

Oracle® Database

2 日でデータ・レプリケーションおよび統合ガイド

11g リリース 1 (11.1)

部品番号 : E05777-03

2008 年 10 月

Oracle Streams、マテリアライズド・ビューおよびその他の分散データベース機能について説明します。

Oracle Database 2 日でデータ・レプリケーションおよび統合ガイド, 11g リリース 1 (11.1)

部品番号 : E05777-03

Oracle Database 2 Day + Data Replication and Integration Guide, 11g Release 1 (11.1)

原本部品番号 : B28324-03

原本著者 : Randy Urbano

原本協力者 : Maria Pratt, Janet Blowney, Steve Fogel, Vira Goorah, Thuvan Hoang, Stella Kister, Sushil Kumar, Patricia McElroy, Colin McGregor, Valarie Moore, Ashish Ray, Subbanarasimha Shastry, Sreejesh Srinivasan, Jim Stamos, Mark Townsend, Byron Wang, Lik Wong, Jingwei Wu, Jun Yuan, Ramkumar Venkatesan

Copyright © 2007, 2008, Oracle. All rights reserved.

制限付権利の説明

このプログラム（ソフトウェアおよびドキュメントを含む）には、オラクル社およびその関連会社に所有権のある情報が含まれています。このプログラムの使用または開示は、オラクル社およびその関連会社との契約に記載された制約条件に従うものとします。著作権、特許権およびその他の知的財産権と工業所有権に関する法律により保護されています。

独立して作成された他のソフトウェアとの互換性を得るために必要な場合、もしくは法律によって規定される場合を除き、このプログラムのリバース・エンジニアリング、逆アセンブル、逆コンパイル等は禁止されています。

このドキュメントの情報は、予告なしに変更される場合があります。誤りを見つけた場合は、オラクル社までご連絡ください。オラクル社およびその関連会社は、このドキュメントに誤りが無いことの保証は致し兼ねます。これらのプログラムのライセンス契約で許諾されている場合を除き、プログラムを形式、手段（電子的または機械的）、目的に関係なく、複製または転用することはできません。

このプログラムが米国政府機関、もしくは米国政府機関に代わってこのプログラムをライセンスまたは使用する者に提供される場合は、次の注意が適用されます。

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, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software--Restricted Rights (June 1987). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

このプログラムは、核、航空、大量輸送、医療あるいはその他の本質的に危険を伴うアプリケーションで使用されることを意図しておりません。このプログラムをかかるとの目的で使用する場合、上述のアプリケーション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。万一かかるプログラムの使用に起因して損害が発生いたしましても、オラクル社およびその関連会社は一切責任を負いかねます。

Oracle, JD Edwards, PeopleSoft, Siebel は米国 Oracle Corporation およびその子会社、関連会社の登録商標です。その他の名称は、他社の商標の可能性があり得ます。

このプログラムは、第三者の Web サイトへリンクし、第三者のコンテンツ、製品、サービスへアクセスすることがあります。オラクル社およびその関連会社は第三者の Web サイトで提供されるコンテンツについては、一切の責任を負いかねます。当該コンテンツの利用は、お客様の責任になります。第三者の製品またはサービスを購入する場合は、第三者と直接の取引となります。オラクル社およびその関連会社は、第三者の製品およびサービスの品質、契約の履行（製品またはサービスの提供、保証義務を含む）に関しては責任を負いかねます。また、第三者との取引により損失や損害が発生いたしましても、オラクル社およびその関連会社は一切の責任を負いかねます。

目次

はじめに	vii
対象読者	viii
ドキュメントのアクセシビリティについて	viii
関連ドキュメント	viii
表記規則	ix
サポートおよびサービス	ix
1 データ・レプリケーションと統合の概要	
このマニュアルについて	1-2
このマニュアルを使用する前に	1-2
このマニュアルの対象外	1-2
データ・レプリケーションと統合の概要	1-3
データ・レプリケーションと統合の機能の概要	1-4
複数のデータベース内の情報のアクセスおよび変更時期	1-4
Oracle Streams でのデータのレプリケート時期	1-5
マテリアライズド・ビューでのデータのレプリケート時期	1-6
データベース間のメッセージの送信時期	1-6
2 データ・レプリケーションおよび統合の一般的なタスク	
GLOBAL_NAMES 初期化パラメータを TRUE に設定	2-2
チュートリアル: Oracle Streams 管理者の作成	2-3
チュートリアル: Oracle Streams 管理者の表領域の作成	2-3
チュートリアル: Oracle Streams 管理者の作成	2-5
ANYDATA キューの作成	2-7
チュートリアル: データベース・リンクの作成	2-8
3 複数のデータベース内の情報のアクセスおよび変更	
複数のデータベース内の情報のアクセスおよび変更の概要	3-2
分散 SQL の概要	3-2
シノニムおよび位置の透過性の概要	3-2
Oracle 以外のデータベース内の情報のアクセスおよび変更の概要	3-3
ストアド・プロシージャの概要	3-3
複数の Oracle Database 内の情報のアクセスおよび変更の準備	3-4
チュートリアル: 複数の Oracle Database の問合せ	3-4
チュートリアル: 複数の Oracle Database 内のデータの変更	3-6
チュートリアル: Remote Oracle Database 内のストアド・プロシージャの実行	3-7

Oracle 以外のデータベースにあるデータの処理	3-8
Oracle 以外のデータベースで動作する Oracle Database の構成	3-9
Oracle 以外のデータベースの処理のベスト・プラクティス	3-9

4 Oracle Streams を使用したデータのレプリケート

Oracle Streams レプリケーションの概要	4-2
変更の取得の概要	4-3
取得プロセスでの変更の取得の概要	4-3
同期取得での変更の取得の概要	4-5
データベース間の変更の伝播の概要	4-6
変更の適用の概要	4-6
取得、伝播および適用の動作を制御するルールの概要	4-7
同一でないコピーのルールベースの変換の概要	4-8
サブメンタル・ロギングの概要	4-9
競合と競合解消の概要	4-9
変更の循環を回避するためのタグの概要	4-10
Oracle Streams レプリケーション環境の一般的なタイプの概要	4-11
2 データベース・レプリケーション環境の概要	4-11
ハブアンドスポーク・レプリケーション環境の概要	4-12
n-way レプリケーション環境の概要	4-14
Oracle Streams レプリケーションの構成プロシージャの概要	4-15
Oracle Streams が提供する主要な PL/SQL パッケージおよび	
データ・ディクショナリ・ビューの概要	4-19
Oracle Streams が提供する主要な PL/SQL パッケージの概要	4-19
Oracle Streams の主要なデータ・ディクショナリ・ビューの概要	4-20
Oracle Streams レプリケーションの準備	4-21
Oracle Streams レプリケーションの構成: 例	4-22
チュートリアル: ローカル取得プロセスを使用した 2 データベース・レプリケーションの構成	4-24
チュートリアル: ダウンストリーム取得プロセスを使用した	
2 データベース・レプリケーションの構成	4-29
チュートリアル: ローカル取得プロセスでのハブアンドスポーク・レプリケーションの構成	4-39
チュートリアル: 同期取得を使用した 2 データベース・レプリケーションの構成	4-45
チュートリアル: 表の最新時刻競合解消の構成	4-55

5 Oracle Streams レプリケーション環境の管理

Oracle Streams レプリケーション環境の管理	5-2
取得プロセスの管理	5-2
取得プロセスの起動と停止	5-2
取得プロセスのパラメータの設定	5-3
伝播の有効化および無効化	5-4
適用プロセスの管理	5-6
適用プロセスの起動と停止	5-6
適用プロセスのパラメータの設定	5-7
Oracle Streams レプリケーション環境の監視	5-9
データベースのレプリケーション・コンポーネントの概要の表示	5-9
データベースの Oracle Streams レプリケーション環境のトポロジの表示	5-10
取得プロセスの監視	5-12
取得プロセスの情報の表示	5-12

取得プロセスに関する統計情報の表示	5-14
伝播の監視	5-16
伝播の情報の表示	5-16
伝播に関する統計情報の表示	5-18
適用プロセスの監視	5-19
適用プロセスの情報の表示	5-20
適用プロセスに関する統計情報の表示	5-21
構成された更新競合ハンドラの表示	5-23
「バッファ・キューの統計」の表示	5-23
取得から適用までの時間の表示	5-25
Oracle Streams レプリケーション環境のトラブルシューティング	5-26
Enterprise Manager での自動化されたアラートへの対応	5-26
取得の中断アラート	5-27
伝播の中断アラート	5-27
適用の中断アラート	5-28
適用エラー・アラート	5-29
Oracle Streams プール・アラート	5-29
適用エラーの管理	5-30
データベース・オブジェクトでの適用エラーの修正	5-30
適用エラー・トランザクションの再試行または削除	5-31
レプリケーション先が使用できない場合のレプリケーション環境の管理	5-32

6 Oracle Streams レプリケーション環境の拡張

Oracle Streams レプリケーション環境の拡張の概要	6-2
チュートリアル: レプリケーション環境へのデータベース・オブジェクトの追加	6-3
チュートリアル: レプリケーション環境へのデータベースの追加	6-7

7 マテリアライズド・ビューを使用したデータのレプリケート

マテリアライズド・ビュー・レプリケーションの概要	7-2
マスター・サイト、マスター表およびマテリアライズド・ビュー・サイトの概要	7-3
マテリアライズド・ビューのリフレッシュの概要	7-3
リフレッシュ・グループの概要	7-3
マテリアライズド・ビュー・レプリケーションの準備	7-4
マテリアライズド・ビュー・サイトの構成	7-4
マスター・サイトでのマテリアライズド・ビュー・ログの構成	7-6
マテリアライズド・ビューを使用した読取り専用データのレプリケート	7-8
マテリアライズド・ビューを使用した読取り専用データのレプリケートの概要	7-8
チュートリアル: マテリアライズド・ビューを使用した 読取り専用データ・レプリケーションの構成	7-9
マテリアライズド・ビューを使用した読取り / 書込みデータのレプリケート	7-12
マテリアライズド・ビューを使用した読取り / 書込みデータのレプリケートの概要	7-13
レプリケーション・グループと更新可能マテリアライズド・ビューの概要	7-14
スケジュールされたリンクと遅延トランザクションの概要	7-15
競合と更新可能マテリアライズド・ビューの概要	7-16
マテリアライズド・ビューを使用した読取り / 書込みデータのレプリケーションの構成	7-16
リフレッシュ・グループの構成	7-23

8 マテリアライズド・ビュー・レプリケーション環境の管理

マテリアライズド・ビュー・レプリケーション環境の管理	8-2
マテリアライズド・ビューのリフレッシュ	8-2
リフレッシュ・グループのリフレッシュ	8-3
マテリアライズド・ビューのリフレッシュ	8-4
リフレッシュ・グループへのマテリアライズド・ビューの追加	8-5
マテリアライズド・ビューの削除	8-7
チュートリアル: マスター・サイトでのマテリアライズド・ビュー・サポートの クリーン・アップ	8-8
マテリアライズド・ビュー・レプリケーション環境の監視	8-10
データベースのレプリケーション・コンポーネントの概要の表示	8-10
マテリアライズド・ビューに関する情報の表示	8-12
現在リフレッシュされているマテリアライズド・ビューの判断	8-14
マテリアライズド・ビュー・グループに関する情報の表示	8-14
更新可能マテリアライズド・ビューの遅延トランザクションに関する情報の表示	8-16
リフレッシュ・グループに関する情報の表示	8-18
マスター・サイトのマテリアライズド・ビュー・ログの表示	8-20
マスター・サイトのマテリアライズド・ビューの表示	8-21
マテリアライズド・ビュー・レプリケーション環境のトラブルシューティング	8-22
マテリアライズド・ビュー・リフレッシュの問題の修正	8-22
マテリアライズド・ビュー・ログが大きくなりすぎるのを防止	8-23

9 Oracle Streams Advanced Queuing を使用したメッセージ送信

メッセージ機能の概要	9-2
メッセージの順序付けの概要	9-3
メッセージ・モードの概要	9-3
メッセージ通知の概要	9-4
伝播の概要	9-4
Oracle Messaging Gateway の概要	9-5
メッセージ機能の準備	9-5
チュートリアル: Oracle Database 間でのメッセージの送信	9-6
タスク 1: 各データベースでのメッセージ・タイプの作成	9-7
タスク 2: キューの構成とキュー間の伝播の構成	9-9
タスク 3: メッセージ・エンキュー・メカニズムの構成	9-10
タスク 4: メッセージをデキューするためのメッセージ・クライアントの構成	9-13
タスク 5: メッセージのエンキュー	9-15
タスク 6: メッセージのデキュー	9-15
チュートリアル: メッセージ通知の構成	9-16
タスク 1: メッセージ・タイプの作成	9-18
タスク 2: キューおよびメッセージ・クライアントの構成	9-19
タスク 3: メッセージをデキューするメカニズムの構成	9-20
タスク 4: メッセージ通知の構成	9-23
タスク 5: メッセージのエンキューとメッセージ通知のチェック	9-23
キューの変更	9-25
キュー表の変更	9-26
伝播の変更	9-28
メッセージング環境の監視	9-30

キュー内のメッセージの表示	9-31
「永続キューの統計」の表示	9-32
メッセージをデキューできるコンシューマの表示	9-33
メッセージング環境のトラブルシューティング	9-34
メッセージのエンキューまたはデキュー中の ORA-01031 エラーの修正	9-34
メッセージのエンキュー中の ORA-24033 エラーの修正	9-35
伝播の ORA-02019 エラーの修正	9-36
デキューしたメッセージがキュー内に残る理由の理解	9-36

10 データの比較および収束

異なるデータベース内のデータの比較および収束の概要	10-2
チュートリアル: データの比較および収束の準備	10-3
チュートリアル: 2つの異なるデータベース内のデータの比較	10-4
チュートリアル: 相違データの収束	10-7

索引

はじめに

ここでは、次の項目について説明します。

- 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
- サポートおよびサービス

対象読者

このマニュアルは、Oracle Database を使用してデータ・レプリケーションおよび統合タスクを実行するすべてのユーザーを対象としています。データ・レプリケーションおよび統合タスクでは、複数のデータベースの情報を統一された方法で使用します。

このマニュアルでは、ベスト・プラクティスを推奨し、データ・レプリケーションおよび統合タスクを実行するための効率的な方法を説明します。また、可能な場合はいつでもタスクを実行できる Oracle Enterprise Manager の使用方法についても説明します。

ドキュメントのアクセシビリティについて

オラクル社は、障害のあるお客様にもオラクル社の製品、サービスおよびサポート・ドキュメントを簡単にご利用いただけることを目標としています。オラクル社のドキュメントには、ユーザーが障害支援技術を使用して情報を利用できる機能が組み込まれています。HTML 形式のドキュメントで用意されており、障害のあるお客様が簡単にアクセスできるようにマークアップされています。標準規格は改善されつつあります。オラクル社はドキュメントをすべてのお客様がご利用できるように、市場をリードする他の技術ベンダーと積極的に連携して技術的な問題に対応しています。オラクル社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/accessibility/> を参照してください。

ドキュメント内のサンプル・コードのアクセシビリティについて

スクリーン・リーダーは、ドキュメント内のサンプル・コードを正確に読めない場合があります。コード表記規則では閉じ括弧だけを行に記述する必要があります。しかし JAWS は括弧だけの行を読まない場合があります。

外部 Web サイトのドキュメントのアクセシビリティについて

このドキュメントにはオラクル社およびその関連会社が所有または管理しない Web サイトへのリンクが含まれている場合があります。オラクル社およびその関連会社は、それらの Web サイトのアクセシビリティに関しての評価や言及は行っておりません。

Oracle サポート・サービスへの TTY アクセス

アメリカ国内では、Oracle サポート・サービスへ 24 時間年中無休でテキスト電話 (TTY) アクセスが提供されています。TTY サポートについては、(800)446-2398 にお電話ください。アメリカ国外からの場合は、+1-407-458-2479 にお電話ください。

関連ドキュメント

詳細は、次の Oracle ドキュメントを参照してください。

- 『Oracle Database 2 日でデータベース管理者』
- 『Oracle Database 管理者ガイド』
- 『Oracle Streams 概要および管理』
- 『Oracle Streams レプリケーション管理者ガイド』
- 『Oracle Streams アドバンスド・キューイング・ユーザーズ・ガイド』
- 『Oracle Database Heterogeneous Connectivity 管理者ガイド』
- 『Oracle Database アドバンスド・レプリケーション』
- 『Oracle Database アドバンスド・レプリケーション・マネージメント API リファレンス』
- 『Oracle Database PL/SQL パッケージ・プロシージャおよびタイプ・リファレンス』
- 『Oracle Database リファレンス』
- Enterprise Manager オンライン・ヘルプ
- 『Oracle Enterprise Manager 概要』

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。

規則	意味
太字	太字は、操作に関連する Graphical User Interface 要素、また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を示します。
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を示します。
固定幅フォント	固定幅フォントは、段落内のコマンド、 URL 、サンプル内のコード、画面に表示されるテキスト、または入力するテキストを示します。

サポートおよびサービス

次の各項に、各サービスに接続するための URL を記載します。

Oracle サポート・サービス

オラクル製品サポートの購入方法、および Oracle サポート・サービスへの連絡方法の詳細は、次の URL を参照してください。

<http://www.oracle.com/lang/jp/support/index.html>

製品マニュアル

製品のマニュアルは、次の URL にあります。

<http://www.oracle.com/technology/global/jp/documentation/index.html>

研修およびトレーニング

研修に関する情報とスケジュールは、次の URL で入手できます。

http://education.oracle.com/pls/web_prod-plq-dad/db_pages.getpage?page_id=3

その他の情報

オラクル製品やサービスに関するその他の情報については、次の URL から参照してください。

<http://www.oracle.com/lang/jp/index.html>

<http://www.oracle.com/technology/global/jp/index.html>

注意： ドキュメント内に記載されている URL や参照ドキュメントには、Oracle Corporation が提供する英語の情報も含まれています。日本語版の情報については、前述の URL を参照してください。

データ・レプリケーションと統合の概要

管理するデータベースが複数あるデータベース管理者は、必要なときに必要な場所で情報を使用可能にする責任を持ちます。

この章は次の項で構成されています。

- [このマニュアルについて](#)
- [データ・レプリケーションと統合の概要](#)
- [データ・レプリケーションと統合の機能の概要](#)

このマニュアルについて

このマニュアルでは、各種のデータ・レプリケーションおよび統合環境を構成および管理するために必要な一般的なタスクの実行方法について説明します。環境のタイプについては、1-3 ページの「[データ・レプリケーションと統合の概要](#)」および1-4 ページの「[データ・レプリケーションと統合の機能の概要](#)」を参照してください。

このマニュアルは、最適なデータ・レプリケーションおよび統合環境を判断するのに役立ちます。また、一般的なタイプのデータ・レプリケーションおよび統合環境を構成、メンテナンス、監視およびトラブルシューティングするための基本的なタスク指向の方法も示します。

このマニュアルで使用するプライマリ・インタフェースは、Oracle Enterprise Manager と SQL*Plus です。

この項の内容は次のとおりです。

- [このマニュアルを使用する前に](#)
- [このマニュアルの対象外](#)

このマニュアルを使用する前に

このマニュアルを使用する前に、次の必要があります。

- 『Oracle Database 2 日でデータベース管理者』を読んで理解していること
- Oracle Database ソフトウェアをインストールし、2 つ以上のデータベースを構成しておくこと

このマニュアルでは、複数のデータベースにおけるデータ・レプリケーションと統合について説明するので、このマニュアルで説明されるほとんどのタスクでは複数のデータベースが必要です。

このマニュアルの対象外

このマニュアルは、タスク指向のマニュアルであり、その目的は、一般的なデータ・レプリケーションと統合のタスクを説明することです。必要に応じて、現在のタスクを理解して実行するために必要な概念を説明します。

データ・レプリケーションおよび統合には、複数の Oracle Database 機能が関係します。これらの機能には、分散 SQL、Oracle Database Gateway、Oracle Streams およびマテリアライズド・ビューが含まれます。このマニュアルでは、これらの機能に関する詳細情報は提供しません。これらの機能の詳しい概念情報およびこれらの機能の詳細な使用方法は、適切な Oracle マニュアルを参照してください。

- 分散 SQL の詳細は、『Oracle Database 管理者ガイド』を参照してください。
- Oracle Database Gateway の詳細は、『Oracle Database Heterogeneous Connectivity 管理者ガイド』を参照してください。
- Oracle Streams の一般的な情報は、『Oracle Streams 概要および管理』を参照してください。
- Oracle Streams をレプリケーションに使用する方法は、『Oracle Streams レプリケーション管理者ガイド』を参照してください。
- マテリアライズド・ビューの詳細は、『Oracle Database アドバンスド・レプリケーション』および『Oracle Database アドバンスド・レプリケーション・マネージメント API リファレンス』を参照してください。
- Oracle Streams をメッセージ・キューイングに使用する方法は、『Oracle Streams アドバンスド・キューイング・ユーザズ・ガイド』を参照してください。

また、このマニュアルでは、Oracle Enterprise Manager で利用できるデータ・レプリケーションおよび統合機能の一部の使用方を説明しますが、Enterprise Manager の詳細な情報は提供しません。Enterprise Manager の詳細は、次の資料を参照してください。

- Enterprise Manager オンライン・ヘルプ
- 『Oracle Enterprise Manager 概要』

データ・レプリケーションと統合の概要

組織が大きくなる場合、複数のデータベースおよびアプリケーションで情報を共有できることが重要になります。データ・レプリケーションおよび統合によって、分散環境において必要なときに必要な場所で情報にアクセスできるようになります。Oracle Database には、データベース、アプリケーションおよびユーザー間での通信を可能にする安全な標準メカニズムが用意されています。これらのメカニズムには、キュー、データ・レプリケーション、メッセージング、および同機種環境と異機種環境両方での分散アクセスが含まれます。

このマニュアルでは、分散 SQL、レプリケーション、メッセージ・キューイングについて説明します。これらの機能を使用して次のタイプのタスクを実行することにより、コンピューティング・リソースを効率よく使用できます。

- データベース間でのデータのレプリケート
- 分散データベースにあるデータへの簡単なアクセス
- Oracle Database と Oracle 以外のデータベース間のデータ交換
- アプリケーション間の通信の有効化
- 顧客、パートナーおよびサプライヤとの情報の交換
- イベント通知およびワークフローの提供

Oracle Database には、特定の要件に対処するために次のタイプのデータ・レプリケーションおよび統合ソリューションが用意されています。

- **統合**：すべてのデータが単一のデータベースに移動され、中央にあるロケーションから管理されます。Oracle Real Application Clusters (Oracle RAC)、グリッド・コンピューティング、仮想プライベート・データベース (VPD) は、可用性が高くスケラブルで安全な単一のデータベースに情報を統合するのに役立ちます。
- **フェデレーション**：データは単一の仮想データベースに統合されているように見えますが、実際には現在の分散位置に残ります。分散問合せ、分散 SQL および Oracle Database Gateway は、フェデレーテッド・データベースの作成に役立ちます。
- **共有**：同じ情報の複数のコピーが複数のデータベースとアプリケーション・データ・ストアでメンテナンスされます。データ・レプリケーションとメッセージングは、複数のデータベースにある情報の共有に役立ちます。

参照：

- 『Oracle Database 2 日で Real Application Clusters ガイド』
- 『Oracle Database 2 日でセキュリティ・ガイド』

データ・レプリケーションと統合の機能の概要

組織は、必ずしもすべてのデータを単一のデータベースに統合できるわけではありません。データが地理的に複数のロケーションに分散され、場所によっては、プライマリ・サイトへの安全な接続が確保できない場合があります。また、データが統合されていても、最適なアプリケーションが相互に通信するためのメソッドが必要な場合があります。ただし、これは、組織が異なる場所やアプリケーション間で情報を共有する必要がある理由の一例にすぎません。

Oracle Database では、組織がデータ・レプリケーションおよび統合を実現するための手段が用意されています。ここでは、組織にとって最適なデータ・レプリケーションおよび統合オプションを決定する際に役立つ情報を示します。

次の項では、データ・レプリケーションおよび統合の様々な機能をいつ使用するかについて説明します。

- 複数のデータベース内の情報のアクセスおよび変更時期
- Oracle Streams でのデータのレプリケート時期
- マテリアライズド・ビューでのデータのレプリケート時期
- データベース間のメッセージの送信時期

前述の機能から要件に合う機能を選択し、個別にまたは組み合わせて使用できます。

注意： このマニュアルで説明されているデータ・レプリケーションと統合の機能に加えて、Oracle Warehouse Builder では、情報の統合に使用できる別のオプションがあります。Oracle Warehouse Builder は様々なデータ統合計画を設計し、デプロイできるフレキシブルなツールです。一般的に Warehouse Builder を使用して実装されるプロジェクトは、ミッション・クリティカルな操作システム、移行シナリオ、異なった操作システムの統合および従来のデータ・ウェアハウスなどです。Oracle Warehouse Builder は、データの統合のソリューションの実装を支援する一連の Graphical User Interface で構成されています。Oracle Warehouse Builder の詳細は、『Oracle Warehouse Builder ユーザーズ・ガイド』を参照してください。

複数のデータベース内の情報のアクセスおよび変更時期

最善の試みで情報を統合したにもかかわらず、多くの組織は複数で分散したデータベースを検出します。これらの組織がこのデータを集中管理する場合でも、少なくとも短期間では不可能な場合があります。このような組織に対して、単一で集中管理されたデータベースであるかのように分散したデータ・ソースにアクセスする方法が必要です。分散 SQL を使用して、単一の Oracle Database であるかのように、アプリケーションおよびユーザーは複数の Oracle または Oracle 以外のデータベースにアクセスし情報を変更できます。

情報を移動またはコピーする必要がないので、分散 SQL を使用して分散したデータ・ソースをフェデレートすることで、組織に最も早く簡単な情報統合のパスを提供します。情報を後で移動する場合はアプリケーションをリライトする必要はありません。統合アプローチに推移している組織には特に有用ですが、現在分散したデータにアクセスする方法が必要です。

たとえば、適切な Oracle Database Gateway での分散 SQL を使用した場合、アプリケーションは、Oracle Database にレガシー・データがインポートされるのを待たずに、レガシー・データに即時アクセスできます。分散 SQL は非定型問合せを実行する組織にも有用で、他の場所に適切に存在するまれにアクセスされるデータを更新します。

参照：

- 第3章「複数のデータベース内の情報のアクセスおよび変更」

Oracle Streams でのデータのレプリケート時期

接続性に問題がない場合、組織はデータをリアルタイムに近い方法でレプリケートする可能性があります。そうすることで、できるかぎり早い時点で、すべてのロケーションにあるデータが最新であることを保証します。Oracle Streams は、組織の特定の要件に応じて、様々な構成でリアルタイムに近いデータのレプリケーションをサポートします。Oracle Streams レプリケーション環境では、自動的にデータベースが相互に変化をプッシュします。

Oracle Streams レプリケーションの一般的な使用例を次に示します。

- 主要なオンライン・トランザクション処理 (OLTP) サイトから処理をオフロードするためのレポート・サイトの作成
- コール・センターまたは同様のアプリケーションへのロード・バランシング、スケーラビリティおよび可用性の提供
- 特定のビジネス要件を満たすためのロケーション間でのサイト自律性の提供
- 支社などの複数のロケーションにあるデータの変換および統合
- 異なるプラットフォーム間、Oracle Database リリース間、および Wide Area Network (WAN) でのデータのレプリケート

Oracle Streams レプリケーション構成には、n-way およびハブアンドスポークという 2 つの一般的なタイプがあります。特に、データのスケラビリティおよび可用性を必要とする組織ではマルチマスター (n-way) 構成が頻繁に使用されます。これらのアプリケーションでよく使用されるのが世界中のレプリカを使用する **follow the sun** モデルです。たとえば、米国、欧州およびアジアにコール・センターがあり、それぞれに顧客データの完全なコピーがあるとします。顧客からの電話は、時刻に応じて適切なコール・センターにルーティングされます。各コール・センターからは、データに対する高速かつローカルなアクセスを確保しています。サイトがなんらかの理由で使用不可能な場合、トランザクションは存続しているロケーションにルーティングされます。このタイプの構成は、複数のロケーション間でロード・バランシングの提供にも使用します。

もう 1 つの一般的な構成として、ハブアンドスポークがあります。たとえば、保険会社が本社と販売代理店で顧客データを共有する場合にこの構成を使用します。この構成のネットワーク・バージョンはエンド・スポークとハブの接続性が制限されている場合、特に有用です。販売代理店には本社に接続する支社へ直接の接続性があっても、本社への直接の接続性がないとします。このネットワーク・ルーティングのタイプは、すべてのロケーションに直接接続する場合に発生する一部の複雑さを排除します。ハブアンドスポーク構成はデータ・ウェアハウス環境にも有用です。データ・ウェアハウス環境では、詳細データが各ストアまたはスポークで維持され、より高レベルのデータはデータ・ウェアハウスまたはハブで共有されます。

n-way およびハブアンドスポークのどちらの場合でも、組織は、複数のロケーションでレプリケート・データを更新できるように Oracle Streams レプリケーションを構成できます。このレプリケーション環境では、データが競合する可能性があります。Oracle Streams には、自動的にこれらの競合を解消する競合解消メソッドが用意されています。

Oracle Streams はメッセージおよびレプリケーションなどの要件を共有するすべての情報のフレキシブルなインフラストラクチャを提供するので、変更が必要なときに組織は簡単に構成を変更できます。

参照:

- 第 4 章「Oracle Streams を使用したデータのレプリケート」

マテリアライズド・ビューでのデータのレプリケート時期

このシステムは1つの場所に格納されたデータにアクセスするすべてのユーザーに対して実用的であるわけではありません。たとえば、外交販売担当者が会社のデータベースに直接アクセスせずに、顧客サイトで価格リストにアクセスする必要がある場合があります。プライマリ・データベースまたはマスター・データベースに接続できなくても、発注処理をする必要がある場合があります。このようなユーザーには、データベースのレプリカまたはデータベースの一部が必要です。

Oracle Streams レプリケーションとは異なり、マテリアライズド・ビューは常にデータを連続的にレプリケートするわけではありません。マテリアライズド・ビューは、トランザクション上一貫性のある時点でリフレッシュされる表または表のサブセットのレプリカです。リフレッシュ中は、変更された行の最終値のみが取得され、マテリアライズド・ビューに適用されます。マスター表の更新回数に関係なく適用されます。これによって、リモート・サイトからマスター・サイトに接続する必要のある時間が削減されます。

マテリアライズド・ビューは、マスター・サイトへの接続が制限されている場所で特に有用です。接続不可能な場合でも、更新可能なマテリアライズド・ビューでは、これらの場所が独立して機能します。複数の場所で更新する場合、更新の競合を回避するため場所間で所有権は通常パーティション化されます。競合が発生した場合でも、競合を自動的に解消できる競合解消メソッドが用意されています。

切断した計算のサポートに加え、組織はマテリアライズド・ビューを使用して、データへのローカル・アクセスを提供し、プライマリ・ロケーションで処理の負荷を取り除き、パフォーマンスおよびスケーラビリティを改善できます。たとえば、1つ以上のマテリアライズド・ビューを使用して、受注システムからレポート・アクティビティの負荷を取り除きます。

参照：

- [第7章「マテリアライズド・ビューを使用したデータのレプリケート」](#)

データベース間のメッセージの送信時期

通常、組織は大きくなるにつれて、処理を自動化したタスクを管理する様々なアプリケーションを開発します。これらアプリケーションがデータを直接共有しなくても、完全に独立して動作するわけではありません。これらのアプリケーションにはタスクを調整し、情報を交換してお互いに通信する方法が必要です。

Oracle Streams Advanced Queuing (AQ) を使用すると、非同期方法でお互いに安全で確実な通信ができます。Oracle Streams AQ は、複数コンシューマのキュー、公開およびサブスクライブ、コンテンツ・ベースのルーティング、インターネット伝播および変換などの、メッセージ・キューイング・システムの標準機能のすべてをサポートします。たとえば、製品が発送されると、発送部門は簡単に請求部門に通知でき、顧客は適宜請求されます。

Oracle Streams AQ と適切なメッセージング・ゲートウェイを結合して、アプリケーションは、TIBCO Rendezvous、IBM Websphere MQ などの、その他のメッセージ・キューイング・システムとの同時運用もできます。この機能はビジネス・パートナーまたは顧客と情報の共有が必要な場合、特に有用です。

参照：

- [第9章「Oracle Streams Advanced Queuing を使用したメッセージ送信」](#)

データ・レプリケーションおよび統合の一般的なタスク

この章では、多くのデータ・レプリケーションおよび統合環境に必要な一般的なタスクの実行方法を説明します。

この章は次の項で構成されています。

- [GLOBAL_NAMES 初期化パラメータを TRUE に設定](#)
- [チュートリアル: Oracle Streams 管理者の作成](#)
- [ANYDATA キューの作成](#)
- [チュートリアル: データベース・リンクの作成](#)

GLOBAL_NAMES 初期化パラメータを TRUE に設定

複数の場所にあるデータにアクセスするには、最初に各場所を一意に識別できることを確認する必要があります。次に、これらの場所の間に通信パスを確立する必要があります。

各データベースの一意識別子を **グローバル・データベース名** と呼びます。初期化パラメータ GLOBAL_NAMES を TRUE に設定することで、分散データベース環境内の各データベースを一意に識別できることが保証されます。データベースは、DB_NAME 初期化パラメータで指定された個々のデータベース名の前に、データベース作成時に DB_DOMAIN 初期化パラメータで指定されたデータベース・ネットワーク・ドメインを付けることで、グローバル・データベース名を形成します。

GLOBAL_NAMES パラメータには、データベース・リンクが接続先データベースと同じ名前を持つ必要があるかどうかを指定します。分散処理を使用する場合は、各データベースでこのパラメータを TRUE に設定し、ネットワーク環境でのデータベースおよびリンクについて一貫したネーミング規則を確実に使用することをお勧めします。

データベースで GLOBAL_NAMES 初期化パラメータを TRUE に設定するには：

1. 初期化パラメータを変更できる管理ユーザーとして、Enterprise Manager にログインします。たとえば、SYSDBA 権限を持つユーザーとしてログインできます。
2. データベース・インスタンスの「データベース」ホームページに移動します。
3. 「サーバー」をクリックして「サーバー」サブページを開きます。
4. 「データベース構成」セクションで「初期化パラメータ」をクリックします。
5. サーバー・パラメータ・ファイルを使用している場合、「SPFile」をクリックしてください。使用していない場合は次の手順に進みます。
6. 「初期化パラメータ」ページで、検索ツールに GLOBAL_NAMES と入力します。

Database Instance: database > Logged in As SYS
Show SQL Revert Apply

Initialization Parameters

Current SPFile

The parameter values listed here are from the SPFILE /ade/rurbano_emru/oracle/dbs/spfileemru.ora

Name	Basic	Dynamic	Category
GLOBAL_NAMES	All	All	All

Filter on a name or partial name

Apply changes in SPFile mode to the current running instance(s). For static parameters, you must restart the database. Show All

Select	Name	Help	Revisions	Value	Comments	Type	Basic	Dynamic	Category
<input checked="" type="radio"/>	db_recovery_file_dest_size	?				Big Integer	✓	✓	Backup and Recovery
<input type="radio"/>	memory_target					Big Integer	✓	✓	Memory

7. 「実行」をクリックします。
8. GLOBAL_NAMES 初期化パラメータを TRUE に設定します。
9. 「適用」をクリックすると変更が保存されます。
パラメータをサーバー・パラメータ・ファイルで永続的に設定するか、初期化パラメータ・ファイルで永続的に設定するかを確認してください。
10. 分散環境の各データベースについて、手順 1～9 を実行します。デフォルトでは、GLOBAL_NAMES 初期化パラメータは FALSE に設定されます。したがって、各データベースで明示的に TRUE に設定する必要があります。

別の項からこの項に移動した場合は、ここでその項に戻ります。

- 「複数の Oracle Database 内の情報のアクセスおよび変更の準備」 (3-4 ページ)
- 「メッセージ機能の準備」 (9-5 ページ)
- 「マテリアライズド・ビュー・レプリケーションの準備」 (7-4 ページ)
- 「Oracle Streams レプリケーションの準備」 (4-21 ページ)

参照：

- 「チュートリアル : データベース・リンクの作成」 (2-8 ページ)
- 『Oracle Database 2 日でデータベース管理者』
- GLOBAL_NAMES 初期化パラメータの詳細は、『Oracle Database リファレンス』を参照してください。

チュートリアル : Oracle Streams 管理者の作成

Oracle Streams のいずれかのコンポーネントを使用することを計画している場合は、Oracle Streams 管理者を構成します。Oracle Streams コンポーネントには次のものがあります。

- キュー
- キュー表
- 取得プロセス
- 伝播
- 適用プロセス
- ルールとルール・セット

Oracle Streams 管理者は、これらのコンポーネントが使用される各データベースでこれらのコンポーネントを構成および管理します。これらのコンポーネントの詳細は、このマニュアルの次の各項を参照してください。

- 第 4 章「Oracle Streams を使用したデータのレプリケート」
- 第 9 章「Oracle Streams Advanced Queuing を使用したメッセージ送信」

Oracle Streams 管理者を構成するには、適切な権限を持つ新規ユーザーを作成するか、これらの権限を既存のユーザーに付与します。SYS または SYSTEM ユーザーを Oracle Streams 管理者として使用しないでください。また、Oracle Streams 管理者は、デフォルト表領域として SYSTEM 表領域を使用しないでください。

strmadmin という名前の新しい Oracle Streams 管理者を作成するには：

1. チュートリアル : Oracle Streams 管理者の表領域の作成
2. チュートリアル : Oracle Streams 管理者の作成

チュートリアル : Oracle Streams 管理者の表領域の作成

Oracle Streams 管理者は、他のユーザーによって使用されない専用表領域を使用する必要があります。キュー表およびその他の Oracle Streams コンポーネントにはディスク容量が必要であり、専用表領域によってこれらの要件が効率的に満たされます。

Oracle Streams 管理者の新規表領域を作成するには：

1. Enterprise Manager に管理ユーザーとしてログインします。
2. データベース・インスタンスの「データベース」ホームページに移動します。
3. 「サーバー」をクリックして「サーバー」サブページを開きます。

4. 「記憶域」セクションの「表領域」をクリックします。
5. 「表領域」 ページで、「作成」 をクリックします。
「表領域の作成」 ページが表示され、「一般」 サブページが表示されます。

6. 「名前」 フィールドに streams_tbs と入力します。
7. 「データファイル」 セクションで「追加」 をクリックして、「データファイルの追加」 ページを開きます。

8. 「ファイル名」 フィールドに streams_tbs.dbf と入力します。
9. 「ファイル・ディレクトリ」 フィールドのディレクトリをチェックし、必要に応じて変更します。

10. 「ファイル・サイズ」フィールドのサイズを 25 に変更し、リストが MB に設定されていることを確認します。
11. 「記憶域」セクションで「フルになった場合に自動的にデータファイルを拡張 (AUTOEXTEND)」を選択します。
12. 「増分」フィールドに 5 と入力し、リストを MB に設定します。
13. 「最大ファイル・サイズ」を設定します。通常は、「無制限」に設定したままにするのが最適です。
14. 「続行」をクリックします。
15. 「表領域の作成」ページで、「OK」をクリックします。
16. 2-3 ページの「チュートリアル : Oracle Streams 管理者の作成」の手順を実行して、Oracle Streams 管理者の作成を完了します。

注意： SQL 文 CREATE TABLESPACE を使用して、表領域を作成することもできます。

チュートリアル : Oracle Streams 管理者の作成

この項では、2-3 ページの「チュートリアル : Oracle Streams 管理者の表領域の作成」で構成した表領域を使用する Oracle Streams 管理者の作成について説明します。

strmadmin という名前の新しい Oracle Streams 管理者を作成するには：

1. Enterprise Manager に管理ユーザーとしてログインします。
2. データベース・インスタンスの「データベース」ホームページに移動します。
3. 「サーバー」をクリックして「サーバー」サブページを開きます。
4. 「セキュリティ」セクションで「ユーザー」をクリックします。
5. 「ユーザー」ページで、「作成」をクリックします。
「ユーザーの作成」ページの「一般」サブページが表示されます。

Database Instance: database > Users > Logged in As SYSTEM

Create User Show SQL Cancel OK

General Roles System Privileges Object Privileges Quotas Consumer Groups Switching Privileges Proxy Users

* Name

Profile

Authentication

* Enter Password

* Confirm Password

For Password choice, the role is authorized via password.

