを実行するのに十分なログを保持します が、LOGRETENTION オプションを SR に設定した TRANLOGOPTIONS パラメータを使用して、通常の リカバリに十分なログを RMAN を介して保持するよう Extract を構成できます。RMAN ログ 保存の使用を無効にするオプションもあります。Extract を構成する前に『Oracle GoldenGate Windows and UNIX リカバリ・ガイド』に記載されている LOGRETENTION のオプションを確認し てください。LOGRETENTION を DISABLED に設定する場合は、6-11 ページの「保存するデータ量 の判断」を参照してください。

注意: Oracle RAC で RMAN ログ保存をサポートするには、Extract グルー プを追加する前に BUGFIX 11879974 で提供されているデータベース・パッチ をダウンロードおよびインストールする必要があります。

RMAN ログ保存機能により、各 Extract グループに基本の (ただし機能しない)Oracle Streams の取得プロセスが作成されます。取得の名前は、関連する Extract グループの名前に基づきま す。ログ保存機能は、他のローカルの Oracle Streams インストールと同時に動作可能です。 Extract グループを作成するとき、ログは現在のデータベース SCN から保存されます。

注意: Oracle Flashback 記憶域がいっぱいの場合は、Extract で必要とされて も RMAN によってアーカイブ・ログがパージされます。この制限は、 Extract(および他の Oracle レプリケーション・コンポーネント)の要件がそ のデータベースへの REDO の可用性を妨げないようにするために存在しま す。

Oracle GoldenGate のキャプチャと Oracle データベース・エンジンをさらに統合するには、 ソース・データベースが Oracle 11.2.0.3 以降の場合、統合キャプチャを使用します。統合キャ プチャ・モードでは、ログ保存は自動的に有効になり、Extract は、REDO ログを直接読み取る のではなく、データ変更をデータベース・ログマイニング・サーバーから直接受信します。4-4 ページの「統合キャプチャについて」を参照してください。

6.4.2.2 その他の Oracle バージョン

Enterprise Edition 10.2 以降以外のバージョンの Oracle の場合は、任意の管理ツールを使用してログ保存プロセスを管理する必要があります。6-11 ページの「保存するデータ量の判断」の説明に従ってください。

6.4.3 保存するデータ量の判断

ログ保存を管理する場合、Extract で通常のリカバリ (制限付きリカバリではなく)を実行するのに必要となるログに迅速にアクセスできるようにしてください。6-9ページの「Extract リカバリ・モード別のログ保存要件」を参照してください。アーカイブをデータベース・システムから移動する必要がある場合、TRANLOGOPTIONSパラメータで別の場所を指定することができます。6-11ページの「アーカイブ場所の指定」を参照してください。

推奨される保存期間は少なくとも 24 時間であり、これはオンラインおよびアーカイブのログを含むトランザクション・データを保存できる時間に相当します。 Extract が任意の時点で必要とする可能性がある最も古いログを判断するには、SHOWTRANS オプションを使用して SEND EXTRACT コマンドを発行します。データ量とビジネス要件を考慮して最適な保存時間を決定するためにテストを実行しなければならない場合があります。

処理時に Extract で必要なデータがオンラインまたはアーカイブのログに保存されなかった場合、次のいずれかの修正処理が必要になる可能性があります。

- Extract を変更して、ログ・データが使用できる後の時点からキャプチャする(さらにターゲットで考えられるデータ損失を受け入れる)。
- ソースおよびターゲットのデータを再同期してから、Oracle GoldenGate 環境の起動をさらに繰り返す。

6.4.4 ログ・アーカイブのパージ

古いアーカイブ・ファイルが新しいバックアップで上書きされる原因となるバックアップグレードまたはアーカイブのオプションを使用していないことを確認します。理想的には、新しいバックアップは、古い名前とは異なる名前の別のファイルにしてください。これによって、Extractで特定のログを探す場合にそれがすでに存在し、さらにサポート事例に必要な場合にデータが使用できるようになります。

6.4.5 アーカイブ場所の指定

アーカイブ・ログが Oracle のデフォルト・ディレクトリ以外の場所にある場合は、Extract パラメータ・ファイルで TRANLOGOPTIONS パラメータの ALTARCHIVELOGDEST オプションを使用してそのディレクトリを指定します。

Oracle パラメータ LOG_ARCHIVE_FORMAT で指定されている形式にサブディレクトリが含まれている場合、TRANLOGOPTIONS の ALTARCHIVEDLOGFORMAT オプションの使用が必要になることもあります。ALTARCHIVEDLOGFORMAT によって、サブディレクトリをパスから削除する代替形式が指定されます。たとえば、%T/log_%t_%s_%r.arc は log_%t_%s_%r.arc に変更されます。ALTARCHIVEDLOGFORMAT を使用するかわりに、手動でサブディレクトリを作成してから、ログ・ファイルをそこに移動することもできます。

6.4.6 その他のプラットフォームに保存されるログのマウント

Extract が作成されるプラットフォームとは異なるプラットフォームに、オンラインおよびアーカイブの REDO ログを保存した場合、次の操作を実行します。

- アーカイブ・ファイルを NFS マウントします。
- Extract パラメータ TRANLOGOPTIONS の LOGSOURCE オプションおよび PATHMAP オプションを使用することで、ファイル構造をソース・システムの構造にマップします。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。

6.5 アーカイブ・ログのみを読み取るための Oracle GoldenGate の構成

Extract は、アーカイブ・ログからのみ読取りを行うよう構成できます。これは、アーカイブ・ ログのみ(ALO) モードと呼ばれます。このモードでは、Extract によって、指定された場所に 保存されているアーカイブ・ログからのみ読み取られます。ALO モードを使用すると、Extract ではデータ・ソースとしてセカンダリ・データベース (スタンバイなど) に転送される本番ログ を使用できます。オンライン・ログは使用されません。Oracle GoldenGate では、必要に応じ て、セカンダリ・データベースに接続してメタデータとその他の必要なデータを取得します。 代替として ALO モードが本番システムでサポートされています。

注意: ALO モードは、統合キャプチャ・モードで動作する Extract と互換性 はありません。

6.5.1 ALO モードの制限および要件

Extract を ALO モードで使用する場合、次のガイドラインに従います。

- スタンバイ・データベースの作成後にソース・データベースでログのリセット (RESETLOG) を実行することはできません。
- 本番システムが Oracle RAC であり、スタンバイ・データベースが RAC 以外である場合、 スタンバイ・データベースで ALO を使用することはできません。両方のシステムを Oracle RAC にすることに加え、各システムのノード数も同一にする必要があります。
- Oracle RAC 上の ALO では、ソース・サーバーへの専用接続が必要です。その接続が失わ れると、Oracle GoldenGate の処理が停止します。
- Oracle RAC では、アーカイブ・ログを格納するディレクトリの名前をすべてのノード間で 一意にする必要があり、そうでない場合には、Extract で SCN が不正であるというエラー が返されます。
- ALO モードでは、ASM モードのアーカイブ・ログ・ファイルはサポートされていません。 Extract での読取りを可能にするには、アーカイブ・ログ・ファイルを ASM 環境外に置く 必要があります。
- Extract が ALO モードの場合、LOGRETENTION パラメータのデフォルトは DISABLED です。必 要に応じて LOGRETENTION 設定を使用し、これをオーバーライドできます。

6.5.2 ALO モードの Extract の構成

Extract を ALO モード用に構成するには、4-1 ページの「ソースおよびターゲットの Oracle データベースでの Oracle GoldenGate の構成」に記載された Oracle GoldenGate の構成プロセ ス全体の一環として次の手順を実行します。

- 1. ソース・データベースの表に対して、表レベルおよびデータベース表のサプリメンタル・ ロギングを有効にします。(5-4ページの「ロギング・プロパティの構成」を参照してくだ さい。)
- 2. Oracle GoldenGate がソース・データベースとは異なるサーバー上で実行されている場合、 TNSNAMES ファイルに正しいエントリを入力するなど、SOL*Net がリモート・サーバーに接 続するように正しく構成されていることを確認します。Extract にソース・データベースへ の SQL*Net 接続を保持する権限を設定しておく必要があります。
- **3.** 次のように、**SOL***Net 接続文字列を使用します。
 - そのデータベースに接続するすべての Oracle GoldenGate プロセスのパラメータ・ ファイル内の USERID パラメータ。
 - GGSCIのDBLOGINコマンド。

USERID 文の例を次に示します。

USERID ggext@ora01, PASSWORD ggs123

注意: Oracle GoldenGate を実行しているサーバーにローカルのスタンバイ・ サーバーがある場合、USERID に接続文字列を使用する必要はありません。 ユーザーのログイン名のみを入力できます。

- 4. Extract パラメータ TRANLOGOPTIONS を、ARCHIVEDLOGONLY を指定して使用します。このオ プションを使用すると、v\$database ビューの db role 列の PRIMARY または LOGICAL STANDBY の値によって決まる、プライマリ・データベースまたはロジカル・スタンバイ・ データベースに対して Extract が強制的に ALO モードで動作します。デフォルトは、オン ライン・ログの読取りになっています。v\$databaseのdb role列のPHYSICAL STANDBYの 値によって決まる、フィジカル・スタンバイ・データベースに対して ALO モードを使用 する場合、TRANLOGOPTIONSで ARCHIVEDLOGONLY を指定する必要はありません。データベー スがフィジカル・スタンバイであることが検出されると、Extract は自動的に ALO モード で動作します。
- **5.** ご使用の環境に対してTRANLOGOPTIONS の他のオプションが必要になる場合があります。 たとえば、使用するコピー・プログラムに応じて、Extract エラーを回避するために COMPLETEARCHIVEDLOGONLY オプションが必要になる場合があります。
- 6. Extract で表名をソース・オブジェクト ID にマップするには、MAP パラメータを使用しま 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。
- 7. BEGIN オプションでタイムスタンプを指定して ADD EXTRACT コマンドを発行するか、ADD EXTRACT で SEQNO および RBA オプションを使用して、Extract グループを追加します。 最適なのは、NOW 引数を使用するのではなく、データの抽出を開始する既知の開始点を Extract に設定することです。NOW の開始時間は、現在のオンライン REDO ログの時間と 一致しますが、ALO の Extract ではオンライン・ログを読み取ることはできないため、 Oracle でのログの切替え時にそのログがアーカイブされるまで待機する必要があります。 切替えのタイミングは REDO ログのサイズとデータベース・アクティビティのボリューム に依存するため、Extract の開始時とデータのキャプチャ時の間にラグが発生することがあ ります。これは、通常および RAC のデータベース構成で発生する可能性があります。

6.6 ログ読取りのボトルネックの回避

Oracle GoldenGate によって REDO ログからデータがキャプチャされると、データベースのロ ギング・メカニズムによって書き込まれている同じファイルが Extract で読み取られるため、 I/O ボトルネックが発生する可能性があります。同じログを読み取る Extract プロセスの数の増 加に伴ってパフォーマンスが低下します。次の操作を実行できます。

- さらに高速なドライバおよびコントローラを使用してみます。さらに高速な I/O システム で Extract とデータベースの両方のロギング・メカニズムの速度が速くなります。
- RAID 0+1 にログを保存します。書き込まれたすべてのブロックでチェックサムを実行し、 高レベルの連続した I/O には不向きな RAID5 は使用しないでください。詳細は、Oracle ド キュメントを参照するか、関連 Web サイトを検索してください。

Oracle データベース用 DDL 同期の構成

この章には、Oracle GoldenGate での DDL サポートの理解および構成に役立つ情報が含まれま

7.1 DDL 同期の概要

Oracle GoldenGate では、DDL 操作のデータベース間の同期がサポートされます。DDL 同期は 次の場合にアクティブになります。

- ビジネス・アプリケーションで、ソース・オブジェクトとターゲット・オブジェクトにア クティブにアクセスおよび更新しています。
- **Oracle GoldenGate** のトランザクション・データ同期がアクティブです。

DDL のレプリケーションをサポートするコンポーネントとトランザクション・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は相互に独立しています。したがっ て、次の同期を行えます。

- DDL 変更のみ
- DML 変更のみ
- DDL と DML の両方

Oracle に対する DDL サポートについて、サポートされているオブジェクトと操作のリストは、 『Oracle GoldenGate *Oracle Installation and Setup Guide*』を参照してください。

7.2 Oracle GoldenGate DDL サポートの制限

この項では、DDL 機能の制限について説明します。このドキュメントのリリース後に見つかっ た追加の制限については、Oracle GoldenGate のリリース・ノートまたはソフトウェアに同梱 されている readme ファイルを参照してください。

7.2.1 DDL 文の長さ

Oracle GoldenGate では、DDL 文の長さが文字ではなくバイトで測定されます。サポートされ る長さは、約2MBです。これには、影響を受けるオブジェクトの名前とそのDDLタイプやそ の他の特性に応じてサイズが異なる内部的なオーバーヘッドが含まれます。DDL がサポートさ れるサイズより長い場合、Extract は警告を出してその DDL 操作を無視します。

無視された DDL は、マーカー表に保存されます。無視された Oracle DDL 文とその他の Oracle DDL 文は、ddl ddl2file.sql スクリプトを使用してキャプチャすることができ、このスクリ プトにより、DDL 操作は Oracle の USER DUMP DEST ディレクトリのテキスト・ファイルに保存 されます。スクリプトから、次の入力を求められます。

- Oracle GoldenGate DDL オブジェクトを含むスキーマの名前 (GLOBALS ファイルに指定され ています)。
- Oracle GoldenGate マーカー順序番号 (Extract パラメータ・ファイルで DDLOPTIONS に REPORT オプションが使用されている場合に Extract レポート・ファイルに記録されます)。
- 出力ファイルの名前。

7.2.2 サポートされているトポロジ

Oracle GoldenGate では、同種構成でのみ DDL 同期がサポートされます。 ソースおよびター ゲットのオブジェクトの定義は同一である必要があります。

Oracle GoldenGate では、スタンバイ・データベースの DDL はサポートされません。

Oracle GoldenGate では、2つ(のみ)のシステム間のすべてのサポートされている1方向構成 および双方向構成で DDL レプリケーションがサポートされます。Oracle アクティブ / アクティ ブ構成で特に考慮する事項については、7-26ページの「アクティブ/アクティブ(双方向)構成 での DDL の伝播」を参照してください。

7.2.3 フィルタリング、マッピングおよび変換

DDL 操作は、Oracle GoldenGate プロセスでは変換されません。ただし、プライマリ Extract 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および フィルタリングできます。データ・ポンプ Extract による DDL のマッピングやフィルタリング はできません。DDLは、プライマリ Extract から受信したとおりに渡されます。これは、 PASSTHRUモードと呼ばれます。

たとえば、ALTER TABLE TableA は、データ・ポンプによって ALTER TABLE TableA として処理 されます。別の内容を指定する TABLE 文に関係なく、そのプロセスによって ALTER TABLE TableB としてマップされることはありません。

7.2.4 名前変更

表に対する RENAME 操作は、同等の ALTER TABLE RENAME に変換されます。たとえば、RENAME tab1 TO tab2 は、ALTER TABLE tab1 RENAME TO tab2に変更されます。この変換は、スキーマ 名の使用が RENAME ではサポートされず、ALTER TABLE RENAME ではサポートされるためです。 Oracle GoldenGate で変換を行い、スキーマ名がターゲットの DDL 文に含まれるようにしま す。変換は、Replicat プロセスのレポート・ファイルにレポートされます。

順序やビューに対する RENAME 操作は、Oracle に順序やビューに対する同等の文がないため、 ALTER 文に変換されません。したがって、順序名の変更は、常にソース DDL と同じ所有者およ びオブジェクト名を使用してターゲットでレプリケートされ、別のものにはマップされません。

7.2.5 表からのフェッチと DDL との相互作用

Oracle GoldenGate では、一部のデータ型は、変更された行を REDO ストリームから特定し、 基になる表に問い合せて変更された列をフェッチすることでサポートされます。たとえば、ク ラシック・キャプチャでは、LOB の部分的な更新 (dbms lob package を介した変更) は、変更 された行と LOB 列を REDO ログから特定し、ベース表に行の LOB 列値を問い合せることでサ ポートされます。同様の手法が、ユーザー定義型のサポートに使用されます(クラシックと統 合の両方のキャプチャ)。

そのようなフェッチベースのサポートは、トランザクションがコミットされた SCN(システム 変更番号)に基づいてデータベースにフラッシュバック問合せを発行することで実装されます。 フラッシュバック問合せ機能にはいくつかの制限があります。一部の DDL 操作が妨げとなり、 その DDL より前のデータを取得するためのフラッシュバック問合せが成功しません。そのよう な DDL の例には、"ALTER TABLE MODIFY COLUMN"や "ALTER TABLE DROP COLUMN"があります。

したがって、Extract のキャプチャにラグがある場合、間にある DDL によって、その DDL よ り前のデータに対するフェッチ・リクエストが失敗することがあります。そのような場合、 Extract は元に戻り、変更された列に対するデータの現在のスナップショットをフェッチしま す。この方法には、2 つの制限があります。1 つ目は、DDL によって、Extract がフェッチする 必要のある列が変更されている可能性があるということです(たとえば、間にある DDL によっ て、キャプチャ対象の UDT に新規属性が追加された場合など)。2 つ目に、DDL によって Extract で論理行識別子として使用される列の1つが変更されている可能性があります。

このようなフェッチに関連する不整合を防ぐには、列の変更の際、次のような予防策を講じま

- **1.** 表に対するすべての DML を一時停止します。
- 2. Extract が残りのすべての REDO のキャプチャを終了するのを待ち、Replicat がキャプチャ されたデータの証跡からの処理を終了するのを待ちます。Replicat が終了したかどうかを 判断するには、処理するデータがこれ以上存在しないことを示すメッセージが表示される まで GGSCI で次のコマンドを発行します。

INFO REPLICAT group

- 3. DDL をソースで実行します。
- **4.** ソースの DML 操作を再開します。

7.2.6 SQL 内のコメント

ソース DDL 文でオブジェクト名の中にコメントが含まれている場合、そのコメントは、ター ゲット DDL 文ではオブジェクト名の最後に表示されます。例を次に示します。

ソース:

CREATE TABLE hr./*comment*/emp ...

ターゲット:

CREATE TABLE hr.emp /*comment*/ ...

これは、DDL 同期の整合性に影響しません。DDL 文の他の箇所のコメントは、レプリケート 時、元のままです。

7.2.7 コンパイル・エラー

トリガー、プロシージャ、ファンクションまたはパッケージに対する CREATE 操作がコンパイ ル・エラーになった場合でも、Oracle GoldenGate は、ターゲットに対してその DDL 操作を実 行します。厳密に言えば、DDL 操作自体は正常に完了し、再帰プロシージャなどでターゲット に対する依存性の実行を可能にするには、これらの操作を伝播する必要があります。

7.2.8 時間隔パーティション化

DDL が暗黙的であるため、DDL レプリケーションは時間隔パーティション化の影響を受けま せん。

7.3 DDL サポートに関する構成のガイドライン

DDL レプリケーションをサポートするように Oracle GoldenGate プロセスを構成する場合に考慮すべきガイドラインは次のとおりです。

7.3.1 データベースの権限

Oracle GoldenGate で DDL のキャプチャとレプリケーションをサポートするために必要なデータベース権限については、4-5ページの「Oracle GoldenGate のデータベース・ユーザーの割当て」を参照してください。

7.3.2 パラレル処理

パラレルの Extract や Replicat プロセスを使用する場合、関連する DDL および DML をすべて 同じプロセス・ストリーム内にまとめ、データの整合性を保証します。次のようになるように プロセスを構成します。

- あるオブジェクトに対するすべての DDL と DML が、同じ Extract グループおよび同じ Replicat グループで処理されます。
- 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ます。

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

Extract グループが、異なる Replicat グループによって読み取られる複数の証跡に書き込む場合、Extract はすべての DDL をすべての証跡に送信します。各 Replicat グループを使用して DDL をフィルタするには、Replicat パラメータ・ファイルで DDL パラメータのフィルタ・オプションを使用します。

7.3.3 データ・ポンプの DDL と DML

データ・ポンプを使用し、オブジェクトが DDL サポートを使用する場合、DML を PASSTHRU モード用に構成します。DDL は PASSTHRU モードでデータ・ポンプに渡されるため、DML も同様である必要があります。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、データ・フィルタリングやデータ・ポンプによる操作が可能になるよう NOPASSTHRU モードで構成できます。

表を PASSTHRU または NOPASSTHRU、あるいはその両方用に設定するには、次のようにします。

- **1.** データ・ポンプのパラメータ・ファイルで、DDL サポートを使用する表が含まれたすべて の TABLE 文の前に PASSTHRU パラメータを指定します。
- 2. データのフィルタリング、マッピングまたは変換を行う場合、同じパラメータ・ファイルで、DDL サポートを使用しない表が含まれた TABLE 文の前に NOPASSTHRU パラメータを指定できます。
- 3. データ・ポンプに対して DDL 構成パラメータ (DDL、DDLOPTIONS、DDLSUBST、 PURGEDDLHISTORY、PURGEMARKERHISTORY、DDLERROR) や DDL 関連のオプションを指定した Oracle GoldenGate トレース・パラメータを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。

7.3.4 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、大 / 小文字および文字セットが保持されます。このサポートによって、データベース階層のすべてのレベルで1バイトおよびマルチバイトの名前、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。

DDL 同期をサポートする構成パラメータで、疑問符 (?) およびアスタリスク (*) のワイルドカー ドを使用して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。ワイルドカードを適切に 処理するために、WILDCARDRESOLVE パラメータはデフォルトで DYNAMIC に設定されています。 WILDCARDRESOLVE がそれ以外の値に設定されている場合、DDL 操作を処理する Oracle GoldenGate プロセスは異常終了し、エラーがプロセス・レポートに書き込まれます。

7.3.5 データ定義

DDL サポートには同種構成が必要なため、Replicat パラメータ・ファイルで 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 サポート用に 構成され、SOURCEDEFS パラメータが使用されている場合、Replicat は異常終了します。 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。

7.3.6 切捨て

TRUNCATE 文は、次のようにサポートされます。

- Oracle GoldenGate の完全 DDL サポートの一環として (TRUNCATE TABLE、ALTER TABLE TRUNCATE PARTITION および他の DDL がサポートされます)。これは、DDL パラメータに よって制御されます (7-9 ページの「DDL サポートの有効化」を参照してください)。
- スタンドアロンの TRUNCATE サポートとして。このサポートによって、TRUNCATE TABLE は レプリケートできますが、他の DDL はできません。GETTRUNCATES パラメータによって、 スタンドアロンの TRUNCATE 機能が制御されます。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。

重複操作によるエラーを回避するために、同時にアクティブにできるのはこれらの機能の1つ だけです。

7.3.7 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されるターゲット・デー タベースから始めます。DDL サポートは、Replicat の初期ロード方法と互換性があります。

初期ロードを実行する前に、DDL の抽出とレプリケーションを無効にします。DDL 処理は、 Extract および Replicat のパラメータ・ファイルの DDL パラメータによって制御されます。

ソース・データとターゲット・データの初期同期ができたら、ソース・アプリケーションを実 行する前に、NEXTVALですべてのソース順序値を少なくとも1回使用します。システム内の各 順序から NEXTVAL を選択するスクリプトを使用できます。これは、Extract の実行中に行われる 必要があります。

7.3.8 CREATE または RENAME の後のデータ継続性

CREATE 操作または RENAME 操作の結果生じた新規の Oracle 表に対する DML 操作をレプリケー トするには、新規表の名前をパラメータ・ファイルの TABLE 文および MAP 文に指定する必要が あります。ワイルドカードを使用して、それらが必ず含まれるようにできます。

CREATE USER を使用して新規ユーザーを作成し、新規または名前が変更された表をそのスキー マに移動するには、新規ユーザー名を TABLE 文および MAP 文に指定する必要があります。新規 ユーザー fin2 を作成し、新規または名前が変更された表をそのスキーマに移動する場合、fin2

オブジェクトをターゲットの同じスキーマにマップするか、異なるスキーマにマップするかに 応じて、パラメータ文は次のようになります。

Extract:

TABLE fin2.*;

Replicat:

MAP fin2*, TARGET different schema.*;

7.4 DDL スコープの理解

データベース・オブジェクトは、スコープで分類されます。スコープは、オブジェクトに対す る DDL 操作の Oracle GoldenGate による処理方法を定義するカテゴリです。スコープは次のと おりです。

- MAPPED
- UNMAPPED
- OTHER

スコープを使用することで、DDL 操作のフィルタリング、文字列の置換およびエラー処理を詳 細に制御できます。

7.4.1 マップされるスコープ

TABLE 文および MAP 文で指定されるオブジェクトは、MAPPED スコープのオブジェクトです。そ れらの文での抽出とレプリケーションの指示は、オーバーライド・ルールが適用されていない かぎり、指定されたオブジェクトに対するデータ (DML) と DDL の両方に適用されます。

TABLE 文と MAP 文内のオブジェクトでは、次の表に示す DDL 操作がサポートされます。

表 7-1 MAP 文と TABLE 文でマップできるオブジェクト

操作	対象のオブジェクト ¹
CREATE	TABLE ³
ALTER	INDEX
DROP	TRIGGER
RENAME	SEQUENCE
COMMENT ON ²	MATERIALIZED VIEW
	VIEW
	FUNCTION
	PACKAGE
	PROCEDURE
	SYNONYM
	PUBLIC SYNONYM ⁴
GRANT	TABLE
REVOKE	SEQUENCE
	MATERIALIZED VIEW
ANALYZE	TABLE
	INDEX
	CLUSTER

- 1 TABLE および MAP では、これらの操作の対象のオブジェクト名に使用される可能性のある一部の特殊文字 がサポートされません。サポートされていない特殊文字が使用されたオブジェクトは、UNMAPPED および OTHER のスコープでサポートされます。
- ² COMMENT ON TABLE、COMMENT ON COLUMN に適用されます。
- 3 AS SELECT を含みます。
- 4 表名は、スキーマ名で修飾される必要があります。

Extract の場合、MAPPED スコープでは、TABLE 文の指示に従って DDL キャプチャ用にオブジェ クトがマークされます。Replicat の場合、MAPPED スコープでは、レプリケーション用に DDL がマークされ、MAP 文の TARGET 句のスキーマと名前によって指定されたオブジェクトにマップ されます。このマッピングを実行するために、Replicat によって ALTER SESSION が発行され、 Replicat セッションのスキーマが TARGET 句で指定されたスキーマに設定されます。修飾されて いないオブジェクトが DDL に含まれている場合、ターゲットで割り当てられるスキーマは、 7-9 ページの「DDL 内の修飾されていないオブジェクト名の正しい識別」に説明されている状 況に応じて異なります。

TABLE 文と MAP 文が次のとおりであるとします。

Extract(ソース)

TABLE fin.expen; TABLE hr.tab*;

Replicat(ターゲット)

MAP fin.expen, TARGET fin2.expen2; MAP hr.tab*, TARGET hrBackup.bak *;

次のソース DDL 文もあるとします。

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

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

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

同様に、例の TABLE 文と MAP 文の2つ目のセットには、次のソースとターゲットの DDL 文が 考えられます。

ソース:

CREATE TABLE hr.tabPayables ...;

ターゲット:

CREATE TABLE hrBackup.bak tabPayables ...;

MAPPED スコープのオブジェクトでは、DDL サポートを詳細に調整しない場合、DDL 構成パラ メータからオブジェクト名を省略できます。TABLE 文と MAP 文でオブジェクト名を変更する必 要がある場合、それらのオブジェクトに対する DDL に変更が自動的に適用されます。

オブジェクトを TABLE 文には含め、MAP 文には含めない場合、そのオブジェクトに対する DDL のスコープは、ソースでは MAPPED ですが、ターゲットでは UNMAPPED です。

7.4.1.1 Oracle クラスタ表と UDT のマッピング

Oracle クラスタ表または Oracle ユーザー定義型 (UDT) は、別のターゲット名にはマップでき ませんが、別のターゲット所有者にはマップできます。これらの特別なオブジェクトは、それ 自体が MAPPED と UNMAPPED の両スコープが混在する可能性のある、基になる表で構成されるた め、名前のマッピングは使用できません。

7.4.1.2 ALTER INDEX のマッピング

ALTER INDEX...RENAME コマンドは、別のターゲット索引名にはマップできませんが、別のター ゲット所有者にはマップできます。

有効な例

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 tqt.*;

また、次のように元の索引名をソースとターゲットの指定に使用し、明示的にマップすること もできます。

MAP src.ind TARGET tqt.ind;

前述のいずれの場合も、ターゲット DDL は次のようになります。

ALTER INDEX tgt.ind RENAME TO indnew;

無効な例:

次のような MAP 文は無効です。

MAP src.ind TARGET tgt.indnew;

その文で古い名前が新しい名前にマップされ、ターゲット DDL は次のようになります。

ALTER INDEX tgt.indnew RENAME TO indnew;

7.4.2 マップされないスコープ

DDL 操作が、TABLE 文または MAP 文での使用についてサポートされていて、そのベース・オブ ジェクト名がそれらのパラメータの1つに含まれない場合、UNMAPPED スコープです。

オブジェクト名のスコープが、ソースでは UNMAPPED (Extract の TABLE 文にない) でも、ター ゲットでは MAPPED (Replicat の MAP 文にある) であることも、その逆もあります。 Oracle DDL のスコープが Replicat 構成で UNMAPPED の場合、Replicat はデフォルトでは次のように動作しま す。

- 1. Replicat セッションの現在の所有者をソース DDL オブジェクトの所有者に設定します。
- 2. DDL をその所有者として実行します。
- **3.** Replicat を Replicat セッションの現在の所有者としてリストアします。

7-9 ページの「DDL 内の修飾されていないオブジェクト名の正しい識別」も参照してください。

7.4.3 他のスコープ

マップできない DDL 操作は、OTHER スコープです。DDL が Replicat 構成で OTHER スコープの 場合、ソース DDL と同じ所有者とオブジェクト名のターゲットに適用されます。

OTHER スコープの例は、データ・ファイル名を操作する DDL などのシステムに固有の参照を行 う DDL 操作です。

OTHERスコープのその他の例は、次のとおりです。

CREATE USER joe IDENTIFIED by joe;

CREATE ROLE ggs_gguser_role IDENTIFIED GLOBALLY;

ALTER TABLESPACE gg_user TABLESPACE GROUP gg_grp_user;

7-9 ページの「DDL 内の修飾されていないオブジェクト名の正しい識別」も参照してください。

7.5 DDL 内の修飾されていないオブジェクト名の正しい識別

Extract では、DDL 操作の実行時に有効な現在のスキーマ(セッション・スキーマとも呼ばれる) をキャプチャします。このスキーマは、DDL 内の修飾されていないオブジェクト名の解決に使 用されます。

次の例で考えてみます。

CONNECT SCOTT/TIGER

CREATE TABLE TAB1 (X NUMBER);

CREATE TABLE SRC1.TAB2(X NUMBER) AS SELECT * FROM TAB1;

いずれの DDL 文でも、修飾されていない表 TAB1 は、DDL の実行時に有効な現在のスキーマ SCOTT に基づいて SCOTT.TAB1 と解決されます。

現在のスキーマを設定する方法には、次の例に示すようにセッションの current schema を設 定するという別の方法もあります。

CONNECT SCOTT/TIGER

ALTER SESSION SET CURRENT SCHEMA=SRC;

CREATE TABLE TAB1 (X NUMBER);

CREATE TABLE SRC1.TAB2(X NUMBER) AS SELECT * FROM TAB1;

いずれの DDL 文でも、修飾されていない表 TAB1 は、DDL の実行時に有効な現在のスキーマ SRC に基づいて SRC.TAB1 と解決されます。

クラシックと統合の両方のキャプチャ・モードで、Extract は DDL の実行時に有効な現在のス キーマをキャプチャし、現在のスキーマを使用して、修飾されていないオブジェクト名(ある 場合)を解決します。その結果、Replicatに対して指定されるMAP文は、修飾されていないオ ブジェクト名を使用する DDL について正しく機能します。

ソース・セッション・スキーマを別のターゲット・セッション・スキーマにマップすることも できます。セッション・スキーマのマッピングは、CREATE TABLE AS SELECT や REFERENCES 句 を使用した CREATE TABLE 操作などの一部の DDL が、マッピングが存在する場合にターゲット で成功するために必要です。このマッピングはグローバルで、同じスキーマ名を含む他のマッ ピングをオーバーライドします。セッション・スキーマをマップするには、MAPSESSIONSCHEMA オプションを指定して DDLOPTIONS パラメータを使用します。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。

DDLOPTIONS パラメータの NOEXPLICITSCHEMAMAPPING オプションを使用してスキーマが明示的 にマップされないようにできます。『Windows and UNIX リファレンス・ガイド』の DDLOPTIONS パラメータに関する項を参照してください。

7.6 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とおりです。

- ソースでは、Oracle GoldenGate DDL サポートはデフォルトで無効です。DDL パラメータを 使用して、DDL をキャプチャするよう Extract を構成する必要があります。
- ターゲットでは、DDL サポートはデフォルトで有効で、レプリケートされるトランザク ション・データの整合性が保たれます。デフォルトでは、Replicat によって証跡に含まれ るすべての DDL 操作を処理します。必要に応じて DDL パラメータを使用し、DDL 操作を 無視またはフィルタするよう Replicat を構成します。

7.7 DDL レプリケーションのフィルタリング

Oracle データベースでは、要件に応じて特定(またはすべて)の DDL がターゲット・データ ベースに適用されるよう、次の方法を使用して DDL 操作をフィルタします。デフォルトでは、 DDL トリガーによってすべての DDL が Extract に渡されます。

- *PL/SOL コードによるフィルタ*。この方法では、DDL 操作が発生すると DDL トリガーに よってコールされる Oracle 関数を使用して DDL を Extract に送信するかどうかを判断しま す。
- 組込みフィルタ・ルールによるフィルタ:この方法では、プロシージャを使用してフィル タ・ルールを Oracle GoldenGate トリガー・ロジックに組み込みます。この方法では、 Extract に送信するオブジェクトのタイプを注意深く制御でき、ルールの評価の順序付けが 可能です。
- ソースまたはターゲット、あるいはその両方でのDDL パラメータによるフィルタ。この方 法は Oracle GoldenGate 内で実行され、Extract と Replicat の両方でフィルタ基準を実行で きます。Extract でフィルタリングを行うか、すべての DDL を証跡に送り、Replicat で フィルタリングを行います。あるいは、異なる場所の組合せでフィルタすることもできま す。DDL パラメータでは、フィルタリングを行う場所を制御でき、DDL スコープに基づい てまとめてフィルタする(すべての MAPPED スコープを含めるなど)機能など、トリガーの 方法よりも多くのフィルタリング・オプションがあります。
- PL/SQL、組込みルールおよびDDL パラメータによるフィルタの組合せ。DDL トリガーま たはフィルタ・ルールによってフィルタされた後で Extract に渡される DDL を、DDL パラ メータを使用して特定のニーズに合せてさらにフィルタできます。

7.7.1 PL/SQL コードによるフィルタ

PL/SQL コードを記述して、DDL を Extract に渡すかどうかを判断する関数に DDL に関する情 報を渡します。Extract に送信する DDL 操作を少なくすることでキャプチャのパフォーマンス を向上させることができます。

- 1. Oracle GoldenGate のインストール・ディレクトリにある ddl filter.sgl ファイルを、こ れから記述するコードをテストできるテスト・マシンにコピーします。
- 2. ファイルを編集用に開きます。filterDDLという名前の PL/SQL 関数が含まれており、こ れを変更して if/then フィルタ基準を指定できます。この関数に渡される情報は、次のと おりです。
 - ora owner: DDL オブジェクトの所有者
 - ora name: オブジェクトの定義されている名前
 - ora objtype: オブジェクトのタイプ (TABLE、INDEX など)
 - ora optype: 操作のタイプ (CREATE、ALTER など)
 - ora login user: DDL を実行したユーザー
 - retVal: Extract の処理に DDL を含める場合は INCLUDE、DDL を除外する場合は EXCLUDE₀

'compute retVal here' コメントの後ろに、フィルタする DDL のタイプごとにフィルタ・ コードを記述します。次に例を示します。

```
if ora_owner='SYS' then
retVal:='EXCLUDE';
end if;
if ora_objtype='USER' and ora_optype ='DROP' then
retVal:='EXCLUDE';
if ora_owner='JOE' and ora_name like 'TEMP%' then
retVal:='EXCLUDE';
end if;
```

この例では、次の DDL が、DDL トリガーによる処理から除外されます。

- SYS によって所有されているオブジェクトに対する DDL
- 任意の DROP USER
- JOE.TEMP% に対する任意の DDL
- 3. (オプション)フィルタリングをトレースする場合は、PL/SQL の各 if/then 文に次の構文 を追加できます。

if ora_owner='JOE' and ora_name like 'TEMP%' then retVal:='EXCLUDE'; if "&gg_user" .DDLReplication.trace_level >= 1 then "&gg_user" .trace_put_line ('DDLFILTER', 'excluded JOE.TEMP%'); end if;

説明:

- &gg_user は、Oracle GoldenGate DDL サポート・オブジェクトのスキーマです。
- .DDLReplication.trace_level は、DDL トレースのレベルです。トリガー・トレース を使用するには、Extract パラメータ・ファイルで DDL または DDLONLY オプションを指 定してTRACE またはTRACE2パラメータを使用する必要があります。 .DDLReplication.trace levelを>=1に設定する必要があります。
- trace_put_line は、Extract がトレース・ファイルに書き込む、フィルタされた DDL のタイプを表すユーザー定義テキスト文字列です。
- 4. コードを保存します。
- 5. テスト・システムで DDL アクティビティを停止します。
- 6. SQL*Plus で、次のように ddl filter.sql ファイルをコンパイルします (schema name は、 Oracle GoldenGate DDL オブジェクトがインストールされているスキーマです)。

@ddl_filter schema_name

注意: これらのオブジェクトの詳細は、『Oracle GoldenGate *Oracle Installation and Setup Guide*』を参照してください。

- 7. テスト環境でテストし、フィルタリングが機能することを確認します。コード内のエラー によってソースとターゲットの DDL が同期しなくなることがあるため、このテストを実行 することは重要です。
- 8. テストが成功したら、ファイルをソース本番システムの Oracle GoldenGate のインストー ル・ディレクトリにコピーします。
- 9. ソース・システムで DDL アクティビティを停止します。
- **10.** 前に行ったように ddl filter.sql ファイルをコンパイルします。 @ddl filter schema name
- **11.** ソース・システムで DDL アクティビティを停止します。

7.7.2 フィルタ・ルールの追加および削除

選択ルールまたは除外ルールを追加して、DDLトリガーによって Extract に送信される DDL 操作を制御できます。ルールを格納し、Extract に送信する DDL 操作を少なくすることでキャ プチャのパフォーマンスを向上させることができます。

- 1. DDLAUX.addRule() 関数を使用して、次の説明に従ったルールを定義します。この関数は、 ddl setup.sql スクリプトを使用して DDL オブジェクトがインストールされた後に Oracle GoldenGate DDL スキーマにインストールされます。
- 2. ルールをアクティブにするには、SQL*Plus で関数を実行するか、SQL ファイルに一連の ルールを入力してそのファイルを SQL*Plus で実行します。

7.7.2.1 DDLAUX.addRule() 関数の定義

FUNCTION addRule(obj name IN VARCHAR2 DEFAULT NULL, base_obj_name IN VARCHAR2 DEFAULT NULL, owner name IN VARCHAR2 DEFAULT NULL, base owner name IN VARCHAR2 DEFAULT NULL, base_obj_property IN NUMBER DEFAULT NULL. obj type IN NUMBER DEFAULT NULL, command IN VARCHAR2 DEFAULT NULL, inclusion IN boolean DEFAULT NULL , sno IN NUMBER DEFAULT NULL) RETURN NUMBER:

7.7.2.2 DDLAUX.addRule() のパラメータ

この関数に渡される情報は次のパラメータで、オブジェクトの属性と関連しています。すべて のパラメータがオプションで、複数のパラメータを指定できます。

sno: ルールを識別するシリアル番号を指定します。ルールの評価は、シリアル番号の小さ いものから大きいものの順に、一致が見つかるまで行われます。sno を使用して除外ルー ルの前に選択ルールを置き、除外ルールの例外とすることができます。これはファンク ションでプロシージャではないため、ルールのシリアル番号を返すので、これを使用して 削除するルールを DDLAUX.dropRule() に指定します。コードの冒頭で DECLARE sno NUMBER; BEGIN sno := 文を使用して指定しないかぎり、シリアル番号は自動的に生成され ます。

例を次に示します。

DECLARE sno NUMBER; BEGIN sno := tkggadmin..DDLAUX.ADDRULE(obj name => 'GGS%' , obj type => TYPE TABLE); END

- obj name: オブジェクト名を指定します。
- owner name: オブジェクトの所有者の名前を指定します。
- base obj name: DDL オブジェクトのベース・オブジェクト名を指定します(オブジェクト が索引の場合のベース表など)。
- base owner name: ベース・オブジェクトの所有者名を指定します。
- base obj property: ベース・オブジェクトのプロパティを指定します。7-13 ページの 「DDLAUX.addRule() に対して有効な DDL コンポーネント」を参照してください。
- obj type: オブジェクト・タイプを指定します。7-13 ページの「DDLAUX.addRule() に対 して有効な DDL コンポーネント」を参照してください。
- command: コマンドを指定します。7-13 ページの「DDLAUX.addRule() に対して有効な DDL コンポーネント」を参照してください。

inclusion = TRUE: 指定されたオブジェクトが DDL トリガーによってキャプチャされるこ とを示します。このパラメータが指定されない場合、ルールは除外ルールとなり、指定さ れたオブジェクトはキャプチャされません。除外ルールと選択ルールの両方を指定できま す。DDL がルールのいずれにも一致しない場合、デフォルトで選択されます (Extract に渡 されます)。パラメータなしでDDLAUX.addRule()をコールすると、すべてのオブジェクト に対するすべての DDL を除外する 空のルールが生成されます。

7.7.2.3 DDLAUX.addRule() に対して有効な DDL コンポーネント

関数コードで指定できる定義済 DDL オブジェクト・タイプ、ベース・オブジェクト・プロパ ティおよび DDL コマンドは次のとおりです。

有効なオブジェクトは次のとおりです。

TYPE INDEX

TYPE TABLE

TYPE VIEW

TYPE SYNONYM

TYPE SEQUENCE

TYPE PROCEDURE

TYPE FUNCTION

TYPE PACKAGE

TYPE TRIGGER

有効なベース・オブジェクト・プロパティは次のとおりです。

TB IOT

TB CLUSTER

TB_NESTED

TB TEMP

TB EXTERNAL

有効なコマンドは次のとおりです。

CMD CREATE

CMD DROP

CMD TRUNCATE

CMD ALTER

7.7.2.4 ルールベースのトリガーのフィルタの例

次の例では、名前が IMPTEMP で始まる表以外のすべての一時表を除外します。

- 1.DDLAUX.ADDRULE(obj name => 'IMPTEMP%', base obj property => TB TEMP, obj type => TYPE TABLE, INCLUSION => TRUE);
- 2.DDLAUX.ADDRULE(base obj property => TB TEMP, obj type => TYPE TABLE);

注意: IMPTEMP% 表を含めるため、そのルールを先にします。

次の例では、名前が 'GGS%' のすべての表を除外します。

DECLARE sno NUMBER; BEGIN sno := DDLAUX.ADDRULE(obj_name => 'GGS%' , obj_type => TYPE_ TABLE); END

次の例では、すべての一時表を除外します。

DDLAUX.ADDRULE(base_obj_property => TB_TEMP, obj_type => TYPE_TABLE);

次の例では、TEMP 表のすべての索引を除外します。

DDLAUX.ADDRULE(base obj property => TB TEMP, obj type => TYPE INDEX);

次の例では、スキーマ TKGGADMIN のすべてのオブジェクトを除外します。

DDLAUX.ADDRULE(obj owner => 'TKGGADMIN');

次の例では、TEMP 表に対して実行される TRUNCATE 操作のすべてのオブジェクトを除外します。

DDLAUX.ADDRULE(base_obj_property => TB_TEMP, obj_type => TYPE_TABLE, command => CMD_ TRUNCATE)

7.7.2.5 フィルタ・ルールの削除

削除するルールを指定して DDLAUX.dropRule() 関数を使用します。この関数は、ddl setup.sql スクリプトを使用して DDL オブジェクトがインストールされた後に Oracle GoldenGate DDL スキーマにインストールされます。入力として、削除するルールのシリアル 番号を指定します。

FUNCTION dropRule (sno IN NUMBER) RETURN BOOLEAN;

7.7.3 DDL パラメータを使用したフィルタリング

DDL パラメータは、Extract および Replicat プロセス内で DDL をフィルタリングするための主 要な Oracle GoldenGate パラメータです。

オプションなしで DDL パラメータを使用すると、フィルタリングは行われず、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・オブジェクトに対 して生成された、サポートされているすべての DDL 操作をキャプチャし、証跡に送信しま す。
- Replicat パラメータとして、Oracle GoldenGate の証跡からすべての DDL 操作をレプリ ケートし、ターゲットに適用します。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。

オプションとともに使用すると、DDLパラメータは、次のものに基づいて DDL 操作を選択また は除外するフィルタリング・エージェントとして機能します。

- 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・ファイルで使用できる DDL パラメータは1つのみですが、複数の選択オプション と除外オプションを組み合せて、必要なレベルまで DDL をフィルタできます。

- DDL フィルタリング・オプションは、トランザクション・ソースからのキャプチャを行う プライマリ Extract に対しては有効ですが、データ・ポンプ Extract に対しては無効です。
- 複数のフィルタ・オプション指定を組み合せると、AND 文として論理的に連結されます。
- レプリケートされる DDL 文では、複数のオプションを使用して指定されたすべてのフィル タ基準が満たされる必要があります。
- 複雑な DDL フィルタリング基準を使用する場合、本番環境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薦めします。

警告: DDL パラメータ、TABLE パラメータ、MAP パラメータ、TABLEEXCLUDE パラメータ、MAPEXCLUDE パラメータのいずれにも、Oracle GoldenGate に よってインストールされる DDL オブジェクトを含めないでください。それ らのパラメータのワイルドカード指定に Oracle GoldenGate によってインス トールされる DDL オブジェクトが含まれていないことを確認します。これ らのオブジェクトは Oracle GoldenGate 構成に含めることはできませんが、 Extract プロセスはこれらのオブジェクトに対する操作を認識している必要が あり、それが EXCLUDE、TABLEEXCLUDE または MAPEXCLUDE のパラメータ文を 使用して、構成からこれらを明示的に除外できない理由です。

注意: DDL パラメータ文を作成する前に、7-28 ページの「処理に対する DDL の評価方法」を参照することをお薦めします。

構文:

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

表 7-2 DDL の選択オプションと除外オプション

オプション 説明 INCLUDE | EXCLUDE INCLUDE および EXCLUDE を使用して、選択句または除外句の始まりを示します。 選択句には、このパラメータの対象の DDL を識別するフィルタリング基準が含まれ ます。 除外句には、特定の DDL をこのパラメータから除外するフィルタリング基準が含ま れます。 選択句または除外句は、INCLUDE または EXCLUDE キーワードに、適用するパラメータの他 のオプションの有効な組合せを続けて構成する必要があります。 EXCLUDE を使用する場合、対応する INCLUDE 句を作成する必要があります。たとえば、次 の例は無効です。 DDL EXCLUDE OBJNAME hr.* しかし、次のいずれも使用できます。 DDL INCLUDE ALL, EXCLUDE OBJNAME hr.* DDL INCLUDE OBJNAME fin.*EXCLUDE OBJNAME fin.ss EXCLUDE は、同じ基準を含む INCLUDE より優先されます。複数の選択句および除外句を使 用できます。 MAPPED | UNMAPPED | MAPPED、UNMAPPED、OTHER および ALL を使用して、DDL 操作のスコープに基づいて OTHER | ALL INCLUDE または EXCLUDE を適用します。 MAPPED では、INCLUDE または EXCLUDE を MAPPED スコープの DDL 操作に適用します。 MAPPED フィルタリングは、他の DDL パラメータ・オプションを使用して指定された フィルタリングの前に実行されます。 UNMAPPED では、INCLUDE または EXCLUDE を UNMAPPED スコープの DDL 操作に適用しま す。 OTHER では、INCLUDE または EXCLUDE を OTHER スコープの DDL 操作に適用します。 ALL では、INCLUDE または EXCLUDE を全スコープの DDL 操作に適用します。 OPTYPE type OPTYPE を使用して、INCLUDE または EXCLUDE を特定のタイプの DDL 操作 (CREATE、 ALTER、RENAME など) に適用します。type には、データベースに対して有効な DDL コマ ンドを使用します。たとえば、ALTER操作を含める場合、正しい構文は次のとおりです。 DDL INCLUDE OPTYPE ALTER OBJTYPE 'type' OBJTYPE を使用して、INCLUDE または EXCLUDE を特定のタイプのデータベース・オブジェ クトに適用します。typeには、TABLE、INDEX、TRIGGER などのデータベースに対して有効 なオブジェクト・タイプを使用します。Oracle マテリアライズド・ビューとマテリアライ ズド・ビュー・ログの場合、正しいタイプはそれぞれ snapshot および snapshot log で す。オブジェクト・タイプの名前を一重引用符で囲みます。例を次に示します。 DDI, INCLUDE OBJTYPE 'INDEX' DDL INCLUDE OBJTYPE 'SNAPSHOT' Oracle オブジェクト・タイプ USER には OBJNAME オプションを使用しないでください。こ れは、OBJNAME には owner.object が必要ですが、USER にはスキーマしかないためです。

表 7-2 DDL の選択オプションと除外オプション (続き)

オプション 説明

OBJNAME name

OBJNAME を使用して、INCLUDE または EXCLUDE をオブジェクトの完全修飾名 (owner.table_name など)に適用します。ワイルドカードは、オブジェクト名にのみ使用できます。

例:

DDL INCLUDE OBJNAME accounts.*

Oracle USER オブジェクトには OBJNAME を使用しないでください。これは、OBJNAME には owner.object が必要ですが、USER にはスキーマしかないためです。

Replicat パラメータ・ファイルで OBJNAME と MAPPED を組み合せて使用する場合、OBJNAME の値は、MAP 文の TARGET 句で指定された名前を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 fin2.*です。

MAP fin.exp_*, TARGET fin2.*;

次の例では、CREATE TABLE 文はソースで次のように実行されます。

CREATE TABLE fin.exp phone;

同じ文が、ターゲットでは次のように実行されます。

CREATE TABLE fin2.exp_phone;

ターゲットの所有者が MAP 文で指定されていない場合、Replicat によって、USERID パラメータで指定されたデータベース・ユーザーにマップされます。

トリガーなどの導出オブジェクトを作成する DDL の場合、OBJNAME の値は、導出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、hr.accounts であり、hr.insert_trig ではありません。

CREATE TRIGGER hr.insert trig ON hr.accounts;

RENAME 操作では、OBJNAME の値は、新規の表名である必要があります。たとえば、次のDDL 文を含める場合、正しい値は、hr.acct です。

ALTER TABLE hr.accounts RENAME TO acct;

INSTR 'string'

INSTR を使用して、INCLUDE または EXCLUDE を、コマンド構文内に特定の文字列を含む (コメント内には含まない) DDL 文に適用します。たとえば、次の例では、索引を作成する DDL が除外されます。

DDL INCLUDE ALL EXCLUDE INSTR 'CREATE INDEX'

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。

INSTR では、文字列内に含まれる一重引用符 (' ') はサポートされない他、NULL 値もサポートされません。

INSTRCOMMENTS 'comment_
string'

INSTRCOMMENTS を使用して、INCLUDE または EXCLUDE を、コメント内に特定の文字列を含む (DDL コマンド自体には含まない) DDL 文に適用します。INSTRCOMMENTS を使用すると、コメントをフィルタリング・エージェントとして使用できます。

たとえば、次の例では、コメントに "source" を含む DDL 文を除外します。

DDL INCLUDE ALL EXCLUDE INSTRCOMMENTS 'SOURCE ONLY'

この例では、次のような DDL 文はレプリケートされません。

CREATE USER john IDENTIFIED BY john /*source only*/;

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。 INSTR と INSTRCOMMENTS を組み合せて、同じ DDL 文のコマンド構文内とコメント内の文字列に対してフィルタできます。

INSTRCOMMENTS では、文字列内に含まれる一重引用符 (' ') はサポートされない他、NULL 値もサポートされません。

表 7-2 DDL の選択オプションと除外オプション(続き)

オプション 説明

INSTRWORDS 'word list'

INSTRWORDS を使用して INCLUDE または EXCLUDE を、特定の語を含む DDL 文に適用しま

word listには、一重引用符内に任意の順序で語を指定します。空白を含めるには、空白 (語がある場合は語も)を二重引用符で囲みます。二重引用符は、文を囲むためにも使用 できます。

指定された語がすべて DDL にある場合、INSTRWORDS は有効です。

ALTER TABLE INCLUDE INSTRWORDS 'ALTER CONSTRAINT " xyz"

この例には、次の両方の文が一致します。

ALTER TABLE ADD CONSTRAINT xyz CHECK

ALTER TABLE DROP CONSTRAINT xyz

INSTRWORDS では、文字列内に含まれる一重引用符('')はサポートされない他、NULL値 もサポートされません。

表 7-2 DDL の選択オプションと除外オプション (続き)

オプション 説明

INSTRCOMMENTSWORDS 'word list'

INSTRWORDS と同様に機能しますが、DDL 文内のコメントにのみ適用され、DDL 構文自体 には適用されません。INSTRCOMMENTS を使用すると、コメントをフィルタリング・エー ジェントとして使用できます。

INSTRCOMMENTSWORDS では、文字列内に含まれる一重引用符('')はサポートされない他、 NULL 値もサポートされません。

INSTRWORDS と INSTRCOMMENTSWORDS を組み合せて、同じ DDL 文のコマンド構文内とコメ ント内の文字列に対してフィルタできます。

STAYMETADATA

Extract と Replicat に有効です。メタデータがレプリケートされないようにします。

Extract は、表に対する DML を初めて検出したとき、その表のメタデータを取得します。 その表に対する DDL が検出されると、古いメタデータは無効になります。その表に対す る次の DML は、新しいメタデータに合せたものになり、ターゲット表の構造は、常に、 ソース表の構造に合った最新のものになります。

ただし、特定の DDL 操作が表のメタデータに影響を及ぼさないことがわかっている場合、 STAYMETADATA を使用して、現在のメタデータが取得またはレプリケートされないように できます。これによってパフォーマンスが向上し、切捨てや制約の無効化などの DDL が ことの多いインポートやエクスポートなどの操作に役立ちます。表の構造は後 続のデータ・レプリケーションの整合性に関係しますが、これらの操作は表の構造に影響 を及ぼさないため、このような場合は無視できます。たとえば、ALTER TABLE ADD FOREIGN KEYによって表のメタデータは影響を受けません。