Expire Password now

Default Tablespace

Temporary Tablespace

Status Locked Unlocked

General Roles System Privileges Object Privileges Quotas Consumer Groups Switching Privileges Proxy Users

Show SQL Cancel OK

6. 「名前」フィールドに strmadmin と入力します。

7. 「パスワードの入力」フィールドと「パスワードの確認」フィールドに新しいユーザーのパスワードを入力します。
管理ユーザーの適切なパスワードを入力します。パスワードの選択の詳細は、『Oracle Database 2 日でセキュリティ・ガイド』を参照してください。
8. 「デフォルト表領域」フィールドの懐中電灯アイコンをクリックして、2-3 ページの「チュートリアル : Oracle Streams 管理者の表領域の作成」で作成された streams_tbs 表領域を選択します。
9. 「一時表領域」フィールドの懐中電灯アイコンをクリックして、新しいユーザーの一時表領域を選択します。
10. 「ロール」をクリックします。
「ユーザーの作成」ページの「ロール」サブページが表示されます。
11. 「リストを編集」をクリックします。
「ロールの変更」ページが表示されます。
12. DBA を「使用可能なロール」リストから「選択したロール」リストに移動します。
13. 「OK」をクリックします。
14. 「ユーザーの作成」ページで、「OK」をクリックしてユーザーを作成します。

注意： SQL 文 CREATE USER を使用してユーザーを作成したり、SQL 文 GRANT を使用してユーザーに権限を付与することもできます。

15. DBMS_STREAMS_AUTH.GRANT_ADMIN_PRIVILEGE プロシージャを使用して、ユーザーに追加の権限を付与します。
 - a. コマンドラインで SQL*Plus を開き、権限を付与できる管理ユーザーとしてデータベースに接続します。
SQL*Plus の起動の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。
 - b. 次のプロシージャを実行します。

```
BEGIN
  DBMS_STREAMS_AUTH.GRANT_ADMIN_PRIVILEGE(
    grantee          => 'stradmin',
    grant_privileges => TRUE);
END;
/
```

別の項からこの項に移動した場合は、ここでその項に戻ります。

- 「メッセージ機能の準備」(9-5 ページ)
- 「チュートリアル : ローカル取得プロセスを使用した 2 データベース・レプリケーションの構成」(4-24 ページ)
- 「チュートリアル : ダウンストリーム取得プロセスを使用した 2 データベース・レプリケーションの構成」(4-29 ページ)
- 「チュートリアル : ローカル取得プロセスでのハブアンドスポーク・レプリケーションの構成」(4-39 ページ)
- 「チュートリアル : 同期取得を使用した 2 データベース・レプリケーションの構成」(4-45 ページ)

ANYDATA キューの作成

キューには、Oracle Streams 環境のメッセージが格納されます。Oracle Streams レプリケーション環境では、キューにはデータベース変更に関する情報を含むメッセージが格納されます。Oracle Streams メッセージ機能環境では、キューにはアプリケーションおよびユーザーによって生成および使用されるメッセージが格納されます。通常、Oracle Streams 環境の各データベースには1つ以上のキューがあります。

ANYDATA キューにより、ほぼすべてのタイプのメッセージを簡単に格納できます。ANYDATA キューを使用する場合は、たとえば複数の異なるタイプのアプリケーション・メッセージを同じキューに格納できます。ANYDATA キューは、Oracle Streams レプリケーション環境のデータベース変更に関する情報の格納に使用する必要があります。

ANYDATA キューおよび関連キュー表を作成するには：

1. Oracle Enterprise Manager で、Oracle Streams 管理者としてハブ・データベースにログインします。
2. 「データベース」 ホームページに移動します。
3. 「データ移動」 をクリックして「データ移動」 サブページを開きます。
4. 「Streams」 セクションの「設定」 をクリックします。
5. Oracle Streams: 「設定オプション」 ページで、「メッセージ機能」 をクリックします。「Streams」 ページが表示され、「メッセージ機能」 サブページが表示されます。
6. 「作成」 をクリックして、「キューの作成: キュー・タイプ」 ページを開きます。

7. 「通常のキュー、SYS.ANYDATA データ型」 を選択します。

ANYDATA データ型のキューにより、単一キュー内のほぼ任意のタイプのメッセージを格納できます。

8. 「続行」 をクリックして、「キューの作成: 通常のキュー、SYS.ANYDATA データ型」 ページを開きます。

9. 「名前」 フィールドにキューの名前を入力します。Oracle Streams 環境で使用される典型的なキュー名は streams_queue ですが、必要に応じて別の名前を入力することもできます。

10. キュー表の所有者の名前とキュー表の名前を「**キュー表**」フィールドに入力します。通常、Oracle Streams 管理者は Oracle Streams キューを所有し、典型的なキュー表名は `streams_queue_table` です。したがって、`strmadmin.streams_queue_table` と入力するか、必要に応じて別の所有者と名前を入力できます。
11. Oracle Streams 管理者の名前が「**キュー・ユーザー**」フィールドに入力されていることを確認します。
12. オプションで、キューの説明を「**説明**」フィールドに入力します。
13. 「**終了**」をクリックしてキュー表とキューを作成します。

注意: `DBMS_STREAMS_ADM.SET_UP_QUEUE` プロシージャを使用して、ANYDATA キューを作成することもできます。

別の項からこの項に移動した場合は、ここでその項に戻ります。

- 「[タスク 2: キューの構成とキュー間の伝播の構成](#)」 (9-9 ページ)
- 「[タスク 2: キューおよびメッセージ・クライアントの構成](#)」 (9-19 ページ)
- 「[チュートリアル: 同期取得を使用した 2 データベース・レプリケーションの構成](#)」 (4-45 ページ)

参照:

- 「[キューの変更](#)」 (9-25 ページ)

チュートリアル: データベース・リンクの作成

分散データベース環境の 2 つの場所の間に通信パスを確立するには、データベース・リンクを作成する必要があります。データベース・リンクは、あるデータベースから別のデータベースへの 1 方向通信パスを定義するポインタです。Oracle Database は、データベース・リンクを使用して、あるデータベース上のユーザーがリモート・データベース内のオブジェクトにアクセスできるようにします。ローカル・ユーザーは、リモート・データベース上のユーザーでない場合でも、リモート・データベースへのデータベース・リンクを使用できます。

データベース・リンクは、複数のデータベースにデータを格納する環境またはデータベース間で情報を共有する環境でほとんどの場合に必要です。これらの環境には、分散 SQL、Oracle Streams レプリケーション、マテリアライズド・ビュー・レプリケーションおよびメッセージ機能を使用する環境があります。

分散環境の各データベースの `GLOBAL_NAMES` 初期化パラメータは `TRUE` に設定されているため、2 つのデータベース間にリンクを確立する場合はグローバル・データベース名を使用する必要があります。これにより、各データベース・リンクが適切なリモート・データベースに接続できます。

2 つのデータベース間にデータベース・リンクを作成する前に、ネットワーク接続を構成して、データベースが相互に通信できるようにする必要があります。データベース間のネットワーク接続の構成の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。

ii1.example.com データベースから ii2.example.com データベースへのデータベース・リンクを作成するには:

1. Oracle Streams 管理者 `strmadmin` や `SYSTEM` などの管理ユーザーとして、Enterprise Manager にログインします。データベース・リンクは、このユーザーのスキーマに作成されます。
2. `ii1.example.com` データベース・インスタンスの「データベース」ホームページに移動します。
3. 「スキーマ」をクリックして「スキーマ」サブページを開きます。

4. 「データベース・オブジェクト」セクションの「データベース・リンク」をクリックします。
5. 「データベース・リンク」ページで、「作成」をクリックして「データベース・リンクの作成」ページを開きます。

6. 「名前」フィールドにデータベース・リンクの名前を入力します。名前は、リンク先のデータベースのグローバル名である必要があります。この例では、データベース・リンク名は `ii2.example.com` です。
7. 「ネット・サービス名」フィールドに、リンク先のデータベースのネット・サービス名を入力します。この例では、ネット・サービス名は `ii2.example.com` です。
8. 「接続モード」セクションで「固定ユーザー」を選択します。
9. 「ユーザー名」フィールドで、データベース・リンクを所有するユーザーのユーザー名を入力します。データベース・リンクは、リモート・データベース上のこのユーザーに接続します。この例では、システムの `SYSTEM` などの管理ユーザー、Oracle Streams 管理者 `strmadmin`、または `oe` などの通常のデータベース・ユーザーを入力できます。
10. 「パスワード」および「パスワードの確認」フィールドで、リモート・データベース上の指定されたユーザーのパスワードを入力します。
11. 「OK」をクリックしてデータベース・リンクを作成します。

注意: SQL 文 `CREATE DATABASE LINK` を使用して、データベース・リンクを作成することもできます。

別の項からこの項に移動した場合は、ここでその項に戻ります。

- [「チュートリアル: 複数の Oracle Database の問合せ」](#) (3-4 ページ)
- [「チュートリアル: 複数の Oracle Database 内のデータの変更」](#) (3-6 ページ)
- [「チュートリアル: Remote Oracle Database 内のストアド・プロシージャの実行」](#) (3-7 ページ)
- [「Oracle 以外のデータベースで動作する Oracle Database の構成」](#) (3-9 ページ)
- [「タスク 2: キューの構成とキュー間の伝播の構成」](#) (9-9 ページ)
- [「チュートリアル: ローカル取得プロセスを使用した 2 データベース・レプリケーションの構成」](#) (4-24 ページ)

- 「チュートリアル: ダウンストリーム取得プロセスを使用した 2 データベース・レプリケーションの構成」 (4-29 ページ)
- 「チュートリアル: ローカル取得プロセスでのハブアンドスポーク・レプリケーションの構成」 (4-39 ページ)
- 「チュートリアル: 同期取得を使用した 2 データベース・レプリケーションの構成」 (4-45 ページ)
- 「チュートリアル: レプリケーション環境へのデータベースの追加」 (6-7 ページ)
- 「チュートリアル: データの比較および収束の準備」 (10-3 ページ)

参照:

- 「GLOBAL_NAMES 初期化パラメータを TRUE に設定」 (2-2 ページ)
- 「データ・レプリケーションと統合の概要」 (1-3 ページ)
- 「データ・レプリケーションと統合の機能の概要」 (1-4 ページ)

複数のデータベース内の情報のアクセスおよび変更

この章では、複数の Oracle および Oracle 以外のデータベース内の情報のアクセス方法および変更方法を説明します。

この章は次の項で構成されています。

- [複数のデータベース内の情報のアクセスおよび変更の概要](#)
- [複数の Oracle Database 内の情報のアクセスおよび変更の準備](#)
- [チュートリアル: 複数の Oracle Database の問合せ](#)
- [チュートリアル: 複数の Oracle Database 内のデータの変更](#)
- [チュートリアル: Remote Oracle Database 内のストアド・プロシージャの実行](#)
- [Oracle 以外のデータベースにあるデータの処理](#)

複数のデータベース内の情報のアクセスおよび変更の概要

Oracle Database に接続している間は、他の Oracle Database 内および Oracle 以外のデータベース内の情報にアクセスして変更できます。2 つ以上のデータベース内の情報が 1 つのデータベースに出現する場合は、これを **フェデレーション** と呼びます。フェデレーションは、情報を元の場所に残し、そこでメンテナンスと更新が行われます。複数のデータ・ソースが 1 つの仮想データベースに統合されているように見えるため、異なる種類のデータベースが 1 つの統合ビューに表示されます。フェデレーテッド構成により、すべてのデータベースがアプリケーションとエンド・ユーザーから 1 つの仮想データベースのように見えるため、分散システムの複雑さが一部緩和されます。

次の各項で、複数のデータベース内の情報のアクセスおよび変更に関する詳細情報を示します。

- [分散 SQL の概要](#)
- [シノニムおよび位置の透過性の概要](#)
- [Oracle 以外のデータベース内の情報のアクセスおよび変更の概要](#)
- [ストアド・プロシージャの概要](#)

参照：

- [「複数のデータベース内の情報のアクセスおよび変更時期」](#)
(1-4 ページ)

分散 SQL の概要

分散 SQL により、アプリケーションおよびユーザーは、複数のデータベース内の情報を単一の SQL 文で問合せまたは変更できます。分散 SQL はデータの物理的な位置をマスクするため、アプリケーションを変更しなくてもデータの位置を変更できます。分散 SQL には、**分散問合せ** (データにアクセスします) と **分散トランザクション** (データを変更します) があります。分散トランザクションでは、**2 フェーズ・コミット・メカニズム**により、分散トランザクションに関係する各データベースでトランザクション内のすべての文を 1 単位としてコミットまたはロール・バックすることでデータの整合性が保証されます。

アプリケーションまたはユーザーが分散トランザクションのコミットを試行する場合、アプリケーションまたはユーザーが接続されているデータベースは **グローバル・コーディネータ** と呼ばれます。グローバル・コーディネータは、次のフェーズを開始することによって 2 フェーズ・コミットを完了します。

- **準備フェーズ:** グローバル・コーディネータは、分散トランザクション内の関係する他のデータベースに対して、障害が発生した場合でもトランザクションをコミットまたはロール・バックできることの確認を求めます。いずれかのデータベースが準備フェーズを完了できない場合は、トランザクションがロール・バックされます。
- **コミット・フェーズ:** 他のすべてのデータベースが、準備されていることをグローバル・コーディネータに通知する場合、グローバル・コーディネータはトランザクションをコミットし、他のすべてのデータベースにトランザクションのコミットを要求します。

参照：

- 『Oracle Database 管理者ガイド』

シノニムおよび位置の透過性の概要

シノニムは、別のデータベース・オブジェクトの別名として機能するデータベース・オブジェクトです。パブリック・シノニムとプライベート・シノニムの両方を作成できます。すべてのデータベース・ユーザーがパブリック・シノニムにアクセスできます。プライベート・シノニムは、特定のユーザーのスキーマにあり、プライベート・シノニムへのアクセス権を付与されたユーザーのみがこれを使用できます。

分散環境では、シノニムはデータベース・オブジェクトの位置の透過性を提供できます。シノニムは、データベース・オブジェクトの位置をアプリケーションおよびユーザーから隠します。データベース・オブジェクトを移動または名前変更する必要がある場合は、シノニムを再定義でき、アプリケーションおよびユーザーは変更を行わずにシノニムを使用し続けることができます。

たとえば、リモート・データベースにある `hr.employees` 表に `ii2.example.com` というグローバル名でアプリケーションがアクセスする必要があるとします。リモート・データベースへのデータベース・リンクが存在し、データベース・リンクの名前は `ii2.example.com` です。この場合は、`hr` スキーマに表 `hr.employees@ii2.example.com` を指す `employees` という名前のシノニムを作成できます。シノニムがある場合、ローカル・データベースのアプリケーションは `hr.employees` を使用してリモート表にアクセスできます。

Oracle 以外のデータベース内の情報のアクセスおよび変更の概要

分散 SQL を使用して、Oracle Database だけでなく Oracle 以外のデータベース内のデータをフェデレートできます。Oracle Database Gateway により、Oracle Database は、Sybase、DB2、Informix、Microsoft SQL Server、INGRES、Teradata データベースなどの多くの Oracle 以外のデータベース内のデータにアクセスして変更できます。このアクセスは、エンド・ユーザーに対して完全に透過的です。つまり、Oracle Database と Oracle 以外のデータベースのどちらのデータにアクセスするかにかかわらず、同じ SQL 文を発行できます。

注意： 特定の Oracle Database Gateway の機能は、アクセスされる Oracle 以外のデータベースの機能によって制限されます。特定の Oracle Database Gateway の制限の詳細は、特定のゲートウェイの Oracle ドキュメントを参照してください。

参照：

- 「Oracle 以外のデータベースにあるデータの処理」(3-8 ページ)

ストアド・プロシージャの概要

フェデレーテッド環境で複雑な操作を実行する際のネットワーク・トラフィックを削減するために、ストアド・プロシージャを使用できます。プロシージャまたはファンクションは、特定の問題を解決するため、または関連する一連のタスクを実行するために実行されるスキーマ・オブジェクトです。ファンクションがコール元に必ず単一の値を返すのに対し、プロシージャは値を返さないという点を除き、プロシージャとファンクションは同一です。一般に、アクションの実行にはプロシージャを使用し、値の計算にはファンクションを使用します。このマニュアルでは、ストアド・プロシージャという一般用語にはプロシージャとファンクションの両方が含まれます。

Oracle Database では、PL/SQL または Java に書き込まれるストアド・プロシージャがサポートされますが、このマニュアルでは、PL/SQL ストアド・プロシージャについてのみ説明します。PL/SQL ストアド・プロシージャは、SQL 文のセットと、グループ化されてデータベースに格納されている他の PL/SQL 構成メンバーから構成されます。ストアド・プロシージャでは、SQL の容易性および柔軟性を構造化プログラミング言語のプロシージャ型機能と組み合わせることができます。

SQL 文と同様に、ストアド・プロシージャを実行するには、物理的な位置を認識する必要はありません。同様に、適切な Oracle Database Gateway を使用して、Oracle 以外のデータベースにあるストアド・プロシージャもコールできます。この場合、ゲートウェイは PL/SQL コールを Oracle 以外のデータベースのストアド・プロシージャにマッピングします。

参照：

- 「チュートリアル：[Remote Oracle Database 内のストアド・プロシージャの実行](#)」(3-7 ページ)
- PL/SQL ストアド・プロシージャの詳細は、『Oracle Database PL/SQL 言語リファレンス』を参照してください。
- Java ストアド・プロシージャの詳細は、『Oracle Database Java 開発者ガイド』を参照してください。

複数の Oracle Database 内の情報のアクセスおよび変更の準備

この項では、他のデータベースにある情報にアクセスしそれを変更するためのデータベースの準備に必要なアクションについて説明します。

複数のデータベース内の情報のアクセスおよび変更を準備するには：

1. 分散環境の各 Oracle Database で GLOBAL_NAMES 初期化パラメータを TRUE に設定します。方法については、2-2 ページの「[GLOBAL_NAMES 初期化パラメータを TRUE に設定](#)」を参照してください。
2. ネットワーク接続を構成して、データベースが相互に通信できるようにします。データベース間のネットワーク接続の構成の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。

チュートリアル：複数の Oracle Database の問合せ

分散問合せは、2 つ以上のデータベース内の情報にアクセスします。シノニムまたは SELECT 文で、名前の最後に `@dblink` を付けることにより、リモート表、ビュー、またはマテリアライズド・ビューを識別できます。`dblink` は、リモート・データベース・オブジェクトを含むデータベースへのデータベース・リンクです。

この項の分散問合せを実行する前に、次の条件を満たします。

- 3-4 ページの「[複数の Oracle Database 内の情報のアクセスおよび変更の準備](#)」で説明する前提条件を満たします。
- ローカル・データベースから、問合せに含まれるデータベース・オブジェクトを含む任意のリモート・データベースへのデータベース・リンクを作成します。この項の例では、`ii1.example.com` データベースの SYSTEM ユーザーが、`ii2.example.com` データベースの SYSTEM ユーザーに接続するデータベース・リンクを使用します。このようなデータベース・リンクの作成の詳細は、2-8 ページの「[チュートリアル：データベース・リンクの作成](#)」を参照してください。
- `hr` サンプル・スキーマがローカル・データベースにインストールされ、`oe` サンプル・スキーマがリモート・データベースにインストールされていることを確認します。これらのサンプル・スキーマは、Oracle Database とともにデフォルトでインストールされます。

この項では、`ii1.example.com` と `ii2.example.com` をサンプル・データベースとして使用します。使用している環境で、これらの条件を満たす任意の 2 つのデータベースをかわりに使用できます。

この例では、次のことを想定します。

- 会社が人事情報を `ii1.example.com` データベースの `hr` スキーマに保持し、注文入力情報を `ii2.example.com` データベースに保持しています。
- `hr.employees` 表の `employee_id` は `oe.orders` 表の `sales_rep_id` に対応します。
- 管理者は、注文に関する質問があり、注文について営業担当者にお問い合わせする必要があります。

この場合、営業担当者の連絡先情報は、`ii1.example.com` データベースの `hr.employees` 表にあり、注文情報は `ii2.example.com` データベースの `oe.orders` 表にあります。

ii1.example.com データベースと ii2.example.com データベースの情報を組み合わせて営業担当者の連絡先情報を表示する分散問合せを実行するには:

1. リモート・データベース・オブジェクトのシノニムを作成します。この例では、ii2.example.com データベースにある oe.orders 表を指す ord というシノニムを hr スキーマ内に作成します。
 - a. SYSTEM ユーザーとして Enterprise Manager にログインします。
 - b. ii1.example.com データベース・インスタンスの「データベース」ホームページに移動します。
 - c. 「スキーマ」をクリックして「スキーマ」サブページを開きます。
 - d. 「データベース・オブジェクト」セクションの「シノニム」をクリックします。
 - e. 「シノニム」ページで、「作成」をクリックします。
 - f. 「シノニムの作成」ページで、「一般」セクションの「名前」フィールドに ord と入力します。
 - g. 「タイプ」に対して「スキーマ」を選択した状態で、「スキーマ」フィールドに hr と入力し、懐中電灯アイコンをクリックして hr スキーマを選択します。
 - h. 「データベース」セクションで、「リモート」を選択します。
 - i. 「サービス名」フィールドの懐中電灯アイコンをクリックして、SYSTEM ユーザーの ii2.example.com データベース・リンクを選択します。
 - j. 「元となるオブジェクト」セクションで、oe.orders と入力します。
 - k. 「OK」をクリックしてシノニムを作成します。

注意: SQL 文 CREATE SYNONYM を使用して、シノニムを作成することもできます。

2. SQL*Plus のコマンドラインを開き、SYSTEM ユーザーとして ii1.example.com データベースに接続します。

SQL*Plus の起動の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。

3. 次の問合せを実行します。

```
COLUMN FIRST_NAME HEADING 'First Name' FORMAT A20
COLUMN LAST_NAME HEADING 'Last Name' FORMAT A20
COLUMN PHONE_NUMBER HEADING 'Phone Number' FORMAT A20

SELECT e.first_name, e.last_name, e.phone_number
FROM hr.employees e, hr.ord o
WHERE o.order_id = 2456 AND
       e.employee_id = o.sales_rep_id;
```

出力は、次のようになります。

First Name	Last Name	Phone Number
Danielle	Greene	011.44.1346.229268

参照:

- 「複数のデータベース内の情報のアクセスおよび変更時期」
(1-4 ページ)

チュートリアル: 複数の Oracle Database 内のデータの変更

分散トランザクションには、2つ以上のデータベース内のデータまたはデータベース・オブジェクトの構造を個別に、またはグループとして変更する1つ以上の文が含まれます。シノニム、またはリモート・データベース内のデータを変更する文では、名前の最後に `@dblink` を追加することでデータベース・オブジェクトを識別できます。`dblink` は、リモート・データベース・オブジェクトを含むデータベースへのデータベース・リンクです。

表のデータを変更する文は、データ操作言語 (DML) と呼ばれます。データベース・オブジェクトの構造を変更する文は、データ定義言語 (DDL) 文と呼ばれます。DML 文と DDL 文は、どちらも分散トランザクションの一部にすることができます。

この項の分散トランザクションを実行する前に、次の条件を満たします。

- 3-4 ページの「[複数の Oracle Database 内の情報のアクセスおよび変更の準備](#)」で説明する前提条件を満たします。
- ローカル・データベースから、トランザクションに含まれるデータベース・オブジェクトを含む任意のリモート・データベースへのデータベース・リンクを作成します。この項の例では、`ii1.example.com` データベースの `SYSTEM` ユーザーが、`ii2.example.com` データベースの `SYSTEM` ユーザーに接続するデータベース・リンクを使用します。このようなデータベース・リンクの作成の詳細は、2-8 ページの「[チュートリアル: データベース・リンクの作成](#)」を参照してください。
- `hr` サンプル・スキーマがローカル・データベースにインストールされ、`oe` サンプル・スキーマがリモート・データベースにインストールされていることを確認します。これらのサンプル・スキーマは、Oracle Database とともにデフォルトでインストールされます。

この項では、`ii1.example.com` と `ii2.example.com` をサンプル・データベースとして使用します。使用している環境で、これらの条件を満たす任意の2つのデータベースをかわりに使用できます。

この例では、次のことを想定します。

- 会社が人事情報を `ii1.example.com` データベースの `hr` スキーマに保持し、注文入力情報を `ii2.example.com` データベースに保持しています。
- `hr.employees` 表の `employee_id` は `oe.orders` 表の `sales_rep_id` に対応します。
- 営業担当者は、営業マネージャにジョブをプロモートしました。`hr.employees` 表内のこの営業担当者の `employee_id` は 154 です。この従業員の `job_id` および `manager_id` データを `hr.employees` 表で変更する必要があります。
- プロモートされた営業担当者の現在の注文は、別の営業担当者に転送されます。`hr.employees` 内で、注文を担当する営業担当者の `employee_id` は 148 です。`sales_rep_id 154` に現在割り当てられている `oe.orders` 表のすべての注文は、`sales_rep_id 148` に変更する必要があります。

会社は、これらの変更を単一の分散トランザクションでコミットすることを希望します。

ii1.example.com データベースと ii2.example.com データベースの両方でデータを変更する分散トランザクションを実行するには:

1. リモート・データベース・オブジェクトのシノニムを作成します。この例では、`ii2.example.com` データベースにある `oe.orders` 表を指す `ord` というシノニムを `hr` スキーマ内に作成します。このシノニムの作成方法は、3-4 ページの「[チュートリアル: 複数の Oracle Database の問合せ](#)」の手順 1 を参照してください。
2. SQL*Plus のコマンドラインを開き、`SYSTEM` ユーザーとして `ii1.example.com` データベースに接続します。

SQL*Plus の起動の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。

3. 各データベースのデータを更新し、変更をコミットします。

```
UPDATE hr.employees SET
  job_id = 'SA_MAN', manager_id = 100
  WHERE employee_id = 154;

UPDATE hr.ord
  SET sales_rep_id = 148
  WHERE sales_rep_id = 154;

COMMIT;
```

参照 :

- 「複数のデータベース内の情報のアクセスおよび変更時期」
(1-4 ページ)

チュートリアル : Remote Oracle Database 内のストアド・プロシージャの実行

リモート・プロシージャ・コール (RPC) は、リモート・データベースのプロシージャまたはファンクションを実行します。RPC は、リモート・プロシージャに定義されている作業を実行します。リモート・プロシージャまたはファンクションを実行するには、名前の最後に `@dblink` を追加することでリモート・プロシージャまたはファンクションを識別できます。`dblink` は、リモート・プロシージャまたはファンクションを含むデータベースへのデータベース・リンクです。リモート・プロシージャまたはファンクションを指すシノニムも作成できます。

この項のサンプル RPC を実行する前に、次の条件を満たします。

- 3-4 ページの「[複数の Oracle Database 内の情報のアクセスおよび変更の準備](#)」で説明する前提条件を満たします。
- ローカル・データベースから、コールされるプロシージャまたはファンクションを含む任意のリモート・データベースへのデータベース・リンクを作成します。この項の例では、`ii1.example.com` データベースの `SYSTEM` ユーザーが、`ii2.example.com` データベースの `SYSTEM` ユーザーに接続するデータベース・リンクを使用します。このようなデータベース・リンクの作成の詳細は、2-8 ページの「[チュートリアル: データベース・リンクの作成](#)」を参照してください。
- `hr` サンプル・スキーマがリモート・データベースにインストールされていることを確認します。`hr` サンプル・スキーマは、Oracle Database とともにデフォルトでインストールされます。

この項では、`ii1.example.com` と `ii2.example.com` をサンプル・データベースとして使用します。使用している環境で、これらの条件を満たす任意の2つのデータベースをかわりに使用できます。

この例では、次のことを想定します。

- 会社が人事情報を `ii2.example.com` データベースの `hr` スキーマに保持しています。
- `hr` スキーマはローカル `ii1.example.com` データベースに存在しません。
- 従業員が会社を退職するときに、`hr` スキーマの `add_job_history` というプロシージャは `hr.job_history` 表に行を挿入します。この行には、従業員の会社での履歴に関する情報が含まれます。
- `employee_id` が 127 の従業員が会社を退職します。
- `ii1.example.com` データベースに接続し、`ii2.example.com` データベースの `add_job_history` プロシージャを実行して、従業員 127 のジョブ履歴を記録します。

RPC を実行して従業員のジョブ履歴を記録するには：

1. SQL*Plus のコマンドラインを開き、SYSTEM ユーザーとして ii1.example.com データベースに接続します。

SQL*Plus の起動の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。

2. ii2.example.com データベースの add_job_history プロシージャを実行して、従業員のジョブ履歴を記録します。

```
exec hr.add_job_history@ii2.example.com(127,'14-JAN-99','26-JUN-06','ST_CLERK',50);
```

3. 変更をコミットします。

```
COMMIT;
```

この手順は、リモート・プロシージャがコミットする場合は不要です。

4. オプションとして、ii2.example.com データベースの hr.job_history 表を問い合わせ、挿入された行を確認します。

```
SELECT * FROM hr.job_history@ii2.example.com ORDER BY employee_id;
```

出力は、次のようになります。

EMPLOYEE_ID	START_DAT	END_DATE	JOB_ID	DEPARTMENT_ID
101	21-SEP-89	27-OCT-93	AC_ACCOUNT	110
101	28-OCT-93	15-MAR-97	AC_MGR	110
102	13-JAN-93	24-JUL-98	IT_PROG	60
114	24-MAR-98	31-DEC-99	ST_CLERK	50
122	01-JAN-99	31-DEC-99	ST_CLERK	50
127	14-JAN-99	26-JUN-06	ST_CLERK	50
176	24-MAR-98	31-DEC-98	SA_REP	80
176	01-JAN-99	31-DEC-99	SA_MAN	80
200	17-SEP-87	17-JUN-93	AD_ASST	90
200	01-JUL-94	31-DEC-98	AC_ACCOUNT	90
201	17-FEB-96	19-DEC-99	MK_REP	20

employee_id が 127 の従業員のジョブ履歴が表に記録されていることに注意してください。

参照：

- 「複数のデータベース内の情報のアクセスおよび変更時期」
(1-4 ページ)

Oracle 以外のデータベースにあるデータの処理

Oracle Database Gateway では、Oracle Database から、Sybase、DB2、Informix、Microsoft SQL Server、Ingres および Teradata データベースなどの多くの Oracle 以外のデータベースにアクセスおよび変更できます。最良のパフォーマンスおよび有用性のために、Oracle 以外のデータベースを処理する際には、確立されたベスト・プラクティスに従う必要があります。

この項の内容は次のとおりです。

- Oracle 以外のデータベースで動作する Oracle Database の構成
- Oracle 以外のデータベースの処理のベスト・プラクティス

参照：

- 「複数のデータベース内の情報のアクセスおよび変更時期」
(1-4 ページ)
- 「Oracle 以外のデータベース内の情報のアクセスおよび変更の概要」
(3-3 ページ)

Oracle 以外のデータベースで動作する Oracle Database の構成

データの間合せ、データの変更、または Oracle 以外のデータベース内のストアド・プロシージャを実行する前に、いくつかのタスクを実行する必要があります。

Oracle 以外のデータベース内のデータの操作を開始するには：

1. Oracle 以外のデータベースごとに、Oracle Database Gateway ソフトウェアをインストールし、構成します。Oracle Database Gateway ソフトウェアは、Oracle Database を実行しているコンピュータ・システム、Oracle 以外のデータベースを実行しているコンピュータ・システムまたは第 3 のコンピュータ・システムにインストールできます。特定の Oracle Database Gateway のインストールおよび構成の詳細は、そのゲートウェイの Oracle ドキュメントを参照してください。
2. Oracle Database が、手順 1 で構成された Oracle Database Gateway のリスナーと通信できるように、Oracle Net Services を構成します。方法については、『Oracle Database Net Services 管理者ガイド』を参照してください。