これが選択的に適用される例は、次のとおりです。

DDL INCLUDE ALL INCLUDE STAYMETADATA OBJNAME xyz

この例は、すべての DLL をレプリケーションに含めながら、オブジェクト xyz を操作す る DDL のみが STAYMETADATA の対象となることを指定しています。

STAYMETADATA は、EXCLUDE 句でも同様に使用できます。

STAYMETADATA は、メタデータの整合性を保証するために、ソースとターゲットで同じよ うに使用される必要があります。

STAYMETADATA が使用される場合、メッセージがレポート・ファイルに追加されます。 DDL レポーティングは、REPORT オプションを指定した DDLOPTIONS パラメータによって 制御されます。

同じ機能は、@ddl staymetadata スクリプトを使用して、ソースで検出されるすべての DDLにグローバルに適用できます。

- @ddl staymetadata onは、メタデータのバージョニングをグローバルで無効にしま す。
- @ddl_staymetadata_off は、メタデータのバージョニングを再度有効にします。

どの DDL がオブジェクトのメタデータに影響するかは必ずしも明白ではないため、この オプションは、Oracle GoldenGate のテクニカル・サポート・スタッフと相談して使用す ることをお薦めします。不適切に使用された場合、レプリケーション環境の整合性が損な われる可能性があります。

表 7-2 DDL の選択オプションと除外オプション (続き)

オプション 説明

EVENTACTIONS (action specification)

イベント・レコードと呼ばれるトランザクション・ログまたは証跡内の DDL レコードに 基づいて、定義されたアクションが Extract または Replicat プロセスで実行されます。 DDL レコードが、DDL パラメータの他のフィルタリング・オプションに従って、Extract またはデータ・ポンプによる証跡への書込みまたは Replicat による実行の対象とされる場 合、DDL イベントがトリガーされます。このシステムを使用して、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。

action specification については、MAP パラメータおよび TABLE パラメータの EVENTACTIONS を参照してください。

DDL レコードに対する EVENTACTIONS の使用のガイドラインは次のとおりです。

- CHECKPOINTBEFORE: 各 DDL レコードは独立しているため、DDL レコードがトランザ クションの始まりであることが保証されます。その結果、CHECKPOINT BEFORE イベン ト・アクションが DDL レコードに対して暗示されます。
- IGNORE: このオプションは、DDL レコードに対して無効です。DDL 操作は独立して いるため、レコードを無視するということは、トランザクション全体を無視すること と同じです。

EVENTACTIONS では、次の DDL オブジェクトは導出オブジェクトのため、サポートされま せん。

- 索引
- トリガー
- シノニム
- 表に対する RENAME および ALTER TABLE RENAME

7.7.4 DDL パラメータ・オプションの組合せ

DDL パラメータ・オプションの組合せ方の例を次に示します。

DDL &

INCLUDE UNMAPPED & OPTYPE alter & OBJTYPE 'table' & OBJNAME users.tab* &

INCLUDE MAPPED OBJINAME * &

EXCLUDE MAPPED OBJNAME temporary.tab"

この文で組み合されたフィルタ基準では、次のように指定されます。

- 表が "users" によって所有され、その名前が "tab" で始まる場合のみ、TABLE または MAP 文で マップされない (UNMAPPED スコープ) 表に対するすべての ALTER TABLE 文を INCLUDE。
- かつ、TABLE または MAP 文でマップされる (MAPPED スコープ) すべての表に対する DDL 操作 タイプをすべて INCLUDE。
- かつ、表が "temporary" によって所有され、その名前が "tab" で始まる場合のみ、MAPPED ス コープのすべての表に対する DDL 操作タイプをすべて EXCLUDE。

7.8 特別なフィルタのケース

次に、フィルタ条件を作成する場合に注意する必要がある特別なケースを示します。

7.8.1 DDL EXCLUDE ALL

DDL EXCLUDE ALL は、DDL 操作自体のレプリケーションは回避しつつ、Oracle GoldenGate の オブジェクト・メタデータを最新に保つ特別な処理オプションです。Oracle GoldenGate 以外 の方法を使用して DDL をターゲットに適用する一方、Oracle GoldenGate によってデータの変 更をターゲット・オブジェクトにレプリケートする場合、DDL EXCLUDE ALLを使用できま

す。現在のメタデータをオブジェクトの変更として Oracle GoldenGate に提供するため、Oracle GoldenGate プロセスの停止と起動の必要がなくなります。次の特別な条件が DDL EXCLUDE ALL に適用されます。

- DDL EXCLUDE ALL では、INCLUDE 句を使用する必要はありません。
- DDL EXCLUDE ALL を使用する場合、WILDCARDRESOLVE パラメータを IMMEDIATE に設定し、必要に応じてただちに DML を解決できるようにします。

すべての DDL メタデータと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。DDL トリガーによる履歴表への DDL 操作の記録は、手動で無効化しないかぎり継続されます。

7.8.2 暗黙的 DDL

ユーザーによって生成された DDL 操作によって、暗黙的 DDL 操作が生成される場合があります。たとえば、次の文では、Oracle DDL トリガーによって 2 つの異なる DDL 操作が処理されます。

CREATE TABLE customers (custID number, name varchar2(50), address varchar2(75), address2 varchar2(75), city varchar2(50), state (varchar2(2), zip number, contact varchar2(50), areacode number(3), phone number(7), primary key (custID));

最初の(明示的)DDL操作は、CREATE TABLE 文自体です。

2つ目の DDL 操作は、暗黙的な CREATE UNIQUE INDEX 文で、主キーの索引を作成します。この操作は、ユーザー・アプリケーションではなく、データベース・エンジンによって生成されます。

暗黙的 DDL のフィルタリングのガイドライン

DDLパラメータを使用して DDL 操作をフィルタする場合、ターゲットで明示的 DDL によって 暗黙的 DDL が生成されるため、デフォルトでは Oracle Golden Gate によって暗黙的 DDL が除外されます。たとえば、前述の例の CREATE TABLE 文が Replicat によって適用されると、ターゲット・データベースで適切な索引が作成されます。

ただし、DDL トリガーを使用して DDL 操作をフィルタする場合、次の事項に基づいて、フィルタ・ルールで暗黙的 DDL を処理する必要があります。

■ フィルタ・ルールで明示的 DDL を伝播から除外する場合、暗黙的 DDL を除外するルール も作成する必要があります。たとえば、次の例の CREATE TABLE 文は除外するが、暗黙的 CREATE UNIQUE INDEX 文は除外しない場合、ターゲット・データベースは、存在しない表 に索引を作成しようとします。

CREATE TABLE customers (custID number, name varchar2(50), address varchar2(75), address2 varchar2(75), city varchar2(50), state (varchar2(2), zip number, contact varchar2(50), areacode number(3), phone number(7), primary key (custID));

■ フィルタリング・ルールで明示的 DDL の伝播が許可されている場合、暗黙的 DDL を除外する必要はありません。Oracle GoldenGate およびターゲット・データベースによって適切に処理されます。

7.9 Oracle GoldenGate による導出オブジェクト名の処理方法

DDL 操作には、ベース・オブジェクト名を含めることができますが、*導出オブジェクト*名も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、ベース・オブジェクトに関連する機能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持つDDL 文は次のとおりです。

- RENAME および ALTER RENAME
- 索引、シノニムまたはトリガーに対する CREATE および DROP

次の DDL 文について考えてみます。

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

この場合、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 スコープでの TABLE または MAP によるマッピングの対象です。導出オブジェクトは索引で、そ の名前 (hr.indexPayrollDate) が導出名です。

導出名は、ベース・オブジェクトとは別の独自の TABLE または MAP 文でマップできます。また、 MAP 文を1つ使用して両方を処理できます。MAP の場合、ターゲットでの導出オブジェクト名の 変換は次のように処理されます。

7.9.1 ベース・オブジェクトに対する MAP はあるが、導出オブジェクトに 対する MAP はない場合

ベース・オブジェクトに対する MAP 文はあるが、導出オブジェクトに対するものはない場合、 結果は導出オブジェクトの暗黙的マッピングです。DDL 文に MAPPED が含まれる場合、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、ソース文と同じです。たとえば、次のようだとします。

Extract(ソース)

Table hr.tab*;

Replicat(ターゲット)

MAP hr.tab*, TARGET hrBackup.*;

ソース DDL 文は次のようだとします。

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

ターゲットでReplicatによって実行されるCREATE INDEX 文は次のとおりです。

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割り 当てるという一般的な方法に基づきます。導出オブジェクトの名前がソースの文で完全修飾さ れていない場合でも、名前が適切に変換されることが保証されます。また、ベース・オブジェ クトと同じターゲット所有者によって索引が所有される場合、暗黙的マッピングによって、導 出オブジェクト名を明示的にマップする必要がなくなります。

7.9.2 ベース・オブジェクトと導出オブジェクトに対する MAP がある場合

ベース・オブジェクトに対する MAP 文があり、導出オブジェクトに対するものもある場合、結 果は明示的マッピングです。DDL 文に MAPPED が含まれる場合、Replicat により、独自の TARGET 句に従って各オブジェクトの所有者と名前が変換されます。たとえば、次のようだとし ます。

Extract(ソース)

TABLE hr.tab*; TABLE hr.index*;

Replicat(ターゲット)

MAP hr.tab*, TARGET hrBackup.*; MAP hr.index*, TARGET hrIndex.*; ソース DDL 文は次のようだとします。

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

ターゲットで Replicat によって実行される CREATE INDEX 文は次のとおりです。

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

ターゲットの索引が、ベース・オブジェクトとは異なる所有者によって所有される必要がある 場合、またはターゲットでの名前がソースの名前と異なる必要がある場合、明示的マッピング を使用します。

7.9.3 導出オブジェクトに対する MAP はあるが、ベース・オブジェクトに 対する MAP はない場合

導出オブジェクトに対する MAP 文はあるが、ベース・オブジェクトに対するものはない場合、 Replicat ではいずれのオブジェクトに対する名前の変換も行われません。ターゲットの 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、次の方法があります。

- ベース・オブジェクトに対する明示的な MAP 文を使用します。
- 名前に問題がなければ、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 文でマップします。
- 名前の変換方法に応じて、各オブジェクトに対する MAP 文を作成します。

7.9.4 導出オブジェクトとしての新規表

次のものから作成された新規の表の Oracle GoldenGate による処理方法について、次に説明し ます。

- RENAME および ALTER RENAME
- CREATE TABLE AS SELECT

7.9.4.1 RENAME および ALTER TABLE RENAME

RENAME および ALTER TABLE RENAME 操作では、ベース・オブジェクトは常に新規の表の名前で す。次の例では、ベース・オブジェクト名は、"index_paydate" とみなされます。

ALTER TABLE hr.indexPayrollDate RENAME TO index paydate;

または

RENAME hr.indexPayrollDate TO index_paydate;

導出オブジェクト名は "hr.indexPayrollDate" です。

7.9.4.2 CREATE TABLE AS SELECT

CREATE TABLE AS SELECT 文には、基になる任意の数のオブジェクトに影響する SELECT 文と INSERT 文が含まれます。ターゲットで、Oracle GoldenGate によって、AS SELECT 句に対する データがターゲット・データベースから取得されます。

注意: このため、CTAS (CREATE TABLE AS SELECT) 文から作成される Oracle XMLType

の表はサポートされません。XMLType 表の場合、行オブジェクト ID がソースと ターゲットの間で一致する必要がありますが、これは、このシナリオでは保持さ れません。空のCTAS 文(新規表にデータを挿入しない)で作成された XMLType 表 では適切に保持されます。

AS SELECT 句のオブジェクトがターゲット・データベースに存在し、その名前がソースの名前 と同一である必要があります。

MAP 文で Oracle GoldenGate は、新規の表の名前 (CREATE TABLE name) のみを TARGET 指定に マップし、AS SELECT 句の基になるオブジェクトの名前はマップしません。それらのオブジェク トに依存性があり、名前が TARGET 指定に変換された場合は、データに矛盾が生じる可能性があ ります。

次に、ソースの CREATE TABLE AS SELECT 文の例とそれが 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を示します。

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

Replicat の MAP 文は次のとおりです。

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

Replicat によって適用されるターゲットの DDL 文は次のとおりです。

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

AS SELECT * FROM 句の表名は、ソースと同じtab2 のまま(xtab2 ではなく)です。

基になるオブジェクトのデータがソースとターゲットで矛盾のないようにするには、Oracle GoldenGate によるデータのレプリケーション用に構成します。前述の例では、次の文を使用し てこの要件を満たすことができます。

ソース

TABLE a.tab*;

ターゲット

MAPEXCLUDE a.tab2 MAP a.tab*, TARGET a.x*; MAP a.tab2, TARGET a.tab2;

7-9ページの「DDL 内の修飾されていないオブジェクト名の正しい識別」も参照してください。

7.9.5 導出オブジェクトのマッピングの無効化

NOMAPDERIVED オプションを指定して DDLOPTIONS パラメータを使用し、これを含む MAP 文の TARGET 句に従って導出オブジェクトの名前が変換されないようにします。NOMAPDERIVED に よって、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 文はオー バーライドされます。導出オブジェクトを含むソース DDL は、ソースと同じ所有者とオブジェ クト名のターゲットにレプリケートされます。

次の表に、MAP 文がベース・オブジェクトのみに対するものか、導出オブジェクトのみに対す るものか、両方に対するものかに基づいて、MAPDERIVED の結果を NOMAPDERIVED と比較して示 します。

表 7-3 マッピング構成に基づいた、ターゲットでの [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	導出オブジェクトが MAP によって変換 されるか	導出オブジェクトに ベース・オブジェクトの 所有者が割り当てられるか
マップされる ¹	マップされる	MAPDERIVED	はい	いいえ
マップされる	マップされない	MAPDERIVED	いいえ	はい
マップされない	マップされる	MAPDERIVED	いいえ	いいえ
マップされない	マップされない	MAPDERIVED	いいえ	いいえ
マップされる	マップされる	NOMAPDERIVED	いいえ	いいえ
マップされる	マップされない	NOMAPDERIVED	いいえ	いいえ
マップされない	マップされる	NOMAPDERIVED	いいえ	いいえ
マップされない	マップされない	NOMAPDERIVED	いいえ	いいえ

 $^{^1}$ 「マップされる」とは、MAP 文に含まれるということです。

次の例で、MAPDERIVED の結果を NOMAPDERIVED と比較して示します。表 7-4 では、ベース名と 導出名の両方が MAPDERIVED によって変換されるため、ターゲットでトリガーと表の両方が rpt によって所有されます。

表 7-4 導出オブジェクト名のデフォルト・マッピング (MAPDERIVED)

MAP 文	Extract によってキャプチャされる ソース DDL 文	Replicat によって適用される 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 7-5 では、NOMAPDERIVED によって変換が行われないため、トリガーは fin によって所有され ます。

表 7-5 NOMAPDERIVED を使用する場合の導出オブジェクト名のマッピング

MAP 文	Extract によってキャプチャされる ソース DDL 文	Replicat によって適用される ターゲット DDL 文
MAP fin.*, TARGET rpt.*;	<pre>CREATE TRIGGER fin.act_trig ON fin.acct;</pre>	CREATE TRIGGER fin.act_trig ON rpt.acct;

注意: RENAME 文の場合、新しい表名がベース表名とみなされ、古い表名が導 出表名とみなされます。

7.10 DDL 文字列置換の使用

Oracle GoldenGate によって処理される際、DDL 操作内で文字列を置換できます。この機能に よって、ディレクトリ名、コメントおよびデータ構造と直接関係ないその他の文字列の変更と マッピングに対する利便性が増します。たとえば、ある表領域名を別のものに置換したり、コ メント内の文字列を置換できます。文字列の置換は、DDLSUBST パラメータによって制御されま 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: DDLSUBST パラメータ文を作成する前に、7-28 ページの「処理に対する DDL の評価方法」を参照することをお薦めします。

7.11 Replicat によって実行される DDL の伝播の制御

Extract と Replicat の両方で DDL 操作が発行されます。

- Extract は ALTER TABLE 文を発行してログ・グループを作成します。
- Replicat は DDL 文をターゲットに適用します。

Oracle GoldenGate DDL 操作を識別するために、次のコメントが各 Extract と Replicat DDL 文 の一部となります。

/* GOLDENGATE DDL REPLICATION */

DDLOPTIONS パラメータによって、Replicat の DDL を伝播するかどうかが制御されます。

- GETREPLICATES および IGNOREREPLICATES オプションによって、Replicat の DDL 操作を Extract でキャプチャするか無視するかが制御されます。デフォルトは IGNOREREPLICATES です。
- GETAPPLOPS および IGNOREAPPLOPS オプションによって、Replicat 以外のアプリケーション(ビジネス・アプリケーション)の DDL をキャプチャするか無視するかが制御されます。

デフォルトでは、DDL がソースに戻されないように、ローカル Replicat によってローカル・ データベースに適用される DDL は Extract で無視されます。しかし、レプリケーション用に構 成されているその他の DDL はすべてキャプチャされます。デフォルトの DDLOPTIONS 構成は次 のとおりです。

DDLOPTIONS GETAPPLOPS, IGNOREREPLICATES

この動作は変更できます。次の項を参照してください。

- 7-26 ページの「アクティブ / アクティブ (双方向)構成での DDL の伝播」
- 7-27 ページの「カスケード構成での DDL の伝播」

7.11.1 アクティブ / アクティブ (双方向)構成での DDL の伝播

Oracle GoldenGate では、2 つのシステム間のアクティブ / アクティブ DDL レプリケーション がサポートされます。アクティブ/アクティブ双方向レプリケーションの場合、Oracle GoldenGate プロセスで次のように構成される必要があります。

- 一方のシステムでビジネス・アプリケーションによって実行される DDL を他方のシステム にレプリケートし、同期を保つ必要があります。この要件を満たすには、両方のシステム の Extract パラメータ・ファイルで DDLOPTIONS 文に GETAPPLOPS オプションを含めます。
- 2. 一方のシステムで Replicat によって適用される DDL は、ローカルの Extract によってキャ プチャされ、他方のシステムに戻される必要があります。この要件を満たすには、両方の システムの Extract パラメータ・ファイルで DDLOPTIONS 文に GETREPLICATES オプションを 使用します。

注意: ループバックが起こらないよう、内部的な Oracle GoldenGate トーク ンによって、実際の Replicat の DDL 文自体は無視されます。 Replicat の DDL を元のシステムに戻す目的は、着信 DML の受信に備えて、そのシステ ムの Replicat がオブジェクト・メタデータ・キャッシュを更新し、新しいメ タデータを持つことです。図 7-1 を参照してください。

3. 各 Replicat は、キャプチャされた Replicat DDL 文がリモート Extract から届くたびに、そのオブジェクト・メタデータ・キャッシュを更新するように構成する必要があります。この要件を満たすには、両方のシステムの Replicat パラメータ・ファイルで DDLOPTIONS 文に UPDATEMETADATA オプションを使用します。

その結果、DDLOPTIONS 文は次のようになります。

Extract(プライマリおよびセカンダリ)

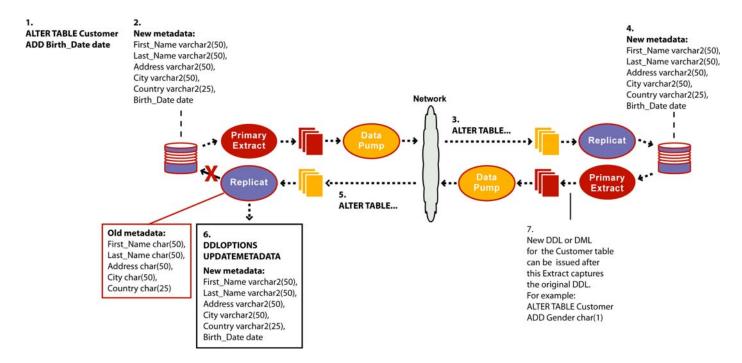
DDLOPTIONS GETREREPLICATES, GETAPPLOPS

Replicat(プライマリおよびセカンダリ)

DDLOPTIONS UPDATEMETADATA

警告:元の DDL と同じオブジェクトに対して DDL または DML を発行する前に、元の DDL がリモート・システムにレプリケートされ、そのシステムの Extract によって再度キャプチャされる時間を考慮します。これによって、各操作が元のシステムの Replicat に正しい順序で届くことを保証し、メタデータの非一貫性によって DML エラーが発生することを防止します。詳細は、図 7-1 を参照してください。

図 7-1 Replicat オブジェクト・メタデータ・キャッシュが更新される DDL のラウンドトリップ・パス



7.11.2 カスケード構成での DDL の伝播

カスケード構成では、各中間システムの Extract パラメータ・ファイルで DDLOPTIONS に次の設定を使用します。この構成では、Extract で中間システムの Replicat から DDL が強制的にキャプチャされ、次のシステム・ダウンストリームにカスケードされます。

DDLOPTIONS GETREPLICATES, IGNOREAPPLOPS

7.12 サプリメンタル・ログ・グループの自動追加

DDLOPTIONS パラメータに ADDTRANDATA オプションを使用し、次ことを実行できます。

- CREATE TABLE で作成された新規表に対して Oracle のサプリメンタル・ロギングを自動的に 有効にします。
- ALTER TABLE の対象の表の Oracle のサプリメンタル・ロギングを更新し、列を追加または 削除します。
- 名前が変更された表の Oracle のサプリメンタル・ロギングを更新します。
- 一意キーまたは主キーが追加または削除された表の Oracle のサプリメンタル・ロギングを 更新します。

デフォルトでは、サプリメンタル・ロギングを追加する ALTER TABLE は、GETREPLICATES パラ メータが使用されていないかぎり、ターゲットにレプリケートされません。

このオプションの詳細は、『Oracle GoldenGate Windows and UNIX リファレンス・ガイド』を 参照してください。

7.13 レプリケートされた DDL からのコメントの削除

DDLOPTIONS パラメータに REMOVE COMMENTS BEFORE および REMOVE COMMENTS AFTER オプションを 使用し、ソース DDL に使用されたコメントがターゲット DDL に含まれないようにできます。 デフォルトでは、コメントは削除されず、文字列の置換に使用できます。

このオプションの詳細は、『Oracle GoldenGate Windows and UNIX リファレンス・ガイド』を 参照してください。

7.14 IDENTIFIED BY パスワードのレプリケート

DDLOPTIONS パラメータに DEFAULTUSERPASSWORD および REPLICATEPASSWORD | NOREPLICATE PASSWORD オプションを使用し、レプリケートされる {CREATE | ALTER} USER name IDENTIFIED BY password 文のパスワードの処理方法を制御します。これらのオプションは一緒 に使用する必要があります。

これらのオプションの詳細は、『Oracle GoldenGate Windows and UNIX リファレンス・ガイド』 を参照してください。

7.15 処理に対する DDL の評価方法

次に、Oracle GoldenGate によるソースおよびターゲット・システムでの DDL 文の処理方法に ついて説明します。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について説明します。

Extract

- 1. Extract は、DDL 文をキャプチャします。
- 2. Extract は、コメントがあれば、メインの文から分離します。
- 3. Extract は、DDL パラメータを検索します。(この例では、存在するものとします。)
- 4. Extract は、IGNOREREPLICATES パラメータを検索します。これがあり、Replicat がこのシス テムでこの DDL を生成した場合、Extract はその DDL 文を無視します。(この例では、こ のシステムでは Replicat 操作がないものとします。)
- 5. Extract は、DDL 文が RENAME かどうかを判断します。そうである場合、名前変更に内部的 にフラグが付けられます。
- **6.** Extract は、ベース・オブジェクト名と導出オブジェクト名(あれば)を取得します。
- 7. 文が RENAME の場合、Extract はこれを ALTER TABLE RENAME に変更します。

- 8. Extract は、DDLOPTIONS REMOVECOMMENTS BEFORE パラメータを検索します。これがある場 合、Extract は DDL 文からコメントを削除しますが、INSTR または INSTRCOMMENTS を使用 する DDL INCLUDE または DDL EXCLUDE 句がある場合はコメントを格納します。
- 9. Extract は DDL のスコープ (MAPPED、UNMAPPED または OTHER) を判断します。
 - 操作とオブジェクト・タイプがマッピングに対してサポートされており、ベース・オ ブジェクト名や導出オブジェクト名 (RENAME の場合) が TABLE パラメータに含まれて いる場合、MAPPEDです。
 - 操作とオブジェクト・タイプがマッピングに対してサポートされておらず、ベース・ オブジェクト名や導出オブジェクト名 (RENAME の場合)が TABLE パラメータに含まれ ていない場合、UNMAPPEDです。
 - これ以外の場合、操作はOTHERと識別されます。
- **10.** Extract は、DDL パラメータに INCLUDE 句と EXCLUDE 句があるかどうかをチェックし、これ らの句の DDL パラメータ基準を評価します。INCLUDE または EXCLUDE が TRUE と評価される には、すべてのオプションが TRUE と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 と評価される場合、Extract は DDL 文を破棄し、別の DDL 文を評価 します。この場合、処理の手順が最初から始まります。
 - INCLUDE 句が TRUE と評価される場合、または DDL パラメータに INCLUDE 句も EXCLUDE 句 も含まれていない場合、Extract は DDL 操作を含めて、処理ロジックが続けられます。
- 11. Extract は、DDLSUBST パラメータを検索し、INCLUDE 句および EXCLUDE 句を評価します。そ れらの句の基準が最終的に TRUE になる場合、Extract は文字列の置換を実行します。 Extract は、パラメータ・ファイル内の各 DDLSUBST パラメータに対して DDL 文を評価しま す。true の DDLSUBST 指定すべてについて、DDLSUBST パラメータがファイルにリストされ ている順に、Extract によって文字列の置換が行われます。
- **12.** DDLSUBT が処理されたため、Extract は REMOVECOMMENTS AFTER パラメータを検索します。 これがある場合、Extract は DDL 文からコメントを削除します。
- 13. Extract は、DDLOPTIONS ADDTRANDATA を検索します。これがある場合、操作が CREATE TABLE であれば、Extract は ALTER TABLE name ADD SUPPLEMENTAL LOG GROUP コマンドを 表に対して発行します。
- **14.** Extract は、DDL 文を証跡に書き込みます。

Replicat

- 1. Replicat は、DDL 文を証跡から読み取ります。
- Replicat は、コメントがあれば、メインの文から分離します。
- Replicat は、DDLOPTIONS REMOVECOMMENTS BEFORE を検索します。これがある場合、 Replicat は DDL 文からコメントを削除します。
- 4. Replicat は、DDL 同期スコープを評価し、DDL が名前のマッピングに適しているかを判断 します。そうでないものは、OTHER スコープです。
- 5. Replicat は、パラメータ・ファイルの MAP 文を評価します。(証跡から読み取った)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 文に含まれる場合、操作は MAPPED スコープとしてマークされます。そうではない場合、UNMAPPED スコープとしてマー クされます。
- 6. Replicat は、ソースのベース・オブジェクト名を、MAP 文の TARGET 句に指定されたベー ス・オブジェクト名で置き換えます。
- 7. 導出オブジェクトがある場合、Replicat は DDLOPTIONS MAPDERIVED を検索します。これが ある場合、Replicat は、ソースの導出名を MAP 文のターゲット導出名で置き換えます。