Oracle Database Gateway との通信が確立された後で、Oracle Database は Oracle 以外のデータベースと通信できます。Oracle Database は、異機種間サービスを使用して Oracle Database Gateway と通信します。異機種間サービスの構成オプションの詳細は、『Oracle Database Heterogeneous Connectivity 管理者ガイド』を参照してください。

3. Oracle 以外のデータベースへのデータベース・リンクを作成します。このデータベース・リンクを作成するときは、明示的に CONNECT TO 句を使用する必要があります。手順 1 および 2 を完了すると、Oracle Database に接続するときと同様にデータベース・リンクを作成できます。方法については、2-8 ページの「チュートリアル: データベース・リンクの作成」を参照してください。

これらの手順を完了した後で、Oracle 以外のデータベース内のデータおよびプロシージャに透過的にアクセスできます。Oracle 以外のデータベースを操作するには、次の項の手順に従って、Oracle 以外のデータベース内のデータベース・オブジェクトを指定します。

- 「チュートリアル: 複数の Oracle Database の間合せ」 (3-4 ページ)
- 「チュートリアル: 複数の Oracle Database 内のデータの変更」 (3-6 ページ)
- 「チュートリアル: Remote Oracle Database 内のストアド・プロシージャの実行」 (3-7 ページ)

Oracle 以外のデータベースの処理のベスト・プラクティス

Oracle Database Gateway のパフォーマンスは、ネットワークのスピード、使用可能なメモリー、データベースからデータベースへ転送されるデータ量および同時セッションの数などの、いくつかの要素から影響を受けます。これらの要素には、より高いパフォーマンスを得るために調整できるものもあります。

次のベスト・プラクティスで、より高いパフォーマンスを実現できます。

- 事後処理の削減
- Oracle 以外のデータベースのチューニング
- 関連する初期化パラメータの設定
- Oracle Database Gateway インストールの場所の確認
- 十分なメモリーの確認
- 大 / 小文字の差異の考慮

事後処理の削減

大量の事後処理がある場合、パフォーマンスに悪影響を与えます。より高いパフォーマンスを実現するには、可能なかぎり SQL 文を Oracle 以外のデータベースで処理するようにします。WHERE 句の一部が Oracle 以外のデータベースへ送信された SQL から欠落したり、結合が分割されると、問合せは事後処理になります。

次のベスト・プラクティスに従って事後処理を削減します。

- 可能であれば、WHERE 句で SQL ファンクションを使用しないでください。WHERE 句の SQL ファンクションはゲートウェイのパフォーマンスに影響を与えます。Oracle 以外のデータベース内で一致しない Oracle ファンクションを使用している場合、ゲートウェイは補正を行います。データは Oracle 以外のデータベースから取得され、ファンクションは Oracle Database に適用されます。
- ヒントを使用して問合せ計画を改善します。ゲートウェイ・トレース・ファイルまたは実行計画を参照して、Oracle 以外のデータベースに送信する SQL を決定します。たとえば、SQL 文に Oracle 以外のデータベースでの表の結合が含まれていて、それらが事後処理である場合、ヒントを使用して、Oracle 以外のデータベースでの結合を実行できます。

Oracle 以外のデータベースのチューニング

オブティマイザに最適な計画を生成する十分な情報がない場合、パフォーマンスは悪影響を受けます。Oracle オプティマイザは、Oracle 以外のデータベースの表および索引統計を使用し最適なパスを決定して、Oracle 以外のデータベースのデータにアクセスします。この情報が欠落または不正確な場合、アクセス・パスは最適ではありません。Oracle 以外のデータベースで索引を定義すると、ゲートウェイのパフォーマンスが向上します。Oracle オプティマイザの詳細は、『Oracle Database 2 日でパフォーマンス・チューニング・ガイド』を参照してください。

関連する初期化パラメータの設定

次の初期化パラメータをカスタマイズして、パフォーマンスを向上できます。

- HS_RPC_FETCH_SIZE
- HS_FDS_FETCH_ROWS
- HS_LANGUAGE

たとえば、Oracle 以外のデータベースでアクセスしているデータが、Oracle Database で使用しているキャラクタ・セットと同じ場合、HS_LANGUAGE 初期化パラメータを Oracle Database のキャラクタ・セットに設定します。

Oracle Database Gateway インストールの場所の確認

Oracle Database Gateway のインストールの場所がパフォーマンスに影響する可能性があります。たとえば、CPU に制約がある場合、Oracle Database を実行中のコンピュータ・システムで Oracle Database Gateway をインストールしないでください。

十分なメモリの確認

Oracle Database Gateway を実行中のコンピュータ・システムに十分なメモリがあることを確認してください。処理中の SQL 文、同時セッションの数、オープン・カーソルの数およびアクセス中の表の列など、いくつかの要素はメモリ要件に影響します。

大 / 小文字の差異の考慮

Oracle 以外のデータベースを操作する場合は、識別子を二重引用符で囲まないかぎり Oracle Database の文字がデフォルトで大文字になることに注意してください。たとえば、emp という名前の Sybase 表を参照するには、次の例のように二重引用符で囲んだ名前を入力します。

```
SELECT * FROM "emp"@SYBS;
```

一方、ユーザー Smith が所有している emp という名前の Sybase 表を Oracle アプリケーションから参照するには、次のように入力します。

```
SELECT * FROM "Smith"."emp"@SYBS;
```

emp という名前の Sybase 表が SMITH (大文字の表所有者名) によって所有されている場合は、次の例に示すように二重引用符を付けずに所有者名を入力できます。

```
SQL> SELECT * FROM SMITH."emp"@SYBS;
```

```
SQL> SELECT * FROM smith."emp"@SYBS;
```

```
SQL> SELECT * FROM Smith."emp"@SYBS;
```

参照：

- 特定の Oracle Database Gateway に関する Oracle ドキュメント
- 『Oracle Database Heterogeneous Connectivity 管理者ガイド』

Oracle Streams を使用したデータのレプリケート

この章では、Oracle Streams レプリケーションに関する概念を示し、データベース間のデータの連続的なレプリケート方法を説明します。

この章は次の項で構成されています。

- [Oracle Streams レプリケーションの概要](#)
- [Oracle Streams レプリケーションの準備](#)
- [Oracle Streams レプリケーションの構成: 例](#)

参照:

- [第5章「Oracle Streams レプリケーション環境の管理」](#)
- [第6章「Oracle Streams レプリケーション環境の拡張」](#)
- [『Oracle Streams レプリケーション管理者ガイド』](#)
- [第7章「マテリアライズド・ビューを使用したデータのレプリケート」](#)

Oracle Streams レプリケーションの概要

レプリケーションは、複数のデータベースでデータベース・オブジェクトとデータを共有するプロセスです。データベース・オブジェクトとデータを複数のデータベースでメンテナンスするために、あるデータベースでのこれらのデータベース・オブジェクトの1つに対する変更が他のデータベースと共有されます。このようにして、レプリケーション環境内のすべてのデータベースでデータベース・オブジェクトとデータの同期が維持されます。

一部のレプリケーション環境は、共有データベース・オブジェクトに対して行われた変更を絶えずレプリケートする必要があります。**Oracle Streams** は、連続レプリケーションのための **Oracle Database** 機能です。通常、このような環境では、共有データベース・オブジェクトを含むデータベースがほぼ常にネットワークに接続され、これらのネットワーク接続上でデータベース変更を絶えずプッシュします。

1つの共有データベース・オブジェクトに対して変更が行われると、**Oracle Streams** は次のアクションを実行して、同じ変更が他の各データベースの対応する共有データベース・オブジェクトに対して行われるようにします。

1. **Oracle Streams** は、変更を自動的に取得し、それをキューにステージングします。
2. **Oracle Streams** は、共有データベース・オブジェクトを含む他の各データベース内のキューに変更を自動的にプッシュします。
3. **Oracle Streams** は、他の各データベースでの変更を自動的に処理します。処理中に、**Oracle Streams** はキューから変更をデキューし、変更を共有データベース・オブジェクトに適用します。

図 4-1 に、**Oracle Streams** 情報フローを示します。

図 4-1 **Oracle Streams** の情報フロー



Oracle Streams レプリケーションを使用して、複数のデータベースでデータを共有し、これらのデータベースでデータを効率よく最新に保つことができます。たとえば、全世界にいくつかのコール・センターを持つ会社は、顧客情報を各コール・センターのローカル・データベースに格納できます。このような環境では、**Oracle Streams** での連続レプリケーションにより、ある場所の顧客データに対して行った変更が、できるだけ早く他のすべての場所にプッシュされるようにすることができます。

Oracle Streams を使用してデータベース・オブジェクトに対する変更を取得する場合、変更は **論理変更レコード (LCR)** に書式設定されます。LCR は、データベース変更を説明する特定の形式のメッセージです。変更がデータ操作言語 (DML) 操作だった場合は、**行 LCR** により、DML 操作の結果として発生した行変更がそれぞれカプセル化されます。1つの DML 操作によって複数の行変更が発生する場合があるため、1つの DML 操作によって複数の行 LCR が発生することがあります。変更がデータ定義言語 (DDL) 操作だった場合、1つの **DDL LCR** が DDL 変更をカプセル化します。

次の各項で、**Oracle Streams** レプリケーションを詳細に説明します。

- [変更の取得の概要](#)
- [データベース間の変更の伝播の概要](#)
- [変更の適用の概要](#)
- [取得、伝播および適用の動作を制御するルールの概要](#)
- [同一でないコピーのルールベースの変換の概要](#)

- サプリメンタル・ロギングの概要
- 競合と競合解消の概要
- 変更の循環を回避するためのタグの概要
- Oracle Streams レプリケーション環境の一般的なタイプの概要
- Oracle Streams レプリケーションの構成プロセスの概要
- Oracle Streams が提供する主要な PL/SQL パッケージおよびデータ・ディクショナリ・ビューの概要

参照：

- 「Oracle Streams でのデータのレプリケート時期」 (1-5 ページ)

変更の取得の概要

Oracle Streams は、データベース変更を自動的に取得する 2 つの方法を提供します。

- **取得プロセス**は、比較的大量の表、スキーマ全体またはデータベースに対するデータ操作言語 (DML) 変更の取得に使用する必要があります。また、取得プロセスは、表およびその他のデータベース・オブジェクトに対するデータ定義言語 (DDL) 変更の取得に使用する必要があります。4-3 ページの「**取得プロセスでの変更の取得の概要**」を参照してください。
- **同期取得**は、比較的少数の表に対する DML 変更の取得に使用する必要があります。4-5 ページの「**同期取得での変更の取得の概要**」を参照してください。

単一の取得プロセスまたは単一の同期取得は、1 つのデータベースのみに対して行われた変更を取得できます。変更が発生したデータベースは、変更の**ソース・データベース**と呼ばれます。

注意： このマニュアルの例では DML の変更のみをレプリケートします。DML を変更する前に、DDL 変更のレプリケートを理解する必要があります。DDL 変更のレプリケートの詳細は、『Oracle Streams レプリケーション管理者ガイド』および『Oracle Database PL/SQL パッケージ・プロセスおよびタイプ・リファレンス』を参照してください。

参照：

- 「取得プロセスの管理」 (5-2 ページ)
- 「Oracle Streams レプリケーションの準備」 (4-21 ページ)

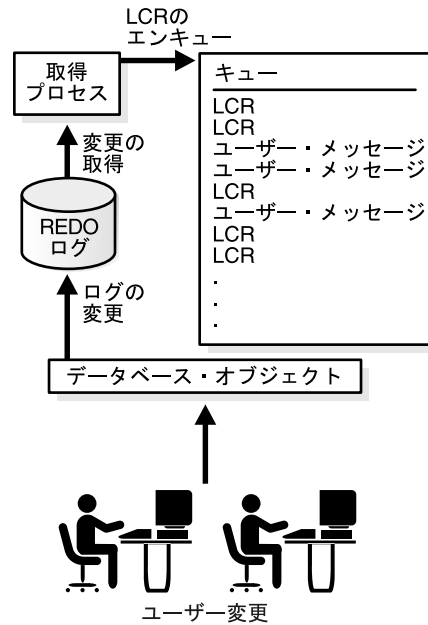
取得プロセスでの変更の取得の概要

取得プロセスは、REDO ログに記録された変更を非同期に取得する Oracle Database のオプションのバックグラウンド処理です。取得プロセスは、データベース変更を取得すると、それを論理変更レコード (LCR) に変換し、LCR をエンキューします。

取得プロセスは、単一のキューに常に関連付けられ、LCR をこのキューにのみエンキューしません。パフォーマンスを向上させるために、取得された LCR は常に**バッファ・キュー**に格納されます。バッファ・キューは、キューに関連付けられているシステム・グローバル領域 (SGA) メモリーです。

図 4-2 では、取得プロセスがどのように動作するかを示します。

図 4-2 取得プロセス



取得プロセスは、ソース・データベースまたはリモート・データベース上で実行できます。ソース・データベース上で実行する場合、取得プロセスは**ローカル取得プロセス**と呼ばれます。リモート・データベース上で実行する場合、取得プロセスは**ダウンストリーム取得プロセス**と呼ばれます。

ダウンストリーム取得では、REDO 転送サービスは、ソース・データベースのログ・ライター・プロセス (LGWR) を使用して、ダウンストリーム取得を実行するデータベースに REDO データを送信します。異なるデータベースが変更を取得するため、ダウンストリーム取得プロセスではソース・データベースのリソースが少なくても済みます。一方、ローカル取得プロセスは、ダウンストリーム取得よりも構成と管理が簡単です。ローカル取得プロセスによって、プラットフォームまたは Oracle Database のバージョンが異なるレプリケーション環境にも柔軟性が提供されます。

参照：

- 「チュートリアル: ダウンストリーム取得プロセスを使用した 2 データベース・レプリケーションの構成」 (4-29 ページ)
- 「チュートリアル: ローカル取得プロセスでのハブアンドスポーク・レプリケーションの構成」 (4-39 ページ)
- 「取得プロセスの管理」 (5-2 ページ)
- 「取得プロセスの監視」 (5-12 ページ)
- 『Oracle Streams 概要および管理』
- 『Oracle Database 2 日でデータベース管理者』

同期取得での変更の取得の概要

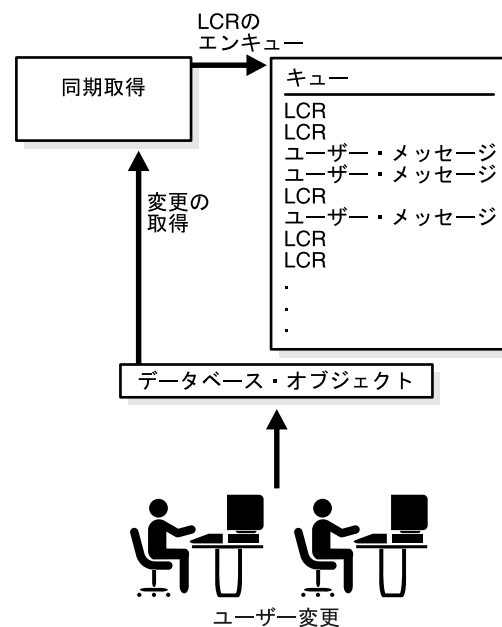
REDO ログの非同期マイニングのかわりに、**同期取得**は内部メカニズムを使用して、データ操作言語（DML）の変更が表に対して行われたときにその変更を取得します。単一の DML 変更によって、表の 1 つ以上の行が変更されることがあります。同期取得はそれぞれの行変更を取得し、それを行の論理変更レコード（LCR）に変換してエンキューします。

同期取得は常に単一のキューに関連付けられ、行 LCR をこのキューにのみエンキューします。同期取得は常に行 LCR を**永続キュー**にエンキューします。永続キューは、メモリーではなくハード・ディスク上のキュー表にメッセージを格納するキューの部分です。

通常、同期取得は、比較的少数の表に対する変更を取得するレプリケーション環境で使用するのが最適です。多くの表、スキーマ全体またはデータベース全体に対する変更を取得する必要がある場合は、同期取得のかわりに取得プロセスを使用する必要があります。

図 4-3 では、同期取得がどのように動作するかを示します。

図 4-3 同期取得



注意： Oracle Database 11g Standard Edition を使用している場合、同期取得はデータベース変更を自動的に取得できる唯一の Oracle Streams コンポーネントです。取得プロセスを使用するには、Oracle Database 11g Enterprise Edition が必要です。

参照：

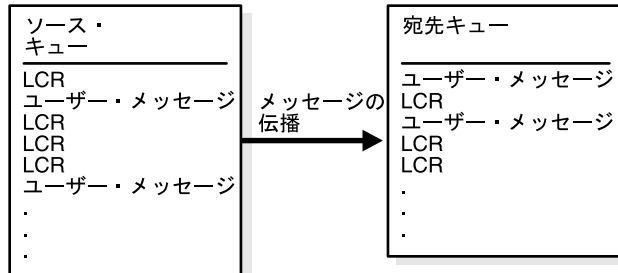
- 「チュートリアル: 同期取得を使用した 2 データベース・レプリケーションの構成」 (4-45 ページ)
- 『Oracle Streams 概要および管理』

データベース間の変更の伝播の概要

伝播は1つのキューから他のキューにメッセージを送信します。Oracle Streams を使用して、同じデータベース内、または異なるデータベース内の2つのキュー間のメッセージ伝播を構成できます。Oracle Streams は、データベース・リンクと Oracle Scheduler ジョブを使用してメッセージを送信します。伝播は常に「ソース・キュー」と「宛先キュー」の間で行われます。Oracle Streams レプリケーション環境では、伝播は通常、ローカル・データベース内のソース・キューからリモート・データベース内の宛先キューにデータベース変更を記述するメッセージを (LCR の形式で) 送信します。

図 4-4 では、伝播を示します。

図 4-4 伝播



参照：

- 「伝播の有効化および無効化」 (5-4 ページ)
- 「伝播の監視」 (5-16 ページ)
- 「チュートリアル: データベース・リンクの作成」 (2-8 ページ)
- Oracle Scheduler については、『Oracle Database 管理者ガイド』を参照してください。

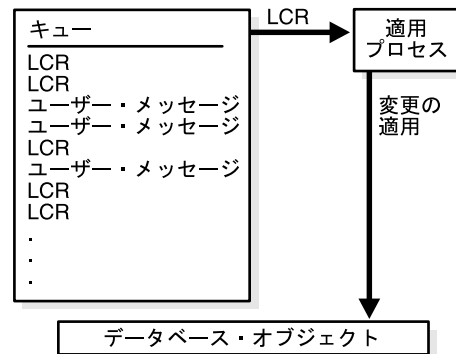
変更の適用の概要

データベース変更が取得されて伝播された後、これらの変更はキューに存在し、レプリケーション・プロセスを完了するために適用する準備ができます。**適用プロセス**は、論理変更レコード (LCR) およびその他のタイプのメッセージを特定のキューからデキューするオプションの Oracle Database バックグラウンド・プロセスです。単純な Oracle Streams レプリケーション環境では、通常、適用プロセスはデキューする LCR 内の変更をローカル・データベースのデータベース・オブジェクトに直接適用します。

適用プロセスは、常に単一のキューに関連付けられ、このキューからのみメッセージをデキューします。単一の適用プロセスは、バッファ・キューまたは永続キューからメッセージをデキューできますが、両方からデキューすることはできません。したがって、適用プロセスが、取得プロセスによって取得された変更を適用する場合は、バッファ・キューから LCR をデキューするように適用プロセスを構成する必要があります。一方、適用プロセスが、同期取得によって取得された変更を適用する場合は、永続キューから LCR をデキューするように適用プロセスを構成する必要があります。

図 4-5 では、適用プロセスがどのように動作するかを示します。

図 4-5 適用プロセス



適用プロセスは、LCR を正常に適用できない場合に、LCR、およびトランザクション内の他のすべての LCR を **エラー・キュー** という特別なキューに移動します。エラー・キューには、データベースの現在の適用エラーがすべて含まれます。データベースに複数の適用プロセスがある場合、エラー・キューには各適用プロセスの適用エラーが含まれます。エラーの原因となった条件を修正してから、エラー・キュー内の対応するトランザクションを再実行して変更を適用します。たとえば、表の行を変更してトランザクション・エラーの原因となった条件を修正し、トランザクションを再実行できます。

適用プロセスが変更をデータベース・オブジェクトに適用するためには、**インスタンス化システム変更番号 (SCN)** をデータベース・オブジェクトに対して設定する必要があります。インスタンス化 SCN は、ソース・データベースの SCN の後にコミットされた変更のみが適用プロセスによって適用されることを指定するデータベース・オブジェクトの SCN です。表のインスタンス化 SCN では、ソース・データベースと宛先データベースで指定された SCN に一貫性があることを前提としています。通常、インスタンス化 SCN は、Oracle Streams レプリケーション環境を構成するときに自動的に設定されます。

注意： 適用プロセスは、デキューするメッセージを、**適用ハンドラ**と呼ばれるカスタム処理のユーザー定義プロシージャにパラメータとして渡すこともできます。適用ハンドラについては、このマニュアルでは説明しません。詳細は、『Oracle Streams 概要および管理』を参照してください。

参照：

- 「適用プロセスの管理」 (5-6 ページ)
- 「適用プロセスの監視」 (5-19 ページ)

取得、伝播および適用の動作を制御するルールの概要

Oracle Streams レプリケーション構成では、レプリケート対象を指定する必要があります。取得プロセス、同期取得、伝播および適用プロセスは、**Oracle Streams クライアント**と呼ばれます。Oracle Streams クライアントがレプリケートする対象は、**ルール**によって決まります。ルールは Oracle Streams クライアントごとに個別に設定でき、異なる Oracle Streams クライアント間でルールが一致している必要はありません。

ルールは**ルール・セット**に編成でき、各 Oracle Streams クライアントの動作は、Oracle Streams クライアントに関連付けられているルール・セット内のルールによって決まります。**ポジティブ・ルール・セット**と**ネガティブ・ルール・セット**を取得プロセス、伝播および適用プロセスに関連付けることができますが、同期取得に関連付けることができるのは**ポジティブ・ルール・セット**のみです。

レプリケーション環境では、データベース変更がルール・セットを満たす場合に Oracle Streams クライアントがそのタスクを実行します。一般に、変更が Oracle Streams クライアントのルール・セットを満たすのは、変更についてネガティブ・ルール・セットのどのルールも TRUE と評価されず、ポジティブ・ルール・セットの 1 つ以上のルールが変更について TRUE と評価される場合です。ネガティブ・ルール・セットが常に最初に評価されます。

Oracle Streams レプリケーション環境でルール・セットを使用すると、具体的には次のことができます。

- 取得プロセスが REDO ログから取得する変更または破棄する変更を指定します。REDO ログ内で検出された変更が取得プロセスのルール・セットを満たしている場合、取得プロセスはその変更を取得します。REDO ログ内で検出された変更が取得プロセスのルール・セットを満たしていない場合、取得プロセスはその変更を破棄します。
- 同期取得が取得する変更を指定します。データ操作言語 (DML) の変更が同期取得のルール・セットを満たす場合、同期取得は変更がコミットされた直後に変更を取得します。表に対して行われた DML 変更が同期取得のルール・セットを満たさない場合、同期取得は変更を取得しません。
- 伝播があるキューから別のキューに送信する変更、または破棄する変更を指定します (LCR にカプセル化されています)。キュー内の LCR が伝播のルール・セットを満たしている場合、伝播はその LCR を送信します。キュー内の LCR が伝播のルール・セットを満たしていない場合、伝播はその LCR を破棄します。
- 適用プロセスがデキューする LCR、または破棄する LCR を指定します。キュー内の LCR が適用プロセスのルール・セットを満たしている場合、適用プロセスはその LCR をデキューして処理します。キュー内の LCR が適用プロセスのルール・セットを満たしていない場合、適用プロセスは LCR を破棄します。

参照：

- ルールの詳細は、『Oracle Streams 概要および管理』を参照してください。

同一でないコピーのルールベースの変換の概要

ルールベース変換は、複数のデータベースでデータベース・オブジェクトが同一でない場合に柔軟性を提供する追加の構成オプションです。ルールベース変換では、各データベースで正常に適用できるように、データベース・オブジェクトに対する変更を修正できます。ルールベース変換は、ポジティブ・ルール・セット内のルールが TRUE と評価された場合のメッセージに対する修正を示します。

たとえば、変更が発生したデータベースで表に 5 つの列があり、別のデータベースの共有表には 5 つの列のうち 4 つしかないとします。データ操作言語 (DML) 操作がソース・データベースの表に対して実行されると、行変更が取得され、行 LCR として書式設定されます。ルールベース変換は、他のデータベースに正常に適用できるようにこれらの行 LCR 内の余分な列を削除できます。行 LCR が変換されない場合、行 LCR には余分な列があるため他のデータベースでの適用プロセスでエラーが発生します。

ルールベース変換には、**宣言**と**カスタム**の 2 種類があります。宣言ルールベース変換には、DML 変更 (行 LCR) の結果として発生した行変更の一般的な変換シナリオのセットが含まれます。カスタム・ルールベース変換には、変換を実行するためのユーザー定義 PL/SQL ファンクションが必要です。このマニュアルでは、宣言ルールベース変換についてのみ説明します。

次の宣言ルールベース変換を使用できます。

- 追加列変換は、列を行 LCR に追加します。
- 削除列変換は、行 LCR から列を削除します。
- 名前変更列変換は、行 LCR 内の列の名前を変更します。
- 名前変更スキーマ変換は、行 LCR 内のスキーマの名前を変更します。
- 名前変更表変換は、行 LCR 内の表の名前を変更します。

これらの宣言ルールベース変換の1つを追加するときに、変換に関連付けるルールを指定します。指定したルールが行 LCR について TRUE と評価される場合、Oracle Streams は行 LCR で宣言変換を内部的に実行します。通常、ルールとルール・セットは Oracle Streams レプリケーション環境を構成するときに自動的に作成されます。

1つのルールに対して複数の宣言ルールベース変換を構成できますが、カスタム・ルールベース変換は、1つのルールに対して1つのみ構成できます。また、1つのルールに対して宣言ルールベース変換とカスタム・ルールベース変換の両方を構成することもできます。

変換は、Oracle Streams 情報フローの任意のステージ（取得中、伝播中または適用中）で発生します。変換がいつ発生するかは、変換が関連付けられているルールによって決まります。たとえば、伝播中に変換を実行するには、変換を伝播のポジティブ・ルール・セットのルールに関連付けます。

参照：

- ルールベースの変換の詳細は、『Oracle Streams 概要および管理』を参照してください。

サブリメンタル・ロギングの概要

サブリメンタル・ロギングは、操作（行の更新など）が実行されるたびに REDO ログに配置される追加の列データのプロセスです。取得プロセスは、この追加情報を取得し、論理変更レコード（LCR）に配置します。これらの LCR を適用する適用プロセスは、データベース変更を正しく適用するためにこの追加情報を必要とします。

参照：

- サブリメンタル・ロギングの詳細は、『Oracle Streams レプリケーション管理者ガイド』を参照してください。

競合と競合解消の概要

競合は、表のデータを共有している2つの異なるデータベースが表の同じ行をほぼ同時に変更した場合に発生します。これらの変更がこれらのデータベースの一方で取得され、もう一方のデータベースに送信されると、適用プロセスは、行 LCR を表に適用しようとしたときに競合を検出します。デフォルトでは、適用エラーがエラー・キューに入れられ、そこで手動で解消できます。適用エラーを回避するには、適用プロセスが環境に最適な方法で競合を処理するように競合解消を構成する必要があります。

Oracle Database には、行の UPDATE が原因で競合が発生した場合に競合を解消する競合ハンドラが組み込まれています。これらのハンドラは、**組込み更新競合ハンドラ**と呼ばれています。

適用プロセスで、デキューされた行 LCR に対する更新競合が発生した場合、2つのデータベースのデータの一貫性を維持するために行 LCR を適用するか破棄する必要があります。更新競合を解消する最も一般的な方法は、最新のタイムスタンプを持つ変更を保持し、古い変更を破棄することです。表に最新の時刻競合解消を構成する方法については、4-55 ページの「[チュートリアル: 表の最新時刻競合解消の構成](#)」を参照してください。

次の各項で、特定のタイプのレプリケーション環境で競合解消を構成する方法を説明します。

- [チュートリアル: ローカル取得プロセスを使用した2データベース・レプリケーションの構成](#)
- [チュートリアル: ローカル取得プロセスでのハブアンドスポーク・レプリケーションの構成](#)
- [チュートリアル: 同期取得を使用した2データベース・レプリケーションの構成](#)

注意： 競合は、単一のデータベースに対する変更のみが取得されるレプリケーション環境では発生しません。このようなレプリケーション環境では、通常、他のデータベースのレプリカは読取り専用になります。

参照：

- 「チュートリアル: 表の最新時刻競合解消の構成」 (4-55 ページ)
- 「構成された更新競合ハンドラの表示」 (5-23 ページ)
- 競合および競合解消の詳細は、『Oracle Streams レプリケーション管理者ガイド』を参照してください。

変更の循環を回避するためのタグの概要

変更循環は、発生元のデータベースに変更を再び送信することを意味します。通常は、変更循環によって各データベース変更の結果が発生元のデータベースに無限にループするため、変更循環を回避する必要があります。このようなループによって、データベース内に意図しないデータが生成され、ネットワークと環境のコンピュータ・リソースに負荷がかかります。デフォルトでは、Oracle Streams は変更循環を回避するように設計されています。

タグは、変更レコード内の追加情報です。データベース変更を記録する各 REDO エントリと、データベース変更をカプセル化する各論理変更レコード (LCR) には、タグが含まれます。タグのデータ型は RAW です。

デフォルトでは、変更レコードには次のタグ値があります。

- ユーザーまたはアプリケーションがデータベース変更を生成する場合、各変更のタグ値は NULL です。特定のデータベース・セッションに対してこのデフォルトを変更できます。
- 適用プロセスがデータベース・オブジェクトにデータベース変更を適用することによってその変更を生成する場合、各変更のタグ値は '00' (ゼロが2つ) と等価の16進数になります。特定の適用プロセスのデフォルトは変更可能です。

LCR のタグ値は、LCR がどのように取得されたかによって決まります。

- 取得プロセスによって取得された LCR には、取得された REDO レコードのタグ値があります。
- 同期取得によって取得された LCR には、変更を行ったデータベース・セッションのタグ値があります。

Oracle Streams クライアントのルールには、タグ値の条件を含めることができます。たとえば、取得プロセスのルールは、REDO レコードのタグ値に基づいて REDO ログ内の変更が取得されるかどうかを決定できます。Oracle Streams レプリケーション環境では、Oracle Streams クライアントはタグとルールを使用して変更循環を回避します。

次の各項で、特定のタイプのレプリケーション環境で変更循環が回避される方法を説明します。

- [チュートリアル: ローカル取得プロセスを使用した2データベース・レプリケーションの構成](#)
- [チュートリアル: ローカル取得プロセスでのハブアンドスポーク・レプリケーションの構成](#)
- [チュートリアル: 同期取得を使用した2データベース・レプリケーションの構成](#)

注意：

- 変更循環は、単一のデータベースに対する変更のみが取得されるレプリケーション環境では発生しません。
 - タグを使用して、特定のセッションで行われた変更のレプリケートを回避することもできます。セッションにタグを設定するには、DBMS_STREAMS.SET_TAG プロシージャを使用します。例については、5-30 ページの「データベース・オブジェクトでの適用エラーの修正」を参照してください。
-

参照：

- 「[適用エラーの管理](#)」(5-30 ページ)
- タグの詳細は、『Oracle Streams レプリケーション管理者ガイド』を参照してください。

Oracle Streams レプリケーション環境の一般的なタイプの概要

Oracle Streams により、多くの異なるタイプのカスタム・レプリケーション環境を構成できます。ただし、最も一般的なのは、2 データベース、ハブアンドスポークおよび n-way の 3 つのタイプのレプリケーション環境です。

次の各項で、これらの一般的なレプリケーション環境タイプについて説明し、どのタイプを使用するのが最適かの決定を支援します。

- [2 データベース・レプリケーション環境の概要](#)
- [ハブアンドスポーク・レプリケーション環境の概要](#)
- [n-way レプリケーション環境の概要](#)

2 データベース・レプリケーション環境の概要

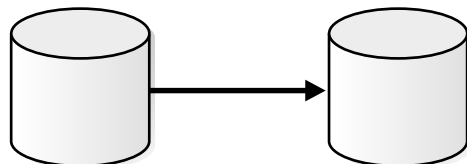
2 データベース・レプリケーション環境は、レプリケートされたデータベース・オブジェクトを 2 つのデータベースのみで共有するレプリケーション環境です。一方のデータベースでレプリケートされたデータベース・オブジェクトに対して行われた変更は、取得された後、もう一方のデータベースに直接送信されて適用されます。2 データベース・レプリケーション環境では、一方のデータベースのみでデータベース・オブジェクトに対する変更が許可される場合も、両方のデータベースでデータベース・オブジェクトに対する変更が許可される場合もあります。

レプリケートされたデータベース・オブジェクトに対する変更が一方のデータベースのみで許可されている場合、もう一方のデータベースには、これらのデータベース・オブジェクトの読み取り専用レプリカが含まれます。これを**単方向レプリケーション環境**と呼び、通常は次の基本コンポーネントが含まれます。

- 1 つ目のデータベースには、レプリケートされたデータベース・オブジェクトに対する変更を取得する取得プロセスまたは同期取得があります。
- 1 つ目のデータベースには、取得した変更を 2 つ目のデータベースに送信する伝播があります。
- 2 つ目のデータベースには、1 つ目のデータベースから変更を適用する適用プロセスがあります。
- パフォーマンスを最適化するために、各取得プロセスと適用プロセスには固有のキューがあります。

図 4-6 に、単方向レプリケーション用に構成された 2 データベース・レプリケーション環境を示します。

図 4-6 2 データベース・レプリケーション環境の単方向レプリケーション

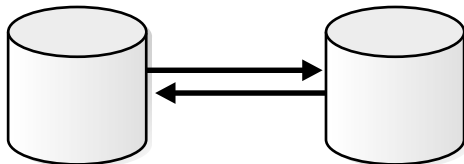


2 データベース・レプリケーション環境では、両方のデータベースで、レプリケートされたデータベース・オブジェクトを変更できます。この場合、データベース・オブジェクトに対する変更は、両方のデータベースで取得され、他方のデータベースに送信されて取得されます。これを**双方向レプリケーション環境**と呼び、通常は次の基本コンポーネントが含まれます。

- 各データベースには、レプリケートされたデータベース・オブジェクトに対する変更を取得する取得プロセスまたは同期取得があります。
- 各データベースには、取得した変更をもう一方のデータベースに送信する伝播があります。
- 各データベースには、もう一方のデータベースから変更を適用する適用プロセスがあります。
- パフォーマンスを最適化するために、各取得プロセスと適用プロセスには固有のキューがあります。

図 4-7 に、双方向レプリケーション用に構成された 2 データベース・レプリケーション環境を示します。

図 4-7 2 データベース・レプリケーション環境の双方向レプリケーション



通常、双方向レプリケーション環境では、レプリケートされたデータベース・オブジェクトの同期化を維持するために競合解消を構成する必要があります。DBMS_STREAMS_ADM パッケージの構成プロシージャを使用して、2 データベース・レプリケーション環境を構成できます。4-15 ページの「[Oracle Streams レプリケーションの構成プロシージャの概要](#)」を参照してください。

参照：

- 「チュートリアル：ローカル取得プロセスを使用した 2 データベース・レプリケーションの構成」 (4-24 ページ)
- 「チュートリアル：ダウンストリーム取得プロセスを使用した 2 データベース・レプリケーションの構成」 (4-29 ページ)
- 「チュートリアル：同期取得を使用した 2 データベース・レプリケーションの構成」 (4-45 ページ)
- 「競合と競合解消の概要」 (4-9 ページ)

ハブアンドスポーク・レプリケーション環境の概要

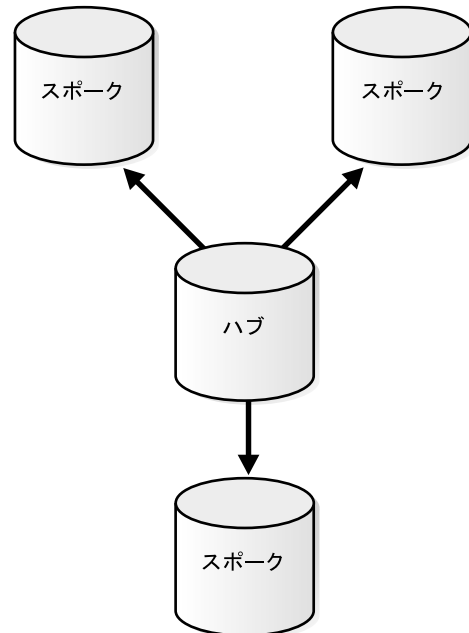
ハブアンドスポーク・レプリケーション環境は、集中データベース（ハブ）がセカンダリ・データベース（スポーク）と通信する環境です。スポークは相互に直接通信しません。ハブアンドスポーク・レプリケーション環境では、スポークはレプリケートされたデータベース・オブジェクトに対する変更を許可する場合と許可しない場合があります。

スポークが変更を許可しない場合、スポークにはハブのデータベース・オブジェクトの読取り専用レプリカが含まれます。このタイプのハブアンドスポーク・レプリケーション環境には、通常は次の基本コンポーネントがあります。

- ハブには、レプリケートされたデータベース・オブジェクトに対する変更を取得する取得プロセスまたは同期取得があります。
- ハブには、取得された変更を各スポークに送信する伝播があります。
- 各スポークには、ハブから変更を適用する適用プロセスがあります。
- パフォーマンスを最適化するために、各取得プロセスと適用プロセスには固有のキューがあります。

図 4-8 に、読取り専用スポークのあるハブアンドスポーク・レプリケーション環境を示します。

図 4-8 読取り専用スポークのあるハブアンドスポーク・レプリケーション環境

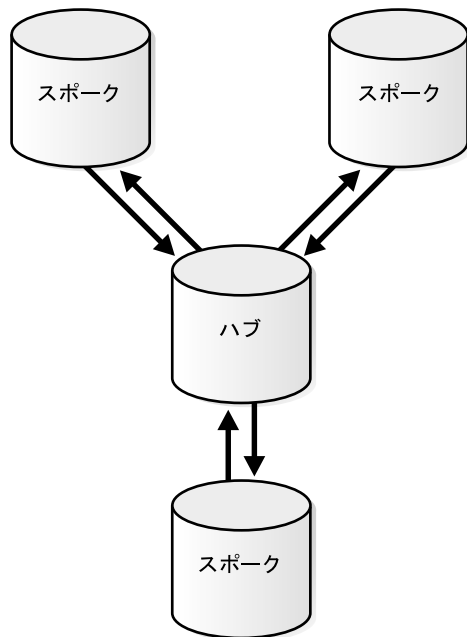


スポークがデータベース・オブジェクトに対する変更を許可する場合、通常は変更が取得されてハブに送信され、ハブが他のスポークに変更をレプリケートします。このタイプのハブアンドスポーク・レプリケーション環境には、通常は次の基本コンポーネントがあります。

- ハブには、レプリケートされたデータベース・オブジェクトに対する変更を取得する取得プロセスまたは同期取得があります。
- ハブには、取得された変更を各スポークに送信する伝播があります。
- 各スポークには、レプリケートされたデータベース・オブジェクトに対する変更を取得する取得プロセスまたは同期取得があります。
- 各スポークには、スポークで行われた変更をハブに送信する伝播があります。
- 各スポークには、ハブからの変更と他のスポークからの変更を適用する適用プロセスがあります。
- ハブには、各スポークからの変更を適用する個別の適用プロセスがあります。各適用プロセスは、それぞれのスポークから変更を適用する必要があります。
- パフォーマンスを最適化するために、各取得プロセスと適用プロセスには固有のキューがあります。

図 4-9 に、読取り / 書込みスポークのあるハブアンドスポーク・レプリケーション環境を示します。

図 4-9 読取り / 書込みスポークのあるハブアンドスポーク・レプリケーション環境



通常、スポーク・データベースでの変更が許可されるハブアンドスポーク・レプリケーション環境では、レプリケートされたデータベース・オブジェクトの同期化を維持するために競合解消を構成する必要があります。ハブアンドスポーク・レプリケーション環境では、レプリケートされたデータベース・オブジェクトに対する変更が一部のスポークでは許可されますが、残りのスポークでは許可されない場合があります。

DBMS_STREAMS_ADM パッケージの構成プロシージャを使用して、ハブアンドスポーク・レプリケーション環境を構成できます。4-15 ページの「Oracle Streams レプリケーションの構成プロシージャの概要」を参照してください。

ハブアンドスポーク・レプリケーションが有効な場合については、1-5 ページの「Oracle Streams でのデータのレプリケート時期」を参照してください。

参照：

- 「チュートリアル：ローカル取得プロセスでのハブアンドスポーク・レプリケーションの構成」 (4-39 ページ)
- 「競合と競合解消の概要」 (4-9 ページ)

n-way レプリケーション環境の概要

n-way レプリケーション環境は、各データベースが環境内の他のデータベースと相互に直接通信する環境です。あるデータベースでのレプリケートされたデータベース・オブジェクトに対する変更は、取得されて環境内の他の各データベースに直接送信され、そこで適用されます。

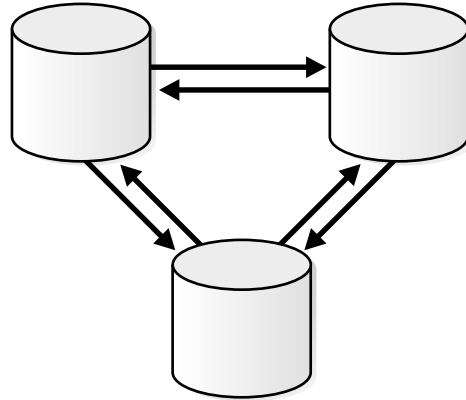
n-way レプリケーション環境には、通常は次の基本コンポーネントがあります。

- 各データベースには、レプリケートされたデータベース・オブジェクトに対する変更を取得する 1 つ以上の取得プロセスまたは同期取得があります。
- 各データベースには、取得された変更を他のデータベースに相互に送信する伝播があります。

- 各データベースには、他の各データベースからの変更を適用する適用プロセスがあります。別の適用プロセスが、各ソース・データベースから変更を適用する必要があります。
- パフォーマンスを最適化するために、各取得プロセスと適用プロセスには固有のキューがあります。

図 4-10 に、n-way レプリケーション環境を示します。

図 4-10 n-way レプリケーション環境



オラクル社が提供する次のパッケージを使用して、n-way レプリケーション環境を構成できます。

- DBMS_STREAMS_ADM を使用して、キューの設定、取得プロセスまたは同期取得の作成、伝播の作成、適用プロセスおよびレプリケーション環境でのルールおよびルール・セットの構成などの、ほとんどの構成アクションが実行できます。
- DBMS_CAPTURE_ADM を使用して、レプリケーション環境で構成した取得プロセスを開始できます。
- DBMS_APPLY_ADM を使用して競合解消を構成し、その他の構成タスクと同様に適用プロセスを開始できます。

n-way レプリケーションが有効な場合については、1-5 ページの「[Oracle Streams でのデータのレプリケート時期](#)」を参照してください。

通常、n-way レプリケーション環境では、レプリケートされたデータベース・オブジェクトの同期化を維持するために競合解消を構成する必要があります。

n-way レプリケーション環境の構成は、このマニュアルの対象外です。n-way レプリケーション環境を構成する例の詳細は、『Oracle Streams レプリケーション管理者ガイド』を参照してください。

Oracle Streams レプリケーションの構成プロシージャの概要

Oracle Streams レプリケーション環境を構成する最も簡単な方法は、DBMS_STREAMS_ADM パッケージにある次のいずれかの構成プロシージャを実行することです。

- MAINTAIN_GLOBAL プロシージャ: データベース・レベルでの変更を 2 つのデータベース間でレプリケートする Oracle Streams 環境を構成します。
- MAINTAIN_SCHEMAS プロシージャ: 指定したスキーマへの変更を 2 つのデータベース間でレプリケートする Oracle Streams 環境を構成します。
- MAINTAIN_SIMPLE_TTS プロシージャ: ソース・データベースからの単純な表領域のクローンを宛先データベースに作成し、Oracle Streams を使用してこの表領域を両方のデータベースで維持します。

- **MAINTAIN_TABLES** プロシージャ: 指定した表への変更を 2 つのデータベース間でレプリケートする Oracle Streams 環境を構成します。
- **MAINTAIN_TTS** プロシージャ: ソース・データベースからの表領域セットのクローンを宛先データベースに作成し、Oracle Streams を使用してこれらの表領域を両方のデータベースで維持します。

これらのプロシージャは、同時に 2 つのデータベースを構成します。一方のデータベースをソース・データベースに、もう一方のデータベースを宛先データベースに指定する必要があります。これらのプロシージャを複数回実行することによって、2 つ以上のデータベースを使用するレプリケーション環境を構成できます。

表 4-1 に、これらのプロシージャに必要なパラメータを示します。

表 4-1 Oracle Streams レプリケーションの構成プロシージャに必要なパラメータ

パラメータ	プロシージャ	説明
source_directory_object	すべてのプロシージャ	ソース・データベースを実行するコンピュータ・システム上のディレクトリのディレクトリ・オブジェクト。このディレクトリには、生成されたデータ・ポンプ・エクスポート・ダンプ・ファイルが保存されます。
destination_directory_object	すべてのプロシージャ	宛先データベースを実行するコンピュータ・システム上のディレクトリのディレクトリ・オブジェクト。このディレクトリには、生成されたデータ・ポンプ・エクスポート・ダンプ・ファイルが転送されます。ダンプ・ファイルは、レプリケートされたデータベース・オブジェクトを宛先データベースでインスタンス化するために使用します。
source_database	すべてのプロシージャ	ソース・データベースのグローバル名。指定されたデータベースには、レプリケートするデータベース・オブジェクトが含まれる必要があります。
destination_database	すべてのプロシージャ	宛先データベースのグローバル名。レプリケートするデータベース・オブジェクトは、宛先データベースではオプションです。宛先データベースにデータベース・オブジェクトが存在しない場合、データベース・オブジェクトはデータ・ポンプ・エクスポート/インポートを使用してインスタンス化されます。 宛先データベースがローカル・データベースでない場合は、ローカル・データベースから宛先データベースへのデータベース・リンクが存在する必要があります。このデータベース・リンクは、宛先データベースのグローバル名と同じ名前を持ち、プロシージャを実行するユーザーからアクセス可能である必要があります。
schema_names	MAINTAIN_SCHEMAS のみ	レプリケーション用に構成されるスキーマ。
tablespace_name	MAINTAIN_SIMPLE_TTS のみ	レプリケーション用に構成される表領域。
table_names	MAINTAIN_TABLES のみ	レプリケーション用に構成される表。
tablespace_names	MAINTAIN_TTS のみ	レプリケーション用に構成される表領域。

また、各プロシージャには、オプションのパラメータもあります。bi_directional パラメータは、重要なオプションのパラメータです。レプリケートされたデータベース・オブジェクトに対する変更を各データベースで取得して、他のデータベースに送信するには、bi_directional パラメータを TRUE に設定する必要があります。このパラメータのデフォルトの設定は FALSE です。bi_directional パラメータが FALSE に設定されていると、プロシージャは、宛先データベースでの変更が取得されない単方向レプリケーション環境を構成します。

これらのプロシージャは、Oracle Streams レプリケーション環境を構成するために複数のタスクを実行します。次のタスクがあります。

- ソース・データベースで、レプリケートされたデータベース・オブジェクトに対するサブリメンタル・ロギングを構成します。4-9 ページの「[サブリメンタル・ロギングの概要](#)」を参照してください。
- bi_directional パラメータが TRUE に設定されている場合に、宛先データベースで、レプリケートされたデータベース・オブジェクトに対するサブリメンタル・ロギングを構成します。
- 宛先データベースでデータベース・オブジェクトをインスタンス化します。宛先データベースにデータベース・オブジェクトが存在しない場合、プロシージャは、データ・ポンプ・エクスポート/インポートを使用して、宛先データベースでデータベース・オブジェクトをインスタンス化します。
- ソース・データベースでレプリケートされたデータベース・オブジェクトに対する変更を取得するように取得プロセスを構成します。この取得プロセスは、ローカル取得プロセスまたはダウストリーム取得プロセスのいずれかです。source_database パラメータで指定されたデータベースでプロシージャを実行すると、このデータベースでローカル取得プロセスが構成されます。source_database パラメータで指定されたデータベース以外のデータベースでプロシージャを実行すると、プロシージャを実行するデータベースでダウストリーム取得プロセスが構成されます。4-3 ページの「[取得プロセスでの変更の取得の概要](#)」を参照してください。
- bi_directional パラメータが TRUE に設定されている場合に、宛先データベースで、レプリケートされたデータベース・オブジェクトに対する変更を取得するための取得プロセスを構成します。この取得プロセスは、ローカル取得プロセスである必要があります。
- 各データベースに、取得した変更を格納するための 1 つ以上のキューを構成します。
- ソース・データベースでデータベース・オブジェクトに対して行われた変更を宛先データベースに送信するための伝播を構成します。4-6 ページの「[データベース間の変更の伝播の概要](#)」を参照してください。
- bi_directional パラメータが TRUE に設定されている場合に、宛先データベースでデータベース・オブジェクトに対して行われた変更をソース・データベースに送信するための伝播を構成します。
- ソース・データベースからの変更を適用する適用プロセスを宛先データベースで構成します。4-6 ページの「[変更の適用の概要](#)」を参照してください。
- bi_directional パラメータが TRUE に設定されている場合に、宛先データベースからの変更を適用する適用プロセスをソース・データベースで構成します。
- 各取得プロセス、伝播および適用プロセスに対して、ルール・セットとルールを構成します。ルールは、指定されたデータベース・オブジェクトに対する変更をレプリケートするよう Oracle Streams クライアントに指示します。4-7 ページの「[取得、伝播および適用の動作を制御するルールの概要](#)」を参照してください。
- 宛先データベースで、レプリケートされたデータベース・オブジェクトのインスタンス化 SCN を設定します。4-6 ページの「[変更の適用の概要](#)」を参照してください。

- `bi_directional` パラメータが `TRUE` に設定されている場合に、ソース・データベースで、レプリケートされたデータベース・オブジェクトのインスタンス化 `SCN` を設定しません。

ヒント: いずれかのプロシージャが実行するすべてのアクションを詳細に表示するには、プロシージャを使用してスクリプトを生成し、テキスト・エディタでスクリプトを表示します。スクリプトの生成には、`perform_actions`、`script_name` および `script_directory_object` パラメータを使用できます。

これらのプロシージャでは、常にハブアンドスポーク・レプリケーション環境用のタグが構成されます。これらのプロシージャおよびタグに関する重要な考慮事項を次に示します。

- 2 データベース・レプリケーション環境は、これらのプロシージャを使用して構成できません。これらのプロシージャで 2 データベース環境のタグを構成することで、変更の循環を回避できます。将来的に 3 つ以上のデータベースを使用するようレプリケーション環境を拡張する予定の場合は、タグの構成方法を理解する必要があります。拡張されたデータベース環境がハブアンドスポーク環境でないと、変更の循環を回避するためにタグの修正が必要になります。
- 3 つ以上のデータベースを含むレプリケーション環境を構成する場合、これらのプロシージャを使用すれば、ハブアンドスポーク・レプリケーション環境を構成できます。これらのプロシージャでは、ハブアンドスポーク環境のタグを構成することで、変更の循環を回避できます。
- これらのプロシージャでは、`n-way` レプリケーション環境を構成できません。これらのプロシージャを使用すると、変更の循環が発生する可能性があります。

タグおよび様々なタイプのレプリケーション環境については、4-10 ページの「[変更の循環を回避するためのタグの概要](#)」および 4-11 ページの「[Oracle Streams レプリケーション環境の一般的なタイプの概要](#)」を参照してください。

注意: 現在、これらの構成プロシージャで構成できるのは、変更を取得する取得プロセスのみです。同期取得を使用するレプリケーション環境をこれらのプロシージャで構成することはできません。同期取得は、`DBMS_STREAMS_ADM` パッケージの `ADD_TABLE_RULES` および `ADD_SUBSET_RULES` プロシージャを使用して構成できます。

参照:

- 「チュートリアル: ローカル取得プロセスを使用した 2 データベース・レプリケーションの構成」 (4-24 ページ)
- 「チュートリアル: ダウンストリーム取得プロセスを使用した 2 データベース・レプリケーションの構成」 (4-29 ページ)
- 「チュートリアル: ローカル取得プロセスでのハブアンドスポーク・レプリケーションの構成」 (4-39 ページ)
- `DBMS_STREAMS_ADM` パッケージのプロシージャの詳細は、『Oracle Database PL/SQL パッケージ・プロシージャおよびタイプ・リファレンス』を参照してください。

Oracle Streams が提供する主要な PL/SQL パッケージおよびデータ・ディクショナリ・ビューの概要

Oracle Enterprise Manager 以外にも、Oracle Streams レプリケーション環境を構成および管理するための PL/SQL パッケージが提供されています。また、Oracle Streams レプリケーション環境を監視するためのデータ・ディクショナリ・ビューも提供されています。

次の各項で、Oracle Streams が提供する主要な PL/SQL パッケージおよびデータ・ディクショナリ・ビューについて説明します。

- [Oracle Streams が提供する主要な PL/SQL パッケージの概要](#)
- [Oracle Streams の主要なデータ・ディクショナリ・ビューの概要](#)

Oracle Streams が提供する主要な PL/SQL パッケージの概要

表 4-2 に、Oracle Streams レプリケーション環境の構成および管理に重要な PL/SQL パッケージを示します。

表 4-2 Oracle Streams が提供する主要な PL/SQL パッケージ

パッケージ	説明
DBMS_STREAMS_ADM	このパッケージを使用すると、Oracle Streams 環境で一般的なタスクを簡単に実行できます。このパッケージには、Oracle Streams レプリケーション環境を構成および維持するためのプロシージャが含まれます。また、表、スキーマおよびデータベース・レベルで、取得プロセス、同期取得、伝播および適用プロセスのための単純なルールを追加および削除するための管理インタフェースも備わっています。さらに、キューの作成や、Oracle Streams メタデータ（データ・ディクショナリ情報など）の管理のためのプロシージャが含まれます。
DBMS_CAPTURE_ADM	このパッケージでは、取得プロセスの起動、停止および構成のための管理インタフェースが提供されます。また、同期取得の構成のための管理インタフェースも提供されます。さらに、宛先データベースでのインスタンス化に備えて、ソース・データベースでデータベース・オブジェクトを準備するための管理プロシージャも提供されます。
DBMS_PROPAGATION_ADM	このパッケージでは、伝播の起動、停止および構成のための管理インタフェースが提供されます。
DBMS_APPLY_ADM	このパッケージでは、適用プロセスの起動、停止および構成のための管理インタフェースが提供されます。また、競合の検出と解消の構成および適用エラーの管理のためのサブプログラムも含まれます。さらに、このパッケージには、適用ハンドラを構成できるプロシージャも含まれます。このパッケージには、宛先データベースでオブジェクトのインスタンス化 SCN を設定する管理プロシージャが用意されています。
DBMS_STREAMS_AUTH	このパッケージでは、Oracle Streams 管理者に対する権限の付与および取消のためのサブプログラムが提供されます。
DBMS_STREAMS_ADVISOR_ADM	このパッケージでは、Oracle Streams 環境についての情報の収集および収集した情報に基づくデータベース管理者へのアドバイスを行うためのインタフェースが提供されます。このパッケージは、Oracle Streams パフォーマンス・アドバイザーの一部です。

参照：

- これらの Oracle Streams パッケージおよびその他の Oracle Streams パッケージの詳細は、『Oracle Streams 概要および管理』を参照してください。
- 「[ハブアンドスポーク・レプリケーション環境の概要](#)」(4-12 ページ)

Oracle Streams の主要なデータ・ディクショナリ・ビューの概要

表 4-3 に、Oracle Streams レプリケーション環境の監視に重要なデータ・ディクショナリ・ビューを示します。

表 4-3 Oracle Streams の主要なデータ・ディクショナリ・ビュー

データ・ディクショナリ・ビュー	説明
ALL_APPLY	適用プロセスに関する情報を表示します。
DBA_APPLY	
ALL_APPLY_CONFLICT_COLUMNS	競合ハンドラに関する情報を表示します。
DBA_APPLY_CONFLICT_COLUMNS	
ALL_APPLY_ERROR	適用プロセスが生成するエラー・トランザクションに関する情報を表示します。
DBA_APPLY_ERROR	
ALL_CAPTURE	取得プロセスに関する情報を表示します。
DBA_CAPTURE	
ALL_PROPAGATION	伝播に関する情報を表示します。
DBA_PROPAGATION	
ALL_STREAMS_COLUMNS	同期取得および適用プロセスでサポートされていない列に関する情報を表示します。
DBA_STREAMS_COLUMNS	
ALL_STREAMS_RULES	取得プロセス、同期取得、伝播および適用プロセスで使用するルールに関する情報を表示します。
DBA_STREAMS_RULES	
ALL_STREAMS_UNSUPPORTED	取得プロセスでサポートされていないデータベース・オブジェクトに関する情報を表示します。
DBA_STREAMS_UNSUPPORTED	
ALL_SYNC_CAPTURE	同期取得に関する情報を表示します。
DBA_SYNC_CAPTURE	
V\$BUFFERED_QUEUES	バッファ・キューおよびバッファ・キュー内のメッセージに関する情報を表示します。
V\$PROPAGATION_RECEIVER	伝播によってバッファ・キュー内に受信されるメッセージに関する情報を表示します。
V\$PROPAGATION_SENDER	伝播によってバッファ・キューから送信されるメッセージに関する情報を表示します。
V\$STREAMS_APPLY_COORDINATOR	有効化されている適用プロセスのコーディネータに関する情報を表示します。
V\$STREAMS_APPLY_READER	有効化されている適用プロセスのリーダー・サーバーに関する情報を表示します。
V\$STREAMS_APPLY_SERVER	有効化されている適用プロセスの適用サーバーに関する情報を表示します。
V\$STREAMS_CAPTURE	有効化されている取得プロセスに関する情報を表示します。
V\$STREAMS_TRANSACTION	取得プロセスまたは適用プロセスによって処理されているトランザクションに関する情報を表示します。このビューを使用すると、長時間実行トランザクションの識別し、各トランザクションで処理されている論理変更レコード (LCR) の数を判断できます。このビューには、取得プロセスで取得された LCR に関する情報のみが表示されます。

参照：

- これらの Oracle Streams データ・ディクショナリ・ビューおよびその他のデータ・ディクショナリ・ビューの詳細は、『Oracle Streams 概要および管理』を参照してください。
- 「Oracle Streams レプリケーション環境の監視」(5-9 ページ)

Oracle Streams レプリケーションの準備

Oracle Streams レプリケーションを構成する前に、レプリケーション環境に参加するデータベースを準備します。

Oracle Streams レプリケーションを準備するには：

1. Oracle Streams でレプリケーション環境を構成する前に、初期化パラメータを正しく設定します。
 - **グローバル名：**Oracle Streams レプリケーション環境に参加する各データベースで GLOBAL_NAMES 初期化パラメータを TRUE に設定します。2-2 ページの「GLOBAL_NAMES 初期化パラメータを TRUE に設定」を参照してください。
 - **互換性：**Oracle Streams の最新の機能を使用するには、COMPATIBLE 初期化パラメータをできるだけ高く設定することをお勧めします。可能な場合は、このパラメータを 11.1.0 以上に設定します。
 - **システム・グローバル領域 (SGA) および Oracle Streams プール：**Oracle Streams プールが、レプリケーション環境に対して作成される Oracle Streams コンポーネントを十分に収容できる大きさであることを確認します。Oracle Streams プールは、システム・グローバル領域 (SGA) の一部です。MEMORY_TARGET 初期化パラメータ (自動メモリー管理)、SGA_TARGET 初期化パラメータ (自動共有メモリー管理) または STREAMS_POOL_SIZE 初期化パラメータを設定することにより、Oracle Streams を管理できます。Oracle Streams プールの詳細は、『Oracle Streams 概要および管理』を参照してください。
- Oracle Streams コンポーネントのメモリー要件は次のとおりです。
- 各キューには、少なくとも 10MB のメモリーが必要です。
 - 各取得プロセスの並列処理には、少なくとも 10MB のメモリーが必要です。取得プロセスが変更を取得する際に使用する処理の数は、取得プロセス・パラメータ parallelism で制御されます。取得プロセスのパフォーマンスは、取得プロセスの並行性を調整すると改善される場合があります。
 - 各伝播には少なくとも 1MB のメモリーが必要です。
 - 各適用プロセスの並列処理には、少なくとも 1MB のメモリーが必要です。適用プロセスが変更を適用する際に使用する処理の数は、適用プロセス・パラメータ parallelism で制御されます。適用プロセスのパフォーマンスは、適用プロセスの並行性を調整すると改善される場合があります。
- **処理とセッション：**Oracle Streams 取得プロセス、伝播および適用プロセスは、バックグラウンドで実行されるプロセスを使用します。これらのプロセスに適応するには PROCESSES および SESSIONS 初期化パラメータの値を増やすことが必要な場合があります。
2. Oracle Streams レプリケーション環境のベスト・プラクティスを確認し、環境を構成する際にはそれに従います。ベスト・プラクティスの詳細は、『Oracle Streams レプリケーション管理者ガイド』を参照してください。

ベスト・プラクティスに従えば、パフォーマンスの最適な環境を構成して問題を回避することができます。DBMS_STREAMS_ADM パッケージの構成プロシージャを使用すれば自動的にベスト・プラクティスに従いますが、構成プロシージャを使用せずに Oracle Streams レプリケーション環境を構成する場合は、ベスト・プラクティスをよく確認し、できるだけそれに従ってください。

次に、Oracle Streams の構成時に従う必要のある重要なベスト・プラクティスを示します。

- Oracle Streams 管理者に個別の表領域を構成します。「チュートリアル: Oracle Streams 管理者の作成」の手順がこのベスト・プラクティスに従っています。
- 取得プロセス、同期取得および適用プロセスに個別のキューを使用します。最適なパフォーマンスを得るには、通常、これらのコンポーネントがキューを共有しないようにする必要があります。
- キュー・ツー・キュー伝播を使用します。

Oracle Streams 環境を構成した後は、Oracle Streams 環境での操作時に次の重要なベスト・プラクティスに従う必要があります。

- パフォーマンスを監視し、必要に応じて調整します。
- キューのサイズを監視します。
- バックアップに関する Oracle Streams のベスト・プラクティスに従います。
- Oracle Streams の情報に関するアラート・ログを確認します。
- 最適なパフォーマンスを得るために、取得プロセスの並列性を設定します。
- 最適なパフォーマンスを得るために、適用プロセスの並列性を設定します。
- 適用エラーを確認し、発生する場合は管理します。

これらのベスト・プラクティスと、その他の Oracle Streams のベスト・プラクティスの詳細は、『Oracle Streams レプリケーション管理者ガイド』を参照してください。

参照:

- 「Oracle Streams レプリケーションの概要」(4-2 ページ)
- 「Oracle Streams レプリケーション環境の管理」(5-1 ページ)
- 『Oracle Database 2 日でデータベース管理者』
- Oracle Streams 環境で重要な初期化パラメータの詳細は、『Oracle Streams 概要および管理』を参照してください。

Oracle Streams レプリケーションの構成 : 例

この項では、Oracle Streams レプリケーション環境の構成方法について、例を使用して説明します。例では、最も一般的なタイプの Oracle Stream レプリケーション環境を構成します。

次に、例について説明します。

- [チュートリアル: ローカル取得プロセスを使用した 2 データベース・レプリケーションの構成](#)

この例では、2つのデータベースで hr スキーマにあるすべての表にデータ操作言語 (DML) 変更をレプリケートする Oracle Streams レプリケーション環境を構成します。構成では、レプリケートされたデータベース・オブジェクトに対する変更を取得する取得プロセスを使用します。

例は、単方向レプリケーションまたは双方向レプリケーション用に環境を構成する方法を示しています。単方向レプリケーションでは、1つのデータベースのみがデータベース・オブジェクトを変更できます。双方向レプリケーションでは、両方のデータベースがデータベース・オブジェクトを変更できます。

この構成は、2つのデータベースのみを含む比較的単純なレプリケーション環境を構成する際に有効です。2つ目のデータベースをレポートまたはデータ分析に使用する場合は、単方向レプリケーション環境を構成します。両方のデータベースでデータの読取り / 書込みが必要な場合は、双方向レプリケーション環境を構成します。たとえば、スケラビリティやデータの可用性を向上させるには、双方向レプリケーションを使用します。

2 データベース・レプリケーション環境の詳細は、4-11 ページの「[2 データベース・レプリケーション環境の概要](#)」を参照してください。

- チュートリアル: ダウンストリーム取得プロセスを使用した 2 データベース・レプリケーションの構成

この例では、2 つのデータベースで hr スキーマにあるすべての表にデータ操作言語 (DML) の変更をレプリケートする Oracle Streams レプリケーション環境を構成します。例は、単方向レプリケーションの構成方法を示していて、レプリケートされたデータベース・オブジェクトに対する変更は、1 つのソース・データベースでのみ許可されます。構成では、ソース・データベースでデータベース・オブジェクトに対して行った変更を取得するために、宛先データベースでダウンストリーム取得プロセスを使用しています。宛先データベースでは、データベース・オブジェクトは読み取り専用です。

この構成は、プライマリ・データベースから別のデータベースにレポートまたはデータ分析をオフロードする際に使用します。この例では、2 つ目のデータベースで取得プロセスを実行し、プライマリ・データベースの負荷を削減しています。

2 データベース・レプリケーション環境の詳細は、4-11 ページの「2 データベース・レプリケーション環境の概要」を参照してください。

- チュートリアル: ローカル取得プロセスでのハブアンドスポーク・レプリケーションの構成

この例では、3 つのデータベースで hr スキーマにあるすべての表にデータ操作言語 (DML) をレプリケートする Oracle Streams レプリケーション環境を構成します。この例では、1 つのハブと 2 つのスポークを持つハブアンドスポーク・レプリケーション環境を構成します。ハブアンドスポーク・レプリケーション環境は、1 つの中央データベース (ハブ) がセカンダリ・データベース (スポーク) と通信する環境です。この例では、スポーク・データベースによって、レプリケートされたデータベース・オブジェクトに対する変更が可能になります。構成では、レプリケートされたデータベース・オブジェクトに対する変更を取得する取得プロセスを使用します。

この構成は、複数のセカンダリ・データベースにデータをレプリケートする必要のある中央データベースが存在する場合に使用します。たとえば、本社に中央データベースがあり、販売代理店に複数のセカンダリ・データベースがある企業にこの構成が有効です。

ハブアンドスポーク・レプリケーション環境の詳細は、4-12 ページの「ハブアンドスポーク・レプリケーション環境の概要」を参照してください。

- チュートリアル: 同期取得を使用した 2 データベース・レプリケーションの構成

この例では、2 つのデータベースで hr スキーマにある employees 表および departments 表にデータ操作言語 (DML) 変更をレプリケートする Oracle Streams レプリケーション環境を構成します。この例では、2 つのデータベース間での双方向レプリケーションを構成しています。このため、データベース・オブジェクトに対する変更は両方のデータベースで許可されています。構成では、レプリケートされたデータベース・オブジェクトに対する変更を取得する同期取得を使用します。

この構成は、2 つのデータベースと少数の表のみを使用する比較的単純なレプリケーション環境を構成する際に使用します。Oracle Database 11g Standard Edition をインストールしている場合は、同期取得を使用することもできます。取得プロセスを使用するには、Oracle Database 11g Enterprise Edition をインストールする必要があります。

2 データベース・レプリケーション環境の詳細は、4-11 ページの「2 データベース・レプリケーション環境の概要」を参照してください。

この項では、表の競合解消を構成する例も示します。4-55 ページの「チュートリアル: 表の最新時刻競合解消の構成」を参照してください。競合解消は、レプリケートされた表に対する DML 変更の実行を 2 つ以上のデータベースに対して許可するレプリケーション環境で使用します。この例は、最新時刻の競合解消を構成しています。このため、表に対する行変更によって競合が発生すると、最新の変更は保持され、古い変更は破棄されます。競合解消の詳細は、4-9 ページの「競合と競合解消の概要」を参照してください。

注意： 一般的な Oracle Streams レプリケーション環境には、他に n-way 環境があります。4-14 ページの「n-way レプリケーション環境の概要」を参照してください。

参照：

- 「Oracle Streams レプリケーション環境の一般的なタイプの概要」
(4-11 ページ)

チュートリアル: ローカル取得プロセスを使用した 2 データベース・レプリケーションの構成

この例では、hr スキーマにあるすべての表に、データ操作言語 (DML) 変更をレプリケートする Oracle Streams レプリケーション環境を構成します。この例では、変更を取得するローカル取得プロセスを使用して 2 データベース・レプリケーション環境を構成します。この例では、グローバル・データベース名 db1.example.com および db2.example.com を使用します。ただし、例を完了するために、ご使用の環境のデータベースに置き換えることもできます。2 データベース・レプリケーション環境の詳細は、4-11 ページの「2 データベース・レプリケーション環境の概要」を参照してください。

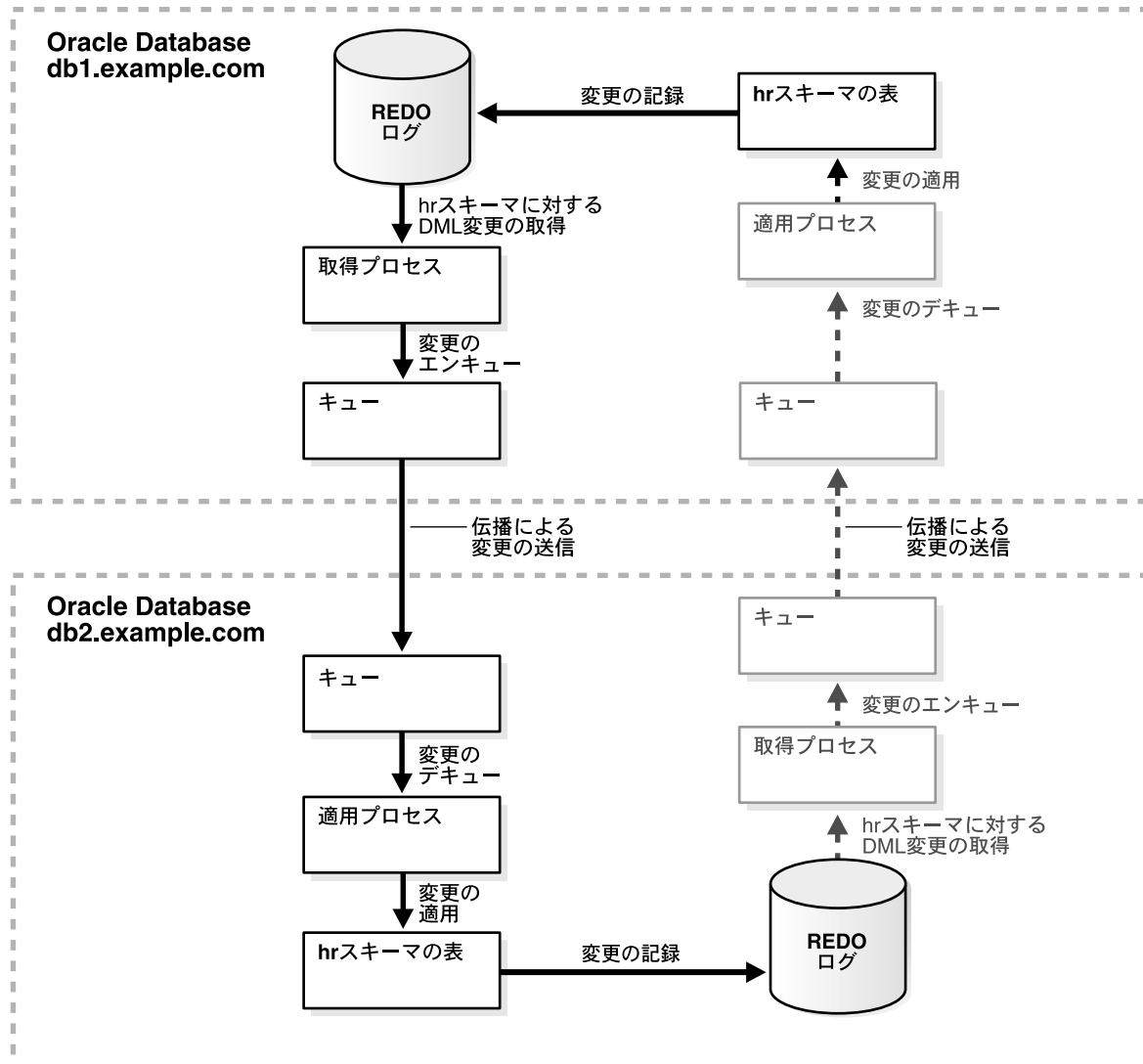
この例では、DBMS_STREAMS_ADM パッケージの MAINTAIN_SCHEMAS プロシージャを使用して 2 データベース・レプリケーション環境を構成します。このプロシージャは、1 つ以上のスキーマをレプリケートする Oracle Streams 環境を構成する最速で最も単純な方法です。また、プロシージャは Oracle Streams レプリケーション環境に対して設定されたベスト・プラクティスに従います。