- 8. Replicat は、DDL パラメータに INCLUDE 句と EXCLUDE 句があるかどうかをチェックし、こ れらの句に含まれる DDL パラメータ基準を評価します。 INCLUDE または EXCLUDE が TRUE と 評価されるには、すべてのオプションが TRUE と評価される必要があります。次のようにな
 - EXCLUDE が TRUE と評価される場合、Replicat は DDL 文を破棄し、別の DDL 文の評価 を始めます。この場合、処理の手順が最初から始まります。
 - INCLUDE 句が TRUE と評価される場合、または DDL パラメータに INCLUDE 句も EXCLUDE 句 も含まれていない場合、Replicat は DDL 操作を含めて、処理ロジックが続けられま す。
- 9. Replicat は、DDLSUBST パラメータを検索し、INCLUDE 句および EXCLUDE 句を評価します。 それらの句のオプションが最終的に TRUE になる場合、Replicat は文字列の置換を実行しま す。Replicat は、パラメータ・ファイル内の各 DDLSUBST パラメータに対して DDL 文を評 価します。true の DDLSUBST 指定すべてについて、DDLSUBST パラメータがファイルにリス トされている順に、Replicat によって文字列の置換が行われます。
- **10.** DDLSUBT が処理されたため、Replicat は REMOVECOMMENTS AFTER パラメータを検索します。 これがある場合、Replicat は DDL 文からコメントを削除します。
- 11. Replicat は、ターゲット・データベースで DDL 操作を実行します。
- 12. エラーがなければ、Replicat は次の DDL 文を処理します。エラーがある場合、Replicat は 次の手順を実行します。
- **13.** Replicat は、Replicat DDLERROR パラメータの INCLUDE ルールと EXCLUDE ルールを、パラ メータ・ファイルに出現する順に分析します。Replicat は、エラー・コードに対するルー ルを検出した場合、指定されたエラー処理を適用し、検出されない場合は DEFAULT 処理を 適用します。
- 14. エラー処理によって DDL 文が正常完了とならない場合、Replicat は、ルールでの指定に応 じて異常終了、操作の無視、または操作の破棄のいずれかを実行します。

注意:同じソースに対して複数のターゲットが MAP 文にある場合、ターゲッ トごとに処理ロジックが実行されます。

7.16 DDL 処理エラーの処理

DDLERROR パラメータを使用して、Extract によって検出されたオブジェクトのメタデータが見 つからない場合のエラーおよび DDL をターゲット・データベースに適用する際に発生する Replicat エラーを処理します。DDLERROR オプションを使用して、ほとんどのエラーをデフォル トの方法(処理の停止など)で処理し、他のエラーを特定の方法で処理できます。予測されるす べてのエラーが処理されるよう、同一のパラメータ・ファイルで複数の DDLERROR インスタンス を使用できます。

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。

7.17 DDL トリガー・エラーの処理

params.sql 非実行可能スクリプトで、ソース DDL の失敗または成功に関連する Oracle GoldenGate DDL トリガーの失敗を処理します。params.sql スクリプトは、ルートの Oracle GoldenGate ディレクトリにあります。使用するパラメータは次のとおりです。

ddl fire error in trigger: TRUE に設定されている場合、Oracle GoldenGate DDL トリガーの失敗は、Oracle GoldenGate エラー・メッセージおよびデータベース・エラー・ メッセージとともにソースのエンドユーザー・アプリケーションに示されます。ソースの 処理は失敗します。

FALSE に設定されている場合、エラーは発生せず、メッセージは、Oracle Golden Gate ディレクトリのトリガー・トレース・ファイルに書き込まれます。ソースの処理は成功しますが、DDL はレプリケートされません。後続のデータ変更が古いターゲット・オブジェクトの構造に適合しない場合、ターゲット・アプリケーション最終的には失敗します。デフォルトは FALSE です。

■ __ddl__cause__error: TRUE に設定されている場合、故意にエラーを発生させ、トリガーのエラー・レスポンスをテストします。エラーを生成するために、Oracle GoldenGate では、例外処理を使用せずに 0 (ゼロ) 行を SELECT しようとします。テストが終了したら、このフラグをデフォルトの FALSE に戻します。

7.18 DDL レポート情報の表示

Oracle GoldenGate では、Extract および Replicat のレポートの最後に、DDL に関する基本的な 統計がデフォルトで表示されます。拡張 DDL レポートを有効にするには、REPORT オプション を指定して DDLOPTIONS パラメータを使用します。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 によって処理された DDL 操作の段階的履歴
- 使用されている DDL のフィルタリングと処理のパラメータ

拡張 DDL レポート情報によって、レポート・ファイルのサイズは大きくなりますが、トラブルシューティングやサプリメンタル・ロギングを追加する ADDTRANDATA がいつ適用されたかを確認する場合などの特定の状況で役立ちます。

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

7.18.1 Extract DDL レポート

Extract レポートには、次の情報がリストされます。

- キャプチャされた各 DDL 操作の構文全体、開始と終了の SCN、Oracle インスタンス、DDL 順序番号(履歴表の SEQNO 列)および操作のサイズ(バイト)。
- 処理基準がどのように操作に適用されたかを示す後続のエントリ(文字列置換、INCLUDE と EXCLUDE のフィルタリングなど)。
- 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。

Extract レポートから取得された次の例は、含まれた操作と除外された操作を示しています。含まれた操作にはレポート・メッセージがありますが、除外された操作にはありません。

2011-01-20 15:11:41 GGS INFO 2100 DDL found, operation [create table myTable (myId number (10) not null,

myNumber number,

myString varchar2(100),

myDate date,

primary key (myId)

)], start SCN [1186754], commit SCN [1186772] instance [test11g (1)], DDL seqno [4134].

2011-01-20 15:11:41 GGS INFO 2100 DDL operation included [INCLUDE OBJNAME myTable*], optype [CREATE], objtype [TABLE], objname [QATEST1.MYTABLE].

2011-01-20 15:11:41 GGS INFO 2100 DDL operation written to extract trail file.

2011-01-20 15:11:42 GGS INFO 2100 Successfully added TRAN DATA for table with the key, table [QATEST1.MYTABLE], operation [ALTER TABLE "QATEST1"."MYTABLE" ADD SUPPLEMENTAL LOG GROUP "GGS_MYTABLE_53475" (MYID) ALWAYS /* GOLDENGATE_DDL_REPLICATION */].

2011-01-20 15:11:43 GGS INFO 2100 DDL found, operation [create table myTableTemp

```
(
    vid varchar2(100),
    someDate date,
   primary key (vid)
)], start SCN [1186777], commit SCN [1186795] instance [test11g (1)], DDL segno
[4137].
2011-01-20 15:11:43 GGS INFO
                                2100 DDL operation excluded [EXCLUDE OBJNAME
myTableTemp OPTYPE CREATE], optype [CREATE], objtype [TABLE], objname
```

7.18.2 プロセス・レポートの統計

GGSCIで SEND コマンドを使用して、DDL 処理に関する現在の統計を Extract および Replicat のレポートに送信できます。

SEND {EXTRACT | REPLICAT} group REPORT 統計には、次の合計が表示されます。

すべての DDL 操作

[QATEST1.MYTABLETEMP].

- MAPPED スコープの操作
- UNMAPPED スコープの操作
- OTHER スコープの操作
- 除外された操作(含まれた操作を引いた操作の数)
- 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)
- 無視されたエラー (Replicat のみ)

7.19 DDL 履歴表のメタデータの表示

GGSCIで DUMPDDL コマンドを使用して、DDL 履歴表に含まれている情報を表示します。この 情報は、独自の形式で格納されていますが、判読可能な形式で画面に出力したり、問合せ可能 な一連の SOL 表にエクスポートすることができます。DDL 履歴表の情報は、Extract プロセス によって使用されるものと同じです。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。

7.20 DDL 処理のトレース

Oracle GoldenGate テクニカル・サポートでサポート・ケースを開く場合、トレースを有効化 するよう求められることがあります。次のパラメータによって DDL トレースが制御されます。

- 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。

7.21 DDL トリガーのトレース

Oracle GoldenGate DDL トリガーのアクティビティをトレースするには、次のツールを使用し

ggs ddl trace.logトレース・ファイル:Oracle GoldenGate では、OracleのUSER DUMP DEST ディレクトリにトレース・ファイルを作成します。RAC では、各ノードにそのノード の DDL トレースを取得する独自のトレース・ファイルが存在します。 トレース・ファイル は、次のように問い合せることができます。

select value from sys.v \$parameter where name = 'user dump dest';

- ddl tracelevel スクリプト:このスクリプトを編集して実行し、トレース・レベルを設定 します。値 None では、致命的エラーとインストール・ログ以外、DDL トレースは生成さ れません。デフォルト値の0では、最小限のトレース情報が生成されます。値1または2 では、はるかに多くの情報がトレース・ファイルに生成されます。サポート・ケースの一 環として Oracle GoldenGate テクニカル・サポートのアナリストから求められないかぎり、 1または2は使用しないでください。
- ddl cleartrace スクリプト: このスクリプトを定期的に実行して、トレース・ファイルの 増大によりディスク領域が過度に使用されることを防止します。ファイルは削除されます が、Oracle GoldenGate によって別のファイルが作成されます。DDL トリガーは、Oracle ディレクトリの領域が少なくなるとトレース・ファイルへの書込みを停止し、領域が再度 使用可能になると書込みを再開します。このスクリプトは、Oracle GoldenGate ディレク トリにあります。スクリプトを実行する前に、トレース・ファイルをバックアップします。

Oracle GoldenGate レプリケーションの インスタンス化と開始

この章では、ターゲット・データの初期ロードの構成、レプリケーションのインスタンス化に 必要なプロセスの追加およびインスタンス化の実行の手順について説明します。これらの手順 に求められる結果は、ソースとターゲットのデータの整合性をとること(初期同期と呼ばれる) と、進行中のトランザクション変更を Oracle Golden Gate でキャプチャおよび配信して整合性 を保つことです。

8.1 基本的な Oracle GoldenGate インスタンス化手順の概要

次の手順では、第4章で構成した基本的なレプリケーション環境をインスタンス化する方法を 示します。手順は次のとおりです。

- インスタンス化の前提条件
- インスタンス化の手順の効率化
- 初期ロードの構成
- Extract のマイニング・データベースでの登録
- 変更キャプチャ・プロセスおよび変更配信プロセスの追加
- ターゲットのインスタンス化の実行
- インスタンス化後の処理の監視
- Oracle GoldenGate 環境のバックアップ

8.2 インスタンス化の前提条件

Oracle GoldenGate プロセスまたはネイティブ・データベース・ロード・プロセスを起動する 前に、次の手順を実行する必要があります。

8.2.1 変更のキャプチャと配信の構成

レプリケーション環境のインスタンス化の準備が整うまでの間に、第4章の説明のように、完 成したパラメータ・ファイルを使用してすべての Extract および Replicat プロセス・グループ が構成されている必要があります。

また、このマニュアルに記載されている他のすべての設定要件が満たされている必要がありま す。

8.2.2 競合処理の追加

ソース・データベースが初期ロード時にアクティブな場合、Replicat パラメータ・ファイルに 競合処理ロジックを追加する必要があります。このロジックでは、静的データのターゲット表 へのロードと、Oracle GoldenGate によるターゲット表へのトランザクション変更のレプリ ケートが同時に行われるために発生する競合が処理されます。

競合を処理するには、HANDLECOLLISIONS パラメータを Replicat パラメータ・ファイルに追加し て次のものを解決します。

- すでに存在する行の INSERT 操作
- 存在しない行の UPDATE および DELETE 操作

HANDLECOLLISIONS 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。

8.2.3 DDL 処理の無効化

初期ロードを実行する前に、DDL の抽出とレプリケーションを無効にします。DDL 処理は、 Extract および Replicat のパラメータ・ファイルの DDL パラメータによって制御されます。 DDL サポートの詳細は、第7章を参照してください。

8.2.4 ターゲット表の準備

ロードを高速に行い、エラーが起こらないようにするための推奨事項は、次のとおりです。

- **データ**: ターゲット表が空であることを確認します。そうでない場合、行の重複エラーや既 存の行とロードされる行との間に競合が発生することがあります。
- 制約:外部キー制約とチェック制約をまだ無効にしていない場合は、無効にします。外部 キー制約によってエラーが発生し、チェック制約によってロード処理が遅くなることがあ ります。その他の要件については、5-1ページの「ソース表とターゲット表の整合性制約の 準備」を参照してください。
- 索引: ターゲット表から索引を削除します。索引は、初期ロード・プロセスによって実行さ れる挿入には不要で、速度を大幅に低下させます。ロードの完了後、索引を再度追加でき ます。
- キー: HANDLECOLLISIONS 関数を使用して、ロードによる増分データ変更を調整するには、 各ターゲット表に主キーと一意キーが必要です。アプリケーションでキーを作成できない 場合、TABLE および MAP パラメータの KEYCOLS オプションを使用し、Oracle GoldenGate の 目的に合せて列を代替キーとして指定します。キーを作成できない場合、影響を受ける ロードのソース表を停止する必要があります。

8.3 インスタンス化の手順の効率化

インスタンス化プロセスをより効率的にするための推奨事項は、次のとおりです。

8.3.1 プロセス・グループ間でのパラメータの共有

変更同期パラメータ・ファイルで使用するパラメータの一部は、初期ロードの Extract および Replicat パラメータ・ファイルにも必要です。共通性を利用するには、次の方法を使用します。

- 共通のパラメータを一方のパラメータ・ファイルから他方にコピーします。
- 共通のパラメータを中央のファイルに格納し、各パラメータ・ファイルで OBEY パラメータ を使用して取得します。
- 共通のパラメータに対する Oracle GoldenGate マクロを作成し、MACRO パラメータを使用し て各パラメータ・ファイルからマクロをコールします。

8.3.2 パラレル・プロセスの使用

パラレル初期ロード・プロセスを構成して、初期ロードをより高速に実行できます。外部キー 関係のある表を同じプロセスのセット内に保持することが重要です。異なるプロセスのセット を使用して大規模な表と小規模な表を分離したり、単にロードを任意の数のプロセス・セット に配分することができます。パラレル・プロセスを適切に構成するには、『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』を参照してください。

8.4 初期ロードの構成

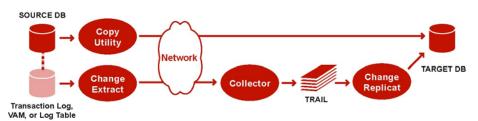
Oracle GoldenGate では、Oracle 専用の次のロード方法がサポートされます。

- データベース・ユーティリティを使用したロード
- 入力ファイルから SOL*Loader へのロード
- データベース・ユーティリティを使用したロード

方法を選択してその構成手順に従い、ロード・プロセスとパラメータ・ファイルを作成します。

8.4.1 データベース・ユーティリティを使用したロード

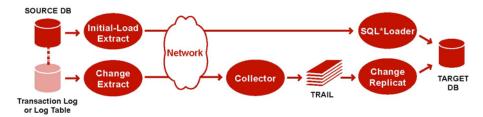
図 8-1



この方法では、データベースのコピー・ユーティリティを使用して、ターゲット・データを構 築します。データベース・ユーティリティでデータの静的コピーを作成および適用しながら、 変更同期 Extract グループを起動して進行中のデータ変更を抽出します。コピーが終了したら、 変更同期 Replicat グループを起動して、コピーの適用中に変更された行を再同期します。これ 以降、Extract と Replicat の両方が継続的に実行され、データ同期が維持されます。

この方法では、初期ロード・プロセスの特別な構成は必要ありません。第4章で構成した変更 同期プロセス・グループを使用するだけです。

8.4.2 SQL*Loader へのダイレクト・バルク・ロード



この方法では、Oracle GoldenGate 初期ロード Extract を構成して実行し、全ソース・レコード を抽出して初期ロード Replicat タスクに直接送信します。初期ロード Replicat タスクは SOL*Loader と通信し、ダイレクトパス・バルク・ロードでデータをロードします。データ・ マッピングと変換は、初期ロード Extract または初期ロード Replicat、あるいはその両方で行え ます。ロード時、第4章で構成した変更同期グループによって増分変更がレプリケートされ、 ロードの結果に反映されます。

制限

- この方法では、LOB または LONG のデータの抽出はサポートされません。かわりの方法につ いては、「入力ファイルから SQL*Loader へのロード」を参照してください。
- この方法では、サイズに関係なく、LOB を含むマテリアライズド・ビューはサポートされま せん。データの暗号化もサポートされません。

SQL*Loader へのダイレクト・バルク・ロードを構成する手順

- 1. LOCK ANY TABLE をターゲット Oracle データベースの Replicat データベース・ユーザーに 付与します。
- 2. ソースおよびターゲット・システムで GGSCI を実行します。
- **3.** 両方のシステムで Manager を起動します。

START MANAGER

4. ソース・システムで、初期ロード Extract を作成します。

ADD EXTRACT initial-load Extract name, SOURCEISTABLE

説明:

- initial-load Extract name は、8 文字以内の初期ロード Extract の名前です。
- SOURCEISTABLE では、Extract に全レコードをソース表から直接読み取るように指示し ます。
- 5. ソース・システムで、初期ロード Extract パラメータ・ファイルを作成します。

EDIT PARAMS initial-load Extract name

6. 次に示す順序で初期ロード Extract パラメータを入力します。パラメータ文ごとに新しい行 を開始します。入力値は環境によって異なります。詳細は、表 8-1 を参照してください。

EXTRACT initext

USERID ogg, PASSWORD AACAAAAAAAAAAJAUEUGODSCVGJEEIUGKJDJTFNDKEJFFFTC & AES128, ENCRYPTKEY securekey1

RMTHOST fin1, MGRPORT 7809 ENCRYPT AES192, KEYNAME securekey2 RMTTASK replicat, GROUP initrep

TABLE hr.*;

表 8-1 SQL*Loader ヘダイレクト・バルク・ロードするための初期ロード Extract パラメータ

パラメータ	説明	
EXTRACT initial-load Extract name	ADD EXTRACT で指定した初期ロード Extract の名前を指定します。	
USERID user id, PASSWORD pw [encryption options]	データベース資格証明と暗号化情報(必要な場合)を指定します。変更同期プロセス用に作成したのと同じユーザーを使用できます。	
RMTHOST hostname, MGRPORT portnumber[, ENCRYPT algorithm KEYNAME keyname]	ターゲット・システム、Manager が実行されるポートおよび TCP/IP 全体でのデータの暗号化 (オプション) を指定しま す。	
RMTTASK replicat, GROUP initial-load Replicat name	プロセス・タイプ (Replicat) と初期ロード Replicat の名前を 指定します。ターゲット・システムの Manager に、初期ロード Replicat を 1 回かぎりのタスクとして動的に起動するよう 指定します。	
TABLE owner.table;	初期データ抽出用の所有者と表またはワイルドカードで指定した表を指定します。ワイルドカード指定からオブジェクトを除外するには、TABLEEXCLUDEパラメータを使用します。	

- 7. ファイルを保存し、閉じます。
- 8. ターゲット・システムで、初期ロード Replicat を作成します。

ADD REPLICAT initial-load Replicat name, SPECIALRUN

説明:

- initial-load Replicat name は、初期ロード Replicat タスクの名前です。
- SPECIALRUN では、初期ロード Replicat を、継続的プロセスではなく、ワンタイム・タ スクとして識別します。
- 9. ターゲット・システムで、初期ロード Replicat パラメータ・ファイルを作成します。 EDIT PARAMS initial-load Replicat name
- 10. 次に示す順序で初期ロード Replicat パラメータを入力します。パラメータ文ごとに新しい 行を開始します。詳細は、表 8-2 を参照してください。

REPLICAT initrep

USERID ogg, PASSWORD AACAAAAAAAAAAAJAUEUGODSCVGJEEIUGKJDJTFNDKEJFFFTC & AES128, ENCRYPTKEY securekey1

BLIL'KL'UAD

ASSUMETARGETDEFS

MAP hr.*, TARGET hr2.*;

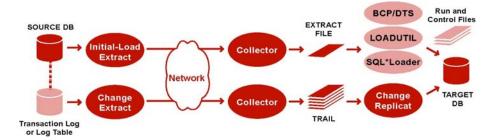
表 8-2 SQL*Loader ヘダイレクト・バルク・ロードするための初期ロード Replicat パラメータ

パラメータ	説明
REPLICAT initial-load Replicat name	ADD REPLICAT で指定した初期ロード Replicat タスクの名前を指定します。
USERID user id, PASSWORD pw [encryption options]	データベース資格証明と暗号化情報(必要な場合)を指定します。 変更同期プロセス用に作成したのと同じユーザーを使用できます。
BULKLOAD	Oracle SQL*Loader インタフェースと直接対話するよう Replicat に 指示します。
ASSUMETARGETDEFS	ソースとターゲットの表が、セマンティクスを含めて同一であるものとします。ソースとターゲットの定義が異なる場合、変更同期と初期ロードの両方のプロセスが使用するソース定義ファイルを作成して指定する必要があります。詳細は、『Windows and UNIX 管理者ガイド』を参照してください。
MAP owner.table, TARGET	ソース表とターゲット表の関係を指定します。
owner.table;	■ owner は、スキーマ名です。
	table は、表の名前または複数の表に対するワイルドカード定 義です。ワイルドカード指定からオブジェクトを除外するに は、MAPEXCLUDEパラメータを使用します。

- 11. パラメータ・ファイルを保存し、閉じます。
- 12. 8-8 ページの「Extract のマイニング・データベースでの登録」に進みます。

8.4.3 入力ファイルから SQL*Loader へのロード

図 8-2



この方法では、初期ロード Extract はソース表からソース・レコードを抽出し、抽出ファイルに 外部 ASCII 形式で書き込みます。ファイルは SQL*Loader によって読み取られます。ロード時、 第4章で構成した変更同期グループによって増分変更がレプリケートされ、ロードの結果に反 映されます。ロード処理の一環として、Oracle GoldenGate で初期ロード Replicat を使用して、 データベース・ユーティリティに必要な実行ファイルと制御ファイルを作成します。制御ファ イルは動的に生成され、変換ルールをあらかじめ構成できないため、データ変換は初期ロード Extract によってソース・システムで行われる必要があります。

ファイルから SQL*Loader へのロードを構成する手順

- ソースおよびターゲット・システムで GGSCI を実行します。
- 2. 両方のシステムで Manager を起動します。

START MANAGER

- 3. ソース・システムで、初期ロード Extract パラメータ・ファイルを作成します。 EDIT PARAMS initial-load Extract name
- 4. 次に示す順序で初期ロード Extract パラメータを入力します。パラメータ文ごとに新しい行 を開始します。入力値は環境によって異なります。詳細は、表 8-3 を参照してください。

SOURCEISTABLE

USERID ogg, PASSWORD AACAAAAAAAAAAJAUEUGODSCVGJEEIUGKJDJTFNDKEJFFFTC & AES128, ENCRYPTKEY securekey1

RMTHOST fin1, MGRPORT 7809 ENCRYPT AES192, KEYNAME securekey2

ENCRYPTTRAIL AES192, KEYNAME mykey1

RMTFILE /ggs/dirdat/ie

FORMATASCII, SQLLOADER

TABLE hr.*;

表 8-3 ファイルから SQL*Loader ヘロードするための初期ロード Extract パラメータ

パラメータ	説明
SOURCEISTABLE	Extract を、ソース表から直接レコードを抽出する初期ロード・プロセスとして指定します。
USERID user id, PASSWORD pw [encryption options]	データベース資格証明と暗号化情報(必要な場合)を指定します。 変更同期プロセス用に作成したのと同じユーザーを使用できます。
RMTHOST hostname, MGRPORT portnumber[, ENCRYPT algorithm KEYNAME keyname]	ターゲット・システム、Manager が実行されるポートおよび TCP/IP 全体でのデータの暗号化 (オプション)を指定します。
ENCRYPTTRAIL encryption options	リモート・ファイルでデータを暗号化します。暗号化のオプション 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。
RMTFILE path name	Extract によって作成され、ロード・データが書き込まれる抽出 ファイルの絶対パスまたはフルパス名を指定します。
FORMATASCII, SQLLOADER	SQL*Loader に対応した、固定長で ASCII 形式のリモート・ファイルを作成します。FORMATASCII の制限とオプションについては、『Oracle GoldenGate Windows and UNIX リファレンス・ガイド』を参照してください。
TABLE owner.table;	初期データ抽出用の所有者と表またはワイルドカードで指定した表を指定します。ワイルドカード指定からオブジェクトを除外するには、TABLEEXCLUDE パラメータを使用します。

- 5. パラメータ・ファイルを保存し、閉じます。
- 6. ターゲット・システムで、初期ロード Replicat パラメータ・ファイルを作成します。 EDIT PARAMS <initial-load Replicat name>
- 7. 次に示す順序で初期ロード Replicat パラメータを入力します。パラメータ文ごとに新しい 行を開始します。詳細は、表 8-4 を参照してください。

GENLOADFILES sqlldr.tpl

USERID ogg, PASSWORD AACAAAAAAAAAAJAUEUGODSCVGJEEIUGKJDJTFNDKEJFFFTC & AES128, ENCRYPTKEY securekey1

DECRYPTTRAIL AES192, KEYNAME mykey1

EXTFILE /ggs/dirdat/ie

ASSUMETARGETDEFS

MAP hr.*, TARGET hr2.*;

表 8-4 ファイルから SQL*Loader ヘロードするための初期ロード Replicat パラメータ

パラメータ	説明
GENLOADFILES template file	データベース・ユーティリティ用の実行ファイルと制御ファイルを 生成します。このパラメータの使用の詳細は、『Oracle GoldenGate <i>Windows and UNIX リファレンス・ガイド</i> 』を参照してください。
USERID user id, PASSWORD pw [encryption options]	USERID では、データベース資格証明と暗号化情報(必要な場合)を 指定します。変更同期プロセス用に作成したのと同じユーザーを使 用できます。
DECRYPTTRAIL encryption options	入力抽出ファイルのデータを復号化します。暗号化のオプションについては、『Oracle GoldenGate <i>Windows and UNIX リファレンス・ガイド</i> 』を参照してください。

表 8-4 ファイルから SQL*Loader ヘロードするための初期ロ	– l⁵ Repl	licat パラメータ	2 (締き)
--------------------------------------	-----------	-------------	--------

パラメータ	説明
EXTFILE path name	Extract パラメータ RMTFILE で指定した抽出ファイルを指定します。
EXTTRAIL path name	
ASSUMETARGETDEFS	ソースとターゲットの表が、セマンティクスを含めて同一であるものとします。ソースとターゲットの定義が異なる場合、変更同期と初期ロードの両方のプロセスが使用するソース定義ファイルを作成して指定する必要があります。詳細は、『Oracle GoldenGate Windows and UNIX 管理者ガイド』を参照してください。
MAP owner.table, TARGET owner.table;	ソース表とターゲット表の関係を指定します。
	■ owner は、スキーマ名です。
	■ table は、表の名前または複数の表に対するワイルドカード定 義です。ワイルドカード指定からオブジェクトを除外するに は、MAPEXCLUDE パラメータを使用します。

- **8.** パラメータ・ファイルを保存し、閉じます。
- 9. 「Extract のマイニング・データベースでの登録」に進みます。

8.5 Extract のマイニング・データベースでの登録

データベース・ログマイニング・サーバーを作成するには、各 Extract プロセスをマイニング・ データベースに登録します。ログマイニング・サーバーの作成によって、ソース・データベー スのスナップショットをソース・データベースの REDO ストリームに抽出します。ソースでの 不要なログマイニング・アクティビティを防止するには、レプリケーションの直前にこの手順 を実行します。