レプリケーション用に構成されているデータベース・オブジェクトは、MAINTAIN_SCHEMAS プロシージャの実行時に、宛先データベースに存在する場合としない場合があります。宛先データベースにデータベース・オブジェクトが存在しない場合、MAINTAIN_SCHEMAS プロシージャはデータ・ポンプ・エクスポート / インポートを使用して、宛先データベースでデータベース・オブジェクトをインスタンス化します。インスタンス化の際に、これらのデータベース・オブジェクトにインスタンス化 SCN が設定されます。宛先データベースにデータベース・オブジェクトが存在する場合、MAINTAIN_SCHEMAS プロシージャは既存のデータベース・オブジェクトを保持して、インスタンス化 SCN を設定します。この例では、MAINTAIN_SCHEMAS プロシージャを実行する前に、db1.example.com データベースおよび db2.example.com データベースの両方に hr スキーマが存在しています。

この例では、単方向レプリケーションまたは双方向レプリケーションの構成方法を示しています。双方向レプリケーションを構成するには、追加の手順を実行して、構成プロシージャの実行時に bi_directional パラメータを TRUE に設定する必要があります。

図 4-11 に、この例で作成した環境の概要を示します。双方向レプリケーションに必要な追加のコンポーネントはグレーで表示され、そのアクションは破線で示されています。

図 4-11 ローカル取得プロセスを使用した 2 データベース・レプリケーション環境



2 データベース・レプリケーション環境を構成するには：

1. 2 データベース・レプリケーション環境を準備するには、次のタスクを完了します。

- db1.example.com データベースが db2.example.com データベースと通信できるように、ネットワーク接続を構成します。

データベース間のネットワーク接続の構成の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。

- レプリケーション環境に参加する各データベースで Oracle Streams 管理者を構成します。方法については、2-3 ページの「チュートリアル: Oracle Streams 管理者の作成」を参照してください。この例では、Oracle Streams 管理者が strmadmin であると想定します。

- c. db1.example.com データベースから db2.example.com データベースへのデータベース・リンクを作成します。
 データベース・リンクは、Oracle Streams 管理者のスキーマに作成する必要があります。また、データベース・リンクは別のデータベースで Oracle Streams 管理者に接続する必要があります。データベース・リンクの名前およびサービス名は、どちらも db2.example.com である必要があります。詳細は、2-8 ページの「チュートリアル: データベース・リンクの作成」を参照してください。
 - d. ARCHIVELOG モードで実行するように db1.example.com データベースを構成します。取得プロセスがソース・データベースで生成された変更を取得するには、ソース・データベースを ARCHIVELOG モードで実行する必要があります。ARCHIVELOG モードで実行するためのデータベースの構成の詳細は、『Oracle Database 管理者ガイド』を参照してください。
2. 双方向レプリケーション環境を構成するには、次の手順を実行します。単方向レプリケーション環境を構成する場合は、これらの手順は不要なため、手順 3 に進みます。
 - a. db2.example.com データベースから db1.example.com データベースへのデータベース・リンクを作成します。
 データベース・リンクは、Oracle Streams 管理者のスキーマに作成する必要があります。また、データベース・リンクは別のデータベースで Oracle Streams 管理者に接続する必要があります。データベース・リンクの名前およびサービス名は、どちらも db1.example.com である必要があります。詳細は、2-8 ページの「チュートリアル: データベース・リンクの作成」を参照してください。
 - b. ARCHIVELOG モードで実行するように db2.example.com データベースを構成します。取得プロセスがソース・データベースで生成された変更を取得するには、ソース・データベースを ARCHIVELOG モードで実行する必要があります。ARCHIVELOG モードで実行するためのデータベースの構成の詳細は、『Oracle Database 管理者ガイド』を参照してください。
 3. Oracle Streams レプリケーション環境に参加する各データベースで初期化パラメータを正しく設定します。方法については、4-21 ページの「Oracle Streams レプリケーションの準備」を参照してください。
 4. コマンドラインで SQL*Plus を開き、Oracle Streams 管理者として db2.example.com データベースに接続します。
 SQL*Plus の起動の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。
 5. MAINTAIN_SCHEMAS プロシージャで生成されるファイル（インスタンス化に使用するデータ・ポンプ・エクスポート・ダンプ・ファイルを含む）を保持するディレクトリ・オブジェクトを作成します。ディレクトリ・オブジェクトは、コンピュータ・システム上の任意のアクセス可能ディレクトリを指すことができます。たとえば、次の文は、/usr/db2_log_files ディレクトリを指す db2_dir という名前のディレクトリ・オブジェクトを作成します。

```
CREATE DIRECTORY db2_dir AS '/usr/db2_log_files';
```
 6. SQL*Plus で、db1.example.com データベースに Oracle Streams 管理者として接続します。
 SQL*Plus でのデータベースへの接続の詳細は、『Oracle Database 管理者ガイド』を参照してください。
 7. MAINTAIN_SCHEMAS プロシージャで生成されるファイル（インスタンス化に使用するデータ・ポンプ・エクスポート・ダンプ・ファイルを含む）を保持するディレクトリ・オブジェクトを作成します。ディレクトリ・オブジェクトは、コンピュータ・システム上の任意のアクセス可能ディレクトリを指すことができます。たとえば、次の文は、/usr/db1_log_files ディレクトリを指す db1_dir という名前のディレクトリ・オブジェクトを作成します。

```
CREATE DIRECTORY db1_dir AS '/usr/db1_log_files';
```

8. MAINTAIN_SCHEMAS プロシージャを実行して、db1.example.com データベースと db2.example.com データベース間で hr スキーマのレプリケーションを構成します。

bi_directional パラメータが、構成するレプリケーション環境用に適切に設定されていることを確認します。このパラメータは、単方向レプリケーションを構成する場合は FALSE に、双方向レプリケーションを構成する場合は TRUE に設定します。

```
BEGIN
  DBMS_STREAMS_ADM.MAINTAIN_SCHEMAS (
    schema_names          => 'hr',
    source_directory_object => 'db1_dir',
    destination_directory_object => 'db2_dir',
    source_database       => 'db1.example.com',
    destination_database  => 'db2.example.com',
    bi_directional        => FALSE); -- Set to TRUE for bi-directional
END;
/
```

MAINTAIN_SCHEMAS プロシージャは、多くの構成タスクを実行するため時間がかかることがあります。プロシージャの実行中は、宛先データベースのレプリケートされたデータベース・オブジェクトにデータ操作言語 (DML) またはデータ定義言語 (DDL) 変更が行われないようにしてください。4-15 ページの「[Oracle Streams レプリケーションの構成プロシージャの概要](#)」を参照してください。

構成プロシージャを実行すると、進捗情報が DBA_RECOVERABLE_SCRIPT、DBA_RECOVERABLE_SCRIPT_PARAMS、DBA_RECOVERABLE_SCRIPT_BLOCKS および DBA_RECOVERABLE_SCRIPT_ERRORS データ・ディクショナリ・ビューに記録されます。プロシージャがエラーによって停止した場合は、DBMS_STREAMS_ADM パッケージ内の RECOVER_OPERATION プロシージャの使用方法について、『[Oracle Streams レプリケーション管理者ガイド](#)』を参照してください。

9. 双方向レプリケーションを構成する場合は、両方のデータベースで hr スキーマ内のすべての表に対して最新時刻の競合解消を構成します。このスキーマには、countries、departments、employees、jobs、job_history、locations および regions 表が含まれます。方法については、4-55 ページの「[チュートリアル: 表の最新時刻競合解消の構成](#)」を参照してください。

例を完了すると、次の特性を持つ 2 データベース・レプリケーション環境が構成されます。

- db1.example.com では、サブリメンタル・ロギングが hr スキーマの表に対して構成されています。
- db1.example.com データベースには次のコンポーネントがあります。
 - システムで生成された名前を持つ取得プロセス。取得プロセスは hr スキーマの DML の変更を取得します。
 - システムで生成された名前を持つキュー。このキューは、データベースの取得プロセスに対するものです。
 - システムで生成された名前を持つ、変更を db1.example.com データベースにあるキューから db2.example.com データベースにあるキューに送信する伝播。
- db2.example.com データベースには次のコンポーネントがあります。
 - システムで生成された名前を持つ、db1.example.com データベースから送信された変更を受信するキュー。このキューは、ローカル・データベースの適用プロセスに対するものです。
 - システムで生成された名前を持つ適用プロセス。適用プロセスはキューから変更をデキューし、それを hr スキーマに適用します。

- 双方向レプリケーション環境の場合は、次のものも構成されます。
 - db2.example.com では、hr スキーマの表に対するサプリメンタル・ロギング。
 - db2.example.com では、システムで生成された名前を持つ取得プロセス。取得プロセスは hr スキーマへの DML 変更を取得します。
 - db2.example.com では、システムで生成された名前を持つキュー。このキューは、データベースの取得プロセスに対するものです。
 - db1.example.com では、システムで生成された名前を持つ、db2.example.com データベースから送信された変更を受信するキュー。このキューは、ローカル・データベースの適用プロセスに対するものです。
 - db1.example.com では、システムで生成された名前を持つ適用プロセス。適用プロセスはキューから変更をデキューし、それを hr スキーマに適用します。
- 双方向レプリケーション環境の場合は、変更の循環を回避するため、次のようにタグが使用されています。
 - 各適用プロセスは適用タグを使用し、適用プロセスによって適用される変更の REDO レコードにはタグが含まれます。各適用プロセスは、レプリケーション環境で一意な適用タグを使用します。
 - 各取得プロセスは、REDO レコード内のタグに関係なく、レプリケートされたデータベース・オブジェクトの変更すべてを取得します。したがって、各取得プロセスは、そのソース・データベースで適用プロセスによって適用された変更を取得します。
 - 各伝播は、レプリケートされたデータベース・オブジェクトに対する変更すべてを、レプリケーション環境のもう一方のデータベースに送信しますが、他のデータベースで発生した変更は除きます。伝播ルールでは、伝播がこれらの変更を破棄するよう指示します。

レプリケーション環境における変更の循環の回避については、4-10 ページの「[変更の循環を回避するためのタグの概要](#)」を参照してください。単方向レプリケーションを構成した場合は、変更は単一のロケーションでのみ取得されるため、変更の循環は発生しません。

Oracle Streams レプリケーション構成を確認するには :

1. db1.example.com データベースで、取得プロセスが有効化され、取得タイプがローカルであることを確認します。確認には、5-12 ページの「[取得プロセスの情報の表示](#)」の手順に従って、「取得」サブページで「ステータス」および「取得タイプ」フィールドを確認します。
2. db1.example.com データベースで、伝播が有効化されていることを確認します。確認には、5-16 ページの「[伝播の情報の表示](#)」の手順に従って、「伝播」サブページで「ステータス」フィールドを確認します。
3. db2.example.com データベースで、適用プロセスが有効化されていることを確認します。確認には、5-20 ページの「[適用プロセスの情報の表示](#)」の手順に従って、「適用」サブページで「ステータス」フィールドを確認します。
4. 双方向レプリケーションを構成した場合は、次の手順を実行します。
 - a. db2.example.com データベースで、取得プロセスが有効化され、取得タイプがローカルであることを確認します。
 - b. db2.example.com データベースで、伝播が有効化されていることを確認します。
 - c. db1.example.com データベースで、適用プロセスが有効化されていることを確認します。

変更をレプリケートするには :

1. hr スキーマに対する変更を取得するデータベースで、DML 変更を hr スキーマのすべての表に対して行います。この例では、db1.example.com データベースが hr スキーマに対する変更を取得しています。双方向レプリケーションを構成した場合は、db2.example.com も hr スキーマに対する変更を取得します。
2. 変更のレプリケーションが完了するまで待つから、SQL*Plus を使用して他のデータベースで変更された表を問い合わせる DML 変更を表示します。

注意： DBMS_STREAMS_ADM パッケージの構成プロシージャは、宛先データベースで読取り専用になるレプリケート表を構成しません。単方向レプリケーションでこれらの表を読取り専用にする必要がある場合、宛先データベースで権限を構成します。ただし、適用プロセスの適用ユーザーは、レプリケートされたデータベース・オブジェクトに対する DML 変更が可能である必要があります。この例では、適用ユーザーは Oracle Streams 管理者です。権限の構成の詳細は、『Oracle Database セキュリティ・ガイド』を参照してください。

参照：

- 「ハブアンドスポーク・レプリケーション環境の概要」 (4-12 ページ)
- 「Oracle Streams でのデータのレプリケート時期」 (1-5 ページ)
- 第 5 章 「Oracle Streams レプリケーション環境の管理」
- MAINTAIN_SCHEMAS プロシージャの使用の詳細は、『Oracle Streams レプリケーション管理者ガイド』を参照してください。
- MAINTAIN_SCHEMAS プロシージャの詳細は、『Oracle Database PL/SQL パッケージ・プロシージャおよびタイプ・リファレンス』を参照してください。

チュートリアル：ダウストリーム取得プロセスを使用した 2 データベース・レプリケーションの構成

この項の例では、hr スキーマにあるすべての表に、データ操作言語 (DML) 変更をレプリケートする Oracle Streams レプリケーション環境を構成します。この例では、宛先データベースでダウストリーム取得プロセスを使用して 2 データベース・レプリケーション環境を構成します。この例では、グローバル・データベース名 src.example.com および dest.example.com を使用します。ただし、例を完了するために、ご使用の環境のデータベースに置き換えることもできます。2 データベース・レプリケーション環境の詳細は、4-11 ページの「2 データベース・レプリケーション環境の概要」を参照してください。

この例では、ダウストリーム取得プロセスは宛先データベース dest.example.com で実行されます。したがって、変更の取得に必要なリソースはソース・データベース src.example.com で解放されます。この例では、アーカイブ・ログ・ダウストリーム取得プロセスではなく、リアルタイム・ダウストリーム取得プロセスを構成します。リアルタイム・ダウストリーム取得は、ソース・データベースでの変更の取得に必要な時間を短縮できるという利点があります。時間を短縮できるのは、データを取得する前に REDO ログ・ファイルがアーカイブされるまで待つ必要がないためです。

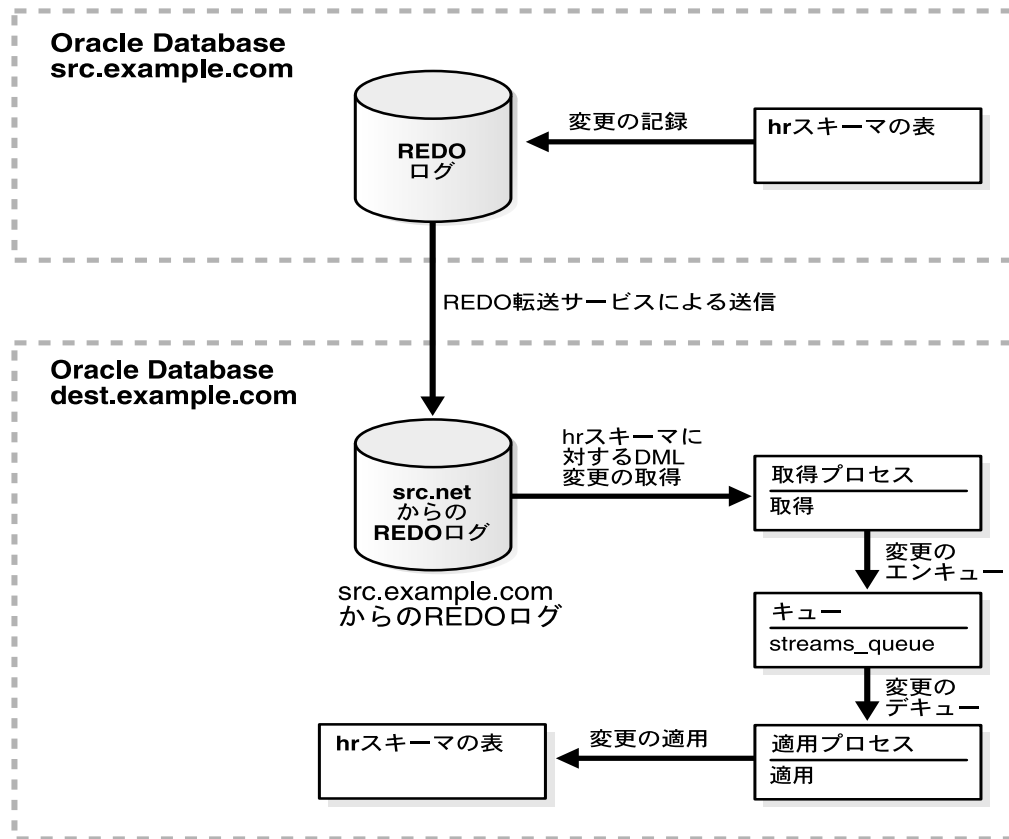
この例では、レプリケートされたデータベース・オブジェクトが宛先データベースでレポートおよび分析に使用されると想定しています。このため、これらのデータベース・オブジェクトは、dest.example.com データベースでは読取り専用であると想定しています。

この例では、DBMS_STREAMS_ADM パッケージの MAINTAIN_SCHEMAS プロシージャを使用して 2 データベース・レプリケーション環境を構成します。このプロシージャは、1 つ以上のスキーマをレプリケートする Oracle Streams 環境を構成する最速で最も単純な方法です。また、プロシージャは Oracle Streams レプリケーション環境に対して設定されたベスト・プラクティスに従います。

レプリケーション用に構成されているデータベース・オブジェクトは、MAINTAIN_SCHEMAS プロシージャの実行時に、宛先データベースに存在する場合としない場合があります。宛先データベースにデータベース・オブジェクトが存在しない場合、MAINTAIN_SCHEMAS プロシージャはデータ・ポンプ・エクスポート / インポートを使用して、宛先データベースでデータベース・オブジェクトをインスタンス化します。インスタンス化の際に、これらのデータベース・オブジェクトにインスタンス化 SCN が設定されます。宛先データベースにデータベース・オブジェクトが存在する場合、MAINTAIN_SCHEMAS プロシージャは既存のデータベース・オブジェクトを保持して、インスタンス化 SCN を設定します。この例では、MAINTAIN_SCHEMAS プロシージャを実行する前に、src.example.com データベースおよび dest.example.com データベースの両方に hr スキーマが存在しています。

図 4-12 に、この例で作成する環境の概要を示します。

図 4-12 ダウンストリーム取得プロセスを使用した 2 データベース・レプリケーション環境



注意： ローカル取得プロセスでは、ダウンストリーム取得プロセスに比べ、プラットフォームまたは Oracle Database のバージョンが異なるレプリケーション環境においてより高い柔軟性が提供されます。詳細は、『Oracle Streams 概要および管理』を参照してください。

2 データベース・レプリケーション環境を構成するには :

1. 2 データベース・レプリケーション環境を準備するには、次のタスクを完了します。

- a. `src.example.com` データベースと `dest.example.com` データベースが相互に通信できるように、ネットワーク接続を構成します。

データベース間のネットワーク接続の構成の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。

- b. レプリケーション環境に参加する各データベースで Oracle Streams 管理者を構成します。方法については、2-3 ページの「チュートリアル: Oracle Streams 管理者の作成」を参照してください。この例では、Oracle Streams 管理者が `strmadmin` であると想定します。

- c. ソース・データベースから宛先データベース、および宛先データベースからソース・データベースへのデータベース・リンクを作成します。この例では、次のデータベース・リンクを作成します。

- `src.example.com` データベースから `dest.example.com` データベースへ。
データベース・リンクの名前とサービス名は、どちらも `dest.example.com` である必要があります。
- `dest.example.com` データベースから `src.example.com` データベースへ。
データベース・リンクの名前とサービス名は、どちらも `src.example.com` である必要があります。

`src.example.com` は、`dest.example.com` データベースにおけるダウンストリーム取得プロセスのソース・データベースなので、`dest.example.com` データベースから `src.example.com` データベースへのデータベース・リンクが必要となります。このデータベース・リンクは取得プロセスの作成および構成を簡略化します。

各データベース・リンクは、Oracle Streams 管理者のスキーマに作成する必要があります。また、各データベース・リンクは、他のデータベースで Oracle Streams 管理者に接続する必要があります。方法については、2-8 ページの「チュートリアル: データベース・リンクの作成」を参照してください。

- d. Oracle Streams レプリケーション環境に参加する各データベースで初期化パラメータを正しく設定します。方法については、4-21 ページの「Oracle Streams レプリケーションの準備」を参照してください。
- e. ARCHIVELOG モードで実行するように両方のデータベースを構成します。ダウンストリーム取得プロセスがソース・データベースで生成された変更を取得するには、ソース・データベースとダウンストリーム取得データベースを ARCHIVELOG モードで実行する必要があります。この例では、`src.example.com` および `dest.example.com` データベースを ARCHIVELOG モードで実行する必要があります。ARCHIVELOG モードで実行するためのデータベースの構成の詳細は、『Oracle Database 管理者ガイド』を参照してください。
- f. REDO データの転送をサポートするために、両方の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・ファイルのいずれか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は、それをダウンストリーム取得データベース・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、ソース・データベースおよびダウンストリーム取得データベースで同じである必要があります。

この例で、ソース・データベースは `src.example.com` で、ダウンストリーム取得データベースは `dest.example.com` です。REDO 転送の認証要件の詳細は、『Oracle Data Guard 概要および管理』を参照してください。

2. ソース・データベース `src.example.com` で次の初期化パラメータを設定し、REDO データがソース・データベースのオンライン REDO ログからダウストリーム・データベース `dest.example.com` のスタンバイ REDO ログに転送されるように REDO 転送サービスを構成します。

- ソース・データベースで、REDO データがダウストリーム・データベースに転送されるように `LOG_ARCHIVE_DEST_n` 初期化パラメータを 1 つ以上構成します。これを行うには、このパラメータに次の属性を設定します。

- `SERVICE`: ダウストリーム・データベースのネットワーク・サービス名を指定します。
- `ASYNC` または `SYNC`: REDO 転送モードを指定します。

`ASYNC` を指定するメリットは、ソース・データベースのパフォーマンスにほとんど影響を与えないことです。ソース・データベースで Oracle Database 10g リリース 1 以上が実行されていて、ダウストリーム・データベースまたはネットワークのパフォーマンスが低いときは、`ASYNC` を使用してソース・データベースのパフォーマンスへの影響を回避することをお勧めします。

`SYNC` を指定するメリットは、`ASYNC` を指定する場合と比べて、ダウストリーム・データベースに REDO データが速く送信されることです。また、`SYNC AFFIRM` を指定すると、`MAXIMUM AVAILABILITY` スタンバイ保護モードの場合と同様に動作するようになります。SQL 文 `ALTER DATABASE STANDBY DATABASE TO MAXIMIZE AVAILABILITY` を指定しても、Oracle Streams 取得プロセスには影響しないことに注意してください。

- `NOREGISTER`: この属性は、アーカイブ REDO ログ・ファイルの場所がダウストリーム・データベース制御ファイルに記録されないように指定します。
- `VALID_FOR`: (`ONLINE_LOGFILE, PRIMARY_ROLE`) または (`ONLINE_LOGFILE, ALL_ROLES`) を指定します。
- `DB_UNIQUE_NAME`: ダウストリーム・データベースの一意の名前です。ダウストリーム・データベースで `DB_UNIQUE_NAME` 初期化パラメータに指定した名前を使用します。

次に、ダウストリーム・データベースを指定する `LOG_ARCHIVE_DEST_n` 設定の例を示します。

```
LOG_ARCHIVE_DEST_2='SERVICE=DEST.EXAMPLE.COM ASYNC NOREGISTER
VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE)
DB_UNIQUE_NAME=dest'
```

- `LOG_ARCHIVE_DEST_STATE_n`: ソース・データベースで、ダウストリーム・データベースの `LOG_ARCHIVE_DEST_n` パラメータに対応する初期化パラメータを `ENABLE` に設定します。

たとえば、ダウストリーム・データベースに `LOG_ARCHIVE_DEST_2` 初期化パラメータが設定されている場合、`LOG_ARCHIVE_DEST_STATE_2` パラメータを次のように設定します。

```
LOG_ARCHIVE_DEST_STATE_2=ENABLE
```

- `LOG_ARCHIVE_CONFIG`: この初期化パラメータの `DB_CONFIG` 属性を設定して、ソース・データベースおよびダウストリーム・データベースの `DB_UNIQUE_NAME` を含めます。

たとえば、ソース・データベースの `DB_UNIQUE_NAME` が `src` で、ダウストリーム・データベースの `DB_UNIQUE_NAME` が `dest` の場合、次のパラメータを指定します。

```
LOG_ARCHIVE_CONFIG='DG_CONFIG=(src,dest)'
```


デフォルトでは、LOG_ARCHIVE_CONFIG パラメータによって、データベースでの REDO の送受信が可能になります。

参照: 初期化パラメータの詳細は、『Oracle Database リファレンス』および『Oracle Data Guard 概要および管理』を参照してください。

3. ダウンストリーム・データベース `dest.example.com` で、次の初期化パラメータを設定し、ローカルに生成される REDO データのアーカイブを構成します。

- LOG_ARCHIVE_DEST_n 初期化パラメータで、少なくとも1つのアーカイブ・ログの保存先を、ダウンストリーム・データベースを実行しているコンピュータ・システム上のディレクトリまたはフラッシュ・リカバリ領域に設定します。これを設定するには、このパラメータの次の属性を設定します。
 - LOCATION: ディスク・ディレクトリへの有効なパス名または USE_DB_RECOVERY_FILE_DEST を指定します。LOCATION 属性に指定する各宛先に、一意のディレクトリ・パス名または USE_DB_RECOVERY_FILE_DEST を指定する必要があります。これは、ローカル・データベースで生成されるアーカイブ REDO ログ・ファイルのローカルの宛先です。
 - VALID_FOR: (ONLINE_LOGFILE, PRIMARY_ROLE) または (ONLINE_LOGFILE, ALL_ROLES) を指定します。

次の例は、リアルタイム・ダウンストリーム取得データベースでローカルに生成される REDO データに対する LOG_ARCHIVE_DEST_n 設定です。

```
LOG_ARCHIVE_DEST_1='LOCATION=/home/arc_dest/local_r1
VALID_FOR=(ONLINE_LOGFILE,PRIMARY_ROLE)'
```

リアルタイム・ダウンストリーム取得構成では、アーカイブ・スタンバイ REDO ログ・ファイルをダウンストリーム・データベースからのアーカイブ・オンライン REDO ログ・ファイルとは別に保持する必要があります。これを実現するには、VALID_FOR 属性の REDO ログ・タイプに ALL_LOGFILES ではなくて ONLINE_LOGFILE を指定します。

必要に応じて、LOG_ARCHIVE_DEST_n 初期化パラメータの他の属性も指定できます。

- この手順で設定済の LOG_ARCHIVE_DEST_n パラメータに対応する LOG_ARCHIVE_DEST_STATE_n 初期化パラメータを ENABLE に設定します。たとえば、LOG_ARCHIVE_DEST_1 初期化パラメータが設定されている場合、LOG_ARCHIVE_DEST_STATE_1 パラメータを次のように設定します。

```
LOG_ARCHIVE_DEST_STATE_1=ENABLE
```

4. ダウンストリーム・データベース `dest.example.com` で、ダウンストリーム・データベースがソース・データベースから REDO データを受信し、ダウンストリーム・データベースのスタンバイ REDO ログにその REDO データを書き込むように、次の初期化パラメータを設定します。

- LOG_ARCHIVE_DEST_n 初期化パラメータで、少なくとも1つのアーカイブ・ログの保存先を、ダウンストリーム・データベースを実行しているコンピュータ・システム上のディレクトリまたはフラッシュ・リカバリ領域に設定します。これを設定するには、このパラメータの次の属性を設定します。
 - LOCATION: ディスク・ディレクトリへの有効なパス名または USE_DB_RECOVERY_FILE_DEST を指定します。LOCATION 属性に指定する各宛先に、一意のディレクトリ・パス名または USE_DB_RECOVERY_FILE_DEST を指定する必要があります。これは、スタンバイ REDO ログから書き込まれたアーカイブ REDO ログ・ファイルのローカルの宛先です。リモート・ソース・データベースからのログ・ファイルは、ローカル・データベースのログ・ファイルとは別に保持する必要があります。
 - VALID_FOR: (STANDBY_LOGFILE, PRIMARY_ROLE) または (STANDBY_LOGFILE, ALL_ROLES) を指定します。

次の例は、リアルタイム・ダウストリーム取得データベースでの LOG_ARCHIVE_DEST_n 設定です。

```
LOG_ARCHIVE_DEST_2='LOCATION=/home/arc_dest/sr1_src1
VALID_FOR=(STANDBY_LOGFILE,PRIMARY_ROLE)'
```

必要に応じて、LOG_ARCHIVE_DEST_n 初期化パラメータの他の属性も指定できます。

- この手順で設定済の LOG_ARCHIVE_DEST_n パラメータに対応する LOG_ARCHIVE_DEST_STATE_n 初期化パラメータを ENABLE に設定します。
たとえば、ダウストリーム・データベースに LOG_ARCHIVE_DEST_2 初期化パラメータが設定されている場合、LOG_ARCHIVE_DEST_STATE_2 パラメータを次のように設定します。

```
LOG_ARCHIVE_DEST_STATE_2=ENABLE
```

- LOG_ARCHIVE_CONFIG: この初期化パラメータの DB_CONFIG 属性を設定して、ソース・データベースおよびダウストリーム・データベースの DB_UNIQUE_NAME を含めます。

たとえば、ソース・データベースの DB_UNIQUE_NAME が src で、ダウストリーム・データベースの DB_UNIQUE_NAME が dest の場合、次のパラメータを指定します。

```
LOG_ARCHIVE_CONFIG='DG_CONFIG=(src,dest)'
```

デフォルトでは、LOG_ARCHIVE_CONFIG パラメータによって、データベースでの REDO の送受信が可能になります。

5. 手順 2、3 または 4 で、データベースのインスタンスが実行中に初期化パラメータをリセットした場合、該当する初期化パラメータ・ファイルでもそれをリセットすれば、データベースの再起動時に新しい値が維持されます。

手順 2、3 または 4 で、インスタンスの実行中には初期化パラメータをリセットしなかったが、初期化パラメータ・ファイルでリセットした場合は、データベースを再起動します。ソース・データベースのグローバル名は、ソース・データベースがオープンな場合にのみダウストリーム・データベースに送信されるため、ダウストリーム・データベースに REDO データを送信するときには、ソース・データベースがオープンである必要があります。

6. ダウストリーム・データベース dest.example.com で、管理ユーザーとして接続してスタンバイ REDO ログ・ファイルを作成します。

注意： 次に示すのは、スタンバイ REDO ログ・ファイルをダウストリーム・データベースに追加する一般的な手順の概略です。スタンバイ REDO ログ・ファイルの追加に使用する実際の手順と SQL 文は、環境によって異なります。たとえば、Oracle Real Application Clusters (Oracle RAC) 環境では手順が異なります。スタンバイ REDO ログ・ファイルをデータベースに追加する手順の詳細は、『Oracle Data Guard 概要および管理』を参照してください。

- a. SQL*Plus を開き、管理ユーザーとしてソース・データベース src.example.com に接続します。

たとえば、管理ユーザーとして src.example.com データベースに接続するには、次のようにします。

```
sqlplus system@src.example.com
Enter password: password
```

SQL*Plus の起動の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。

- b. ソース・データベース `src.example.com` で使用するログ・ファイルのサイズを決定します。スタンバイ・ログ・ファイルのサイズは、ソース・データベースのログ・ファイル・サイズと正確に同じかそれより大きくなるようにします。たとえば、ソース・データベースのログ・ファイル・サイズが `500MB` の場合、スタンバイ・ログ・ファイルのサイズも `500MB` 以上にします。ソース・データベースにおける REDO ログ・ファイルのサイズ (バイト単位) は、ソース・データベースで `V$LOG` ビューを問い合わせて確認することができます。

たとえば、次のように `V$LOG` ビューを問い合わせます。

```
SELECT BYTES FROM V$LOG;
```

- c. ダウンストリーム・データベース `dest.example.com` で必要なスタンバイ・ログ・ファイル・グループの数を決定します。スタンバイ・ログ・ファイル・グループの数は、ソース・データベースでのオンライン・ログ・ファイル・グループの数より 1 以上大きくする必要があります。たとえば、ソース・データベースに 2 つのオンライン・ログ・ファイル・グループがある場合、ダウンストリーム・データベースには少なくとも 3 つのスタンバイ・ログ・ファイル・グループが必要です。ソース・データベースのオンライン・ログ・ファイル・グループの数は、ソース・データベースで `V$LOG` ビューを問い合わせて確認することができます。

たとえば、`SQL*Plus` で管理ユーザーとしてまだ `src.example.com` に接続している状態で、次のように `V$LOG` ビューを問い合わせます。

```
SELECT COUNT(GROUP#) FROM V$LOG;
```

- d. `SQL*Plus` で、管理ユーザーとしてダウンストリーム・データベース `dest.example.com` に接続します。

```
CONNECT system@dest.example.com
Enter password: password
```

- e. `SQL` 文、`ALTER DATABASE ADD STANDBY LOGFILE` を使用して、ダウンストリーム・データベース `dest.example.com` にスタンバイ・ログ・ファイル・グループを追加します。

たとえば、ソース・データベースに 2 つのオンライン REDO ログ・ファイル・グループがあり、ログ・ファイルのサイズが `500MB` であるとしします。この場合、次の文を使用して適切なスタンバイ・ログ・ファイル・グループを作成します。

```
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/slog4a.rdo', '/oracle/dbs/slog4b.rdo') SIZE 500M;
```

```
ALTER DATABASE ADD STANDBY LOGFILE GROUP 5
  ('/oracle/dbs/slog5a.rdo', '/oracle/dbs/slog5b.rdo') SIZE 500M;
```