1. マイニング・データベースにログインします。データベース・ログマイニング・サーバー をソース・マイニング・データベースに作成するか、ダウンストリーム・マイニング・ データベースに作成するかに応じて、使用するコマンドは異なります。

ソース・デプロイ用のコマンド:

DBLOGIN USERID user, PASSWORD password [encryption options]

ダウンストリーム・デプロイ用のコマンド:

MININGDBLOGIN USERID user, PASSWORD password [encryption options]

説明: encryption options では、パスワードの暗号化オプション(オプション)を指定し ます。詳細は、『Oracle GoldenGate Windows and UNIX リファレンス・ガイド』の DBLOGINに関する項を参照してください。

2. Extract プロセスをマイニング・データベースに登録します。 複数の Extract をダウンスト リーム・データベースに登録するには、Extract ごとにコマンドを発行します。

REGISTER EXTRACT group DATABASE

REGISTER コマンドがすぐに復帰した場合でも、プロセスの登録は、完了に数分かかる場合があ ります。

8.6 変更キャプチャ・プロセスおよび変更配信プロセスの追加

注意: 初期ロードと変更のキャプチャを開始する準備ができたとき、または その直前に次の手順を実行します。初期ロード手順の際にこれらのプロセス を起動します。

次の手順では、第4章で構成した Oracle GoldenGate の Extract、データ・ポンプおよび Replicat のプロセスを作成します。まとめて"変更同期"プロセスと呼ばれるこれらのプロセス では、次の処理を行います。

- ターゲットでロードを実行する間、進行中のソースの変更をキャプチャして適用します。
- 発生した競合を解決します。

8.6.1 RMAN アーカイブ・ログ削除ポリシーの設定

RMAN のアーカイブ・ログ削除ポリシーを次の値に設定します。

CONFIGURE ARCHIVELOG DELETION POLICY TO APPLIED ON ALL STANDBY

これは、プライマリ Extract を追加する前に行う必要があります。

8.6.2 プライマリ Extract の追加

次の手順では、変更データをキャプチャするプライマリ Extract を追加します。

- 1. GGSCI を実行します。
- 2. ADD EXTRACT コマンドを発行して、プライマリ Extract グループを追加します。

ADD EXTRACT group name

- {, TRANLOG | , INTEGRATED TRANLOG}
- {, BEGIN {NOW | yyyy-mm-dd [:hh:mi:[ss[.ccccc]]]} |
- {, EXTSEQNO seqno, EXTRBA relative byte address}
- [, THREADS n]

説明:

- group name は、Extract グループの名前です。
- TRANLOG では、トランザクション・ログをデータ・ソースとして指定します(クラシッ ク・キャプチャの場合のみ)。
- INTEGRATED TRANLOG では、Extract がデータベース・ログマイニング・サーバーを介し て論理変更レコードを受信することを指定します(統合キャプチャの場合のみ)。
- BEGIN では、特定の時間にデータのキャプチャを開始することを指定します。
 - NOW では、3 の発行と同じ時間をタイムスタンプとする最初のレコードから開始し ます。
 - yyyy-mm-dd[:hh:mi:[ss[.cccccc]]]では、明示されたタイムスタンプから開始 します。このタイムスタンプからログが使用可能になります。
- EXTSEQNO segno, EXTRBA relative byte addressでは、REDO ストリーム内の特定 の位置(ログ順序番号と RBA) からデータのキャプチャを開始するよう指定します。
- THREADS nは、Oracle Real Application Cluster (RAC) に対するクラシック・キャプ チャ・モードに必要で、クラスタによって使用される REDO ログ・スレッドの数を指 定します。Extract で各スレッドを読み取って調整し、トランザクションの整合性を保 ちます。統合キャプチャには必要ありません。

注意:これは、最低限必要な構文です。その他のオプションを使用できます。 『Oracle GoldenGate Windows and UNIX リファレンス・ガイド』を参照して ください。

例 8-1 タイムスタンプ開始点を指定したクラシック・キャプチャ

ADD EXTRACT finance, TRANLOG, BEGIN 2011-01-01 12:00:00.000000

例 8-2 ADD EXTRACT タイムスタンプ開始点を指定した統合キャプチャ

ADD EXTRACT finance, INTEGRATED TRANLOG, BEGIN NOW

例 8-3 ログ順序 /RBA 開始点を指定した統合キャプチャ

ADD EXTRACT finance, INTEGRATED TRANLOG, EXTSEQNO 2952, EXTRBA 7598080

8.6.3 ローカル証跡の追加

次の手順では、プライマリ Extract によってキャプチャされたデータが書き込まれるローカル証 跡を追加します。

ソース・システムの GGSCI で、ADD EXTTRAIL コマンドを発行します。

ADD EXTTRAIL pathname, EXTRACT group name

説明:

- EXTTRAILでは、証跡がローカル・システムに作成されるよう指定します。
- pathname は、2 文字の名前を含めた、証跡の相対名または完全修飾名です。
- EXTRACT group name は、プライマリ Extract グループの名前です。

例 8-4

ADD EXTTRAIL /ggs/dirdat/lt, EXTRACT finance

8.6.4 データ・ポンプ Extract グループの追加

次の手順では、ローカル証跡を読み取り、データをターゲットに送信するデータ・ポンプを追 加します。

ソース・システムの GGSCI で、ADD EXTRACT コマンドを発行します。

ADD EXTRACT group name, EXTTRAILSOURCE trail name

説明:

- group name は、Extract グループの名前です。
- EXTTRAILSOURCE trail name は、ローカル証跡の相対名または完全修飾名です。

例 8-5

ADD EXTRACT financep, EXTTRAILSOURCE c:\ggs\dirdat\lt

8.6.5 リモート証跡の追加

次の手順では、リモート証跡を追加します。Replicat によって準備されますが、この証跡は データ・ポンプと関連付けられる必要があるため、ターゲットではなく、ソース・システムで 追加される必要があります。

ソース・システムの GGSCI で、次のコマンドを発行します。

ADD RMTTRAIL pathname, EXTRACT group name

説明:

- RMTTRAIL では、証跡がターゲット・システムに作成されるよう指定します。
- pathname は、2 文字の名前を含めた、証跡の相対名または完全修飾名です。
- EXTRACT group name は、データ・ポンプ Extract グループの名前です。

例 8-6

ADD RMTTRAIL /ggs/dirdat/rt, EXTRACT financep

8.6.6 Replicat グループの追加

次の手順では、リモート証跡(ターゲットに自動的に作成される)を読み取り、データ変更を ターゲットの Oracle データベースに適用する Replicat グループを追加します。

- 1. ターゲット・システムで GGSCI を実行します。
- **2.** ADD REPLICAT コマンドを発行します。

ADD REPLICAT group name, EXTTRAIL pathname

説明:

- group name は、Replicat グループの名前です。
- EXTTRAIL pathname は、2 文字の名前を含めた、リモート証跡の相対名または完全修飾 名です。

例 8-7

ADD REPLICAT financer, EXTTRAIL c:\qqs\dirdat\rt

8.7 ターゲットのインスタンス化の実行

この手順では、ターゲット表のインスタンス化の間に Oracle GoldenGate でソースの進行中の トランザクション変更をキャプチャし、ターゲットに適用されるまで格納します。ターゲット 表のインスタンス化を実行するまでに、Oracle GoldenGate 環境全体が変更のキャプチャと配 信用に構成されている必要があります。Oracle GoldenGate を初期ロード・ユーティリティと して使用する場合は初期ロード・プロセスも同様です。

8.7.1 データベース・ユーティリティを使用する場合のインスタンス化の実 行手順

1. ソースおよびターゲットのシステムで、GGSCI を実行し、Manager プロセスを起動しま す。

START MANAGER

注意: Windows クラスタで、クラスタ・アドミニストレータから Manager リソースを起動します。

2. プライマリ変更キャプチャ Extract グループを起動します。

START EXTRACT Extract group name

3. データ・ポンプ Extract グループを起動します。

START EXTRACT data pump name

注意: Extract が新しい Oracle GoldenGate 構成で初めて起動されたとき、開 いているトランザクションはスキップされます。Extract の起動後に開始され たトランザクションのみキャプチャされます。

- 4. 順序値をレプリケートする場合、次のようにします。
 - update. Sequence に対する EXECUTE 権限を持つユーザーとして DBLOGIN コマンドを発行し ます。

DBLOGIN USERID DBLOGINuser, PASSWORD password [encryption options]

次のコマンドを発行して、各ソース順序を更新し、REDO を生成します。REDO から、 Replicat でターゲットに対して順序の初期同期を実行します。順序名には任意の文字 またはすべての文字を表すアスタリスクのワイルドカードを使用できますが、所有者 には使用できません。

FLUSH SEQUENCE owner.sequence

- 5. データベース・ユーティリティを使用してコピーを開始します。
- 6. コピーが完了するまで待ち、完了時間を記録します。
- 7. ターゲット・システムで、Replicat パラメータ・ファイルを表示し、HANDLECOLLISIONS パ ラメータがリストされていることを確認します。そうではない場合、EDIT PARAMS コマンド を使用してパラメータを追加します。

VIEW PARAMS group name

EDIT PARAMS group name

8. Replicat を開始します。

START REPLICAT group name

9. INFO REPLICAT コマンドを発行します。初期ロード中に生成された変更データを Replicat ですべて適用したと表示されるまで、コマンドの発行を続けます。たとえば、初期ロード Extract が 12:05 に停止した場合、Replicat がその時間までのデータを適用したことを確認 します。

INFO REPLICAT group name

10. 変更配信 Replicat の HANDLECOLLISIONS をオフにし、初期ロードのエラー処理を無効にし ます。

SEND REPLICAT group name, NOHANDLECOLLISIONS

11. 変更配信 Replicat パラメータ・ファイルを編集し、HANDLECOLLISIONS パラメータを削除し ます。

EDIT PARAMS group name

12. パラメータ・ファイルを保存し、閉じます。

これ以降、Oracle GoldenGate でデータ変更の同期が続けられます。

8.7.2 SQL*Loader へのダイレクト・バルク・ロードを使用する場合のイン スタンス化の実行手順

- 1. ソース・システムで GGSCI を実行します。
- プライマリ変更キャプチャ Extract グループを起動します。

START EXTRACT group name

3. データ・ポンプ Extract グループを起動します。

START EXTRACT data pump name

- 順序値をレプリケートする場合、次のようにします。
 - update.Sequence に対する EXECUTE 権限を持つユーザーとして DBLOGIN コマンドを発行し ます。

DBLOGIN USERID DBLOGINuser, PASSWORD password [encryption options]

次のコマンドを発行して、各ソース順序を更新し、REDO を生成します。REDO から、 Replicat でターゲットに対して順序の初期同期を実行します。順序名には任意の文字 またはすべての文字を表すアスタリスクのワイルドカードを使用できますが、所有者 には使用できません。

FLUSH SEQUENCE owner.sequence

初期ロード Extract を起動します。

START EXTRACT initial-load Extract name

警告:初期ロード Replicat は起動しないでください。Manager プロセスで自 動的に起動し、ロードの完了時に終了します。

- 6. ターゲット・システムで GGSCI を実行します。
- 7. VIEW REPORT コマンドを発行し、SQL*Loaderへの初期ロードがいつ完了したかを確認しま す。

VIEW REPORT initial-load Extract name

8. ロードが完了したら、変更データ Replicat グループを起動します。

START REPLICAT group name

INFO REPLICAT コマンドを発行します。初期ロード中に生成された変更データを Replicat ですべて適用したと表示されるまで、コマンドの発行を続けます。たとえば、初期ロード Extract が 12:05 に停止した場合、Replicat がその時間までのデータを適用したことを確認 します。

INFO REPLICAT group name

10. 変更配信 Replicat の HANDLECOLLISIONS をオフにし、初期ロードのエラー処理を無効にし ます。

SEND REPLICAT group name, NOHANDLECOLLISIONS

11. 変更配信 Replicat パラメータ・ファイルを編集し、HANDLECOLLISIONS パラメータを削除し ます。

EDIT PARAMS group name

12. パラメータ・ファイルを保存し、閉じます。

これ以降、Oracle GoldenGate でデータ変更の同期が続けられます。

8.8 入力ファイルから SQL*Loader の場合のインスタンス化の 実行手順

- 1. ソース・システムで GGSCI を実行します。
- 2. プライマリ変更キャプチャ Extract グループを起動します。

START EXTRACT group name

3. データ・ポンプ Extract グループを起動します。

START EXTRACT data pump name

- 4. 順序値をレプリケートする場合、次のようにします。
 - update.Sequence に対する EXECUTE 権限を持つユーザーとして DBLOGIN コマンドを発行し ます。

DBLOGIN USERID DBLOGINuser, PASSWORD password [encryption options]

次のコマンドを発行して、各ソース順序を更新し、REDO を生成します。REDO から、 Replicat でターゲットに対して順序の初期同期を実行します。順序名には任意の文字 またはすべての文字を表すアスタリスクのワイルドカードを使用できますが、所有者 には使用できません。

FLUSH SEQUENCE owner.sequence

5. ソース・システムの Oracle GoldenGate のインストール・ディレクトリで、オペレーティ ング・システムのコマンドライン (GGSCI ではない) から初期ロード Extract を起動しま す。

UNIX および Linux:

\$ /OGG directory/extract paramfile dirprm/initial-load Extract name.prm reportfile path name

Windows

C:\> OGG directory\extract paramfile dirprm\initial-load Extract name.prm reportfile path name

説明:initial-load Extract name は、初期ロード Extract の名前で、path name は、 Extract レポート・ファイルが作成される場所の相対パスまたは完全修飾パスです。

- 6. ソースからの初期抽出が完了するまで待ちます。コマンドラインから Extract レポート・ ファイルを表示し、進捗と結果を確認します。
- 7. ターゲット・システムで、初期ロード Replicat を起動します。

UNIX および Linux:

\$ /OGG directory/replicat paramfile dirprm/initial-load Replicat name.prm reportfile path name

C:\> OGG directory\replicat paramfile dirprm\initial-load Replicat name.prm reportfile path name

説明:initial-load Extract name は、初期ロード Replicat の名前で、path name は、 Replicat レポート・ファイルが作成される場所の相対パスまたは完全修飾パスです。

- 初期ロード Replicat が停止したら、コマンドラインから Replicat レポート・ファイルを表 示し、結果を確認します。
- 9. ASCII 形式のファイルおよび初期ロード Replicat が作成した実行ファイルと制御ファイル を使用し、SQL*Loaderを介してデータをロードします。
- 10. ロードが完了したら、変更配信 Replicat グループを起動します。

START REPLICAT group name

11. INFO REPLICAT コマンドを発行します。初期ロード中に生成された変更データを Replicat ですべて適用したと表示されるまで、コマンドの発行を続けます。たとえば、初期ロード Extract が 12:05 に停止した場合、Replicat がその時間までのデータを適用したことを確認 します。

INFO REPLICAT group name

12. 変更配信 Replicat の HANDLECOLLISIONS をオフにし、初期ロードのエラー処理を無効にし ます。

SEND REPLICAT group name, NOHANDLECOLLISIONS

13. 変更配信 Replicat パラメータ・ファイルを編集し、HANDLECOLLISIONS パラメータを削除し ます。

EDIT PARAMS group name

14. パラメータ・ファイルを保存し、閉じます。

これ以降、Oracle GoldenGate でデータ変更の同期が続けられます。

8.9 インスタンス化後の処理の監視

ターゲットをインスタンス化し、レプリケーションが有効になったら、レプリケーション環境 のステータス、ラグおよび全体的な正常性を確認し、プロセスが適切に実行されていること、 Oracle GoldenGate エラー・ログに警告がないこと、およびラグが許容可能なレベルであるこ とをチェックします。次のものから Oracle GoldenGate プロセスを確認できます。

- *GGSCI*: 『Oracle GoldenGate *Windows and UNIX 管理者ガイド*』を参照してください。
- Oracle Golden Gate Monitor: この製品の管理ドキュメントとオンライン・ヘルプを参照して ください。

キャプチャと配信がすべての適切な表に対して実行されていることと、データの同期が保たれ ていることも確認します。この目的には Oracle Golden Gate Veridata 製品を使用できます。

8.10 Oracle GoldenGate 環境のバックアップ

Oracle GoldenGate の処理を開始したら、障害に備えて処理の状態を残しておく上で、効果的 なバックアップ・ルーチンが不可欠です。Oracle GoldenGate の作業ファイルをリストアでき ない場合、レプリケーション環境全体を再インスタンス化し、初期ロードを新たに完了する必 要があります。

ベスト・プラクティスとして、バックアップ・ルーチンに Oracle GoldenGate ホーム・インス トール全体を含めます。ディレクトリのルートに重要なサブディレクトリ、ファイルおよびプ ログラムが非常に多くあり、個別に追跡することは困難です。いずれにしても、最も重要な ファイルがバックアップ領域の大半を占めるため、単純にインストール・ディレクトリ全体を バックアップすることは、高速で簡単なリカバリのためには理にかなっています。

プロセスの制御

オンライン・プロセスを制御する標準の方法は、GGSCIを介する方法です。その他の方法につ いては、『Oracle GoldenGate Windows and UNIX 管理者ガイド』を参照してください。

9.1 プロセスを起動するとき

通常、Oracle GoldenGate プロセスが本番設定で初めて起動されるのは、(ソース・ユーザー・ アプリケーションがアクティブのままであるとすると) 初期同期プロセス時です。ターゲット にソース・データがロードされている間、Oracle GoldenGate で進行中のユーザー変更をキャ プチャし、ロードの結果に反映します。

注意: Extract が新しい Oracle GoldenGate 構成で初めて起動されたとき、開 いているトランザクションはスキップされます。Extract の起動後に開始され たトランザクションのみキャプチャされます。

9.2 インスタンス化の完了後のプロセスの起動

次の手順は、日常的に必要に応じてプロセスを起動するためのものです。基本的な構文を示し ます。この他に使用可能な構文もあります (『Oracle GoldenGate Windows and UNIX リファレ ンス・ガイ / 』を参照してください)。

Manager を起動する手順

- 1. Oracle GoldenGate ディレクトリから GGSCI を実行します。
- 2. GGSCIで、次のコマンドを発行します。

START MANAGER

注意: ユーザー・アカウント制御が有効な Windows Server 2008 でコマンド ラインまたは GGSCI から Manager を起動すると、プログラムの実行を許可 または拒否するよう求める UAC プロンプトが表示されます。

Extract または Replicat を起動する手順

START {EXTRACT | REPLICAT} group_name

group name は、Extract または Replicat グループの名前またはグループのワイルドカード・ セット(*、fin* など)です。

Extract または Replicat を正常に停止する手順

STOP {EXTRACT | REPLICAT} group_name

group name は、Extract または Replicat グループの名前またはグループのワイルドカード・ セット(*、fin* など)です。

Replicat を強制的に停止する手順

STOP REPLICAT group_name!

現在のトランザクションが中止され、プロセスがただちに停止します。Extract は強制的に停止 できません。

STOP で停止できないプロセスの強制終了

KILL {EXTRACT | REPLICAT} group_name

強制終了すると、プロセスは正常に停止せず、チェックポイント情報が失われることがありま す。

一度に複数のプロセスを制御する手順

command ER wildcard specification

説明:

- command は、KILL、START または STOP です。
- wildcard specificationは、コマンドを適用するプロセス・グループの名前のワイルド カード指定です。コマンドは、ワイルドカードに適合する各 Extract および Replicat グ ループに適用されます。Oracle GoldenGate では、最大 100,000 個のワイルドカード・エン トリがサポートされます。

Extract グループを削除する手順

- 1. GGSCI を実行します。
- 2. プロセスを停止します。

STOP EXTRACT group name

3. 次のコマンドを発行します。

DELETE EXTRACT group name !

! 引数では、ワイルドカードに適合するすべての Extract グループがプロンプトなしで削除 されます。

Replicat グループを削除する手順

1. プロセスを停止します。

STOP REPLICAT group name

2. このグループでチェックポイント表を使用している場合、GGSCIから次のコマンドを発行 し、データベースにログインします。

DBLOGIN USERID user, PASSWORD password [encryption options]

説明:

- USERID user, PASSWORD passwordでは、データベース・ログイン資格証明を指定しま
- encryption optionsでは、パスワードを暗号化するオプションの1つを指定します。
- 3. 次のコマンドを発行してグループを削除します。

DELETE REPLICAT group name

注意: その他のコマンドとオプションについては、『Oracle GoldenGate Windows and UNIX リファレンス・ガイド』を参照してください。

プロセス・グループを削除しても、パラメータ・ファイルは失われません。同じパラメータ・ ファイルを使用して同じグループを再度作成することも、パラメータ・ファイルを削除してグ ループの構成を完全に削除することもできます。

Oracle DDL レプリケーション環境の管理

この章では、Oracle GoldenGate の DDL オブジェクトがシステムにまだ存在する場合に、デー タベース環境または Oracle GoldenGate 環境を変更する手順について説明します。

Oracle GoldenGate の DDL サポートを構成する手順については、『Oracle GoldenGate Windows and UNIX 管理者ガイト』を参照してください。

10.1 DDL トリガーの有効化と無効化

Oracle GoldenGate 内で構成を変更せずに DDL 操作をキャプチャするトリガーを有効および無 効にすることができます。次のスクリプトによって、DDL トリガーが制御されます。

- ddl disable: トリガーを無効にします。トリガーを無効にすると、DDL 操作はそれ以降 キャプチャまたはレプリケートされません。
- ddl enable: トリガーを有効にします。トリガーを有効にすると、Oracle GoldenGate に よって現在の DDL 変更がキャプチャされますが、トリガーを無効にしている間に生成され た DDL はキャプチャされません。

これらのスクリプトを実行する前に、Oracle GoldenGate のプロセス、SQL*Plus、ビジネス・ アプリケーションおよび Oracle を使用するその他のソフトウェアのセッションなど、これまで DDL を発行したすべてのセッションを無効にします。そうしない場合、データベースによって ORA-04021 エラーが生成されることがあります。 ソース・システムおよびターゲット・システ ムで DDL の一貫性を維持する場合には、これらのスクリプトを使用しないでください。

10.2 DDL マーカー表の保持

マーカー表から行をいつでもパージできます。DDL 履歴は保持されません。マーカー表をパー ジするには、Manager パラメータ PURGEMARKERHISTORY を使用します。Manager によって、次 のいずれかからマーカー表の名前が取得されます。

- 1. GLOBALS ファイルの MARKERTABLE パラメータで指定した名前 (指定した場合)。
- 2. デフォルト名である GGS MARKER。

PURGEMARKERHISTORY には、最終変更日に基づいて、行を保持する時間の最大時間と最小時間を 指定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。

10.3 DDL マーカー表の削除

DDL の同期を中止する場合を除き、DDL マーカー表を削除しないでください。マーカー表と DDL トリガーは互いに依存します。DDL トリガーが有効である場合にマーカー表をドロップ すると失敗します。これは、トリガーが無効になり、DDL 操作が見つからないことを回避する ための安全対策です。マーカー表を削除すると、次のエラーが生成されます。

"ORA-04098: trigger 'SYS.GGS DDL TRIGGER BEFORE' is invalid and failed re-validation"

Oracle GoldenGate の DDL オブジェクトを削除する正しい方法は、残りの DDL 環境をどうす るかによって異なります。正しい手順を選択するには、次のいずれかを参照してください。

- 10-4 ページの「既存の DDL 環境のクリーン状態への復元」
- 10-6 ページの「システムからの DDL オブジェクトの削除」

10.4 DDL 履歴表の保持

DDL 履歴表をパージしてそのサイズを制御できますが、これは慎重に実行してください。DDL 履歴表によって、DDL 同期環境の整合性が維持されます。この表へのパージは、Oracle GoldenGate のインタフェースを介してリカバリすることはできません。

- 1. DDL の履歴を損失するあらゆる可能性を回避するには、定期的に履歴表の完全バックアッ プを作成します。
- 2. パージされた DDL を確実にリカバリできるようにするには、履歴テーブルに対して Oracle Flashback を有効にします。フラッシュバック保存時間は、それが必要になる可能 性がある時点を大幅にさかのぼって設定します。たとえば、完全バックアップが最大1週 間分である場合は、2週間分のフラッシュバックを保持します。Oracle GoldenGate の位置 を、過去にさかのぼってフラッシュバックに特定し、再処理することが可能です。
- 3. 可能な場合、DDL 履歴表を手動でパージして、必須の行が誤ってパージされることがない ようにします。自動パージ・メカニズムが必要な場合、Managerパラメータ・ファイルで PURGEDDLHISTORY パラメータを使用します。行を保持する最大時間および最小時間を指定 できます。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: パフォーマンスを向上させるために Oracle GoldenGate によって作成 される一時表は、同じルールに従って DDL 履歴表と同時にパージされる場合 があります。これらの表の名前は履歴表の名前から取得され、それらのパー ジが Manager レポート・ファイルで報告されます。これが通常の動作です。

10.5 DDL 履歴表の削除

DDL の同期を中止する場合を除き、DDL 履歴表を削除しないでください。 履歴表には、発行 された DDL 操作のレコードが含まれています。

履歴表と DDL トリガーは互いに依存します。DDL トリガーが有効である場合に履歴表をド ロップすると失敗します。これは、トリガーが無効になり、DDL操作が見つからないことを回 避するための安全対策です。履歴表を削除すると、次のエラーが生成されます。

"ORA-04098: trigger 'SYS.GGS_DDL_TRIGGER_BEFORE' is invalid and failed re-validation"

Oracle GoldenGate の DDL オブジェクトを削除する正しい方法は、残りの DDL 環境をどうす るかによって異なります。正しい手順を選択するには、次のいずれかを参照してください。

- 10-4 ページの「既存の DDL 環境のクリーン状態への復元」
- 10-6 ページの「システムからの DDL オブジェクトの削除」

10.6 DDL トレース・ファイルのパージ

DDL トレース・ファイルでディスク容量を必要以上に消費しないようにするには、定期的に ddl cleartrace スクリプトを実行します。このスクリプトによってトレース・ファイルは削除 されますが、Oracle GoldenGate によって再作成されます。

DDL トレース・ファイルのデフォルト名は ggs ddl trace.log です。これは Oracle の USER_ DUMP DEST ディレクトリにあります。ddl cleartrace スクリプトは Oracle GoldenGate ディレ クトリにあります。

10.7 DDL サポートが有効である場合のデータベースのパッチお よびアップグレードの適用

通常、データベースのパッチおよびアップグレードによって Oracle GoldenGate の DDL トリ ガーと Oracle GoldenGate のその他の DDL オブジェクトは無効にされます。データベースの パッチを適用する前に、次の手順を実行します。

- 1. SYSDBA 権限を持つユーザーとして SQL*Plus にログインします。
- 2. SQL*Plus で ddl disable スクリプトを実行して、Oracle GoldenGate の DDL トリガーを 無効にします。
- 3. パッチを適用します。
- **4.** SQL*Plus で ddl enable スクリプトを実行して、の DDL トリガーを有効にします。

注意: 通常、データベースのアップグレードおよびパッチは Oracle のオブ ジェクトに対して動作します。Oracle GoldenGate によってそれらのオブ ジェクトは自動的にフィルタ処理で除外されるため、レプリケーションの再 開時に、それらの手順から DDL はレプリケートされません。

パッチまたはアップグレード後の再コンパイル・エラー (手順の前にトリガーを無効にしてい ない場合に発生)を回避するには、スクリプト内の適切な場所に@ddl disable および@ddl enable のコールを追加することを検討してください。

10.8 DDL サポートが有効である場合の Oracle GoldenGate の パッチおよびアップグレードの適用

注意: リリースに付随するリリース・ノートまたはアップグレード手順に次 のような手順がある場合には、次の手順ではなく、それらの手順を実行して ください。30Kよりも大きい DDL 文をサポートしていない Oracle GoldenGate のバージョン (リリース 10.4 より前)からのアップグレードにこ の手順を使用しないでください。その場合にアップグレードするには、10-4 ページの「既存の DDL 環境のクリーン状態への復元」の手順を実行します。

パッチまたはアップグレードを DDL オブジェクトに適用するには、次の手順を実行します。こ の手順では、新しいビルドにクリーン・インストールが必要であるかどうかに応じて、現在の DDL 同期構成が保持される、あるいは保持されない場合があります。

- 1. GGSCI を実行します。この手順の間、セッションを開いたままにしてください。
- **2.** DDL キャプチャを停止するために Extract を停止します。

STOP EXTRACT group

3. DDL レプリケーションを停止するために Replicat を停止します。

STOP REPLICAT group

- **4.** Oracle GoldenGate によって示される指示に従って、パッチまたはアップグレードのファ イルをダウンロードあるいは抽出します。
- **5.** 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。
- 6. SYSDBA 権限を持つユーザーとして SQL*Plus にログインします。
- 7. Oracle GoldenGate のプロセス、SOL*Plus、ビジネス・アプリケーションおよび Oracle を 使用するその他のソフトウェアのセッションなど、これまで DDL を発行したすべてのセッ ションを切断します。そうしない場合、データベースによって ORA-04021 エラーが生成さ れることがあります。
- **8.** ddl disable スクリプトを実行して DDL トリガーを無効にします。
- 9. ddl setup スクリプトを実行します。Oracle GoldenGate DDL のスキーマ名の入力が求め られます。スキーマ名を変更した場合は、新しい名前を使用してください。
- **10.** ddl enable.sql スクリプトを実行して DDL トリガーを有効にします。
- 11. GGSCI で Extract を開始して DDL キャプチャを再開します。 START EXTRACT group
- 12. DDL レプリケーションを開始するために Replicat を開始します。 START REPLICAT group

10.9 既存の DDL 環境のクリーン状態への復元

Oracle GoldenGate の DDL オブジェクトを完全に削除してから再インストールするには、次の 手順を実行します。この手順によって新しい DDL 環境が作成されますが、現在の DDL 履歴は 削除されます。

注意: この手順では、オブジェクトの相互依存性のため、すべてのオブジェ クトを削除して再インストールする必要があります。

- **1.** Oracle GoldenGate の新しいバージョンのインストールと同時にこの手順を実行する場合、 Oracle GoldenGate のファイルをダウンロードしてインストールし、必要に応じてプロセ ス・グループとパラメータ・ファイルを作成または更新します。
- 2. (オプション)ソースおよびターゲットの構造の連続性を維持するには、DDLアクティビ ティを停止してから Replicat によって証跡内の DDL と DML データの処理が完了されてい ることを確認します。Replicat の終了時間を判断するには、処理するデータがこれ以上存 在しないことを示すメッセージが表示されるまで次のコマンドを発行します。

INFO REPLICAT group

注意: INFO REPLICAT を使用するかわりに、TABLE および MAP の EVENTACTIONS オプションを使用して、DDL と DML の処理後に Extract およ び Replicat のプロセスを停止できます。

- **3.** GGSCI を実行します。
- **4.** DDL キャプチャを停止するために Extract を停止します。

STOP EXTRACT group

- **5.** DDL レプリケーションを停止するために Replicat を停止します。 STOP REPLICAT group
- 6. 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。
- 7. SYSDBA 権限を持つユーザーとして SQL*Plus にログインします。
- **8.** Oracle GoldenGate のプロセス、SQL*Plus、ビジネス・アプリケーションおよび Oracle を 使用するその他のソフトウェアのセッションなど、これまで DDL を発行したすべてのセッ ションを切断します。そうしない場合、データベースによって ORA-04021 エラーが生成さ れることがあります。
- 9. ddl disable スクリプトを実行して DDL トリガーを無効にします。
- **10.** ddl remove スクリプトを実行して、Oracle GoldenGate の DDL トリガー、DDL 履歴表お よびマーカー表とその他の関連オブジェクトを削除します。デバッグで必要となる場合に 備え、このスクリプトにより、スクリプト出力を記録する ddl_remove_spool.txt ファイ ルと、環境設定を記録するddl remove set.txt ファイルが作成されます。
- 11. marker remove スクリプトを実行して Oracle GoldenGate マーカー・サポート・システム を削除します。デバッグで必要となる場合に備え、このスクリプトにより、スクリプト出 力を記録する marker remove spool.txt ファイルと、環境設定を記録する marker remove set.txt ファイルが作成されます。
- **12.** このインストール用の DDL スキーマを変更する場合、次の権限を Oracle GoldenGate ス キーマに付与します。

GRANT EXECUTE ON utl file TO schema;

13. このインストール用の DDL スキーマを変更する場合、スキーマのデフォルト表領域をその スキーマ専用にする必要があります。他のスキーマと共有することはできません。この表 領域に対して AUTOEXTEND を ON に設定し、表領域のサイズを GGS DDL HIST 表および GGS MARKER 表の拡張に対応できるサイズに設定する必要があります。特に、GGS DDL HIST 表は DDL アクティビティ全体に比例して増加します。

注意: DDL 表領域が一杯になると、Extract は DDL のキャプチャを停止し ます。その場合にユーザー DDL アクティビティが失敗するようにするには、 params.sql スクリプトを編集して ddl_fire_error_in_trigger パラメータ を TRUE に設定します。ユーザー DDL を停止することで表領域のサイズを拡 張する時間ができ、DDL キャプチャが失われることを防ぎます。ただし、表 領域のサイズをこのように管理する場合、ビジネスの混乱を防ぐためにビジ ネス・アプリケーションと Extract を絶えず監視する必要があります。かわり に、表領域を適切なサイズに設定し、表領域が一杯にならないように AUTOEXTEND を ON に設定することをお薦めします。

警告:記載された手順に従ってオブジェクト名を変更する必要がある場合以 外、params.sql の他のパラメータを編集しないでください。

14. このインストール用の DDL スキーマを変更する場合、GLOBALS ファイルを編集し、次のパ ラメータに新規スキーマ名を指定します。

GGSCHEMA schema name

- **15.** marker setup スクリプトを実行して Oracle GoldenGate マーカー・サポート・システムを 再インストールします。Oracle GoldenGate のスキーマ名の入力が求められます。
- **16.** ddl setup スクリプトを実行します。Oracle GoldenGate DDL のスキーマ名の入力が求め られます。
- **17.** role setup スクリプトを実行して Oracle GoldenGate の DDL ロールを再作成します。
- **18.** Oracle GoldenGate のプロセス (Extract、Replicat、GGSCI および Manager) を実行するす べての Oracle GoldenGate ユーザーにロールを付与します。プロセスに異なるユーザー名 がある場合、複数の付与が必要になることがあります。
- **19.** ddl enable.sql スクリプトを実行して DDL トリガーを有効にします。

10.10 システムからの DDL オブジェクトの削除

この手順によって、DDL 環境が削除され、ソースとターゲットの DDL 操作間で連続性を維持 する履歴が削除されます。

注意: オブジェクトの相互依存性のため、すべてのオブジェクトを削除する 必要があります。

- 1. GGSCI を実行します。
- **2.** DDL キャプチャを停止するために Extract を停止します。

STOP EXTRACT group

3. DDL レプリケーションを停止するために Replicat を停止します。

STOP REPLICAT group

- **4.** 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。
- **5.** SOL*Plus を実行し、SYSDBA 権限を持つユーザーでログインします。
- **6.** Oracle GoldenGate のプロセス、SQL*Plus、ビジネス・アプリケーションおよび Oracle を 使用するその他のソフトウェアのセッションなど、これまで DDL を発行したすべてのセッ ションを切断します。そうしない場合、データベースによって ORA-04021 エラーが生成さ れることがあります。
- 7. ddl disable スクリプトを実行して DDL トリガーを無効にします。
- 8. ddl remove スクリプトを実行して、Oracle GoldenGate の DDL トリガー、DDL 履歴表お よびマーカー表と関連オブジェクトを削除します。デバッグで必要となる場合に備え、こ のスクリプトにより、スクリプト出力を記録するddl remove spool.txt ファイルと、現 行ユーザーの環境設定を記録するddl_remove_set.txtファイルが作成されます。
- 9. marker remove スクリプトを実行して Oracle GoldenGate マーカー・サポート・システム を削除します。デバッグで必要となる場合に備え、このスクリプトにより、スクリプト出 力を記録する marker_remove_spool.txt ファイルと、環境設定を記録する marker_ remove_set.txt ファイルが作成されます。

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になり、現在の Oracle GoldenGate 環境を保持する必要がなくなった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。

注意: 以降の項を記載の順に実行します。

11.1 プロセスの停止

この手順では、Extract および Replication プロセスを停止します。停止の指示があるまで、Manager は実行したままにします。

すべてのシステムで:

- 1. コマンド・シェルを実行します。
- **2.** 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. 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。
- 4. GGSCI を実行します。
- **5.** Oracle GoldenGate プロセスをすべて停止します。

STOP ER *

6. Manager プロセスを停止します。

STOP MANAGER

11.2 DDL 環境の削除

この手順では、ソース・システムの DDL スキーマからすべての Oracle GoldenGate DDL オブジェクトを削除します。

- 1. SYSDBA 権限を持つユーザーとして SQL*Plus にログインします。
- 2. Oracle GoldenGate のプロセス、SQL*Plus、ビジネス・アプリケーションおよび Oracle を 使用するその他のソフトウェアのセッションなど、これまで DDL を発行したすべてのセッションを切断します。そうしない場合、データベースによって ORA-04021 エラーが生成されることがあります。
- 3. ddl disable スクリプトを実行して DDL トリガーを無効にします。
- **4.** ddl_remove スクリプトを実行して、Oracle GoldenGate の DDL トリガー、DDL 履歴表およびマーカー表とその他の関連オブジェクトを削除します。デバッグで必要となる場合に

備え、このスクリプトにより、スクリプト出力を記録する ddl remove spool.txt ファイ ルと、環境設定を記録するddl remove set.txt ファイルが作成されます。

5. marker remove スクリプトを実行して Oracle GoldenGate マーカー・サポート・システム を削除します。デバッグで必要となる場合に備え、このスクリプトにより、スクリプト出 力を記録する marker remove spool.txt ファイルと、環境設定を記録する marker remove set.txt ファイルが作成されます。

11.3 データベース・オブジェクトの削除

次の手順に従って、Oracle データベース内に構成されている Oracle GoldenGate オブジェクト を削除します。構成によっては、特定のステップやコマンドが該当しない場合もあります。

ソース・システムで:

DBLOGIN を使用してログインする前に、この手順の GGSCI コマンドを確認します。特別な権限 が必要な場合があります。

- 1. すべての Extract および Replicat プロセスが停止していることを確認します。
- 2. DBLOGIN (ダウンストリーム・マイニング・データベースからデータベース・ログマイニン グ・サーバーを削除する必要がある場合は MININGDBLOGIN コマンド) を使用してデー タベースにログインします。

[MINING] DBLOGIN USERID user, PASSWORD password [encryption options]

- **3.** GGSCI を実行します。
- 4. GGSCIで、構成に応じて次のコマンドのいずれか、またはすべてを実行します。
 - スキーマレベルのサプリメンタル・ロギングを無効にします(ワイルドカードは使用で きません)。

DELETE SCHEMATRANDATA schema

表レベルのサプリメンタル・ロギングを無効にします。

DELETE TRANDATA schema.*

(双方向構成)Oracleトレース表を削除します。

DELETE TRACETABLE schema.table

(クラシック・キャプチャ構成)ログ保存を無効にし、基になる Oracle Streams キャプ チャ・プロセスを削除します。DBLOGIN には、6-10 ページの表 6-3 に示す権限が必要 です。

UNREGISTER EXTRACT group LOGRETENTION

(統合キャプチャ構成)Oracle マイニング・データベースから、ログマイニング・サー バーを削除します。[MINING]DBLOGINには、dbms goldengate auth.grant admin privilege プロシージャで付与される権限が必要です。

DELETE EXTRACT group

UNREGISTER EXTRACT group DATABASE

Replicat チェックポイントが存在する任意のシステムで:

- **1.** Replicat が停止していることを確認します。
- 2. DBLOGIN コマンドを使用してデータベースにログインします。

DBLOGIN USERID user, PASSWORD password [encryption options]

3. DELETE CHECKPOINTTABLE コマンドを実行して、Replicat チェックポイント表を削除しま す。

DELETE CHECKPOINTTABLE schema.table

11.4 (Windows) Oracle GoldenGate Windows コンポーネント の削除

この手順では、Windows クラスタ・リソースである Oracle GoldenGate をソースまたはター ゲットの Windows システムから削除し、Oracle GoldenGate イベントが Windows イベント・ マネージャにレポートされないようにして、Manager サービスを削除します。ソース・システ ムおよびターゲット・システムで、次の手順を実行します。

- 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. (クラスタ)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 ノードから GGSCI を実行し、すべての Extract および Replicat のプロセスが停止している ことを確認します。稼動しているものがあれば停止します。

STATUS ER *

STOP ER *

- 3. (クラスタ) 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。
- **4.** (クラスタ) 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。
- 5. 「スタート」、「ファイル名を指定して実行」の順にクリックし、「ファイル名を指定して実 行」ダイアログ・ボックスに cmd と入力してコマンド・コンソールを開きます。
- **6.** 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。
- 7. 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

- 8. Windows の SYSTEM32 フォルダから CATEGORY.DLL ファイルと GGSMSG.DLL ファイルを削除 します。
- 9. (クラスタ) クラスタ・グループをクラスタ内の次のノードに移動し、ステップ 5 から繰り 返します。

11.5 Oracle GoldenGate ファイルの削除

次の手順をすべてのシステムで実行し、Oracle GoldenGate インストール・ディレクトリを削 除します。

1. GGSCIですべてのプロセスが停止していることを確認します。稼動しているものがあれば 停止します。

STATUS MANAGER

STATUS ER *

STOP MANAGER

STOP ER *

2. GGSCI を終了します。

EXIT

3. Oracle GoldenGate インストール・ディレクトリを削除します。

統合キャプチャ用のダウンストリーム・ マイニング・データベースの構成

この章では、統合キャプチャ・モードの Extract をサポートするようダウンストリーム Oracle マイニング・データベースを準備する手順について説明します。

- ダウンストリーム・デプロイ用のキャプチャ・オプションの評価
- ダウンストリーム・デプロイ用のソース・データベースの準備
- ダウンストリーム・マイニング・データベースの準備

これらの手順で使用されるパラメータの詳細は、『Oracle Database リファレンス 11g リリース 2 (11.2)』および『Oracle Data Guard 概要および管理 11g リリース 2 (11.2)』を参照してくださ

統合キャプチャの詳細は、4-3ページの「使用するキャプチャ方法の決定」を参照してくださ 11,

ダウンストリーム・マイニング構成の例は、B-1ページの「統合キャプチャ用のダウンスト リーム・マイニング・データベースの構成例」を参照してください。

A.1 ダウンストリーム・デプロイ用のキャプチャ・オプション の評価

ダウンストリーム・デプロイによってソース・データベースの負荷を軽減できます。ソース・ データベースは REDO ログをダウンストリーム・データベースに送り、Extract はダウンスト リーム・データベースでログマイニング・サーバーを使用して REDO ログをマイニングしま す。ダウンストリーム・マイニング・データベースでは、アーカイブ・ログとオンライン REDO ログの両方をソース・データベースから受け入れることができます。

複数のソース・データベースの REDO データを 1 つのダウンストリーム・データベースに送信 できます。ただし、ダウンストリーム・マイニング・データベースは、それらのソース・デー タベースの1つからのみオンラインREDO ログを受け入れることができます。残りのソース・ データベースはアーカイブ・ログを送る必要があります。

オンライン・ログがダウンストリーム・データベースに送られると、Extract による リアルタイ ム・キャプチャが可能です。Extract でソース・ログからの読取りと同様に変更がキャプチャさ れます。オンライン REDO ログをソース・データベースから受け入れるには、ダウンストリー ム・マイニング・データベースにスタンバイ REDO ログが構成されている必要があります。

ダウンストリーム・マイニング構成を使用する場合、ソース・データベースとマイニング・ データベースは同じプラットフォームのものである必要があります。たとえば、ソース・デー タベースが Linux 64 ビットで稼働している場合、ダウンストリーム・データベースのプラット フォームも Linux 64 ビットである必要があります。

A.2 ダウンストリーム・デプロイ用のソース・データベースの 準備

この項では、次のプロセスについて説明します。

- ソース・ユーザー・アカウントの作成
- ソースからダウンストリーム・マイニング・データベースへの REDO 転送の構成

A.2.1 ソース・ユーザー・アカウントの作成

ソース・データベースに Extract ユーザーが必要です。Extract ではこのユーザーの資格証明を使用してメタデータ問合せを行い、必要に応じてソース・データベースから列値をフェッチします。ソース・ユーザーは USERID パラメータによって指定されます。

4-5 ページの「Oracle GoldenGate のデータベース・ユーザーの割当て」の手順に従っていた場合、すでにこのユーザーを作成し、必要な権限を割り当てている可能性があります。このユーザーを作成していない場合、ここで作成し、次の権限を割り当てます。

- 1. Extract が統合キャプチャ・モードで動作するために必要な権限を付与します。
 - Oracle 11.2.0.3 以降の場合、dbms_goldengate_auth.grant_admin_privilege プロシージャを使用します。
 - dbms_goldengate_auth.grant_admin_privilege プロシージャがない、それより前の Oracle リリースの場合、dbms streams auth.grant admin privilege を使用します。
- 2. V \$DATABASE に対する SELECT 権限をダウンストリーム・マイニング・ユーザーに付与します。

GRANT SELECT ON V \$DATABASE TO user;

3. 4-5 ページの「Oracle Golden Gate のデータベース・ユーザーの割当て」に示されている適切で基本的なユーザー権限を割り当てます。

A.2.2 ソースからダウンストリーム・マイニング・データベースへの REDO 転送の構成

次の手順を実行して、ソース・データベースからダウンストリーム・マイニング・データベースへの REDO ログ・ファイルの転送を設定し、ダウンストリーム・マイニング・データベースでこれらの REDO ログ・ファイルを受け入れる準備をします。

注意: ソース・データベースから送られたアーカイブ・ログは、*外部アーカイブ・ログ*と呼ばれます。外部アーカイブ・ログの格納にダウンストリーム・マイニング・データベースのリカバリ領域を使用することはできません。 そのような構成は、統合キャプチャでサポートされません。

これらの手順では、必要に応じて複数のソースから REDO を転送するための要件が考慮されています。ソースごとに Extract プロセスを構成します。複数のソースから 1 つのダウンストリーム・マイニング・データベースへの REDO の送信をサポートするためのルールを次に要約します。

- オンラインREDO をダウンストリーム・マイニング・データベースのスタンバイ REDO ログに送信するよう構成できるソース・データベースは1つのみです。このソース・データベースの log archive dest n設定に TEMPLATE 句を含めることはできません。
- オンライン REDO をダウンストリーム・マイニング・データベースのスタンバイ REDO ログに送信 しないソース・データベースには、 $\log_{\text{archive_dest_n}}$ パラメータに TEMPLATE 句を指定する必要があります。
- REDO をダウンストリーム・マイニング・データベースに送信する各ソース・データベースは一意の DBID を持つ必要があります。これらのソース・データベースの v\$database ビューから DBID 列を選択し、DBID が一意であることを確認します。

REDO 転送を構成する手順

- 1. 各ソース・データベースがマイニング・データベースと通信できるよう Oracle Net を構成します。詳細は、『Oracle Database Net Services 管理者ガイド11g リリース 2 (11.2)』を参照してください。
- 2. REDO データの転送をサポートするよう、各ソース・データベースとダウンストリーム・マイニング・データベースで認証を構成します。REDO 転送セッションは、Secure Sockets Layer(SSL) プロトコルまたはリモート・ログイン・パスワード・ファイルを使用して認証されます。ソース・データベースにリモート・ログイン・パスワード・ファイルがある場合、マイニング・データベース・システムの適切なディレクトリにコピーします。パスワード・ファイルは、すべてのソース・データベースとマイニング・データベースで同じである必要があります。REDO 転送の認証要件の詳細は、『Oracle Data Guard 概要および管理11g リリース2(11.2)』のスタンバイ・データベースの作成用のプライマリ・データベースの準備に関する項を参照してください。
- 3. 各ソース・データベースで、REDO データをダウンストリーム・マイニング・データベースに転送するために LOG_ARCHIVE_DEST_n 初期化パラメータを 1 つ構成します。リアルタイム・キャプチャ・モードを使用するか、アーカイブログのみキャプチャ・モードを使用するかに応じて、次の例のいずれかのようにこのパラメータの属性を設定します。
 - ソース・データベースがオンライン REDO ログをダウンストリーム・データベースに 送信する場合のダウンストリーム・マイニング・サーバーでのリアルタイム・キャプ チャの例:

ALTER SYSTEM

SET LOG_ARCHIVE_DEST_2='SERVICE=DBMSCAP.EXAMPLE.COM ASYNC NOREGISTER VALID_FOR=(ONLINE_LOGFILES, PRIMARY_ROLE)DB_UNIQUE_NAME=dbmscap'

ダウンストリーム・ログマイニング・サーバーでのアーカイブログのみキャプチャの例:

ALTER SYSTEM SET

LOG_ARCHIVE_DEST_2='SERVICE=DMBSCAP.EXAMPLE.COM ASYNC NOREGISTER VALID_FOR=(ONLINE_LOGFILES, PRIMARY_ROLE)
TEMPLATE=/usr/oracle/log_for_dbms1/dbms1_arch_%t_%s_%r.log
DB UNIQUE NAME=dbmscap'

注意:アーカイブ・ログのみのダウンストリーム・マイニング・データベースを使用する場合、TEMPLATE 属性の値を指定する必要があります。ソース・データベースでTEMPLATE 句を使用して、すべてのリモート・ソース・データベースのログ・ファイルが、ローカル・データベース・ログ・ファイルと分けて、さらにログ・ファイル同士とも分けて保管されるようにすることをお薦めします。

4. ソース・データベースで、次の例に示すように、ダウンストリーム・マイニング・データベースの宛先に相当する LOG_ARCHIVE_DEST_n パラメータに対応する LOG_ARCHIVE_DEST_STATE n 初期化パラメータに値 ENABLE を設定します。

ALTER SYSTEM SET LOG_ARCHIVE_DEST_STATE 2=ENABLE

5. ソース・データベースとダウンストリーム・マイニング・データベースで、次の例に示すように、ソース・データベースとダウンストリーム・データベースの DB_UNIQUE_NAME を含むよう LOG_ARCHIVE_CONFIG 初期化パラメータの DG_CONFIG 属性を設定します。

ALTER SYSTEM SET LOG ARCHIVE CONFIG='DG CONFIG=(dbms1,dbmscap)'

A.3 ダウンストリーム・マイニング・データベースの準備

- ダウンストリーム・マイニング・ユーザー・アカウントの作成
- ローカル REDO ログ・ファイルをアーカイブするためのマイニング・データベースの構成
- リアルタイム・キャプチャ用のダウンストリーム・マイニング・データベースの準備

A.3.1 ダウンストリーム・マイニング・ユーザー・アカウントの作成

ダウンストリーム・マイニング構成を使用する場合、ダウンストリーム・データベースにExtract マイニング・ユーザーが必要です。マイニング Extract プロセスはこのユーザーの資格証明を使用して、ダウンストリーム・ログマイニング・サーバーと対話します。ダウンストリーム・マイニング・ユーザーは、MININGUSER オプションを使用した TRANLOGOPTIONS パラメータで指定します。ダウンストリーム・マイニング・データベースでこのユーザーを作成し、次の権限を割り当てます。

- 1. dbms_goldengate_auth.grant_admin_privilege プロシージャを実行して、ダウンストリーム・マイニング・ユーザーが統合キャプチャ・モードで操作するために必要な権限を付与します。
- 2. V_\$DATABASE に対する SELECT 権限をダウンストリーム・マイニング・ユーザーに付与します。

GRANT SELECT ON V \$DATABASE TO user;

3. 4-5 ページの「Oracle GoldenGate のデータベース・ユーザーの割当て」に示されている基本的な権限をダウンストリーム・マイニング・ユーザーに割り当てます。

A.3.2 ローカル REDO ログ・ファイルをアーカイブするためのマイニング・ データベースの構成

この手順では、REDO データをオンライン REDO ログにアーカイブするようダウンストリーム・マイニング・データベースを構成します。これらは、ダウンストリーム・マイニング・データベースで生成される REDO ログです。

Extract をリアルタイム統合キャプチャ・モードで実行する場合、ダウンストリーム・マイニング・データベースでアーカイブを有効にする必要がありますが、これはアーカイブ・ログのみキャプチャの場合も推奨されます。統合キャプチャ・モードの Extract はデータベースの状態情報を書き込みます。ダウンストリーム・マイニング・データベースでディスクの障害や破損があった場合、アーカイブと通常のバックアップによって、この状態情報をリカバリできます。

ローカル REDO ログ・ファイルをアーカイブする手順

1. ダウンストリーム・マイニング・データベースをアーカイブ・ログ・モードに変更します。 これは、次の DDL コマンドを発行して行います。

STARTUP MOUNT;

ALTER DATABASE ARCHIVELOG;

ALTER DATABASE OPEN;

2. ダウンストリーム・マイニング・データベースで、次の例に示すように、1つ目のアーカイブ・ログの宛先をLOG ARCHIVE DEST n 初期化パラメータに設定します。

ALTER SYSTEM SET

LOG_ARCHIVE_DEST_1='LOCATION=/home/arc_dest/local VALID FOR=(ONLINE LOGFILE, PRIMARY ROLE)'

あるいは、次の例のようなコマンドを使用します。

ALTER SYSTEM SET

LOG_ARCHIVE_DEST_1='LOCATION="USE_DB_RECOVERY_FILE_DEST" valid_for=(ONLINE_LOGFILE, PRIMARY ROLE)'

注意: ダウンストリーム・マイニング・データベースによって生成されるオ ンライン REDO ログはリカバリ領域にアーカイブできます。ただし、外部 アーカイブ・ログのステージングやスタンバイ REDO ログのアーカイブにダ ウンストリーム・マイニング・データベースのリカバリ領域は使用できませ ん。高速リカバリ領域の構成の詳細は、『Oracle Database バックアップおよび *リカバリ・ユーザーズ・ガイド11g リリース2(11.2)』を参*照してください。

3. ローカル・アーカイブの宛先を有効にします。

ALTER SYSTEM SET LOG ARCHIVE DEST STATE 1=ENABLE

これらの初期化パラメータの詳細は、『Oracle Data Guard *概要および管理11g リリース 2 (11.2)』* を参照してください。

A.3.3 リアルタイム・キャプチャ用のダウンストリーム・マイニング・デー タベースの準備

この手順は、ダウンストリーム・マイニング・データベースでリアルタイム・キャプチャを使 用する場合にのみ必要です。アーカイブ・ログのみキャプチャ・モードを使用する場合は必要 ありません。リアルタイム・キャプチャを使用する場合、A-4ページの「ローカル REDO ロ グ・ファイルをアーカイブするためのマイニング・データベースの構成」に示すようにローカ ル REDO データをアーカイブするようダウンストリーム・データベースがすでに構成されてい るものとします。