- f. `dest.example.com` で次の問合せを実行し、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;
```

出力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

- g. ソース・データベースのログ・ファイルが、手順 4 で LOCATION 属性に指定したディレクトリに表示されることを確認します。ディレクトリ内のファイルを表示するために、ソース・データベースにあるログ・ファイルを切り替える必要がある場合があります。
7. SQL*Plus で、src.example.com データベースに Oracle Streams 管理者として接続します。
- SQL*Plus でのデータベースへの接続の詳細は、『Oracle Database 管理者ガイド』を参照してください。
8. MAINTAIN_SCHEMAS プロシージャで生成されるファイル（インスタンス化に使用するデータ・ポンプ・エクスポート・ダンプ・ファイルを含む）を保持するディレクトリ・オブジェクトを作成します。ディレクトリ・オブジェクトは、コンピュータ・システム上の任意のアクセス可能ディレクトリを指すことができます。たとえば、次の文は、/usr/src_log_files ディレクトリを指す src_dir という名前のディレクトリ・オブジェクトを作成します。
- ```
CREATE DIRECTORY src_dir AS '/usr/src_log_files';
```
9. SQL\*Plus で、dest.example.com データベースに Oracle Streams 管理者として接続します。
10. MAINTAIN\_SCHEMAS プロシージャで生成されるファイル（インスタンス化に使用するデータ・ポンプ・エクスポート・ダンプ・ファイルを含む）を保持するディレクトリ・オブジェクトを作成します。ディレクトリ・オブジェクトは、コンピュータ・システム上の任意のアクセス可能ディレクトリを指すことができます。たとえば、次の文は、/usr/dest\_log\_files ディレクトリを指す dest\_dir という名前のディレクトリ・オブジェクトを作成します。
- ```
CREATE DIRECTORY dest_dir AS '/usr/dest_log_files';
```
11. dest.example.com データベースにまだ Oracle Streams 管理者として接続しているときに、MAINTAIN_SCHEMAS プロシージャを実行して src.example.com データベースと dest.example.com データベース間のレプリケーションを構成します。

```
BEGIN
  DBMS_STREAMS_ADM.MAINTAIN_SCHEMAS (
    schema_names           => 'hr',
    source_directory_object => 'src_dir',
    destination_directory_object => 'dest_dir',
    source_database        => 'src.example.com',
    destination_database   => 'dest.example.com',
    capture_name           => 'capture',
    capture_queue_table    => 'streams_queue_qt',
    capture_queue_name     => 'streams_queue',
    apply_name             => 'apply',
    apply_queue_table      => 'streams_queue_qt',
    apply_queue_name       => 'streams_queue');
END;
/
```

MAINTAIN_SCHEMAS プロシージャは、多くの構成タスクを実行するため時間がかかることがあります。プロシージャの実行中は、宛先データベースのレプリケートされたデータベース・オブジェクトにデータ操作言語 (DML) またはデータ定義言語 (DDL) 変更が行われないようにしてください。

MAINTAIN_SCHEMAS プロシージャに必要なパラメータは、schema_names、source_directory_object、destination_directory_object、source_database および destination_database です。4-15 ページの「[Oracle Streams レプリケーションの構成プロシージャの概要](#)」を参照してください。

この例では、取得プロセス、取得プロセスのキュー表、取得プロセスのキュー、適用プロセス、適用プロセスのキュー表および適用プロセスのキューの名前を選択できることを示すために他のパラメータを指定しています。これらのパラメータを指定しないと、システムで生成された名前が使用されます。

構成プロシージャを使用してダウンストリーム取得を構成する際は、キューおよびキュー表の名前を指定するパラメータは重要です。このような構成では、キュー間で変更が伝播されないようにするために、取得プロセスおよび適用プロセスがダウンストリーム取得プロセスで同じキューを使用するとより効率的です。このサンプル構成の効率を向上させるために、`capture_queue_name` および `apply_queue_name` パラメータの両方に `streams_queue` が指定されていることに注意してください。また、`capture_queue_table` および `apply_queue_table` パラメータの両方に `streams_queue qt` が指定されています。

構成プロシージャを実行すると、進捗情報が `DBA_RECOVERABLE_SCRIPT`、`DBA_RECOVERABLE_SCRIPT_PARAMS`、`DBA_RECOVERABLE_SCRIPT_BLOCKS` および `DBA_RECOVERABLE_SCRIPT_ERRORS` データ・ディクショナリ・ビューに記録されます。プロシージャがエラーによって停止した場合は、`DBMS_STREAMS_ADM` パッケージ内の `RECOVER_OPERATION` プロシージャの使用方法について、『Oracle Streams レプリケーション管理者ガイド』を参照してください。

プロシージャが正常に完了してから、次の手順を実行します。

12. `dest.example.com` データベースにまだ Oracle Streams 管理者として接続しているときに、`downstream_real_time_mine` 取得プロセスのパラメータを `Y` に設定します。

```
BEGIN
  DBMS_CAPTURE_ADM.SET_PARAMETER(
    capture_name => 'capture',
    parameter    => 'downstream_real_time_mine',
    value        => 'Y');
END;
/
```

取得プロセスのパラメータの設定に Enterprise Manager を使用する方法については、5-3 ページの「取得プロセスのパラメータの設定」を参照してください。

13. SQL*Plus で、管理ユーザーとしてソース・データベース `src.example.com` に接続します。
14. ソース・データベースで現行のログ・ファイルを次のようにアーカイブします。

```
ALTER SYSTEM ARCHIVE LOG CURRENT;
```

ソース・データベースで現行のログ・ファイルをアーカイブすると、ソース・データベースの REDO ログのリアルタイム・マイニングが開始されます。

例を完了すると、次の特性を持つ 2 データベース・レプリケーション環境が構成されます。

- `src.example.com` データベースでは、サブリメンタル・ロギングが `hr` スキーマの表に対して構成されています。
- `dest.example.com` データベースには次のコンポーネントがあります。
 - `capture` という名前のダウンストリーム取得プロセス。この取得プロセスは、ソース・データベース `src.example.com` から送信される REDO ログ情報で、`hr` スキーマに対する変更を取得します。
 - `streams_queue qt` という名前のキュー表を使用する `streams_queue` という名前のキュー。このキューは、データベースの取得プロセスおよび適用プロセスに対するものです。
 - `apply` という名前の適用プロセス。この適用プロセスは、変更を `hr` スキーマに適用します。

Oracle Streams レプリケーション構成を確認するには :

1. `dest.example.com` データベースで、取得プロセスが有効化され、取得タイプがダウンストリームであることを確認します。確認には、5-12 ページの「[取得プロセスの情報の表示](#)」の手順に従って、「取得」サブページで「ステータス」および「取得タイプ」フィールドを確認します。
2. `dest.example.com` データベースで、適用プロセスが有効化されていることを確認します。確認には、5-20 ページの「[適用プロセスの情報の表示](#)」の手順に従って、「適用」サブページで「ステータス」フィールドを確認します。

変更をレプリケートするには :

1. `src.example.com` データベースで、`hr` スキーマの任意の表に対して DML 変更を行った後、変更をコミットします。
2. 変更のレプリケーションが完了するまで待つから、SQL*Plus を使用して `dest.example.com` データベースで変更された表を問い合わせ DML 変更を表示します。

注意： DBMS_STREAMS_ADM パッケージの構成プロシージャは、宛先データベースで読取り専用になるレプリケート表を構成しません。これらの表を読取り専用にする必要がある場合、宛先データベースで権限を構成します。ただし、適用プロセスの適用ユーザーは、レプリケートされたデータベース・オブジェクトに対する DML 変更が可能である必要があります。この例では、適用ユーザーは Oracle Streams 管理者です。権限の構成の詳細は、『Oracle Database セキュリティ・ガイド』を参照してください。

参照：

- 「[ハブアンドスポーク・レプリケーション環境の概要](#)」 (4-12 ページ)
- 「[Oracle Streams でのデータのレプリケート時期](#)」 (1-5 ページ)
- [第 5 章「Oracle Streams レプリケーション環境の管理」](#)
- ダウンストリーム取得プロセスの詳細は、『Oracle Streams 概要および管理』を参照してください。
- MAINTAIN_SCHEMAS プロシージャの使用の詳細は、『Oracle Streams レプリケーション管理者ガイド』を参照してください。
- MAINTAIN_SCHEMAS プロシージャの詳細は、『Oracle Database PL/SQL パッケージ・プロシージャおよびタイプ・リファレンス』を参照してください。

チュートリアル: ローカル取得プロセスでのハブアンドスポーク・レプリケーションの構成

この項の例では、hr スキーマにあるすべての表に、データ操作言語 (DML) 変更をレプリケートする Oracle Streams ハブアンドスポーク・レプリケーション環境を構成します。この例では、各データベースで取得プロセスを使用してこれらの変更を取得します。ハブアンドスポーク・レプリケーションとは、中央のハブ・データベースが 1 つ以上のスポーク・データベースでの変更をレプリケートすることを意味します。スポーク・データベースは、相互に直接通信しません。このサンプル構成では、ハブ・データベースが、1 つのスポーク・データベースで生成された変更を他のスポーク・データベースに送信します。ハブアンドスポーク・レプリケーション環境の詳細は、4-12 ページの「ハブアンドスポーク・レプリケーション環境の概要」を参照してください。

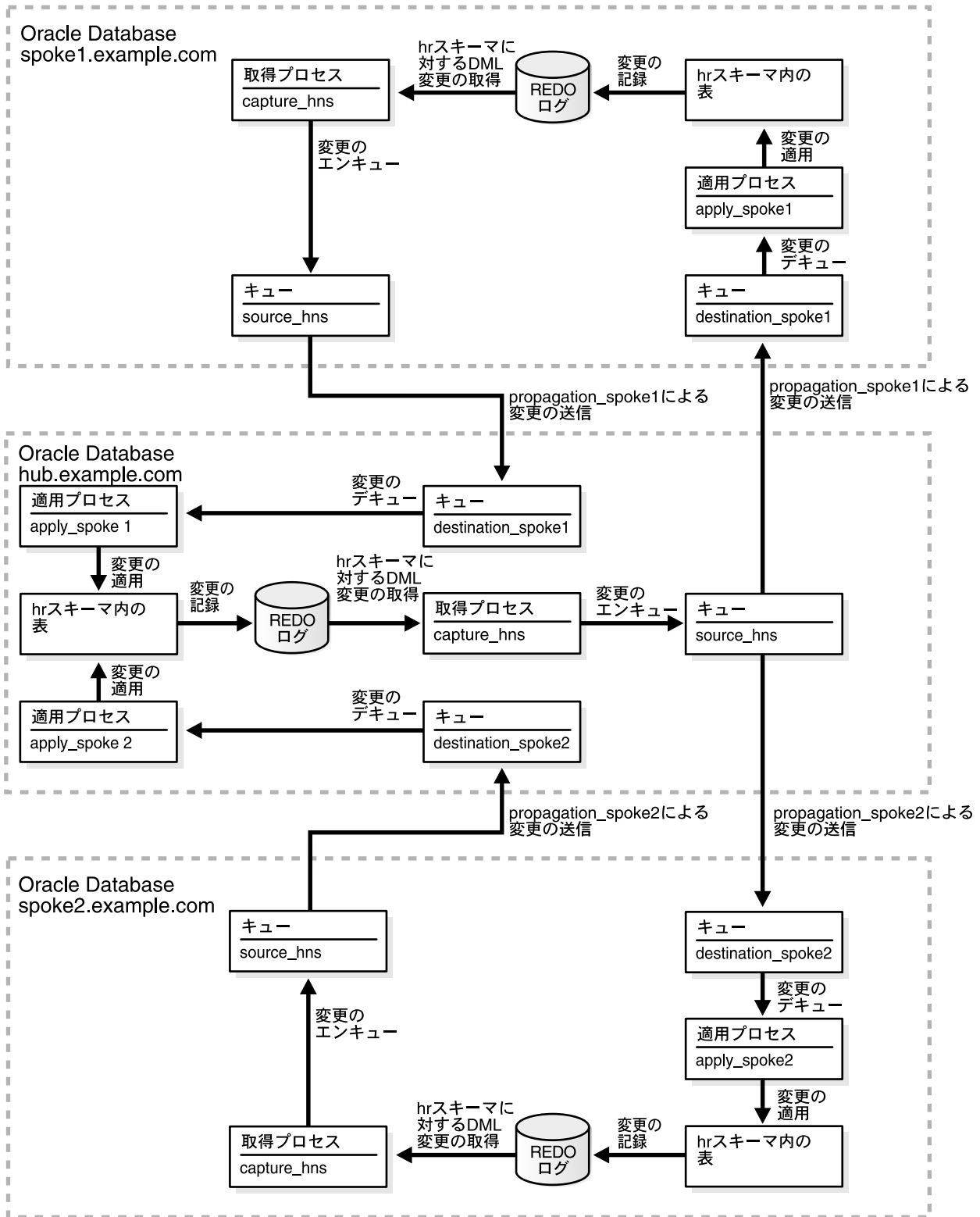
この例では、DBMS_STREAMS_ADM パッケージの MAINTAIN_SCHEMAS プロシージャを使用してハブアンドスポーク・レプリケーション環境を構成します。このプロシージャは、1 つ以上のスキーマをレプリケートする Oracle Streams 環境を構成する最速で最も単純な方法です。また、プロシージャは Oracle Streams レプリケーション環境に対して設定されたベスト・プラクティスに従います。

この例では、ハブ・データベースのグローバル名は hub.example.com、スポーク・データベースのグローバル名は spoke1.example.com および spoke2.example.com です。ただし、使用している環境のデータベースをかわりに使用して例を完了することができます。

レプリケーション用に構成されているデータベース・オブジェクトは、MAINTAIN_SCHEMAS プロシージャの実行時に、宛先データベースに存在する場合としない場合があります。宛先データベースにデータベース・オブジェクトが存在しない場合、MAINTAIN_SCHEMAS プロシージャはデータ・ポンプ・エクスポート / インポートを使用して、宛先データベースでデータベース・オブジェクトをインスタンス化します。インスタンス化の際に、これらのデータベース・オブジェクトにインスタンス化 SCN が設定されます。宛先データベースにデータベース・オブジェクトが存在する場合、MAINTAIN_SCHEMAS プロシージャは既存のデータベース・オブジェクトを保持して、インスタンス化 SCN を設定します。この例では、MAINTAIN_SCHEMAS プロシージャを実行する前に、各データベースに hr スキーマが存在しています。

図 4-13 に、この例で作成する環境の概要を示します。

図 4-13 取得プロセスと読取り/書き込みスポークを持つハブアンドスポーク環境の例



読取り/書き込みスポークのあるこのハブアンドスポーク・レプリケーション環境を構成するには:

1. ハブアンドスポーク・レプリケーション環境を準備するには、次のタスクを完了します。
 - a. 次のデータベースが相互に通信できるように、ネットワーク接続を構成します。
 - hub.example.com データベースと spoke1.example.com データベース
 - hub.example.com データベースと spoke2.example.com データベース
 データベース間のネットワーク接続の構成の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。
 - b. レプリケーション環境に参加する各データベースで Oracle Streams 管理者を構成します。方法については、2-3 ページの「チュートリアル: Oracle Streams 管理者の作成」を参照してください。この例では、Oracle Streams 管理者が strmadmin であると想定します。
 - c. ハブ・データベースから各スポーク・データベースおよび各スポーク・データベースからハブ・データベースへのデータベース・リンクを作成します。この例では、次のデータベース・リンクを作成します。
 - hub.example.com データベースから spoke1.example.com データベースへ。データベース・リンクの名前とサービス名は、どちらも spoke1.example.com である必要があります。
 - hub.example.com データベースから spoke2.example.com データベースへ。データベース・リンクの名前とサービス名は、どちらも spoke2.example.com である必要があります。
 - spoke1.example.com データベースから hub.example.com データベースへ。データベース・リンクの名前とサービス名は、どちらも hub.example.com である必要があります。
 - spoke2.example.com データベースから hub.example.com データベースへ。データベース・リンクの名前とサービス名は、どちらも hub.example.com である必要があります。
 各データベース・リンクは、Oracle Streams 管理者のスキーマに作成する必要があります。また、各データベース・リンクは宛先データベースで Oracle Streams 管理者に接続する必要があります。方法については、2-8 ページの「チュートリアル: データベース・リンクの作成」を参照してください。
 - d. Oracle Streams レプリケーション環境に参加する各データベースで初期化パラメータを正しく設定します。方法については、4-21 ページの「Oracle Streams レプリケーションの準備」を参照してください。
 - e. ARCHIVELOG モードで実行するように各ソース・データベースを構成します。取得プロセスがソース・データベースで生成された変更を取得するためには、ソース・データベースを ARCHIVELOG モードで実行する必要があります。この例では、すべてのデータベースを ARCHIVELOG モードで実行する必要があります。ARCHIVELOG モードで実行するためのデータベースの構成の詳細は、『Oracle Database 管理者ガイド』を参照してください。
2. コマンドラインで SQL*Plus を開き、Oracle Streams 管理者として spoke1.example.com データベースに接続します。

SQL*Plus の起動の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。

3. MAINTAIN_SCHEMAS プロシージャで生成されるファイル（インスタンス化に使用するデータ・ポンプ・エクスポート・ダンプ・ファイルを含む）を保持するディレクトリ・オブジェクトを作成します。ディレクトリ・オブジェクトは、コンピュータ・システム上の任意のアクセス可能ディレクトリを指すことができます。たとえば、次の文は、`/usr/spoke1_log_files` ディレクトリを指す `spoke1_dir` という名前のディレクトリ・オブジェクトを作成します。

```
CREATE DIRECTORY spoke1_dir AS '/usr/spoke1_log_files';
```

4. SQL*Plus で、`spoke2.example.com` データベースに Oracle Streams 管理者として接続します。

SQL*Plus でのデータベースへの接続の詳細は、『Oracle Database 管理者ガイド』を参照してください。

5. MAINTAIN_SCHEMAS プロシージャで生成されるファイル（インスタンス化に使用するデータ・ポンプ・エクスポート・ダンプ・ファイルを含む）を保持するディレクトリ・オブジェクトを作成します。ディレクトリ・オブジェクトは、コンピュータ・システム上の任意のアクセス可能ディレクトリを指すことができます。たとえば、次の文は、`/usr/spoke2_log_files` ディレクトリを指す `spoke2_dir` という名前のディレクトリ・オブジェクトを作成します。

```
CREATE DIRECTORY spoke2_dir AS '/usr/spoke2_log_files';
```

6. SQL*Plus で、`hub.example.com` データベースに Oracle Streams 管理者として接続します。

7. MAINTAIN_SCHEMAS プロシージャで生成されるファイル（インスタンス化に使用するデータ・ポンプ・エクスポート・ダンプ・ファイルを含む）を保持するディレクトリ・オブジェクトを作成します。ディレクトリ・オブジェクトは、コンピュータ・システム上の任意のアクセス可能ディレクトリを指すことができます。たとえば、次の文は、`/usr/hub_log_files` ディレクトリを指す `hub_dir` という名前のディレクトリ・オブジェクトを作成します。

```
CREATE DIRECTORY hub_dir AS '/usr/hub_log_files';
```

8. SQL*Plus で `hub.example.com` データベースにまだ Streams 管理者として接続しているときに、MAINTAIN_SCHEMAS プロシージャを実行して `hub.example.com` データベースと `spoke1.example.com` データベース間のレプリケーションを構成します。

```
BEGIN
  DBMS_STREAMS_ADM.MAINTAIN_SCHEMAS (
    schema_names          => 'hr',
    source_directory_object => 'hub_dir',
    destination_directory_object => 'spoke1_dir',
    source_database       => 'hub.example.com',
    destination_database  => 'spoke1.example.com',
    capture_name          => 'capture_hns',
    capture_queue_table   => 'source_hns_qt',
    capture_queue_name    => 'source_hns',
    propagation_name     => 'propagation_spoke1',
    apply_name            => 'apply_spoke1',
    apply_queue_table     => 'destination_spoke1_qt',
    apply_queue_name      => 'destination_spoke1',
    bi_directional        => TRUE);
END;
/
```

MAINTAIN_SCHEMAS プロシージャは、多くの構成タスクを実行するため時間がかかることがあります。プロシージャの実行中は、宛先データベースのレプリケートされたデータベース・オブジェクトにデータ操作言語（DML）またはデータ定義言語（DDL）変更が行われないようにしてください。

MAINTAIN_SCHEMAS プロシージャに必要なパラメータは、`schema_names`、`source_directory_object`、`destination_directory_object`、`source_database` および `destination_database` です。また、構成プロシージャを使用して双方向レプリケーションを構成する際は、`bi_directional` パラメータを TRUE に設定する必要があります。4-15 ページの「[Oracle Streams レプリケーションの構成プロシージャの概要](#)」を参照してください。

この例では、取得プロセス、取得プロセスのキュー表、取得プロセスのキュー、伝播、適用プロセス、適用プロセスのキュー表および適用プロセスのキューの名前を選択できることを示すために他のパラメータを指定しています。これらのパラメータを指定しないと、システムで生成された名前が使用されます。

構成プロシージャを実行すると、進捗情報が `DBA_RECOVERABLE_SCRIPT`、`DBA_RECOVERABLE_SCRIPT_PARAMS`、`DBA_RECOVERABLE_SCRIPT_BLOCKS` および `DBA_RECOVERABLE_SCRIPT_ERRORS` データ・ディクショナリ・ビューに記録されます。プロシージャがエラーによって停止した場合は、`DBMS_STREAMS_ADM` パッケージ内の `RECOVER_OPERATION` プロシージャの使用方法について、『[Oracle Streams レプリケーション管理者ガイド](#)』を参照してください。

9. SQL*Plus で `hub.example.com` データベースにまだ Streams 管理者として接続しているときに、`MAINTAIN_SCHEMAS` プロシージャを実行して `hub.example.com` データベースと `spoke2.example.com` データベース間のレプリケーションを構成します。

```
BEGIN
  DBMS_STREAMS_ADM.MAINTAIN_SCHEMAS (
    schema_names           => 'hr',
    source_directory_object => 'hub_dir',
    destination_directory_object => 'spoke2_dir',
    source_database         => 'hub.example.com',
    destination_database    => 'spoke2.example.com',
    capture_name            => 'capture_hns',
    capture_queue_table     => 'source_hns_qt',
    capture_queue_name      => 'source_hns',
    propagation_name        => 'propagation_spoke2',
    apply_name              => 'apply_spoke2',
    apply_queue_table       => 'destination_spoke2_qt',
    apply_queue_name        => 'destination_spoke2',
    bi_directional          => TRUE);
END;
/
```

10. `hub.example.com`、`spoke1.example.com`、`spoke2.example.com` の各データベースで `hr` スキーマ内のすべての表に対して最新時刻競合解消を構成します。このスキーマには、`countries`、`departments`、`employees`、`jobs`、`job_history`、`locations` および `regions` の各表が含まれます。方法については、4-55 ページの「[チュートリアル: 表の最新時刻競合解消の構成](#)」を参照してください。

例を完了すると、次の特性を持つハブアンドスポーク・レプリケーション環境が構成されます。

- 各データベースで、サプリメンタル・ロギングが `hr` スキーマの表に対して構成されています。
- 各データベースには、`capture_hns` という名前の取得プロセスがあります。取得プロセスは、データベースの `hr` スキーマに対する変更を取得します。
- 各データベースには、`source_hns_qt` という名前のキュー表を使用する `source_hns` という名前のキューがあります。このキューは、データベースの取得プロセスに対するものです。
- ハブ・データベース `hub.example.com` には、次の追加コンポーネントがあります。
 - `apply_spoke1` という名前の適用プロセス。この適用プロセスは、`spoke1.example.com` データベースから送信された `hr` スキーマに対する変更を適用します。

- destination_spoke1_qt という名前のキュー表を使用する destination_spoke1 という名前のキュー。このキューは、データベースの apply_spoke1 適用プロセスに対するものです。
 - apply_spoke2 という名前の適用プロセス。この適用プロセスは、spoke2.example.com データベースから送信された hr スキーマに対する変更を適用します。
 - destination_spoke2_qt という名前のキュー表を使用する destination_spoke2 という名前のキュー。このキューは、データベースの apply_spoke2 適用プロセスに対するものです。
 - propagation_spoke1 という名前の伝播。この伝播は、hr スキーマに対する変更を source_hns キューから spoke1.example.com データベースの destination_spoke1 キューに送信します。
 - propagation_spoke2 という名前の伝播。この伝播は、hr スキーマに対する変更を source_hns キューから spoke2.example.com データベースの destination_spoke2 キューに送信します。
- スポーク・データベース spoke1.example.com には、次の追加コンポーネントがありません。
 - apply_spoke1 という名前の適用プロセス。この適用プロセスは、hub.example.com データベースから送信された hr スキーマに対する変更を適用します。
 - destination_spoke1_qt という名前のキュー表を使用する destination_spoke1 という名前のキュー。このキューは、データベースの apply_spoke1 適用プロセスに対するものです。
 - propagation_spoke1 という名前の伝播。この伝播は、hr スキーマに対する変更を source_hns キューから hub.example.com データベースの destination_spoke1 キューに送信します。
- スポーク・データベース spoke2.example.com には、次の追加コンポーネントがありません。
 - apply_spoke2 という名前の適用プロセス。この適用プロセスは、hub.example.com データベースから送信された hr スキーマに対する変更を適用します。
 - destination_spoke2_qt という名前のキュー表を使用する destination_spoke2 という名前のキュー。このキューは、データベースの apply_spoke2 適用プロセスに対するものです。
 - propagation_spoke2 という名前の伝播。この伝播は、hr スキーマに対する変更を source_hns キューから hub.example.com データベースの destination_spoke2 キューに送信します。
- 変更の循環を回避するため、次のようにタグが使用されています。
 - 各適用プロセスは適用タグを使用し、適用プロセスによって適用される変更の REDO レコードにはタグが含まれます。各適用プロセスは、レプリケーション環境で一意的な適用タグを使用します。
 - 各取得プロセスは、REDO レコード内のタグに関係なく、レプリケートされたデータベース・オブジェクトの変更すべてを取得します。したがって、各取得プロセスは、そのソース・データベースで適用プロセスによって適用された変更を取得します。
 - 各伝播は、レプリケートされたデータベース・オブジェクトに対する変更すべてを、レプリケーション環境の他のデータベースに送信しますが、他のデータベースで発生した変更は除きます。伝播ルールでは、伝播がこれらの変更を破棄するよう指示しません。

レプリケーション環境で変更の循環を回避する方法の詳細は、4-10 ページの「[変更の循環を回避するためのタグの概要](#)」および『[Oracle Database PL/SQL パッケージ・プロシージャおよびタイプ・リファレンス](#)』を参照してください。

Oracle Streams レプリケーション構成を確認するには :

1. 各データベースで、取得プロセスが有効化され、取得タイプがローカルであることを確認します。確認には、5-12 ページの「[取得プロセスの情報の表示](#)」の手順に従って、「取得」サブページの「ステータス」および「取得タイプ」フィールドをチェックします。
2. 各データベースで、伝播が有効化されていることを確認します。確認には、5-16 ページの「[伝播の情報の表示](#)」の手順に従って、「伝播」サブページで「ステータス」フィールドを確認します。ハブ・データベースには2つの伝播が必要で、両方の伝播が有効化されている必要があります。各スポーク・データベースには、有効化されている1つの伝播が必要です。
3. 各データベースで、適用プロセスが有効化されていることを確認します。確認には、5-20 ページの「[適用プロセスの情報の表示](#)」の手順に従って、「適用」サブページで「ステータス」フィールドを確認します。ハブ・データベースには2つの適用プロセスが必要で、両方の適用プロセスが有効化されている必要があります。各スポーク・データベースには、有効化されている1つの適用プロセスが必要です。

変更をレプリケートするには :

1. データベースの1つで、DML 変更を hr スキーマの任意の表に対して行います。
2. 変更のレプリケーションが完了するまで待ってから、SQL*Plus を使用して他のデータベースで変更された表を問い合せて DML 変更を表示します。

参照 :

- 「[ハブアンドスポーク・レプリケーション環境の概要](#)」 (4-12 ページ)
- 「[Oracle Streams でのデータのレプリケート時期](#)」 (1-5 ページ)
- [第5章「Oracle Streams レプリケーション環境の管理」](#)
- MAINTAIN_SCHEMAS プロシージャの使用の詳細は、『[Oracle Streams レプリケーション管理者ガイド](#)』を参照してください。
- MAINTAIN_SCHEMAS プロシージャの詳細は、『[Oracle Database PL/SQL パッケージ・プロシージャおよびタイプ・リファレンス](#)』を参照してください。

チュートリアル : 同期取得を使用した2データベース・レプリケーションの構成

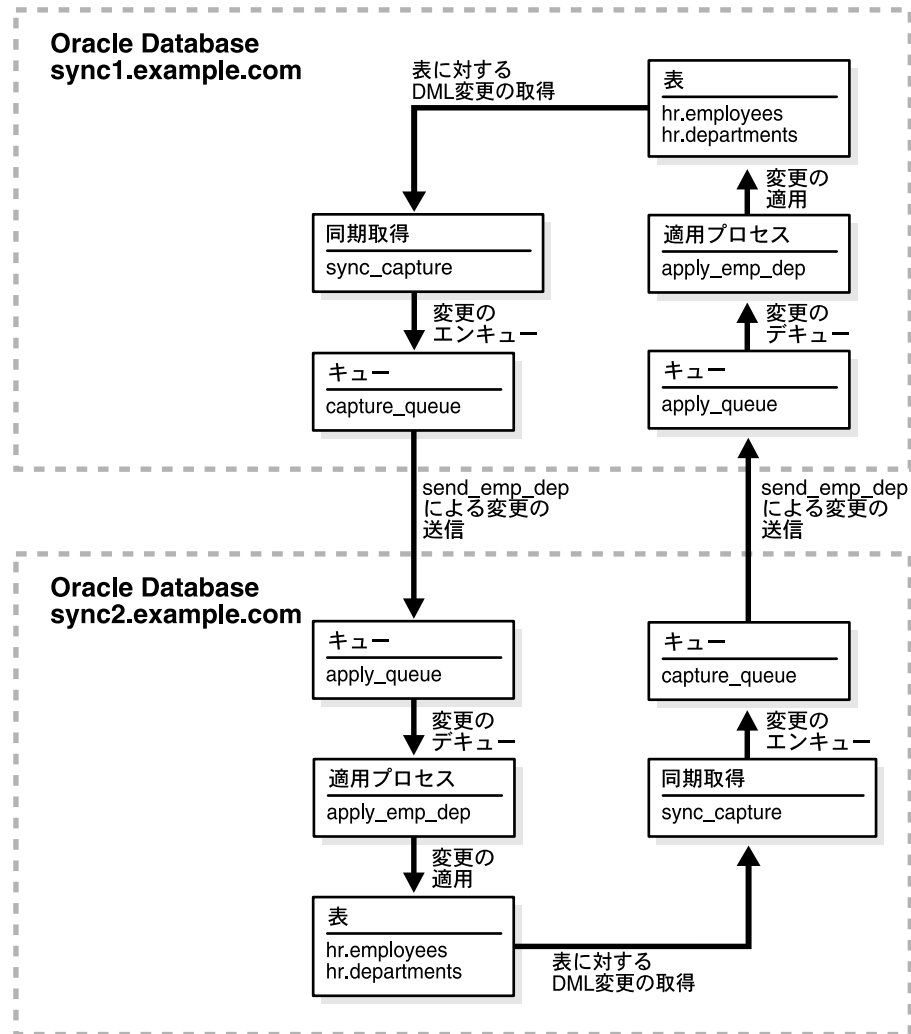
この項の例では、hr スキーマにある2つの表に、データ操作言語 (DML) 変更をレプリケートする Oracle Streams レプリケーション環境を構成します。この例では、各データベースで同期取得を使用して変更を取得します。この例では、Oracle Streams レプリケーション環境で、データベースのグローバル名に sync1.example.com および sync2.example.com を使用します。ただし、例を完了するために、ご使用の環境の2つのデータベースに置き換えることもできます。2データベース・レプリケーション環境の詳細は、4-11 ページの「[2データベース・レプリケーション環境の概要](#)」を参照してください。

具体的に、この例では、sync1.example.com および sync2.example.com データベースで hr.employees および hr.departments 表を共有する Oracle Streams の2データベース・レプリケーション環境を構成します。2つのデータベースは、すべての DML 変更をこれらの表にレプリケートします。hr サンプル・スキーマは、Oracle Database に付属してデフォルトでインストールされます。

注意： 同期取得は、表レベルでのみ変更を取得できます。スキーマ・レベルまたはデータベース・レベルでの変更の取得はできません。同期取得は、DBMS_STREAMS_ADM パッケージの ADD_TABLE_RULES および ADD_SUBSET_RULES プロシージャを使用して構成できます。

図 4-14 に、この例で作成する環境の概要を示します。

図 4-14 同期取得を使用した 2 データベース・レプリケーション環境



このレプリケーション環境を同期取得で構成するには：

1. 2 データベース・レプリケーション環境を準備するには、次のタスクを完了します。
 - a. ネットワーク接続を構成して、2つのデータベースが相互に通信できるようにします。データベース間のネットワーク接続の構成の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。
 - b. レプリケーション環境に参加する各データベースで Oracle Streams 管理者を構成します。方法については、2-3 ページの「チュートリアル: Oracle Streams 管理者の作成」を参照してください。この例では、Oracle Streams 管理者が strmadmin であると想定します。
 - c. Oracle Streams レプリケーション環境に参加する各データベースで初期化パラメータを正しく設定します。方法については、4-21 ページの「Oracle Streams レプリケーションの準備」を参照してください。
 - d. hr.employees および hr.departments 表が 2 つのデータベースに存在し、これらのデータベースで一貫していることを確認します。データベース・オブジェクトが 1 つのデータベースにのみ存在する場合は、エクスポート / インポートを使用してこれらを他のデータベースで作成および移入できます。エクスポート / インポートの詳細は、『Oracle Database ユーティリティ』を参照してください。

2. 各データベースに2つの ANYDATA キューを作成します。この例では、各データベースに次の2つのキューを作成します。

- Oracle Streams 管理者 `strmadmin` が所有する、`capture_queue` という名前のキュー。このキューはデータベースでの同期取得によって使用されます。
- Streams 管理者 `strmadmin` が所有する、`apply_queue` という名前のキュー。このキューはデータベースでの適用プロセスによって使用されます。

方法については、2-7 ページの「[ANYDATA キューの作成](#)」を参照してください。

3. 各データベースから他のデータベースへのデータベース・リンクを作成します。
- a. `sync1.example.com` データベースから `sync2.example.com` データベースに、データベース・リンクを作成します。データベース・リンクは、Oracle Streams 管理者のスキーマに作成する必要があります。また、データベース・リンクは `sync2.example.com` データベースで Oracle Streams 管理者に接続する必要があります。データベース・リンクの名前およびサービス名は、どちらも `sync2.example.com` である必要があります。
 - b. `sync2.example.com` データベースから `sync1.example.com` データベースに、データベース・リンクを作成します。データベース・リンクは、Oracle Streams 管理者のスキーマに作成する必要があります。また、データベース・リンクは `sync1.example.com` データベースで Oracle Streams 管理者に接続する必要があります。データベース・リンクの名前およびサービス名は、どちらも `sync1.example.com` である必要があります。

方法については、2-8 ページの「[チュートリアル: データベース・リンクの作成](#)」を参照してください。

4. `sync1.example.com` データベースで適用プロセスを構成します。この適用プロセスは、`sync2.example.com` データベースで取得され、`sync1.example.com` データベースに伝播された共有表への変更を適用します。

- a. SQL*Plus を開き、`sync1.example.com` データベースに Oracle Streams 管理者として接続します。

SQL*Plus の起動の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。

- b. 適用プロセスを作成します。

```
BEGIN
  DBMS_APPLY_ADM.CREATE_APPLY(
    queue_name      => 'strmadmin.apply_queue',
    apply_name      => 'apply_emp_dep',
    apply_captured => FALSE);
END;
/
```

適用プロセスは永続キュー内の変更を適用するため、`apply_captured` パラメータが `FALSE` に設定されます。これらは、同期取得によって取得された変更です。`apply_captured` パラメータは、適用プロセスが取得プロセスによって取得された変更を適用する場合にのみ `TRUE` に設定する必要があります。

適用プロセスは起動しないでください。

- c. 適用プロセス・ルール・セットにルールを追加します。