A.3.3.1 スタンバイ REDO ログ・ファイルの作成

次の手順では、スタンバイ REDO ログ・ファイルをダウンストリーム・マイニング・データ ベースに追加する手順を概説します。スタンバイ REDO ログの作成ルールを次に要約します。

- 各スタンバイ REDO ログ・ファイルのサイズは、少なくとも REDO ソース・データベース の最大REDOログ・ファイルと同程度である必要があります。管理を簡単にするために、 ソース・データベースのすべての REDO ログ・ファイルとダウンストリーム・マイニン グ・データベースのスタンバイ REDO ログ・ファイルを同じサイズにすることをお薦めし ます。
- スタンバイ REDO ログは、ソース・データベースの REDO スレッドごとにソース・データ ベースのREDO ログより1つ以上多い数のREDO ログ・グループを持つ必要があります。

スタンバイ REDO ログ・ファイルの追加に必要な特定の手順や SQL 文は、環境によって異な ります。スタンバイ REDO ログ・ファイルのデータベースへの追加の詳細は、『Oracle Data

注意:1つのダウンストリーム・マイニング・データベースに REDO を送信 するソース・データベースが複数ある場合、それらのソースのうち1つのみ がマイニング・データベースのスタンバイ REDO ログに REDO を送信でき ます。このソース・データベースからの REDO をマイニングする Extract プ ロセスはリアルタイム・モードで実行できます。他のソース・データベース はすべてアーカイブ・ログのみをダウンストリーム・マイニング・データ ベースに送信し、このデータを読み取る Extract はアーカイブ・ログのみモー ドで実行されるよう構成される必要があります。

スタンバイ REDO ログ・ファイルを作成する手順

- 1. SQL*Plus で、管理ユーザーとしてソース・データベースに接続します。
- 2. ソース・ログ・ファイルのサイズを確認します。結果を控えておきます。 SELECT BYTES FROM V\$LOG;

 ソース・データベースに構成されているオンライン・ログ・ファイル・グループの数を確 認します。結果を控えておきます。

SELECT COUNT (GROUP#) FROM V\$LOG;

- **4.** 管理ユーザーとしてダウンストリーム・マイニング・データベースに接続します。
- 5. スタンバイ・ログ・ファイル・グループをマイニング・データベースに追加します。スタ ンバイ・ログ・ファイルのサイズは、ソース・ログ・ファイルのサイズ以上である必要が あります。スタンバイ・ログ・ファイル・グループの数は、ソース・オンライン・ログ・ ファイル・グループの数より1つ以上多い数である必要があります。これは、RAC インス トールの各インスタンス (スレッド) に適用されます。ソース・データベースに n 個のス レッドがあり、それぞれ m 個の REDO ログ・グループがある場合、n*(m+1) 個の REDO ログ・グループをダウンストリーム・マイニング・データベースで構成する必要がありま す。

次の例は、3つのスタンバイ・ログ・グループを示しています。

ALTER DATABASE ADD STANDBY LOGFILE GROUP 3

('/oracle/dbs/slog3a.rdo', '/oracle/dbs/slog3b.rdo') SIZE 500M;

ALTER DATABASE ADD STANDBY LOGFILE GROUP 4

('/oracle/dbs/slog4.rdo', '/oracle/dbs/slog4b.rdo') SIZE 500M;

ALTER DATABASE ADD STANDBY LOGFILE GROUP 5

- ('/oracle/dbs/slog5.rdo', '/oracle/dbs/slog5b.rdo') SIZE 500M;
- **6.** スタンバイ・ログ・ファイル・グループが正常に追加されたことを確認します。

SELECT GROUP#, THREAD#, SEQUENCE#, ARCHIVED, STATUS FROM V\$STANDBY LOG;

結果は次のようになります。

GROUP#	THREAD#		SEQUENCE#	ARC	STATUS
	3	0	0	YES	UNASSIGNED
	4	0	0	YES	UNASSIGNED
	5	0	0	YES	UNASSIGNED

7. ソース・データベースのログ・ファイルが、設定したローカルの LOG ARCHIVE DEST nの LOCATION 属性で指定した場所にあることを確認します。ディレクトリ内のファイルを確認 するために、ソース・データベースでログ・ファイルの切り替えが必要な場合があります。

A.3.3.2 スタンバイ REDO ログ・ファイルをローカルにアーカイブする ためのデータベースの構成

この手順では、ソース・データベースのオンライン REDO ログから REDO データを受信する スタンバイ REDO ログをアーカイブするようダウンストリーム・マイニング・データベースを 構成します。外部アーカイブ・ログは、ダウンストリーム・マイニング・データベースのリカ バリ領域にアーカイブできないことに注意してください。

スタンバイ REDO ログをローカルでアーカイブする手順

1. ダウンストリーム・マイニング・データベースで、次の例に示すように、2つ目のアーカ イブ・ログの宛先をLOG ARCHIVE DEST n初期化パラメータに設定します。

ALTER SYSTEM SET

LOG ARCHIVE DEST 2='LOCATION=/home/arc dest/srl dbms1 VALID FOR=(STANDBY LOGFILE, PRIMARY ROLE)'

外部アーカイブ・ログ(リモート・ソース・データベースからのログ)はローカル・マイニ ング・データベース・ログ・ファイルと分けて保管し、ログ・ファイル同士も分けて保管 することをお薦めします。外部アーカイブ・ログのステージングにダウンストリーム・マ イニング・データベースのリカバリ領域を使用することはできません。高速リカバリ領域 の構成の詳細は、『Oracle Database バックアップおよびリカバリ・ユーザーズ・ガイド11g UU-Z(11.2)』を参照してください。

注意:別のデータベースによって生成された REDO をダウンストリーム・マ イニング・データベースのスタンバイ REDO ログで受け入れる場合、前述の とおりスタンバイ REDO ログ用に log_archive_dest_2 を構成する必要があ ります。

2. 次の例に示すように、前の手順で設定した LOG_ARCHIVE_DEST_2 パラメータを有効にしま す。

ALTER SYSTEM SET LOG_ARCHIVE_DEST_STATE_2=ENABLE

これらの初期化パラメータの詳細は、『Oracle Data Guard 概要および管理11g リリース2 (11.2)』を参照してください。

統合キャプチャ用のダウンストリーム・ マイニング・データベースの構成例

この付録では、統合キャプチャ・モードの Extract をサポートするようダウンストリーム Oracle マイニング・データベースを準備する場合の例を示します。ダウンストリーム・マイニ ング・データベースの構成の詳細は、A-1ページの「統合キャプチャ用のダウンストリーム・ マイニング・データベースの構成」を参照してください。

B.1 例 1: リアルタイム・モードでの 1 つのソース・データベー スからのキャプチャ

注意: この例では、必要なスタンバイ REDO ログ・ファイルを A-1 ページの 「統合キャプチャ用のダウンストリーム・マイニング・データベースの構成」 に示すように作成してあるものとします。

この例では、統合キャプチャ・セッションをダウンストリーム・マイニング・データベース DBMSCAP にデプロイし、ソース・データベース DBMS1 から変更をキャプチャします。次の ユーザーが存在するものとします。

- DBMS1 にユーザー GGADM1。Extract はこの資格証明を使用して、DBMS1 からデータと メタデータをフェッチします。ユーザー GGADM1 は、ソース・データベースで V \$DATABASE ビューに対する SELECT 権限を持っています。DBMS GOLDENGATE AUTH.GRANT ADMIN_PRIVILEGE() プロシージャをコールして、ソース・データベースで適切な権限がこ のユーザーに付与されているものとします。
- DBMSCAP にユーザー GGADMCAP。Extract はこの資格証明を使用して、論理変更レ コードをダウンストリーム・マイニング・データベース DBMSCAP のログマイニング・ サーバーから取得します。DBMS GOLDENGATE AUTH.GRANT ADMIN PRIVILEGE() プロシー ジャをコールして、マイニング・データベースで適切な権限がこのユーザーに付与されて いるものとします。ユーザー GGADMCAP は、ダウンストリーム・マイニング・データ ベースで V \$DATABASE ビューに対する SELECT 権限を持っています。

B.1.1 ローカル REDO をアーカイブするためのマイニング・データベース の準備

1. ダウンストリーム・マイニング・データベースはアーカイブ・ログ・モードである必要が あります。これは、次の DDL コマンドを発行して行います。

STARTUP MOUNT:

ALTER DATABASE ARCHIVELOG;

ALTER DATABASE OPEN;

2. ダウンストリーム・マイニング・データベースで log archive dest 1をアーカイブ・ ローカル REDO に設定します。

ALITER SYSTEM SET

LOG_ARCHIVE_DEST_1='LOCATION=/home/arc_dest/local

VALID FOR=(ONLINE LOGFILE, PRIMARY ROLE)'

3. log_archive_dest_1 を有効にします。
ALTER SYSTEM SET LOG ARCHIVE DEST STATE 1=ENABLE

B.1.2 ソース・データベースから受信した REDO をスタンバイ REDO ログ にアーカイブするためのマイニング・データベースの準備

1. ダウンストリーム・マイニング・データベースで、次の例に示すように log_archive_dest_2 を設定します。

ALTER SYSTEM SET

LOG_ARCHIVE_DEST_2='LOCATION=/home/arc_dest/srl_dbms1

VALID FOR=(STANDBY LOGFILE, PRIMARY ROLE)'

- 次の例に示すように log_archive_dest_2 を有効にします。
 ALTER SYSTEM SET LOG_ARCHIVE_DEST_STATE_2=ENABLE
- **3.** ダウンストリーム・マイニング・データベースで DG_CONFIG を設定します。 ALTER SYSTEM SET LOG ARCHIVE CONFIG='DG CONFIG=(dbms1,dbmscap)'

B.1.3 REDO をマイニング・データベースに送信するためのソース・データベースの準備

1. ソース・データベースが、必要とされる互換性で稼働していることを確認します。 select name, value from v\$parameter where name = 'compatible';

統合キャプチャに必要な最低限の互換性設定は、10.2.0.0.0です。

- ソース・データベースで DG_CONFIG を設定します。
 ALTER SYSTEM SET LOG_ARCHIVE_CONFIG='DG_CONFIG=(dbms1,dbmscap)';
- 3. ソース・データベースで REDO 転送を設定します。

ALTER SYSTEM

SET LOG_ARCHIVE_DEST_2='SERVICE=DBMSCAP.EXAMPLE.COM ASYNC OPTIONAL NOREGISTER VALID FOR=(ONLINE LOGFILES, PRIMARY ROLE) DB UNIQUE NAME=dbmscap;

ダウンストリームの宛先を有効にします。
 ALTER SYSTEM SET LOG_ARCHIVE_DEST_STATE_2=ENABLE;

B.1.4 DBMSCAP での統合キャプチャ (ext1) の設定

- 1. Extract をダウンストリーム・マイニング・データベースに登録します。
 - GGSCI> DBLOGIN USERID ggadml@dbms1 PASSWORD ggadmlpw GGSCI> MININGDBLOGIN USERID ggadmcap@dbmscap PASSWORD ggadmcappw GGSCI> REGISTER EXTRACT extl DATABASE
- 2. Extract をダウンストリーム・マイニング・データベースで作成します。 GGSCI> ADD EXTRACT ext1 INTEGRATED TRANLOG BEGIN NOW

3. Extract パラメータ・ファイル ext1.prm を編集します。リアルタイム・キャプチャを使用 するには、次の行が必要です。

USERID ggadm1@dbms1 PASSWORD ggadm1pw TRANLOGOPTIONS MININGUSER qqadmcap@dbmscap MININGPASSWORD qqadmcappw TRANLOGOPTIONS INTEGRATEDPARAMS (downstream real time mine Y)

4. Extract を開始します。

GGSCI> START EXTRACT ext1

注意:同一のソース・データベースからデータをキャプチャする(前述の例 でデータベース DBMS1 の変更をキャプチャするなど)かぎり、リアルタイ ム統合キャプチャ・モードで実行される Extract をダウンストリーム・マイニ ング・データベースに複数作成できます。

B.2 例 2: アーカイブログのみモードでの複数のソースからの キャプチャ

次の例では、統合キャプチャ・セッションをダウンストリーム・マイニング・データベース DBMSCAP にデプロイし、データベース DBMS1 および DBMS2 から変更をキャプチャします。 次のユーザーが存在するものとします。

- DBMS1 にユーザー GGADM1。Extract はこの資格証明を使用して、DBMS1 からデータと メタデータをフェッチします。ユーザー GGADM1 は、DBMS1 で v \$database に対する SELECT 権限を持っています。DBMS GOLDENGATE AUTH.GRANT ADMIN PRIVILEGE()プロ シージャをコールして、DBMS1で適切な権限がこのユーザーに付与されているものとしま
- DBMS2 にユーザー GGADM2。Extract はこの資格証明を使用して、DBMS2 からデータと メタデータをフェッチします。ユーザー GGADM2 は、DBMS2 で v \$database に対する SELECT 権限を持っています。DBMS GOLDENGATE AUTH.GRANT ADMIN PRIVILEGE()プロ シージャをコールして、DBMS2で適切な権限がこのユーザーに付与されているものとしま
- DBMSCAP にユーザー GGADMCAP。Extract はこの資格証明を使用して、論理変更レ コードをダウンストリーム・マイニング・データベースのログマイニング・サーバーから 取得します。DBMS GOLDENGATE AUTH.GRANT ADMIN PRIVILEGE() プロシージャをコールし て、ダウンストリーム・マイニング・データベース DBMSCAP で適切な権限がこのユー ザーに付与されているものとします。ユーザー GGADMCAP は、DBMSCAP で v \$database に対する SELECT 権限を持っています。

この手順では、ダウンストリーム・マイニング・データベースがアーカイブ・ログ・モードで 構成されていることも前提とします。

B.2.1 ローカル REDO をアーカイブするためのマイニング・データベース の準備

1. ダウンストリーム・マイニング・データベースはアーカイブ・ログ・モードである必要が あります。これは、次の DDL コマンドを発行して行います。

STARTUP MOUNT;

ALTER DATABASE ARCHIVELOG;

ALTER DATABASE OPEN;

2. ダウンストリーム・マイニング・データベースで log archive dest 1をアーカイブ・ ローカル REDO に設定します。

ALTER SYSTEM SET

LOG_ARCHIVE_DEST_1='LOCATION=/home/arc_dest/local VALID_FOR=(ONLINE_LOGFILE, PRIMARY_ROLE)'

log_archive_dest_1 を有効にします。
 ALTER SYSTEM SET LOG_ARCHIVE_DEST_STATE 1=ENABLE

B.2.2 ソース・データベースからの REDO をアーカイブするためのマイニング・データベースの準備

ダウンストリーム・マイニング・データベースで DG_CONFIG を設定します。 ALTER SYSTEM SET LOG_ARCHIVE_CONFIG='DG_CONFIG=(dbms1,dbms2, dbmscap)'

B.2.3 REDO をマイニング・データベースに送信するための 1 つ目のソース・データベースの準備

1. DBMS1 ソース・データベースが、必要とされる互換性で稼働していることを確認します。 select name, value from v\$parameter where name = 'compatible';

統合キャプチャに必要な最低限の互換性設定は、10.2.0.0.0です。

- 2. DBMS1 ソース・データベースで DG_CONFIG を設定します。 ALTER SYSTEM SET LOG_ARCHIVE_CONFIG='DG_CONFIG=(dbms1, dbmscap)';
- 3. DBMS1 ソース・データベースで REDO 転送を設定します。REDO データをダウンストリーム・マイニング・データベースの外部アーカイブ・ログに直接送信する場合、TEMPLATE 句は必須です。

ALTER SYSTEM

SET LOG_ARCHIVE_DEST_2='SERVICE=DBMSCAP.EXAMPLE.COM ASYNC OPTIONAL NOREGISTER TEMPLATE='/usr/orcl/arc_dest/dbms1/dbms1_arch_%t_%s_%r.log VALID_FOR=(ONLINE_LOGFILES, PRIMARY ROLE)DB UNIQUE NAME=dbmscap;

4. ダウンストリームの宛先を有効にします。

ALTER SYSTEM SET LOG_ARCHIVE_DEST_STATE_2=ENABLE;

B.2.4 REDO をマイニング・データベースに送信するための 2 つ目のソース・データベースの準備

1. DBMS2 ソース・データベースが、必要とされる互換性で稼働していることを確認します。 select name, value from v\$parameter where name = 'compatible';

統合キャプチャに必要な最低限の互換性設定は、10.2.0.0.0です。

- 2. DBMS2 ソース・データベースで DG_CONFIG を設定します。
 - ALTER SYSTEM SET LOG_ARCHIVE_CONFIG='DG_CONFIG=(dbms2, dbmscap)';
- 3. DBMS2 ソース・データベースで REDO 転送を設定します。REDO データをダウンストリーム・マイニング・データベースの外部アーカイブ・ログに直接送信する場合、TEMPLATE 句は必須です。

ALTER SYSTEM

SET LOG_ARCHIVE_DEST_2='SERVICE=DBMSCAP.EXAMPLE.COM ASYNC OPTIONAL NOREGISTER TEMPLATE='/usr/orcl/arc_dest/dbms2/dbms2_arch_%t_%s_%r.log VALID_FOR=(ONLINE_

LOGFILES, PRIMARY ROLE) DB UNIQUE NAME=dbmscap';

4. ダウンストリームの宛先を有効にします。 ALTER SYSTEM SET LOG ARCHIVE DEST STATE 2=ENABLE;

B.2.5 ダウンストリーム・マイニング・データベースでの Extract の設定

この手順では、DBMS1 および DBMS2 によって送信されたアーカイブ・ログからキャプチャす るようダウンストリーム・データベースで Extract を設定します。

B.2.5.1 DBMS1 によって送信されたアーカイブ・ログから変更をキャプ チャするための Extract (ext1) の設定

DBMSCAP ダウンストリーム・マイニング・データベースで次の手順を実行します。

- 1. DBMS1 ソース・データベース用に DBMSCAP に Extract を登録します。
 - GGSCI> DBLOGIN USERID ggadm1@dbms1 PASSWORD ggadm1pw
 - GGSCI> MININGDBLOGIN USERID ggadmcap@dbmscap PASSWORD ggadmcappw
 - GGSCI> REGISTER EXTRACT ext1 DATABASE
- **2.** Extract をマイニング・データベース DBMSCAP で追加します。
 - GGSCI> ADD EXTRACT ext1 INTEGRATED TRANLOG BEGIN NOW
- 3. Extract パラメータ・ファイル ext1.prm を編集します。

USERID ggadm1@dbms1 PASSWORD ggadm1pw TRANLOGOPTIONS MININGUSER ggadmcap@dbmscap, MININGPASSWORD ggadmcappw TRANLOGOPTIONS INTEGRATEDPARAMS (downstream real time mine N)

4. Extract を開始します。

GGSCI > START EXTRACT ext1

B.2.5.2 DBMS2 によって送信されたアーカイブ・ログから変更をキャプ チャするための Extract (ext2) の設定

ダウンストリーム DBMSCAP マイニング・データベースで次の手順を実行します。

- 1. ソース・データベース DBMS2 用にマイニング・データベースに Extract を登録します。
 - GGSCI> DBLOGIN USERID ggadm2@dbms2, PASSWORD ggadm2pw
 - GGSCI> MININGDBLOGIN USERID ggadmcap@dbmscap, PASSWORD ggadmcappw
 - GGSCI> REGISTER EXTRACT ext2 DATABASE
- 2. Extract をマイニング・データベースで作成します。
 - GGSCI> ADD EXTRACT ext2 INTEGRATED TRANLOG, BEGIN NOW
- 3. Extract パラメータ・ファイル ext2.prm を編集します。

USERID ggadm2@dbms2, PASSWORD ggadm2pwd TRANLOGOPTIONS MININGUSER ggadmcap@dbmscap, MININGPASSWORD ggadmcappw TRANLOGOPTIONS INTEGRATEDPARAMS (downstream_real_time_mine N)

4. Extract を開始します。

GGSCI > START EXTRACT ext2

注意: ダウンストリーム・マイニング・データベースでアーカイブログのみ モードで実行されている場合、同一のソース・データベースからデータを キャプチャする Extract を複数作成できます。この例の場合、データベース DBMS1 および DBMS2 の変更をキャプチャする他の Extract をダウンスト リーム・マイニング・データベースに作成できます。

B.3 例 3: リアルタイム・モードとアーカイブログのみモードが 混在する複数ソースからのキャプチャ

注意:この例では、必要なスタンバイ REDO ログ・ファイルを A-1 ページの 「統合キャプチャ用のダウンストリーム・マイニング・データベースの構成」 に示すように作成してあるものとします。

次の例では、統合キャプチャ・セッションをダウンストリーム・マイニング・データベース DBMSCAP にデプロイし、データベース DBMS1、DBMS2 および DBMS3 から変更をキャプチャします。次のユーザーが存在するものとします。

- DBMS1 にユーザー GGADM1。Extract はこの資格証明を使用して、DBMS1 からデータとメタデータをフェッチします。ユーザー GGADM1 は、DBMS1 で v_\$database に対する SELECT 権限を持っています。DBMS_GOLDENGATE_AUTH.GRANT_ADMIN_PRIVILEGE() プロシージャをコールして、DBMS1 で適切な権限がこのユーザーに付与されているものとします。
- DBMS2 にユーザー GGADM2。Extract はこの資格証明を使用して、DBMS2 からデータとメタデータをフェッチします。ユーザー GGADM2 は、DBMS2 で v_\$database に対する SELECT 権限を持っています。DBMS_GOLDENGATE_AUTH.GRANT_ADMIN_PRIVILEGE() プロシージャをコールして、DBMS2 で適切な権限がこのユーザーに付与されているものとします。
- DBMS3 にユーザー GGADM3。Extract はこの資格証明を使用して、DBMS3 からデータとメタデータをフェッチします。ユーザー GGADM3 は、DBMS3 で v_\$database に対する SELECT 権限を持っています。DBMS_GOLDENGATE_AUTH.GRANT_ADMIN_PRIVILEGE() プロシージャをコールして、DBMS3 で適切な権限がこのユーザーに付与されているものとします。
- DBMSCAP にユーザー GGADMCAP。Extract はこの資格証明を使用して、論理変更レコードをダウンストリーム・マイニング・データベースのログマイニング・サーバーから取得します。DBMS_GOLDENGATE_AUTH.GRANT_ADMIN_PRIVILEGE() プロシージャをコールして、ダウンストリーム・マイニング・データベース DBMSCAP で適切な権限がこのユーザーに付与されているものとします。ユーザー GGADMCAP は、DBMSCAP でv_\$database に対する SELECT 権限を持っています。

この手順では、ダウンストリーム・マイニング・データベースがアーカイブ・ログ・モードで構成されていることも前提とします。

この例では、DBMS3 によって送信される REDO データはリアルタイム・モードでマイニングされ、DBMS1 および DBMS2 から送信される REDO データはアーカイブログのみモードでマイニングされます。

B.3.1 ローカル REDO をアーカイブするためのマイニング・データベース の準備

1. ダウンストリーム・マイニング・データベースはアーカイブ・ログ・モードである必要があります。これは、次の DDL コマンドを発行して行います。

STARTUP MOUNT; ALITER DATABASE ARCHIVELOG; ALITER DATABASE OPEN; 2. ダウンストリーム・マイニング・データベースで log_archive_dest_1 をアーカイブ・ローカル REDO に設定します。

ALITER SYSTEM SET

LOG_ARCHIVE_DEST_1='LOCATION=/home/arc_dest/local

VALID FOR=(ONLINE LOGFILE, PRIMARY ROLE)'

log_archive_dest_1 を有効にします。
 ALTER SYSTEM SET LOG ARCHIVE DEST STATE 1=ENABLE

B.3.2 ソース・データベースから REDO を受け入れるためのマイニング・ データベースの準備

REDO データは、ダウンストリーム・マイニング・データベースのスタンバイ REDO ログに受け入れられるため、正しいサイズに設定されたスタンバイ REDO ログが適切な数存在する必要があります。スタンバイ・ログを構成していない場合、A-1ページの「統合キャプチャ用のダウンストリーム・マイニング・データベースの構成」を参照してください。

1. ダウンストリーム・マイニング・データベースで、次の例に示すように、2つ目のアーカイブ・ログの宛先を LOG_ARCHIVE_DEST_n 初期化パラメータに設定します。これは、アーカイブ・スタンバイ REDO ログを処理するために必要です。

ALTER SYSTEM SET
LOG ARCHIVE DEST 2='LOCATION=/home/s

LOG_ARCHIVE_DEST_2='LOCATION=/home/arc_dest/srl_dbms3
VALID_FOR=(STANDBY_LOGFILE, PRIMARY_ROLE)'

 次の例に示すように、LOG_ARCHIVE_DEST_2 パラメータに対応する LOG_ARCHIVE_DEST_ STATE_2 初期化パラメータを有効にします。

ALTER SYSTEM SET LOG_ARCHIVE_DEST_STATE_2=ENABLE

3. ダウンストリーム・マイニング・データベースですべてのソース・データベースから REDO データを受け入れるよう DG_CONFIG を設定します。

ALTER SYSTEM SET LOG_ARCHIVE_CONFIG='DG_CONFIG=(dbms1, dbms2, dbms3, dbmscap)'

B.3.3 REDO をマイニング・データベースに送信するための 1 つ目のソース・データベースの準備

1. DBMS1 ソース・データベースが、必要とされる互換性で稼働していることを確認します。 select name, value from v\$parameter where name = 'compatible';

統合キャプチャに必要な最低限の互換性設定は、10.2.0.0.0です。

DBMS1 ソース・データベースで DG_CONFIG を設定します。
 ALTER SYSTEM SET LOG ARCHIVE CONFIG='DG CONFIG=(dbms1, dbmscap)';

3. DBMS1 ソース・データベースで REDO 転送を設定します。REDO データをダウンストリーム・マイニング・データベースの外部アーカイブ・ログに直接送信する場合、TEMPLATE 句は必須です。

ALTER SYSTEM

SET LOG_ARCHIVE_DEST_2='SERVICE=DBMSCAP.EXAMPLE.COM ASYNC OPTIONAL NOREGISTER TEMPLATE='/usr/orcl/arc_dest/dbms1/dbms1_arch_%t_%s_%r.log VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE)DB_UNIQUE_NAME=dbmscap';

4. ダウンストリームの宛先を有効にします。

ALTER SYSTEM SET LOG_ARCHIVE_DEST_STATE_2=ENABLE;

B.3.4 REDO をマイニング・データベースに送信するための 2 つ目のソー ス・データベースの準備

1. DBMS2 ソース・データベースが、必要とされる互換性で稼働していることを確認します。 select name, value from v\$parameter where name = 'compatible';

NAME: VALUE compatible 10.2.0.3.0

統合キャプチャに必要な最低限の互換性設定は、10.2.0.0.0です。

2. DBMS2 ソース・データベースで DG CONFIG を設定します。

ALTER SYSTEM SET LOG_ARCHIVE_CONFIG='DG_CONFIG=(dbms2, dbmscap)';