```
BEGIN
  DBMS_STREAMS_ADM.ADD_TABLE_RULES(
    table_name      => 'hr.employees',
    streams_type    => 'apply',
    streams_name    => 'apply_emp_dep',
    queue_name      => 'strmadmin.apply_queue',
    source_database => 'sync2.example.com');
END;
/
```

このルールは、`apply_queue` キューに出現する `hr.employees` 表に対するすべての DML 変更を適用するよう適用プロセス `apply_emp_dep` に指示します。ルールは、適用プロセスが `sync2.example.com` ソース・データベースで取得された変更のみ適用することも指定します。

- d. 適用プロセス・ルール・セットにルールを追加します。

```
BEGIN
  DBMS_STREAMS_ADM.ADD_TABLE_RULES(
    table_name      => 'hr.departments',
    streams_type    => 'apply',
    streams_name    => 'apply_emp_dep',
    queue_name      => 'strmadmin.apply_queue',
    source_database => 'sync2.example.com');
END;
/
```

このルールは、`apply_queue` キューに出現する `hr.departments` 表に対するすべての DML 変更を適用するよう適用プロセス `apply_emp_dep` に指示します。ルールは、適用プロセスが `sync2.example.com` ソース・データベースで取得された変更のみ適用することも指定します。

5. `sync2.example.com` データベースで適用プロセスを構成します。この適用プロセスは、`sync1.example.com` データベースで取得され、`sync2.example.com` データベースに伝播された変更を適用します。

- a. SQL*Plus で、`sync2.example.com` データベースに Oracle Streams 管理者として接続します。

SQL*Plus でのデータベースへの接続の詳細は、『Oracle Database 管理者ガイド』を参照してください。

- b. 適用プロセスを作成します。

```
BEGIN
  DBMS_APPLY_ADM.CREATE_APPLY(
    queue_name      => 'strmadmin.apply_queue',
    apply_name      => 'apply_emp_dep',
    apply_captured => FALSE);
END;
/
```

適用プロセスは永続キュー内の変更を適用するため、`apply_captured` パラメータが `FALSE` に設定されます。これらの変更は、同期取得によって取得されました。`apply_captured` パラメータは、適用プロセスが取得プロセスによって取得された変更を適用する場合にのみ `TRUE` に設定する必要があります。

適用プロセスは起動しないでください。

- c. 適用プロセス・ルール・セットにルールを追加します。

```
BEGIN
  DBMS_STREAMS_ADM.ADD_TABLE_RULES(
    table_name      => 'hr.employees',
    streams_type    => 'apply',
    streams_name    => 'apply_emp_dep',
    queue_name      => 'strmadmin.apply_queue',
    source_database => 'sync1.example.com');
END;
/
```

このルールは、`apply_queue` キューに出現するすべての DML 変更を `hr.employees` 表に適用するよう適用プロセス `apply_emp_dep` に指示します。ルールは、適用プロセスが `sync1.example.com` ソース・データベースで取得された変更のみ適用することも指定します。

- d. 適用プロセス・ルール・セットにルールを追加します。

```
BEGIN
  DBMS_STREAMS_ADM.ADD_TABLE_RULES(
    table_name      => 'hr.departments',
    streams_type    => 'apply',
    streams_name    => 'apply_emp_dep',
    queue_name      => 'strmadmin.apply_queue',
    source_database => 'sync1.example.com');
END;
/
```

このルールは、`apply_queue` キューに出現するすべての DML 変更を `hr.departments` 表に適用するよう適用プロセス `apply_emp_dep` に指示します。ルールは、適用プロセスが `sync1.example.com` ソース・データベースで取得された変更のみ適用することも指定します。

6. `sync1.example.com` データベースのキューから `sync2.example.com` データベースのキューに変更を送信する伝播を作成します。

- a. SQL*Plus で、`sync1.example.com` データベースに Oracle Streams 管理者として接続します。
- b. `sync2.example.com` データベースに対する変更を送信する伝播の作成を作成します。

```
BEGIN
  DBMS_STREAMS_ADM.ADD_TABLE_PROPAGATION_RULES(
    table_name      => 'hr.employees',
    streams_name    => 'send_emp_dep',
    source_queue_name => 'strmadmin.capture_queue',
    destination_queue_name => 'strmadmin.apply_queue@sync2.example.com',
    source_database => 'sync1.example.com',
    queue_to_queue  => TRUE);
END;
/
```

`ADD_TABLE_PROPAGATION_RULES` プロシージャは、伝播およびポジティブ・ルール・セットを作成します。このプロシージャは、`sync2.example.com` データベース内の `apply_queue` キューの `hr.employees` 表に DML 変更を送信するよう指示する伝播ルール・セットにルールの追加もします。

- c. 伝播ルール・セットにルールを追加します。

```
BEGIN
  DBMS_STREAMS_ADM.ADD_TABLE_PROPAGATION_RULES(
    table_name      => 'hr.departments',
    streams_name    => 'send_emp_dep',
    source_queue_name => 'strmadmin.capture_queue',
    destination_queue_name => 'strmadmin.apply_queue@sync2.example.com',
    source_database => 'sync1.example.com',
    queue_to_queue  => TRUE);
END;
/
```

`ADD_TABLE_PROPAGATION_RULES` プロシージャは、`sync2.example.com` データベース内の `apply_queue` キューの `hr.departments` 表に DML 変更を送信するよう指示する伝播ルール・セットにルールを追加します。

7. sync2.example.com データベースのキューから sync1.example.com データベースのキューに変更を送信する伝播を作成します。

- a. SQL*Plus で、sync2.example.com データベースに Oracle Streams 管理者として接続します。
- b. sync1.example.com データベースに対する変更を送信する伝播の作成を作成します。

```
BEGIN
  DBMS_STREAMS_ADM.ADD_TABLE_PROPAGATION_RULES (
    table_name          => 'hr.employees',
    streams_name        => 'send_emp_dep',
    source_queue_name   => 'strmadmin.capture_queue',
    destination_queue_name => 'strmadmin.apply_queue@sync1.example.com',
    source_database     => 'sync2.example.com',
    queue_to_queue      => TRUE);
END;
/
```

ADD_TABLE_PROPAGATION_RULES プロシージャは、伝播およびポジティブ・ルール・セットを作成します。このプロシージャでは、sync1.example.com データベース内の apply_queue キューの hr.employees 表に DML 変更を送信するよう指示する伝播ルール・セットにルールを追加します。

- c. 伝播ルール・セットにルールを追加します。

```
BEGIN
  DBMS_STREAMS_ADM.ADD_TABLE_PROPAGATION_RULES (
    table_name          => 'hr.departments',
    streams_name        => 'send_emp_dep',
    source_queue_name   => 'strmadmin.capture_queue',
    destination_queue_name => 'strmadmin.apply_queue@sync1.example.com',
    source_database     => 'sync2.example.com',
    queue_to_queue      => TRUE);
END;
/
```

ADD_TABLE_PROPAGATION_RULES プロシージャは、sync1.example.com データベース内の apply_queue キューの hr.departments 表に DML 変更を送信するよう指示する伝播ルール・セットにルールを追加します。

8. sync1.example.com データベースで同期取得を構成します。

- a. SQL*Plus で、sync1.example.com データベースに Oracle Streams 管理者として接続します。
- b. ADD_TABLE_RULES プロシージャを実行して、同期取得を作成し、hr.employees 表に対する変更を取得するよう指示するルールを追加します。

```
BEGIN
  DBMS_STREAMS_ADM.ADD_TABLE_RULES (
    table_name      => 'hr.employees',
    streams_type    => 'sync_capture',
    streams_name    => 'sync_capture',
    queue_name      => 'strmadmin.capture_queue');
END;
/
```

- c. 同期取得ルール・セットにルールを追加します。

```
BEGIN
  DBMS_STREAMS_ADM.ADD_TABLE_RULES(
    table_name    => 'hr.departments',
    streams_type  => 'sync_capture',
    streams_name  => 'sync_capture',
    queue_name    => 'strmadmin.capture_queue');
END;
/
```

これらのプロシージャを実行して次のアクションを実行します。

- 現在のデータベースで `sync_capture` という名前の同期取得を作成します。同じ名前の同期取得は存在できません。
- 同期取得を有効にします。同期取得は無効にできません。
- `strmadmin` が所有する `capture_queue` という名前の既存のキューで同期取得を関連付けます。
- 同期取得 `sync_capture` のポジティブ・ルール・セットを作成します。ルール・セットにはシステムで作成された名前があります。
- `hr.employees` 表に対する DML 変更を取得するルールを作成し、同期取得のポジティブ・ルール・セットにルールを追加します。ルールにはシステムで生成された名前があります。
- 表に対して `DBMS_CAPTURE_ADM.PREPARE_SYNC_INSTANTIATION` ファンクションを自動的に実行することで、インスタンス化用に `hr.employees` 表を準備します。
- `hr.departments` 表に対する DML 変更を取得するルールを作成し、同期取得のポジティブ・ルール・セットにルールを追加します。ルールにはシステムで生成された名前があります。
- 表に対して `DBMS_CAPTURE_ADM.PREPARE_SYNC_INSTANTIATION` ファンクションを自動的に実行することで、インスタンス化用に `hr.departments` 表を準備します。

9. `sync2.example.com` データベースで同期取得を構成します。

- a. SQL*Plus で、`sync2.example.com` データベースに Oracle Streams 管理者として接続します。
- b. `ADD_TABLE_RULES` プロシージャを実行して、同期取得を作成し、`hr.employees` 表に対する変更を取得するよう指示するルールを追加します。

```
BEGIN
  DBMS_STREAMS_ADM.ADD_TABLE_RULES(
    table_name    => 'hr.employees',
    streams_type  => 'sync_capture',
    streams_name  => 'sync_capture',
    queue_name    => 'strmadmin.capture_queue');
END;
/
```

- c. 同期取得ルール・セットにルールを追加します。

```
BEGIN
  DBMS_STREAMS_ADM.ADD_TABLE_RULES(
    table_name    => 'hr.departments',
    streams_type  => 'sync_capture',
    streams_name  => 'sync_capture',
    queue_name    => 'strmadmin.capture_queue');
END;
/
```

手順 8 では、これらのプロシージャによって現在のデータベースで実行されるアクションについて説明します。

10. sync2.example.com データベースで表のインスタンス化 SCN を設定します。
- SQL*Plus で、sync1.example.com データベースに Oracle Streams 管理者として接続します。
 - sync2.example.com データベースで hr.employees 表のインスタンス化 SCN を設定します。

```
DECLARE
    iscn NUMBER;    -- Variable to hold instantiation SCN value
BEGIN
    iscn := DBMS_FLASHBACK.GET_SYSTEM_CHANGE_NUMBER();
    DBMS_APPLY_ADM.SET_TABLE_INSTANTIATION_SCN@sync2.example.com(
        source_object_name => 'hr.employees',
        source_database_name => 'sync1.example.com',
        instantiation_scn   => iscn);
END;
/
```

- sync2.example.com データベースで hr.departments 表のインスタンス化 SCN を設定します。

```
DECLARE
    iscn NUMBER;    -- Variable to hold instantiation SCN value
BEGIN
    iscn := DBMS_FLASHBACK.GET_SYSTEM_CHANGE_NUMBER();
    DBMS_APPLY_ADM.SET_TABLE_INSTANTIATION_SCN@sync2.example.com(
        source_object_name => 'hr.departments',
        source_database_name => 'sync1.example.com',
        instantiation_scn   => iscn);
END;
/
```

インスタンス化 SCN は、適用プロセスが表に変更を適用できる最小の SCN です。適用プロセスが sync2.example.com データベースの共有表への変更を適用する前に、インスタンス化 SCN を各表に対して設定する必要があります。

11. sync1.example.com データベースで表のインスタンス化 SCN を設定します。
- SQL*Plus で、sync2.example.com データベースに Oracle Streams 管理者として接続します。
 - sync1.example.com データベースで hr.employees 表のインスタンス化 SCN を設定します。

```
DECLARE
    iscn NUMBER;    -- Variable to hold instantiation SCN value
BEGIN
    iscn := DBMS_FLASHBACK.GET_SYSTEM_CHANGE_NUMBER();
    DBMS_APPLY_ADM.SET_TABLE_INSTANTIATION_SCN@sync1.example.com(
        source_object_name => 'hr.employees',
        source_database_name => 'sync2.example.com',
        instantiation_scn   => iscn);
END;
/
```

- c. sync2.example.com データベースで hr.departments 表のインスタンス化 SCN を設定します。

```
DECLARE
  iscn NUMBER;    -- Variable to hold instantiation SCN value
BEGIN
  iscn := DBMS_FLASHBACK.GET_SYSTEM_CHANGE_NUMBER();
  DBMS_APPLY_ADM.SET_TABLE_INSTANTIATION_SCN@sync1.example.com(
    source_object_name => 'hr.departments',
    source_database_name => 'sync2.example.com',
    instantiation_scn => iscn);
END;
/
```

12. 各データベースで適用プロセスを起動します。

- a. SQL*Plus で、sync1.example.com データベースに Oracle Streams 管理者として接続します。
- b. 適用プロセスを開始します。

```
BEGIN
  DBMS_APPLY_ADM.START_APPLY(
    apply_name => 'apply_emp_dep');
END;
/
```

- c. SQL*Plus で、sync2.example.com データベースに Oracle Streams 管理者として接続します。
- d. 適用プロセスを開始します。

```
BEGIN
  DBMS_APPLY_ADM.START_APPLY(
    apply_name => 'apply_emp_dep');
END;
/
```

適用プロセスの開始に Enterprise Manager を使用する方法については、5-6 ページの「[適用プロセスの起動と停止](#)」を参照してください。

13. sync1.example.com および sync2.example.com の各データベースの hr.departments および hr.employees の各表に対して、最新時刻競合解消を構成します。方法については、4-55 ページの「[チュートリアル: 表の最新時刻競合解消の構成](#)」を参照してください。

次のような特性を持つ 2 データベース・レプリケーション環境が構成されます。

- 各データベースには、sync_capture という名前の同期取得があります。この同期取得は、hr.employees 表および hr.departments 表に対するすべての DML 変更を取得します。
- 各データベースには、capture_queue という名前のキューがあります。このキューは、データベースでの同期取得用です。
- 各データベースには、apply_emp_dep という名前の適用プロセスがあります。この適用プロセスは、hr.employees 表および hr.departments 表に対するすべての DML 変更を適用します。
- 各データベースには、apply_queue という名前のキューがあります。このキューは、データベースでの適用プロセス用です。

- 各データベースには、send_emp_dep という名前の伝播があります。この伝播は、ローカル・データベース内の capture_queue から、他のデータベース内の apply_queue に変更を送信します。この伝播では、hr.employees 表および hr.departments 表に対するすべての DML 変更が送信されます。
 - 変更の循環を回避するため、次のようにタグが使用されています。
 - 各適用プロセスでは、デフォルトの適用タグを使用します。デフォルトの適用タグは、'00' (ゼロが2つ) と等価の 16 進数になります。
 - 各同期取得は、NULL タグを含むセッションの変更のみを取得します。したがって、同期取得は、ローカル適用プロセスによって適用される変更は取得しません。同期取得のルールは、これらの変更を取得しないよう同期取得に指示します。
- レプリケーション環境で変更の循環を回避する方法の詳細は、4-10 ページの「[変更の循環を回避するためのタグの概要](#)」を参照してください。

Oracle Streams レプリケーション構成を確認するには :

1. 各データベースで、次の手順を実行して同期取得が構成されることを確認します。
 - a. SQL*Plus を起動して、データベースに Oracle Streams 管理者として接続します。
SQL*Plus の起動の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。
 - b. 次のように ALL_SYNC_CAPTURE データ・ディクショナリ・ビューを問い合わせます。

```
SELECT CAPTURE_NAME FROM ALL_SYNC_CAPTURE;
```


各データベースに sync_capture という名前の同期取得が存在していることを確認します。
2. 各データベースで、伝播が有効化されていることを確認します。確認には、5-16 ページの「[伝播の情報の表示](#)」の手順に従って、「伝播」サブページで「ステータス」を確認します。
3. 各データベースで、適用プロセスが有効化されていることを確認します。確認には、5-20 ページの「[適用プロセスの情報の表示](#)」の手順に従って、「適用」サブページで「ステータス」を確認します。

変更をレプリケートするには :

1. データベースの1つで、hr.employees 表または hr.departments 表に対して DML 変更を行います。
2. 変更のレプリケーションが完了するまで待つから、SQL*Plus を使用して他のデータベースの hr.employees 表または hr.departments 表を問い合わせて変更を表示します。

参照 :

- 「[同期取得での変更の取得の概要](#)」 (4-5 ページ)
- 「[ハブアンドスポーク・レプリケーション環境の概要](#)」 (4-12 ページ)
- 「[Oracle Streams でのデータのレプリケート時期](#)」 (1-5 ページ)
- [第5章「Oracle Streams レプリケーション環境の管理」](#)

チュートリアル : 表の最新時刻競合解消の構成

レプリケーション環境では、競合解消によって競合が自動的に解消されます。競合解消の詳細は、4-9 ページの「[競合と競合解消の概要](#)」を参照してください。

更新競合を解消する最も一般的な方法は、最新のタイムスタンプを持つ変更を保持し、古い変更を破棄することです。この方法では、適用中に競合が検出された場合、適用プロセスは、変更のタイムスタンプ列が表の対応する行よりも新しい場合に変更を適用します。表のタイムスタンプ列が最新の場合、適用プロセスは変更を破棄します。

この項の例では、次のアクションを完了することにより、`hr.departments` 表の最新時刻競合解消を構成します。

- `TIMESTAMP WITH TIME ZONE` データ型の `time` 列を表に追加する
- 行が変更されたときに最新の時刻を持つ行の時間列を更新するようにトリガーを構成する
- 表の列にサブリメンタル・ロギングを追加する
- `DBMS_APPLY_ADM` パッケージの `SET_UPDATE_CONFLICT_HANDLER` プロシージャを実行して、表の競合解消を構成する

この項の手順を使用して、任意の表の競合解消を構成できます。これを構成するには、`hr` をスキーマ名で置換し、`departments` を表名で置換します。また、`SET_UPDATE_CONFLICT_HANDLER` プロシージャを実行するときに、`hr.departments` 表の列を、使用する表の列で置換します。

`hr.departments` 表の最新時刻競合解消を構成するには :

1. 表に `time` 列を追加します。
 - a. `SQL*Plus` で、Oracle Streams 管理者や `SYSTEM` などの管理ユーザーとしてデータベースに接続します。別の方法として、時間列が追加される表を所有しているユーザーとして接続することもできます。

`SQL*Plus` の起動の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。
 - b. `SQL` 文 `ALTER TABLE` を使用して、表に `time` 列を追加します。この例では、次の文を実行して `time` 列を `hr.departments` 表に追加します。


```
ALTER TABLE hr.departments ADD (time TIMESTAMP WITH TIME ZONE);
```

2. トリガーを作成して変更が発生した時刻で各マスター表の `time` 列を更新します。

ヒント: トリガーを使用して `time` 列を更新するかわりに、アプリケーションで表に対して列を修正または挿入するたびに、`time` 列を移入できません。

- a. Oracle Enterprise Manager で、`SYSTEM` または Oracle Streams 管理者などの管理者ユーザーとしてデータベースにログインします。
- b. 「データベース」ホームページに移動します。
- c. 「スキーマ」をクリックして「スキーマ」サブページを開きます。
- d. 「プログラム」セクションで「トリガー」をクリックします。

- e. 「トリガー」 ページで、「作成」 をクリックします。
「トリガーの作成」 ページが表示され、「一般」 サブページが表示されます。

Database Instance: database > Triggers > Create Trigger

Logged in As STRMADMIN

General Event Advanced

* Name

* Schema STRMADMIN

Replace If Exists Enable

* Trigger Body

General Event Advanced

Show SQL Cancel OK

- f. 「名前」 フィールドにトリガーの名前を入力します。この例では、insert_departments_time を入力します。
- g. 表を所有しているスキーマを「スキーマ」 フィールドに入力します。この例では、「スキーマ」 フィールドには hr と入力します。
- h. 「トリガー本体」 フィールドに次のように入力します。
- ```
BEGIN
 -- Consider time synchronization problems. The previous update to this
 -- row might have originated from a site with a clock time ahead of
 -- the local clock time.
 IF :OLD.TIME IS NULL OR :OLD.TIME < SYSTIMESTAMP THEN
 :NEW.TIME := SYSTIMESTAMP;
 ELSE
 :NEW.TIME := :OLD.TIME + 1 / 86400;
 END IF;
END;
```
- i. 「イベント」 をクリックして「イベント」 サブページを開きます。
- j. 「対象オブジェクト」 リストで「表」 が選択されていることを確認します。
- k. 「表 (Schema.Table)」 フィールドに *schema.table* の形式で表名を入力するか、懐中電灯アイコンを使用してデータベース・オブジェクトを検索します。この例では、hr.departments を入力します。
- l. 「起動タイミング」 に「前」 が選択されていることを確認してください。
- m. 「イベント」 に「選択」 および「列の更新」 を選択します。  
表内の列が表示されます。
- n. 新規の time 列を除く表のすべての列を選択します。
- o. 「拡張」 をクリックして、「拡張」 サブページを開きます。
- p. 「行レベル・トリガー」 を選択します。
- q. 「OK」 をクリックしてトリガーを作成します。



---

**注意：** SQL 文 CREATE TRIGGER を使用して、トリガーを作成することもできます。

---

3. SQL\*Plus で、データベースに Oracle Streams 管理者として接続します。

SQL\*Plus の起動の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。

4. 表の列にサブリメンタル・ロギングを追加します。

```
ALTER TABLE hr.departments ADD SUPPLEMENTAL LOG DATA (ALL) COLUMNS;
```

サブリメンタル・ロギングは、適用中の競合解消に必要です。

5. SET\_UPDATE\_CONFLICT\_HANDLER プロシージャを実行して、表の最新時刻競合解消を構成します。

たとえば、次のプロシージャを実行して、hr.departments 表の最新時刻競合解消を構成します。