3. DBMS2 ソース・データベースで REDO 転送を設定します。REDO データをダウンスト リーム・マイニング・データベースの外部アーカイブ・ログに直接送信する場合、 TEMPLATE 句は必須です。

ALTER SYSTEM

SET LOG_ARCHIVE_DEST_2='SERVICE=DBMSCAP.EXAMPLE.COM ASYNC OPTIONAL NOREGISTER TEMPLATE='/usr/orcl/arc_dest/dbms2/dbms2_arch_%t_%s_%r.log VALID_FOR=(ONLINE_ LOGFILES, PRIMARY_ROLE) DB_UNIQUE_NAME=dbmscap';

4. ダウンストリームの宛先を有効にします。

ALTER SYSTEM SET LOG ARCHIVE DEST STATE 2=ENABLE;

B.3.5 REDO をマイニング・データベースに送信するための3つ目のソー ス・データベースの準備

1. DBMS3 ソース・データベースが、必要とされる互換性で稼働していることを確認します。 select name, value from v\$parameter where name = 'compatible';

NAME VALUE compatible 11.2.0.3.0

統合キャプチャに必要な最低限の互換性設定は、10.2.0.0.0です。

2. DBMS3 ソース・データベースで DG CONFIG を設定します。

ALTER SYSTEM SET LOG ARCHIVE CONFIG='DG CONFIG=(dbms3, dbmscap)';

3. DBMS3 ソース・データベースで REDO 転送を設定します。DBMS3 はオンライン REDO ログをダウンストリーム・マイニング・データベースのスタンバイ REDO ログに送信する ソースであるため、TEMPLATE 句は指定しないでください。

ALTER SYSTEM

SET LOG ARCHIVE DEST 2='SERVICE=DBMSCAP.EXAMPLE.COM ASYNC OPTIONAL NOREGISTER VALID FOR= (ONLINE LOGFILES, PRIMARY ROLE) DB UNIQUE NAME=dbmscap;

4. ダウンストリームの宛先を有効にします。

ALTER SYSTEM SET LOG ARCHIVE DEST STATE 2=ENABLE;

B.3.6 ダウンストリーム・マイニング・データベースでの Extract の設定

この手順では、DBMS1 および DBMS2 によって送信されたアーカイブ・ログからキャプチャす るようダウンストリーム・データベースで Extract を設定します。

B.3.6.1 DBMS1 によって送信されたアーカイブ・ログから変更をキャプ チャするための Extract (ext1) の設定

DBMSCAP ダウンストリーム・マイニング・データベースで次の手順を実行します。

- 1. DBMS1 ソース・データベース用に DBMSCAP に Extract を登録します。
 - GGSCI> DBLOGIN USERID ggadm1@dbms1 PASSWORD ggadm1pw
 - GGSCI> MININGDBLOGIN USERID ggadmcap@dbmscap PASSWORD ggadmcappw
 - GGSCI> REGISTER EXTRACT ext1 DATABASE
- **2.** Extract をマイニング・データベース DBMSCAP で追加します。
 - GGSCI> ADD EXTRACT ext1 INTEGRATED TRANLOG BEGIN NOW
- 3. Extract パラメータ・ファイル ext1.prm を編集します。
 - USERID ggadm1@dbms1 PASSWORD ggadm1pw TRANLOGOPTIONS MININGUSER ggadmcap@dbmscap, MININGPASSWORD ggadmcappw TRANLOGOPTIONS INTEGRATEDPARAMS (downstream real time mine N)
- 4. Extract を開始します。
 - GGSCI> START EXTRACT ext1

B.3.6.2 DBMS2 によって送信されたアーカイブ・ログから変更をキャプ チャするための Extract (ext2) の設定

DBMSCAP ダウンストリーム・マイニング・データベースで次の手順を実行します。

- 1. ソース・データベース DBMS2 用にマイニング・データベースに Extract を登録します。
 - GGSCI> DBLOGIN USERID ggadm2@dbms2, PASSWORD ggadm2pw
 - GGSCI> MININGDBLOGIN USERID ggadmcap@dbmscap, PASSWORD ggadmcappw
 - GGSCI > REGISTER EXTRACT ext2 DATABASE
- 2. Extract をマイニング・データベースで作成します。
 - GGSCI> ADD EXTRACT ext2 INTEGRATED TRANLOG, BEGIN NOW
- 3. Extract パラメータ・ファイル ext2.prm を編集します。
 - USERID ggadm2@dbms2, PASSWORD ggadm2pwd TRANLOGOPTIONS MININGUSER ggadmcap@dbmscap, MININGPASSWORD ggadmcappw TRANLOGOPTIONS INTEGRATEDPARAMS (downstream_real_time_mine N)
- **4.** Extract を起動します。
 - GGSCI > START EXTRACT ext2

B.3.6.3 DBMS3 によって送信されたオンライン・ログからリアルタイ ム・モードで変更をキャプチャするための Extract (ext3) の設定

DBMSCAP ダウンストリーム・マイニング・データベースで次の手順を実行します。

- 1. ソース・データベース DBMS3 用にマイニング・データベースに Extract を登録します。
 - GGSCI> DBLOGIN USERID ggadm3@dbms3, PASSWORD ggadm3pw
 - GGSCI> MININGDBLOGIN USERID ggadmcap@dbmscap, PASSWORD ggadmcappw
 - GGSCT > REGISTER EXTRACT ext3 DATABASE

2. Extract をマイニング・データベースで作成します。

GGSCI> ADD EXTRACT ext3 INTEGRATED TRANLOG, BEGIN NOW

3. Extract パラメータ・ファイル ext3.prm を編集します。リアルタイム・マイニングを有効 にするには、downstream_real_time_mineを指定する必要があります。

USERID ggadm3@dbms3, PASSWORD ggadm3pwd ${\tt TRANLOGOPTIONS\ MININGUSER\ ggadmcap@dbmscap,\ MININGPASSWORD\ ggadmcappw}$ TRANLOGOPTIONS INTEGRATEDPARAMS (downstream_real_time_mine Y)

4. Extract を起動します。

GGSCI> START EXTRACT ext3

注意:同一のソース・データベースからデータをキャプチャする(前述の例 でデータベース DBMS3 の変更をキャプチャするなど) かぎり、リアルタイ ム統合キャプチャ・モードで実行される Extract をダウンストリーム・マイニ ング・データベースに複数作成できます。

XML スキーマの変更のサポート

この項では、XML スキーマの変更をサポートするための手順について説明します。クラシッ ク・キャプチャと統合キャプチャの両方のモードとも、XML スキーマへの変更のキャプチャは サポートしません。

C.1 RegisterSchema のサポート

RegisterSchema は、XML スキーマを参照する表を作成する前に、ソースとターゲットの両方 のデータベースでスキーマ定義を登録することで処理されます。

C.2 DeleteSchema のサポート:

まず、ソース・データベースで DeleteSchema を発行します。Replicat でソース・データベース の変更に追い付いたら、ターゲット・データベースで DeleteSchema コールを発行します。

C.3 CopyEvolve のサポート

CopyEvolve プロシージャではスキーマを展開(変更)し、列を追加または削除することで表を 変更できます。XML ドキュメントが有効かどうかの変更にも使用できます。CopyEvolve の処 理には、さらに調整が必要です。ソース・データベースで CopyEvolve を発行する場合、次の手 順を使用します。

- 1. ソース・データベースで独立した表への変更を停止します。
- 2. プライマリまたはソースのデータベースで CopyEvolve を実行します。
- 3. Replicat がそれらの表からのデータをすべてターゲット・データベースに適用し終えるの を待ちます。
- **4.** Replicat を停止します。
- 5. ターゲット・データベースで CopyEvolve を適用します。
- **6.** Replicat を再起動します。

Oracle GoldenGate とのアクティブ/ アクティブ伝播のための DBFS の準備

この章では、Oracle Database File System (DBFS) が両方(またはすべて)のシステムで使用さ れているアクティブ/アクティブの双方向環境または多方向環境内で機能するよう、Oracle GoldenGate を構成する手順について説明します。

D.1 サポートされている操作および前提条件

Oracle GoldenGate では DBFS 向けに次のものがサポートされています。

- DBFS オブジェクトに対する、CREATE 文以外のサポートされている DDL(TRUNCATE、ALTER など)。DBFS に対する CREATE は、作成された DBFS オブジェクトを保持するすべてのス キーマ同様、構成から除外する必要があります。CREATE を除外するのは、DBFS のメタ データを SYS ディクショナリ表 (それ自体がデフォルトで Oracle GoldenGate キャプチャ から除外される)に正しく移入する必要があるためです。
- DBFS ファイル・システムの基礎となる表での DML のキャプチャおよびレプリケーショ

後述の手順では、Oracle GoldenGate がアクティブ / アクティブ構成をサポートするよう、正し く構成されていることを前提とします。つまり、次のように設定されている必要があります。

- このガイド内の指示に従ってインストールされています。
- 『Oracle GoldenGate Windows and UNIX 管理者ガイド』の指示に従って構成されています。

D.2 必要なパッチの適用

両方のデータベースに Oracle Bug#9651229 用の Oracle DBFS パッチを適用します。パッチがイ ンストールされているかどうかを判断するには、次の問合せを実行します。

connect / as sysdba select procedure name dba procedures where object name = 'DBMS DBFS SFS ADMIN' and procedure name = 'PARTITION SEQUENCE';

問合せでは1行が返されます。それ以外の場合は、パッチが適用された適切なバージョンの DBFS がデータベースに存在しないことを示します。

D.3 これらのプロシージャで使用されている例

次のプロシージャでは、2つのシステムがあると仮定し、両方のシステムの DBFS ユーザーが、 Oracle GoldenGate との同期が維持されている同一の DBFS ファイル、ディレクトリおよびコ ンテンツを参照できるように環境を構成します。これらの概念を、3 つ以上のピア・システム のサポートに適用することも可能です。

D.4 DBFS 順序番号のパーティション化

DBFSでは内部順序番号ジェネレータを使用して、一意の名前と一意の ID が作成されます。こ れらの手順では、データベース間で競合することがないように順序が識別可能な範囲にパー ティション化されます。これが実行された後は、DMLの伝播中に名前、主キーまたは ID が競 合することなく、他の DBFS 操作(新しいファイル・システムの作成とその後のファイル・シ ステム操作の両方)を実行できます。

1. 各データベースに sysdba として接続します。

次の問合せを各データベースで発行します。

```
select last_number
from dba sequences
where sequence_owner = 'SYS'
and sequence name = 'DBFS SFS $FSSEQ'
```

- 2. この問合せから、両方のシステム間でLAST NUMBER の最大値を選択するか、いずれかのシ ステム上にある順序の現在値よりも大幅に大きい値を選択します。
- 3. 次のプロシージャの両方で、この値(ここではプレースホルダとして "maxval" を使用)を 置換します。これらのプロシージャでは、各システムは myid=0 および myid=1 として論理 的に索引付けされます。

ノード1

```
declare
   begin
   dbms dbfs sfs admin.partition sequence(nodes => 2, myid => 0, newstart => :maxval);
   commit:
   end:
ノード2
   declare
   dbms_dbfs_sfs_admin.partition_sequence( nodes => 2, myid => 1, newstart =>
    :maxval);
   commit;
   end;
```

注意: myid パラメータに指定された値の違いに注意してください。これが索 引値の違いです。

3つ以上のデータベース間での多方向構成の場合は、次のように変更できます。

- maxval に設定されている最大値を適切に増加して調整し、すべてのノードでその値を 使用します。
- プロシージャ内の myid の値として、最初のノードには 0、2番目のノードには 1、3番 目には2、などのように異なる値を設定します。

4. (推奨) DBFS 順序ジェネレータがパーティション化された後でのみ、各システム上に新し い DBFS ファイル・システムを作成し、Oracle GoldenGate を使用した DML 伝播にこれら のファイル・システムのみを使用します。D-3ページの「DBFS ファイル・システムの構 成しを参照してください。

注意: Oracle Bug#9651229 のパッチが適用される前または DBFS 順序番号が 調整される前に作成された DBFS ファイル・システムを伝播用に構成できま すが、このドキュメントで説明されていない追加の手順が必要になります。 古いファイル・システムを保持している場合は、Oracle Support でサービ ス・リクエストをオープンしてください。

D.5 DBFS ファイル・システムの構成

DBFS ファイル・システム操作をレプリケートするには、DML 用の標準の双方向構成と同様の 構成を使用します。

- 構造が同一である表の組合せを使用します。
- 各データベースが組合せのもう一方の表に対する書込み権限を持つように設定し、もう一 方を読取り専用に設定します。例を次に示します。
 - ノード 1 がローカル表 t1 に書き込み、これらの変更がノード 2 の t1 にレプリケートさ れます。
 - ノード2がローカル表 t2 に書き込み、これらの変更がノード1の t2 にレプリケートさ れます。
 - ノード1では、t2は読取り専用です。ノード2では、t1は読取り専用です。

DBFS ファイル・システムでこのような表の組合せを簡単に行える理由は、次のとおりです。

- DBFS ファイル・システムの基礎となる表が同じ構造である。
- これらの表は、高レベルのファイル・システム操作中に、従来型の単純な DML により変更 される。
- DBFS コンテンツ API により、読取り / 書込みまたは読取り専用として修飾可能なマウン ト・ポイントを使用して、個別の DBFS ストアのネームスペースを統一する方法が提供さ れる。

次の手順では、2 つの DBFS ファイル・システム (この場合は FS1 および FS2 という名前)を作 成し、必要に応じてそれらを読取り/書込みまたは読取りに設定します。

次のプロシージャを実行して、2つのファイル・システムを作成します。(FS1 および FS2 をご使用のストア名に置換します。)

例 D-1

```
declare
dbms_dbfs_sfs.createFilesystem('FS1');
dbms_dbfs_sfs.createFilesystem('FS2');
dbms_dbfs_content.registerStore('FS1',
'posix', 'DBMS DBFS SFS');
dbms dbfs content.registerStore('FS2',
'posix', 'DBMS DBFS SFS');
commit;
end;
```

2. 次のプロシージャを実行し、各ファイル・システムに適切なアクセス権限を付与します。 (FS1 および FS2 をご使用のストア名に置換します。)

例 D-2 ノード1

```
dbms dbfs content.mountStore('FS1', 'local');
dbms dbfs content.mountStore('FS2', 'remote',
read only => true);
commit:
end;
例 D-3 ノード2
declare
dbms dbfs content.mountStore('FS1', 'remote',
read only => true);
dbms dbfs content.mountStore('FS2', 'local');
commit:
end:
```

この例において、ノード1ではストア FS1 は読取り/書込み、ストア FS2 は読取り専用で あるのに対し、ノード2ではストア FS1 が読取り専用、ストア FS2 が読取り/書込みとい うように逆になる点に注意してください。

また、読取り/書込みストアは*ローカル*としてマウントされ、読取り専用ストアは*リモー* **/としてマウントされる点にも注意してください。これにより、読取り操作と書込み操作** に対して、同一のネームスペースおよび同一のセマンティクスが各システム上のユーザー に提供されます。ローカルのパス名は変更できますが、リモートのパス名は変更できませ No.

D.6 ローカル・ピアとリモート・ピアの正しいマップ

DBFS ファイル・システムの基礎となる表の名前は内部的および動的に生成されます。前の例 に引き続き、次のものがあるとします。

- 2つのノード(この例ではノード1とノード2)。
- 各ノードに2つずつ(この例ではFS1とFS2)、4つのストア。
- 各ストアに2つずつ(table と ptable)、基礎となる8つの表。これらの表は、Extract の TABLE 文で識別、指定され、Replicat の MAP 文でマップされる必要があります。
- 1. 各ファイル・システムの基礎となる表の名前を識別するには、次の問合せを発行します。 (FS1 および FS2 をご使用のストア名に置換します。)

例 D-4

```
select fs.store name, tb.table name, tb.ptable name
from table (dbms dbfs sfs.listTables) tb,
table(dbms dbfs sfs.listFilesystems) fs
where fs.schema name = tb.schema name
and fs.table_name = tb.table_name
and fs.store_name in ('FS1', 'FS2')
```

出力は次の例のようになります。

例 D-5 出力例: ノード 1(実際の表名は異なります。)

STORE NAME	TABLE_NAME	PTABLE_NAME
FS1	SFS\$_FST_100	SFS\$_FSTP_100
FS2	SFS\$_FST_118	SFS\$_FSTP_118

例 D-6 出力例: ノード 2(実際の表名は異なります。)

STORE NAME	TABLE_NAME	PTABLE_NAME
FS1	SFS\$_FST_101	SFS\$_FSTP_101
FS2	SFS\$_FST_119	SFS\$_FSTP_119

2. Extract パラメータ・ファイルで次の TABLE 文を作成することにより、Extract に対して ローカルで読取り / 書込みである表を識別します。(ご使用の所有者名および表名に置き換 えます。)

例 D-7 ノード1

```
TABLE owner.SFS$ FST 100
TABLE owner.SFS$ FSTP 100;
```

例 D-8 ノード2

TABLE owner.SFS\$ FST 119 TABLE owner.SFS\$ FSTP 119;

3. Replicat パラメータ・ファイルに次の MAP 文を作成することにより、各リモート・ファイ ル・システム上の変更を対応するローカル・ファイル・システムにリンク付けします。(ご 使用の所有者名および表名に置き換えます。)

例 D-9 ノード1

```
MAP owner.SFS$ FST 119, TARGET owner.SFS$ FST 118;
MAP owner.SFS$ FSTP 119, TARGET owner.SFS$ FSTP 118
```

例 D-10 ノード2

MAP owner.SFS\$ FST 100, TARGET owner.SFS\$ FST 101; MAP owner.SFS\$ FSTP 100, TARGET owner.SFS\$ FSTP 101;

このマッピングにより、ローカルの読取り/書込みソース表がキャプチャされ、リモート の読取り専用のピア表にレプリケートされます。

- ノード1の FS1 に行われたファイル・システムの変更は、ノード2の FS1 に伝播されま
- ノード2の FS2 に行われたファイル・システムの変更は、ノード1の FS2 に伝播されま す。

ファイル・システムへの変更は、データベースの DBFS コンテンツ API(パッケージ DBMS DBFS CONTENT) または dbfs client マウントおよび従来のファイル・システム・ツールを 使用して行うことができます。

すべての変更は両方のディレクトリに伝播されます。

- 各システム上の DBFS ネームスペースの仮想ルートで、ユーザーは同一の内容を確認 できます。
- 可変操作の場合は、各システム上の /local サブディレクトリを使用します。
- 読取り操作の場合は、ローカル・コンテンツとリモート・コンテンツのどちらを確認 するかに応じて、/local か /remote のいずれかのサブディレクトリを使用できます。

インストールされる Oracle GoldenGate コンポーネント

この付録では、Oracle GoldenGate ソフトウェアによって Oracle GoldenGate インストール・ディレクトリで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。

E.1 Oracle Goldengate プログラムとユーティリティ

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すべての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キャプチャまたは配信のみが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 E-1 インストールされる Oracle Goldengate プログラムとユーティリティ

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 DDL に従ってターゲット・データベース表の定 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	データ定義を生成します。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。
emsclnt	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	コマンドの発行またはパラメータ・ファイルの管理に使用する Oracle Goldengate のユーザー・インタフェース。

表 E-1 インストールされる Oracle Goldengate プログラムとユーティリティ(続き)

プログラム	説明
ggsmgr.jcl	z/OS システムでバッチ・ジョブまたはオペレータ・コンソールから
ggsmgr.proc	Oracle GoldenGate Manager プロセスを起動します。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。
ggsmgrst.jcl	y y Example of the second of t
ggsmgrst.proc	
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	データ暗号化鍵を生成します。
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	(Manager) 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 グに使用する制御プロセス。
replicat	データをターゲット・データベース表に適用します。
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。
triggen	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、トリガーベースの抽出方法をサポートします。
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。

E.2 Oracle Goldengate 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 E-2 インストールされる Oracle GoldenGate サブディレクトリ

ディレクトリ	説明
br	制限付きリカバリ機能用のチェックポイント・ファイルを格納します。
cfg	Oracle GoldenGate Monitor の構成に使用されるプロパティおよび XML ファイルを格納します。
dirdb	Oracle GoldenGate Monitor アプリケーションまたは Oracle Enterprise Manager 内で使用するために Oracle GoldenGate インスタンスから収集された情報の永続化に使用されるデータ・ストアを格納します。

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

ディレクトリ	説明
dirchk	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ールト・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 の内部フォーマットで書き込まれます。
	ファイル名の形式は、group_name+sequence_number.ext です (sequence_number は古いファイルに付加される順序番号で、ext は、Extract チェックポイント・ファイルの場合は cpe、Replicat チェックポイント・ファイルの場合は cpr です)。
	これらのファイルを編集しないでください。
	例:
	ext1.cpe
	rep1.cpr
dirdat	Extract プロセスによって作成される Oracle GoldenGate の証跡ファイルと抽出ファイルのデフォルトの場所。抽出されたデータを Replicat プロセスや他のアプリケーション、ユーティリティでさらに処理するために格納します。 Oracle GoldenGate の内部フォーマットで書き込まれます。
	ファイル名フォーマットは、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。
	これらのファイルを編集しないでください。
	例:
	rt000001
	finance
dirdef	異機種間の同期化環境で使用されるソース・データおよびターゲット・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。ファイル名の形式は、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。
	新たに作成された表の定義を追加するために、これらのファイルを編集できます。定義ファイルの編集方法がわからない場合には、Oracle GoldenGate テクニカル・サポートに問い合せてください。
	例:
	defs.dat
dirjar	Oracle GoldenGate Monitor をサポートする Java 実行可能ファイルを格納します。
dirout	このディレクトリは使用されません。
dirpcs	ステータス・ファイルのデフォルトの場所です。ファイル名の形式は、group.extensionです(groupはグループの名前で、extensionは、pce(Extract)、pcr(Replicat)またはpcm(Manager)のいずれかです)。
	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。
	これらのファイルを編集しないでください。
	例:
	mgr.pcm
	ext.pce

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

ディレクトリ	説明
dirprm	Oracle GoldenGate の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 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。外部 ASCII 形式で書き込まれます。ファイル名の形式は、group name/user-defined name.prmまたはmgr.prmです。
	プロセスの停止後、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。これらは、テキスト・エディタで直接編集するか、GGSCIで EDIT PARAMS を使用して編集できます。
	例:
	defgen.prm
	finance.prm
dirrec	Oracle GoldenGate では使用しません。
dirrpt	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。
	ファイル名の形式は、group name+sequence number.rpt です (sequence number は古いファイルに付加される順序番号です)。
	これらのファイルを編集しないでください。
	例:
	fin2.rpt
	mgr4.rpt
dirsql	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 ユーティリティにより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・スクリプトおよびユーザー が作成した SQL スクリプトの格納に使用されます。
dirtmp	キャッシュ・マネージャに割り当てられているメモリー・サイズを超えた場合にトランザクション・データを格納するデフォルトの場所。これらのファイルを編集しないでください。
dirwlt	Oracle GoldenGate Monitor をサポートする Oracle Wallet を格納します。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が実行されるまでインストールされません。
UserExitExamples	ユーザー・イグジットの作成に便利なサンプル・ファイルを格納しま す。

E.3 その他の Oracle GoldenGate のファイル

この項では、ルートの 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 E-3 その他の Oracle GoldenGate のファイル

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

表 E-3 その他の Oracle GoldenGate のファイル(続き)

コンポーネント	説明
cagent.dll	Oracle GoldenGate Monitor C サブエージェントの Windows ダイナミック・リンク・ライブラリを格納します。
category.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。
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。
db2cntl.tpl	IBM LOADUTIL 一括ロード・ユーティリティ用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 DDL を Microsoft Access 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 DDL を DB2 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 DDL を DB2 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	マーカー表からの DDL をファイルに保存するスクリプト。
ddl_disable.sql	Oracle GoldenGate DDL トリガーを無効にするスクリプト。(Oracle インストール)
ddl_enable.sql	Oracle GoldenGate DDL トリガーを有効にするスクリプト。(Oracle インストール)
ddl_filter.sql	Oracle GoldenGate での 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ます。手動で実行しないでください。
ddl_informix.tpl	ソース DDL を Informix 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 DDL を SQL Server 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 DDL を MySQL 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・スタッフによって使用される空のスク リプト・ファイル。
ddl_nssql.tpl	ソース DDL を NonStop SQL 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 DDL サポートの一部としてプログラムで実行される
ddl_ora10.sql	スクリプト。これらのスクリプトは実行しないでください。
ddl_oral1.sql	
ddl_ora10upCommon.sql	
ddl_oracle.tpl	ソース DDL を Oracle 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_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 トリガーをピン留めするスクリプト (Oracle のインストール)。
ddl_ purgeRecyclebin.sql	DDL レプリケーション機能をサポートするために Oracle のごみ箱を パージするスクリプト。

表 E-3 その他の Oracle GoldenGate のファイル(続き)

コンポーネント	説明
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	Oracle 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
ddl_session1.sql	リプトはプログラムで実行されます。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 インストール)
ddl_sqlmx.tpl	Tandem Enscribe DDL を NonStop SQL/MX 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 インストール)
ddl_staymetadata_ off.sql	Oracle DDL トリガーが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ます。手動で実行しな
ddl_staymetadata_ on.sql	いでください。
ddl_sybase.tpl	ソース DDL を Sybase 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 DDL を NonStop SQL 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 トレースがオンであるか、オフであるかを制御するスクリプト。
ddl_trace_on.sql	
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	チュートリアルおよび基本的なテストで使用するためのデモンストレーション表を作成し、移入するスクリプト。
.dmp files	トレースの目的で Oracle GoldenGate のプロセスで作成されるダンプ・ファイル。
ENCKEYS	暗号化鍵を保存する、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	UTF16でエンコードされたデータを、ユーザー・イグジットとプロセスの間で交換される情報に対するコールバック構造で使用する方法を示すユーザー・イグジット例。
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、情報メッセージおよび警告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起動時に確認されます。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る必要があります。
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。

表 E-3 その他の Oracle GoldenGate のファイル(続き)

コンポーネント	説明
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、ユーザーが作成するファイル。
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 の Windows 共有ライブラリ。
icuin38.dll	
icuuc38.dll	
jagent.bat	Oracle GoldenGate Monitor の Java エージェントの Windows バッチ・ファイル。
jagent.log	Oracle GoldenGate Monitor エージェントのログ・ファイル。
jagentjni.log	
jagent.sh	Oracle GoldenGate Monitor の Java エージェントの 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 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 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 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。
prvtclkm.plb	Oracle の暗号化された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 エージェントをサポートするスクリプト・
pw_agent_util.sh	ファイル。
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 インストール)
sampleodbc.ini	UNIX 上の Ingres 2.6 用サンプル 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。

表 E-3 その他の Oracle GoldenGate のファイル(続き)

コンポーネント	説明
start.prm	Manager プロセスを起動および停止する z/OS paramlib のメンバー。
stop.prm	
startmgr	GGSSCI から Manager プロセスを開始する z/OS Unix System Services
stopmgr	のスクリプト。
startmgrcom	Manager プロセス用の z/OS システム入力コマンド。
stopmgrcom	
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 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。

E.4 Oracle GoldenGate チェックポイント表

データベース・チェックポイントを使用する場合、ADD CHECKPOINTTABLE コマンドを実行する と、Oracle GoldenGate によりユーザー定義名を使用してデータベースにチェックポイント表 が作成されます。または、ユーザーが $chkpt_db_create.sql$ スクリプトを使用して表を作成す ることもできます (db はスクリプトでサポートされるデータベースのタイプの省略形です)。

この表の列の名前または属性を変更しないでください。必要に応じて表記憶域属性を変更しま

表 E-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)	同じ表に書き込む Replicat グループの数にかかわらず、GROUPNAME とともに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。