```
DECLARE
 cols DBMS_UTILITY.NAME_ARRAY;
BEGIN
 cols(1) := 'department_id';
 cols(2) := 'department_name';
 cols(3) := 'manager_id';
 cols(4) := 'location_id';
 cols(5) := 'time';
 DBMS_APPLY_ADM.SET_UPDATE_CONFLICT_HANDLER(
 object_name => 'hr.departments',
 method_name => 'MAXIMUM',
 resolution_column => 'time',
 column_list => cols);
END;
```

cols 列リストの表内のすべての列を含みます。

6. レプリケーション環境で競合解消が必要な表について、これらの手順を繰り返します。複数のデータベースで表の競合解消を構成することが必要な場合があります。

レプリケーション環境を構成または拡張する例を実行している場合は、適切な表に対して最新時刻競合解消を構成します。

- 4-24 ページの「チュートリアル: ローカル取得プロセスを使用した 2 データベース・レプリケーションの構成」では、db1.example.com および db2.example.com データベースで hr スキーマのすべての表に対して競合解消を構成します。このスキーマには、countries、departments、employees、jobs、job\_history、locations および regions 表が含まれます。
- 4-39 ページの「チュートリアル: ローカル取得プロセスでのハブアンドスポーク・レプリケーションの構成」では、hub1.example.com、spoke1.example.com、spoke2.example.com の各データベースで hr スキーマ内のすべての表に対して競合解消を構成します。このスキーマには、countries、departments、employees、jobs、job\_history、locations および regions 表が含まれます。
- 4-45 ページの「チュートリアル: 同期取得を使用した 2 データベース・レプリケーションの構成」では、sync1.example.com および sync2.example.com データベースで hr.departments および hr.employees 表に対して競合解消を構成します。

- 6-3 ページの「チュートリアル: レプリケーション環境へのデータベース・オブジェクトの追加」では、hub.example.com、spoke1.example.com および spoke2.example.com データベースで oe.orders および oe.order\_items 表に対して競合解消を構成します。
- 6-7 ページの「チュートリアル: レプリケーション環境へのデータベースの追加」では、spoke3.example.com データベースで hr スキーマのすべての表に対して競合解消を構成します。このスキーマには、countries、departments、employees、jobs、job\_history、locations および regions 表が含まれます。

例からこの項に移動した場合は、ここでその例に戻ります。

**参照:**

- 「競合と競合解消の概要」(4-9 ページ)
- 「構成された更新競合ハンドラの表示」(5-23 ページ)

---

## Oracle Streams レプリケーション環境の管理

この章では、Oracle Streams レプリケーション環境の管理、監視およびトラブルシューティング方法を説明します。

この章は次の項で構成されています。

- [Oracle Streams レプリケーション環境の管理](#)
- [Oracle Streams レプリケーション環境の監視](#)
- [Oracle Streams レプリケーション環境のトラブルシューティング](#)

**参照：**

- [第 4 章「Oracle Streams を使用したデータのレプリケート」](#)
- [第 6 章「Oracle Streams レプリケーション環境の拡張」](#)
- 『Oracle Streams 概要および管理』
- 『Oracle Streams レプリケーション管理者ガイド』

## Oracle Streams レプリケーション環境の管理

Oracle Streams レプリケーション環境では、管理はそれほど必要ありません。環境が正しく構成されている場合は、必要な最小限の管理作業で変更がデータベース・オブジェクトに自動的にレプリケートされます。この項では、Oracle Streams レプリケーション環境で必要になることのある管理タスクの実行方法を説明します。

次の各項で、Oracle Streams レプリケーション環境の管理を説明します。

- [取得プロセスの管理](#)
- [伝播の有効化および無効化](#)
- [適用プロセスの管理](#)

### 参照：

- [第4章「Oracle Streams を使用したデータのレプリケート」](#)
- [「Oracle Streams レプリケーション環境の監視」](#) (5-9 ページ)
- [「Oracle Streams レプリケーション環境のトラブルシューティング」](#) (5-26 ページ)

## 取得プロセスの管理

Enterprise Manager を使用して、取得プロセスを管理できます。この項では、取得プロセスの最も一般的な管理タスクを完了する方法を説明します。

次の各項で、取得プロセスの管理について説明します。

- [取得プロセスの起動と停止](#)
- [取得プロセスのパラメータの設定](#)

### 参照：

- [「取得プロセスでの変更の取得の概要」](#) (4-3 ページ)
- [「取得プロセスの監視」](#) (5-12 ページ)

## 取得プロセスの起動と停止

この項では、Enterprise Manager で取得プロセスを起動および停止する方法を説明します。

取得プロセスは、サポートされていないデータ型などのエラーを検出した場合に自動的に停止することがあります。この状況が発生した場合は、Enterprise Manager の「Streams」ページの「取得」サブページにエラーが表示されます。この場合は、エラーを解決して取得プロセスを再起動する必要があります。

また、取得プロセスは、取得プロセスが停止していたときにデータベース・オブジェクトに対して行われた変更を取得できることにも注意してください。これらの変更は REDO ログに記録され、取得プロセスは、再起動後に REDO ログ内で取得するように構成されている変更を検索します。データベース・オブジェクトに対する特定の変更を取得しないようにするには、タグを使用する必要があります。4-10 ページの「[変更の循環を回避するためのタグの概要](#)」を参照してください。

### 取得プロセスを起動または停止するには：

1. Oracle Enterprise Manager で、Oracle Streams 管理者としてハブ・データベースにログインします。
2. 「データベース」ホームページに移動します。
3. 「データ移動」をクリックして「データ移動」サブページを開きます。
4. 「Streams」セクションの「管理」をクリックします。  
「Streams」ページが表示され、「概要」サブページが表示されます。

5. 「取得」をクリックして「取得」サブページを開きます。

| Select                           | Name         | Source Database | Positive Rule Set | Negative Rule Set | Queue         | State             | Status  | First SCN | Start SCN | Capture Type | Error |
|----------------------------------|--------------|-----------------|-------------------|-------------------|---------------|-------------------|---------|-----------|-----------|--------------|-------|
| <input checked="" type="radio"/> | CAPTURE_SIMP | III.EXAMPLE.COM | RULESET\$ 6       | n/a               | STREAMS_QUEUE | CAPTURING CHANGES | ENABLED | 683625    | 683625    | LOCAL        |       |

6. 起動または停止する取得プロセスを選択します。
7. 無効化または中断した取得プロセスを起動するには「**起動**」をクリックし、有効になっている取得プロセスを停止するには「**停止**」をクリックします。
8. 確認ページで「**はい**」をクリックして、取得プロセスの起動または停止を終了します。

**注意：** DBMS\_CAPTURE\_ADM パッケージの START\_CAPTURE プロシージャおよび STOP\_CAPTURE プロシージャを使用して、取得プロセスを起動および停止することもできます。

**参照：**

- 「取得プロセスの監視」 (5-12 ページ)
- 「取得プロセスでの変更の取得の概要」 (4-3 ページ)

## 取得プロセスのパラメータの設定

この項では、Enterprise Manager で取得プロセス・パラメータを設定する方法を説明します。取得プロセス・パラメータは、取得プロセスの動作を制御します。パラメータを設定して、取得プロセスの特定の動作を変更できます。たとえば、PARALLELISM 取得プロセス・パラメータを変更して、変更を取得するプロセス数を制御できます。通常は、取得プロセスの並列性を調整して、取得プロセスのパフォーマンスを最適化します。

### 取得プロセス・パラメータを設定するには：

1. Oracle Enterprise Manager で、Oracle Streams 管理者としてハブ・データベースにログインします。
2. 「データベース」 ホームページに移動します。
3. 「データ移動」をクリックして「データ移動」サブページを開きます。
4. 「Streams」セクションの「管理」をクリックします。  
「Streams」ページが表示され、「概要」サブページが表示されます。
5. 「取得」をクリックして「取得」サブページを開きます。

| Select                           | Name         | Source Database | Positive Rule Set | Negative Rule Set | Queue         | State             | Status  | First SCN | Start SCN | Capture Type | Error |
|----------------------------------|--------------|-----------------|-------------------|-------------------|---------------|-------------------|---------|-----------|-----------|--------------|-------|
| <input checked="" type="radio"/> | CAPTURE_SIMP | III.EXAMPLE.COM | RULESET\$ 6       | n/a               | STREAMS_QUEUE | CAPTURING CHANGES | ENABLED | 683625    | 683625    | LOCAL        |       |

6. 変更する取得プロセスを選択します。

7. 「編集」をクリックして「取得の編集」ページを開きます。

Database Instance: database > Streams > Logged in As STRMADMIN  
 Edit Capture: CAPTURE\_SIMP Revert Apply

Parameters Restore Defaults

| Name                        | Value    | Set by user |
|-----------------------------|----------|-------------|
| DISABLE_ON_LIMIT            | N        | NO          |
| DOWNSTREAM_REAL_TIME_MINE   | Y        | NO          |
| MAXIMUM_SCN                 | INFINITE | NO          |
| MESSAGE_LIMIT               | INFINITE | NO          |
| MESSAGE_TRACKING_FREQUENCY  | 2000000  | NO          |
| PARALLELISM                 | 1        | NO          |
| SKIP_AUTOFILTERED_TABLE_DDL | Y        | NO          |
| STARTUP_SECONDS             | 0        | NO          |
| TIME_LIMIT                  | INFINITE | NO          |
| TRACE_LEVEL                 | 0        | NO          |
| WRITE_ALERT_LOG             | Y        | NO          |

▶ Start and First SCN Revert Apply

8. 「パラメータ」セクションで、1つ以上の取得プロセス・パラメータを変更します。

パラメータの詳細は、『Oracle Database PL/SQL パッケージ・プロシージャおよびタイプ・リファレンス』を参照してください。parallelismパラメータを変更した場合は、「適用」をクリックしたときに取得プロセスが自動的に停止し、再起動します。

9. 「適用」をクリックすると変更が保存されます。

---

**注意：** DBMS\_CAPTURE\_ADM.SET\_PARAMETER プロシージャを使用して、取得プロセス・パラメータを設定することもできます。

---

**参照：**

- 「取得プロセスの監視」(5-12 ページ)
- 「取得プロセスでの変更の取得の概要」(4-3 ページ)

## 伝播の有効化および無効化

この項では、Enterprise Manager で伝播を有効または無効にする方法を説明します。

伝播がメッセージを送信するデータベースが停止している場合、またはネットワークの問題のため伝播によるメッセージの送信が不能な場合は、伝播を無効にできます。このような状況では、伝播を無効にし、問題が解決したら有効にできます。

また、16 回連続試行してもメッセージの送信に失敗した場合は、伝播が自動的に無効になります。この状況が発生した場合は、Enterprise Manager の「Streams」ページの「伝播」サブページにエラーが表示されます。この場合は、エラーを解決し、伝播を有効にする必要があります。

**伝播を有効または無効にするには：**

1. Oracle Enterprise Manager で、Oracle Streams 管理者としてハブ・データベースにログインします。
2. 「データベース」ホームページに移動します。

3. 「データ移動」をクリックして「データ移動」サブページを開きます。
4. 「Streams」セクションの「管理」をクリックします。  
「Streams」ページが表示され、「概要」サブページが表示されます。
5. 「伝播」をクリックして「伝播」サブページを開きます。

Database Instance: database > Logged in As STRMADMIN

### Streams

Overview Capture **Propagation** Apply Messaging

Last Refresh May 8, 2008 10:36:54 AM PDT Setup Propagation

View Delete Statistics

| Select                           | Name         | Positive Rule Set | Negative Rule Set | Source Queue  | Destination DB Link | Status  | Failures Since Startup | Error |
|----------------------------------|--------------|-------------------|-------------------|---------------|---------------------|---------|------------------------|-------|
| <input checked="" type="radio"/> | STR1_TO_STR2 | RULESET\$ 3       | n/a               | STREAMS_QUEUE | II2.EXAMPLE.COM     | Enabled | 0                      |       |

6. 有効または無効にする伝播の「ステータス」フィールドにあるリンクをクリックします。  
リンク・テキストは、「有効」または「無効」です。  
伝播の「ステータス」ページが表示されます。

Database Instance: database > Streams > Logged in As STRMADMIN

Status: STR1\_TO\_STR2 Cancel OK

#### Current Status

Disabled  
 Enabled

**Start Time**  
Time Oct 18, 2006 8:57:04 AM

**Duration of the Propagation**  
 Infinite  
 Finite  
Duration  Seconds

**Next Time**  
 Simple  
Repeat Propagation  By Seconds  
Frequency   
 Advanced  
  
Enter function to evaluate the next propagation time.

**Latency**  
During a propagation window, if there are no messages to be propagated to the specified destination, then the queue will not be checked for messages by the job queue process for the latency specified. If the latency is 0, then a job queue process will be waiting for messages to be enqueued for the destination. As soon as a message is enqueued, it is propagated. Specify this parameter to optimize resource consumption.  
Latency  3 Seconds  
Specify the maximum time to wait before propagating a message after it is enqueued.

Cancel OK

7. 「現行のステータス」セクションで、ステータスを「有効」または「無効」に変更します。  
「次回」、「伝播の期間」または「待機時間」セクションの設定は変更しないでください。  
一般に、これらのセクションのデフォルト値で、最も効率的な伝播が行われます。伝播スケジュールの変更については、このマニュアルで説明しません。
8. 「OK」をクリックして変更を保存します。

---

**注意：** DBMS\_PROPAGATION\_ADM パッケージの START\_PROPAGATION プロシージャおよび STOP\_PROPAGATION プロシージャを使用して、伝播を起動および停止することもできます。

---

**参照：**

- 「データベース間の変更の伝播の概要」 (4-6 ページ)
- 「伝播の監視」 (5-16 ページ)
- 伝播および伝播スケジュールの管理の詳細は、『Oracle Streams アドバンスト・キューイング・ユーザーズ・ガイド』を参照してください。

**適用プロセスの管理**

Enterprise Manager を使用して、適用プロセスを管理できます。この項では、適用プロセスの最も一般的な管理タスクを完了する方法を説明します。

次の各項で、適用プロセスの管理について説明します。

- [適用プロセスの起動と停止](#)
- [適用プロセスのパラメータの設定](#)

**参照：**

- 「変更の適用の概要」 (4-6 ページ)
- 「適用プロセスの監視」 (5-19 ページ)
- 「適用エラーの管理」 (5-30 ページ)

**適用プロセスの起動と停止**

この項では、Enterprise Manager で適用プロセスを起動および停止する方法を説明します。

適用プロセスは、表に存在しない行の更新試行などのエラーが発生した場合に自動的に停止します。この状況が発生した場合は、Enterprise Manager の「Streams」ページの「適用」サブページで、適用プロセスのステータスが「中断」になります。この場合は、エラーを解決して適用プロセスを再起動する必要があります。

**適用プロセスを起動または停止するには：**

1. Oracle Enterprise Manager で、Oracle Streams 管理者としてハブ・データベースにログインします。
2. 「データベース」ホームページに移動します。
3. 「データ移動」をクリックして「データ移動」サブページを開きます。
4. 「Streams」セクションの「管理」をクリックします。  
「Streams」ページが表示され、「概要」サブページが表示されます。
5. 「適用」をクリックして「適用」サブページを開きます。

| Database Instance: database >    |            |                          |                         |                             |                       |                           | Logged in As STRMADMIN                          |                        |         |                  |
|----------------------------------|------------|--------------------------|-------------------------|-----------------------------|-----------------------|---------------------------|-------------------------------------------------|------------------------|---------|------------------|
| Streams                          |            |                          |                         |                             |                       |                           |                                                 |                        |         |                  |
|                                  |            | <a href="#">Overview</a> | <a href="#">Capture</a> | <a href="#">Propagation</a> | <b>Apply</b>          | <a href="#">Messaging</a> |                                                 |                        |         |                  |
|                                  |            |                          |                         |                             |                       |                           | Last Refresh <b>May 8, 2008 10:01:51 AM PDT</b> |                        |         |                  |
|                                  |            | <a href="#">View</a>     | <a href="#">Edit</a>    | <a href="#">Statistics</a>  | <a href="#">Start</a> | <a href="#">Stop</a>      | <a href="#">Errors</a>                          | <a href="#">Delete</a> |         |                  |
| Select                           | Name       | Source Database          | Positive Rule Set       | Negative Rule Set           | Queue                 | Handlers                  | Apply Captured                                  | Apply Tag              | Status  | State            |
| <input checked="" type="radio"/> | APPLY_SIMP | III.EXAMPLE.COM          | RULESET\$ 3             | n/a                         | STREAMS_QUEUE         | n/a                       | YES                                             | 00                     | ENABLED | DEQUEUE MESSAGES |



6. 起動または停止する適用プロセスを選択します。
7. 無効化または中断した適用プロセスを起動するには「**起動**」をクリックし、有効になっている適用プロセスを停止するには「**停止**」をクリックします。
8. 確認ページで「**はい**」をクリックして、適用プロセスの起動または停止を終了します。

---

**注意：** DBMS\_APPLY\_ADM パッケージの START\_APPLY プロシージャおよび STOP\_APPLY プロシージャを使用して、適用プロセスを起動および停止することもできます。

---

**参照：**

- 「適用プロセスの監視」 (5-19 ページ)
- 「適用エラーの管理」 (5-30 ページ)
- 「変更の適用の概要」 (4-6 ページ)

### 適用プロセスのパラメータの設定

この項では、Enterprise Manager で適用プロセス・パラメータを設定する方法を説明します。適用プロセス・パラメータは、適用プロセスの動作を制御します。パラメータを設定して、適用プロセスの特定の動作を変更できます。たとえば、parallelism 適用プロセス・パラメータを変更して、変更を適用するプロセス数を制御できます。通常は、適用プロセスの並列性を調整して、適用プロセスのパフォーマンスを最適化します。

#### 適用プロセス・パラメータを設定するには：

1. Oracle Enterprise Manager で、Oracle Streams 管理者としてハブ・データベースにログインします。
2. 「データベース」 ホームページに移動します。
3. 「**データ移動**」をクリックして「データ移動」サブページを開きます。
4. 「Streams」 セクションの「**管理**」をクリックします。  
「Streams」 ページが表示され、「概要」サブページが表示されます。
5. 「**適用**」をクリックして「適用」サブページを開きます。

| Database Instance: database >                 |            |                 |                   |                   |               |          |                |           |         | Logged in As STRMADMIN                          |                             |              |                           |
|-----------------------------------------------|------------|-----------------|-------------------|-------------------|---------------|----------|----------------|-----------|---------|-------------------------------------------------|-----------------------------|--------------|---------------------------|
| Streams                                       |            |                 |                   |                   |               |          |                |           |         |                                                 |                             |              |                           |
| <a href="#">Overview</a>                      |            |                 |                   |                   |               |          |                |           |         | <a href="#">Capture</a>                         | <a href="#">Propagation</a> | <b>Apply</b> | <a href="#">Messaging</a> |
|                                               |            |                 |                   |                   |               |          |                |           |         | Last Refresh <b>May 8, 2008 10:01:51 AM PDT</b> |                             |              |                           |
| View Edit Statistics Start Stop Errors Delete |            |                 |                   |                   |               |          |                |           |         |                                                 |                             |              |                           |
| Select                                        | Name       | Source Database | Positive Rule Set | Negative Rule Set | Queue         | Handlers | Apply Captured | Apply Tag | Status  | State                                           |                             |              |                           |
| <input checked="" type="radio"/>              | APPLY_SIMP | II1.EXAMPLE.COM | RULESET\$ 3       | n/a               | STREAMS_QUEUE | n/a      | YES            | 00        | ENABLED | DEQUEUE MESSAGES                                |                             |              |                           |

6. 変更する適用プロセスを選択します。

7. 「編集」をクリックして「適用の編集」ページを開きます。

Database Instance: database > Streams > Logged in As STRMADMIN  
 Edit Apply: APPLY\_SIMP Revert Apply

**Parameters** Restore Defaults

| Name                         | Value    | Set by user |
|------------------------------|----------|-------------|
| ALLOW_DUPLICATE_ROWS         | N        | NO          |
| COMMIT_SERIALIZATION         | FULL     | NO          |
| DISABLE_ON_ERROR             | N        | YES         |
| DISABLE_ON_LIMIT             | N        | NO          |
| MAXIMUM_SCN                  | INFINITE | NO          |
| PARALLELISM                  | 1        | NO          |
| PRESERVE_ENCRYPTION          | Y        | NO          |
| RTRIM_ON_IMPLICIT_CONVERSION | Y        | NO          |
| STARTUP_SECONDS              | 0        | NO          |
| TIME_LIMIT                   | INFINITE | NO          |
| TRACE_LEVEL                  | 0        | NO          |
| TRANSACTION_LIMIT            | INFINITE | NO          |
| TXN_LCR_SPILL_THRESHOLD      | 10000    | NO          |
| WRITE_ALERT_LOG              | Y        | NO          |

[▶ Apply Tag](#)

Revert Apply

8. 「パラメータ」セクションで、1つ以上の適用プロセス・パラメータを変更します。  
 パラメータの詳細は、『Oracle Database PL/SQL パッケージ・プロシージャおよびタイプ・リファレンス』を参照してください。parallelism パラメータを変更した場合は、「適用」をクリックしたときに適用プロセスが自動的に停止し、再起動します。
9. 「適用」をクリックすると変更が保存されます。

---

**注意：** DBMS\_APPLY\_ADM.SET\_PARAMETER プロシージャを使用して、適用プロセス・パラメータを設定することもできます。

---

**参照：**

- 「適用プロセスの監視」(5-19 ページ)
- 「適用エラーの管理」(5-30 ページ)
- 「変更の適用の概要」(4-6 ページ)

## Oracle Streams レプリケーション環境の監視

この項では、Enterprise Manager と SQL\*Plus を使用して、レプリケーション・コンポーネントとレプリケーション・トポロジに関する一般情報を表示する方法を説明します。取得プロセス、伝播および適用プロセスの監視方法も説明します。

次の各項で、Oracle Streams レプリケーション環境の監視を説明します。

- データベースのレプリケーション・コンポーネントの概要の表示
- データベースの Oracle Streams レプリケーション環境のトポロジの表示
- 取得プロセスの監視
- 伝播の監視
- 適用プロセスの監視
- 「バッファ・キューの統計」の表示
- 取得から適用までの時間の表示

### 参照：

- 「Oracle Streams レプリケーション環境の管理」(5-2 ページ)
- 第 4 章「Oracle Streams を使用したデータのレプリケート」

## データベースのレプリケーション・コンポーネントの概要の表示

Enterprise Manager の「Streams」ページの「概要」サブページには、現在のデータベース内の Oracle Streams コンポーネントに関する情報が表示されます。次のような情報があります。

- ローカル・データベース内の取得プロセス、伝播、適用プロセス、キューおよびキュー表の数。
- 現在エラーになっている取得プロセス、伝播および適用プロセスの数。

番号をクリックしてドリルダウンし、コンポーネントに関する詳細情報を表示したり、コンポーネントを管理したりできます。

### データベースのレプリケーション・コンポーネントの概要を表示するには：

1. Oracle Enterprise Manager で、Oracle Streams 管理者としてハブ・データベースにログインします。
2. 「データベース」ホームページに移動します。
3. 「データ移動」をクリックして「データ移動」サブページを開きます。

4. 「Streams」セクションの「管理」をクリックします。

「Streams」ページが表示され、「概要」サブページが表示されます。Oracle Streams の「概要」ページの詳細は、「ヘルプ」をクリックしてください。

The screenshot shows the Oracle Streams Overview page. At the top, it indicates the database instance is 'database' and the user is logged in as 'STRMADMIN'. The page is divided into four main sections: Capture, Propagation, Apply, and Messaging. Each section displays the number of processes and errors. For example, under 'Capture', there is 1 Capture Process and 0 errors. The 'Overview' section on the right explains that Oracle Streams enables information sharing and provides details on capture, propagation, and apply processes.

| Section     | Metric                          | Value |
|-------------|---------------------------------|-------|
| Capture     | Capture Processes               | 1     |
|             | Capture Processes Having Errors | 0     |
| Propagation | Propagation Jobs                | 1     |
|             | Propagation Errors              | 0     |
| Apply       | Apply Processes                 | 1     |
|             | Apply Processes Having Errors   | 0     |
| Messaging   | Queue Tables                    | 21    |
|             | Queues                          | 41    |
|             | Total Propagation Errors        | 0     |

参照：

- 「データベースの Oracle Streams レプリケーション環境のトポロジの表示」 (5-10 ページ)

## データベースの Oracle Streams レプリケーション環境のトポロジの表示

Oracle Streams トポロジには、ローカル・データベース、および Oracle Streams 環境でローカル・データベースと対話する他のデータベースのグラフィカル表現が表示されます。レプリケーション環境では、トポロジは、表示されているデータベースに関する次の情報を示します。

- 現在のデータベースから Oracle Streams 環境の他のデータベースへの伝播に使用されるデータベース・リンク。現在のデータベースを始点とする各矢印は、現在のデータベースから別のデータベースへの伝播に使用されるデータベース・リンクを示します。レプリケーション環境では、データベース・リンクを使用して、レプリケートされたオブジェクトに対する変更を他のデータベースに送信します。
- Oracle Streams 環境の他のデータベースから現在のデータベースへの伝播に使用されるデータベース・リンク。現在のデータベースの適用プロセスは、伝播されるメッセージを現在のデータベースに適用します。現在のデータベースを終点とする各矢印は、他のデータベースから、メッセージが適用される現在のデータベースへの伝播に使用されるデータベース・リンクを示します。

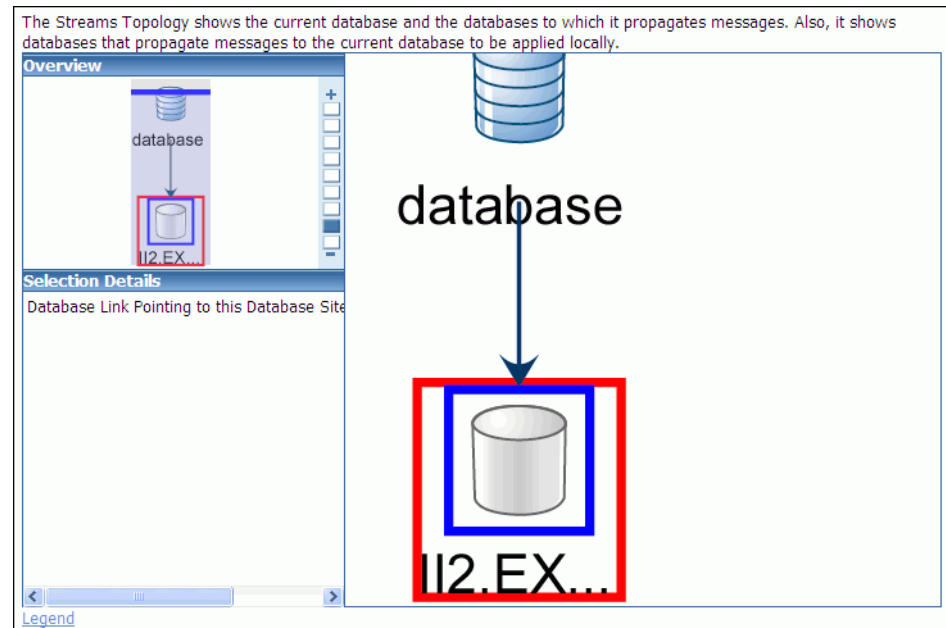
**Oracle Streams トポロジを表示するには：**

1. Oracle Enterprise Manager で、Oracle Streams 管理者としてハブ・データベースにログインします。
2. 「データベース」ホームページに移動します。
3. 「データ移動」をクリックして「データ移動」サブページを開きます。
4. 「Streams」セクションの「管理」をクリックします。  
「Streams」ページが表示され、「概要」サブページが表示されます。
5. Oracle Streams トポロジが表示されるまで下にスクロールします。

「概要」 ウィンドウで、トポロジの特定部分を選択して詳細に表示したり、ズーム・イン/ズーム・アウトできます。トポロジ内のデータベースを選択し、データベースに関する情報を示す「選択の詳細」 ウィンドウを表示することもできます。

Oracle Streams トポロジの詳細は、「ヘルプ」または「凡例」リンクをクリックしてください。

次の図は、Oracle Streams レプリケーション環境の Oracle Streams トポロジの例を示しています。



前の図の Oracle Streams トポロジには、次の要素が含まれています。

- 現在のデータベースの名前は、database です。
- ii2.example.com データベースは、現在のデータベースと相互に作用するレプリケーション環境の一部です。
- 現在のデータベースによって、変更が ii2.example.com データベースに送信されます。これらの変更は、適用プロセスによって ii2.example.com データベースで適用されます。

Enterprise Manager での Oracle Streams トポロジのグラフィック表示だけでなく、DBMS\_STREAMS\_ADVISOR\_ADM パッケージを使用して Oracle Streams トポロジに関する情報を収集することもできます。この情報を収集した後で、次のデータ・ディクショナリ・ビューを問い合わせることで Oracle Streams トポロジを表示できます。

- DBA\_STREAMS\_TP\_COMPONENT には、各データベースの各 Oracle Streams コンポーネントに関する情報が含まれています。
- DBA\_STREAMS\_TP\_COMPONENT\_LINK には、Oracle Streams コンポーネント間でのメッセージ・フローの方法に関する情報が含まれています。
- DBA\_STREAMS\_TP\_COMPONENT\_STAT には、各 Oracle Streams コンポーネントに関する統計が含まれています。
- DBA\_STREAMS\_TP\_DATABASE には、Oracle Streams コンポーネントを含む各データベースに関する情報が含まれています。
- DBA\_STREAMS\_TP\_PATH\_BOTTLENECK には、ストリーム・フローの速度を低下させる可能性のある Oracle Streams コンポーネントに関する情報が含まれています。
- DBA\_STREAMS\_TP\_PATH\_STAT には、Oracle Streams トポロジ内に存在する各ストリームに関する統計が含まれています。

DBMS\_STREAMS\_ADVISOR\_ADM パッケージを使用して情報を収集すると、Oracle Streams パフォーマンス・アドバイザが Oracle Streams のパフォーマンスに関する情報をこれらのビューに配置します。これらのビューを問い合わせれば、Oracle Streams コンポーネントの現在のパフォーマンス状況を判断し、そのパフォーマンスを向上する方法を決定できます。

**参照：**

- 「データベースのレプリケーション・コンポーネントの概要の表示」 (5-9 ページ)
- DBMS\_STREAMS\_ADVISOR\_ADM パッケージ、トポロジ・データ・ディクショナリ・ビューおよび Oracle Streams パフォーマンス・アドバイザの使用の詳細は、『Oracle Streams 概要および管理』を参照してください。
- 『Oracle Database リファレンス』

## 取得プロセスの監視

Enterprise Manager を使用して、取得プロセスに関する詳細情報を表示できます。取得プロセスの統計も表示できます。

次の各項で、取得プロセスの監視について説明します。

- 取得プロセスの情報の表示
- 取得プロセスに関する統計情報の表示

**参照：**

- 「取得プロセスでの変更の取得の概要」 (4-3 ページ)
- 「取得プロセスの管理」 (5-2 ページ)

## 取得プロセスの情報の表示

Enterprise Manager を使用して、取得プロセスに関する情報を表示できます。この情報には、取得プロセスのステータスと状態、取得プロセスで使用されるルールおよび取得プロセスに関するその他の情報が含まれます。

### Enterprise Manager で取得プロセスに関する詳細情報を表示するには：

1. Oracle Enterprise Manager で、Oracle Streams 管理者としてハブ・データベースにログインします。
2. 「データベース」 ホームページに移動します。
3. 「データ移動」 をクリックして「データ移動」 サブページを開きます。
4. 「Streams」 セクションの「管理」 をクリックします。  
「Streams」 ページが表示され、「概要」 サブページが表示されます。
5. 「取得」 をクリックして「取得」 サブページを開きます。

The screenshot shows the Oracle Enterprise Manager interface for Streams. The page title is "Streams" and it is logged in as STRMADMIN. There are tabs for Overview, Capture, Propagation, Apply, and Messaging. The "Capture" tab is active. Below the tabs, there are buttons for View, Edit, Statistics, Start, Stop, and Delete. A table lists the capture processes with columns: Select, Name, Source Database, Positive Rule Set, Negative Rule Set, Queue, State, Status, First SCN, Start SCN, Capture Type, and Error. One process is listed: CAPTURE\_SIMP, III.EXAMPLE.COM, RULESET\$ 6, n/a, STREAMS\_QUEUE, CAPTURING CHANGES, ENABLED, 683625, 683625, LOCAL.

| Select                              | Name         | Source Database | Positive Rule Set | Negative Rule Set | Queue         | State             | Status  | First SCN | Start SCN | Capture Type | Error |
|-------------------------------------|--------------|-----------------|-------------------|-------------------|---------------|-------------------|---------|-----------|-----------|--------------|-------|
| <input checked="" type="checkbox"/> | CAPTURE_SIMP | III.EXAMPLE.COM | RULESET\$ 6       | n/a               | STREAMS_QUEUE | CAPTURING CHANGES | ENABLED | 683625    | 683625    | LOCAL        |       |

「取得」サブページには、データベース内の各取得プロセスに関する一般情報が表示されます。この情報には、取得プロセスのルール・セット、キュー、状態およびステータスが含まれます。「取得」サブページの詳細は、「ヘルプ」をクリックしてください。

6. 監視する取得プロセスを選択します。
7. 「表示」をクリックして「取得の詳細を表示」ページを開きます。

Database Instance: database > Streams > Logged in As STRMADMIN

**View Capture Details: CAPTURE\_SIMP**

---

**General**

Name **CAPTURE\_SIMP**  
Queue **STREAMS\_QUEUE**  
Queue Schema **STRMADMIN**  
Status **ENABLED**  
Error Message  
Status Change Time **May 8, 2008 6:17:05 AM PDT**

**Rules**

Positive Rule Set [RULESETS\\_6](#)  
Positive Rule Set Owner **STRMADMIN**  
Negative Rule Set [n/a](#)  
Negative Rule Set Owner **n/a**

---

**Database**

Source Database **II1.EXAMPLE.COM**  
Source Database ID **2310028912**  
Capture Type **LOCAL**  
Use Database Link **NO**  
Logfile Assignment **IMPLICIT**  
LogMiner ID **1**

**SCN**

First SCN [683625](#)  
Start SCN [683625](#)  
Captured SCN **720152**  
Applied SCN **720010**  
Max Checkpoint SCN **720152**  
Required Checkpoint SCN **717262**

---

**Objects**

Search

This table lists all the objects captured by this process

| Name                | Rule Set Type | Change Type | Rule Name | Rule Type | Same Rule Condition |
|---------------------|---------------|-------------|-----------|-----------|---------------------|
| No Search Conducted |               |             |           |           |                     |

「取得の詳細を表示」ページには、取得プロセスに関する詳細情報が表示されます。また、取得プロセスが変更を取得するデータベース・オブジェクトを表示できます。ルールでは、取得プロセスで取得されるデータベース変更が制御されます。「オブジェクト」セクションの検索ツールを使用して、取得プロセス・ルールを表示します。

- ポジティブ・ルール・セットのすべてのルールを表示するには、「**ポジティブ・ルール・タイプ**」を選択し、「**実行**」をクリックします。ポジティブ・ルールは、データベース・オブジェクトに対する変更を取得するよう取得プロセスに指示します。
- ネガティブ・ルール・セットのすべてのルールを表示するには、「**ネガティブ・ルール・タイプ**」を選択し、「**実行**」をクリックします。ネガティブ・ルールは、データベース・オブジェクトに対する変更を取得しないよう取得プロセスに指示します。

このページの詳細を表示するには、「**ヘルプ**」をクリックします。

---

**注意：** 取得プロセスに関する情報を、ALL\_CAPTURE データ・ディクショナリ・ビューに問い合わせることもできます。

---

**参照：**

- 「取得プロセスでの変更の取得の概要」(4-3 ページ)
- 「取得、伝播および適用の動作を制御するルールの概要」(4-7 ページ)
- 「取得プロセスの管理」(5-2 ページ)

## 取得プロセスに関する統計情報の表示

Enterprise Manager を使用して、取得プロセスの統計を表示できます。統計には、取得プロセス・キュー内のメッセージ数、取得プロセスで取得およびエンキューされたメッセージ数、および取得プロセスに関連するその他の統計が含まれます。

### Enterprise Manager で取得プロセスの統計を表示するには：

1. Oracle Enterprise Manager で、Oracle Streams 管理者としてハブ・データベースにログインします。
2. 「データベース」 ホームページに移動します。
3. 「データ移動」 をクリックして「データ移動」 サブページを開きます。
4. 「Streams」 セクションの「管理」 をクリックします。  
「Streams」 ページが表示され、「概要」 サブページが表示されます。
5. 「取得」 をクリックして「取得」 サブページを開きます。

The screenshot shows the Oracle Enterprise Manager interface for Streams. At the top, it says "Database Instance: database >" and "Logged in As STRMADMIN". Below that is the "Streams" section with tabs for "Overview", "Capture", "Propagation", "Apply", and "Messaging". The "Capture" tab is selected. There are buttons for "View", "Edit", "Statistics", "Start", "Stop", and "Delete". A "Last Refresh" timestamp is shown as "May 8, 2008 7:48:26 AM PDT". Below these buttons is a table with the following data:

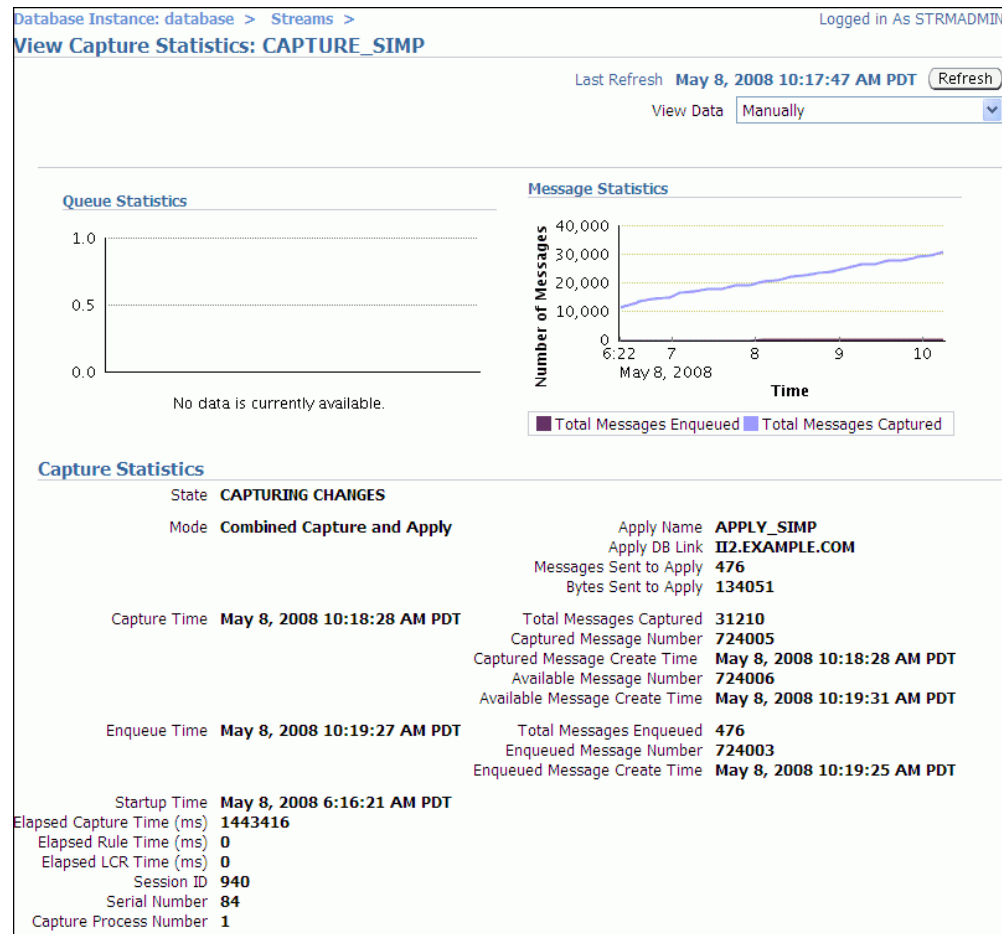
| Select                           | Name         | Source Database | Positive Rule Set | Negative Rule Set | Queue         | State             | Status  | First SCN | Start SCN | Capture Type | Error |
|----------------------------------|--------------|-----------------|-------------------|-------------------|---------------|-------------------|---------|-----------|-----------|--------------|-------|
| <input checked="" type="radio"/> | CAPTURE_SIMP | III.EXAMPLE.COM | RULESET\$ 6       | n/a               | STREAMS_QUEUE | CAPTURING CHANGES | ENABLED | 683625    | 683625    | LOCAL        |       |

「取得」 サブページには、データベース内の各取得プロセスに関する一般情報が表示されます。この情報には、取得プロセスのルール・セット、キュー、状態およびステータスが含まれます。「取得」 サブページの詳細は、「ヘルプ」 をクリックしてください。

6. 監視する取得プロセスを選択します。



7. 「統計」をクリックして「取得の統計を表示」ページを開きます。



「取得の統計を表示」ページには、次の情報が表示されます。

- 「キュー統計」グラフでは、取得プロセス・キュー内の現在のメッセージ数を表示します。グラフの「エンキューされたメッセージ数」行には、現在バッファ・キュー入っているメッセージの合計数が表示されます。グラフの「オーバーフローしたメッセージ数」行には、メモリーから永続キュー表にオーバーフローしたメッセージの合計数が表示されます。
- 「メッセージの統計」グラフは、取得プロセスが前回起動した後にエンキューおよび取得された変更の合計数を示します。「エンキューされた合計メッセージ数」は、取得プロセスによってエンキューされた変更の数を示します。「取得された合計メッセージ数」は、取得プロセス・ルールと比較して詳細に評価された変更の数を示します。変更が取得プロセス・ルールを満たさない場合、変更はエンキューされません。
- 「取得の統計」セクションには、取得プロセスの現在の状態と、時間および取得された変更（メッセージ）に関連する統計が表示されます。これらの統計の詳細は、『Oracle Database リファレンス』で V\$STREAMS\_CAPTURE 動的パフォーマンス・ビューに関する項を参照してください。

このページの詳細を表示するには、「ヘルプ」をクリックします。

**参照：**

- 「取得プロセスでの変更の取得の概要」(4-3 ページ)
- 「取得プロセスの管理」(5-2 ページ)

## 伝播の監視

Enterprise Manager を使用して、伝播に関する詳細情報を表示できます。伝播の統計も表示できます。

次の各項で、伝播の監視について説明します。

- [伝播の情報の表示](#)
- [伝播に関する統計情報の表示](#)

### 参照：

- [「データベース間の変更の伝播の概要」](#) (4-6 ページ)
- [「伝播の有効化および無効化」](#) (5-4 ページ)

## 伝播の情報の表示

Enterprise Manager を使用して、伝播に関する情報を表示できます。この情報には、伝播のステータス、伝播が失敗したかどうか、伝播で使用されるルールおよび伝播に関するその他の情報が含まれます。

### Enterprise Manager で伝播に関する情報を表示するには：

1. Oracle Enterprise Manager で、Oracle Streams 管理者としてハブ・データベースにログインします。
2. 「データベース」 ホームページに移動します。
3. 「データ移動」 をクリックして「データ移動」 サブページを開きます。
4. 「Streams」 セクションの「管理」 をクリックします。  
「Streams」 ページが表示され、「概要」 サブページが表示されます。
5. 「伝播」 をクリックして「伝播」 サブページを開きます。

The screenshot shows the Oracle Enterprise Manager interface for Streams propagation. The page title is "Streams" and the user is logged in as STRMADMIN. The "Propagation" tab is selected. The table below lists the propagation rules.

| Select                           | Name         | Positive Rule Set | Negative Rule Set | Source Queue  | Destination DB Link | Status  | Failures Since Startup | Error |
|----------------------------------|--------------|-------------------|-------------------|---------------|---------------------|---------|------------------------|-------|
| <input checked="" type="radio"/> | STR1_TO_STR2 | RULESET\$ 3       | n/a               | STREAMS_QUEUE | II2.EXAMPLE.COM     | Enabled | 0                      |       |

6. 監視する伝播を選択します。

7. 「表示」をクリックして、伝播の「伝播の詳細を表示」ページを開きます。

Database Instance: database > Streams > Logged in As STRMADMIN

### View Propagation Details: STR1\_TO\_STR2

**General**

|                           |                            |
|---------------------------|----------------------------|
| Propagation Name          | STR1_TO_STR2               |
| Source Queue              | STREAMS_QUEUE              |
| Source Queue Schema       | STRMADMIN                  |
| Destination Queue         | STREAMS_QUEUE              |
| Destination Queue Schema  | STRMADMIN                  |
| Destination Database Link | II2.EXAMPLE.COM            |
| Number Of Errors          | 0                          |
| Positive Rule Set Name    | <a href="#">RULESETS_3</a> |
| Positive Rule Set Owner   | STRMADMIN                  |
| Negative Rule Set Name    | n/a                        |
| Negative Rule Set Owner   | n/a                        |
| Start Date                |                            |
| Last Error Time           | n/a                        |
| Last Error Message        | n/a                        |

**Objects**

Search

| Name                | Rule Set Type | Change Type | Rule Name | Rule Type | Same Rule Condition |
|---------------------|---------------|-------------|-----------|-----------|---------------------|
| No Search Conducted |               |             |           |           |                     |

「伝播の詳細を表示」ページには、伝播プロセスに関する詳細情報が表示されます。また、伝播が変更を送信するデータベース・オブジェクトを表示できます。ルールでは、伝播によって送信されるデータベース変更が制御されます。「オブジェクト」セクションの検索ツールを使用して、伝播ルールを表示します。

- ポジティブ・ルール・セットのすべてのルールを表示するには、「**ポジティブ・ルール・タイプ**」を選択し、「**実行**」をクリックします。ポジティブ・ルールは、データベース・オブジェクトに対する変更を宛先キューに送信するよう伝播に指示します。
- ネガティブ・ルール・セットのすべてのルールを表示するには、「**ネガティブ・ルール・タイプ**」を選択し、「**実行**」をクリックします。ネガティブ・ルールは、データベース・オブジェクトに対する変更を宛先キューに送信しないよう伝播に指示します。

このページの詳細を表示するには、「**ヘルプ**」をクリックします。

---

**注意：** 伝播に関する情報を、ALL\_PROPAGATION データ・ディクショナリ・ビューに問い合わせることもできます。

---

**参照：**

- 「データベース間の変更の伝播の概要」(4-6 ページ)
- 「取得、伝播および適用の動作を制御するルールの概要」(4-7 ページ)
- 「伝播の有効化および無効化」(5-4 ページ)

## 伝播に関する統計情報の表示

Enterprise Manager を使用して、伝播の統計を表示できます。統計には、伝播ソース・キュー内のメッセージ数、伝播によって送信されたメッセージ数および伝播に関連するその他の統計が含まれます。

**注意：** 取得と適用の組合せを使用する場合、伝播の統計は計算されません。

### Enterprise Manager で伝播の統計を表示するには：

1. Oracle Enterprise Manager で、Oracle Streams 管理者としてハブ・データベースにログインします。
2. 「データベース」 ホームページに移動します。
3. 「データ移動」 をクリックして「データ移動」 サブページを開きます。
4. 「Streams」 セクションの「管理」 をクリックします。  
「Streams」 ページが表示され、「概要」 サブページが表示されます。
5. 「伝播」 をクリックして「伝播」 サブページを開きます。

The screenshot shows the Oracle Enterprise Manager interface for Streams. The 'Streams' section is active, and the 'Propagation' tab is selected. A table lists the propagation details for 'STR1\_TO\_STR2 RULESET\$ 3'. The table has columns for Name, Positive Rule Set, Negative Rule Set, Source Queue, Destination DB Link, Status, Failures Since Startup, and Error. The 'Status' is 'Enabled' and 'Failures Since Startup' is '0'.

| Select                              | Name                     | Positive Rule Set | Negative Rule Set | Source Queue  | Destination DB Link | Status  | Failures Since Startup | Error |
|-------------------------------------|--------------------------|-------------------|-------------------|---------------|---------------------|---------|------------------------|-------|
| <input checked="" type="checkbox"/> | STR1_TO_STR2 RULESET\$ 3 |                   | n/a               | STREAMS_QUEUE | III.EXAMPLE.COM     | Enabled | 0                      |       |

6. 監視する伝播を選択します。
7. 「統計」 をクリックして、伝播の「伝播の統計を表示」 ページを開きます。

「伝播の統計を表示」 ページには、次のサブページが表示されます。

- 「永続キュー」 サブページには、伝播で使用されるキューの永続キュー部分にエンキューされたメッセージの伝播統計が表示されます。このサブページを使用して、伝播が同期取得で取得された変更を送信する場合の伝播統計を表示します。
- 「バッファ・キュー」 サブページには、伝播で使用されるキューのバッファ・キュー部分にエンキューされたメッセージの伝播統計が表示されます。このサブページを使用して、伝播が取得プロセスで取得された変更を送信する場合の伝播統計を表示します。

どちらのサブページにも、キュー内のメッセージ数と、伝播により送信されたメッセージ数を数時間にわたって示すグラフがあります。両方のサブページには、メッセージの合計数や、伝播が前回起動した後に伝播されたバイト数など、その他の伝播統計も含まれます。現在のサブページの統計の詳細は、「ヘルプ」 をクリックしてください。



**注意：** バッファ・キュー統計および伝播統計を、次の動的パフォーマンス・ビューに問い合わせることもできます。

- V\$BUFFERED\_QUEUES
- V\$PROPAGATION\_SENDER
- V\$PROPAGATION\_RECEIVER

**参照：**

- 「データベース間の変更の伝播の概要」(4-6 ページ)
- 「伝播の有効化および無効化」(5-4 ページ)
- 取得と適用の複合の詳細は、『Oracle Streams 概要および管理』を参照してください。

## 適用プロセスの監視

Enterprise Manager を使用して、適用プロセスに関する詳細情報を表示できます。適用プロセスの統計も表示できます。

次の各項で、適用プロセスの監視について説明します。

- 適用プロセスの情報の表示
- 適用プロセスに関する統計情報の表示
- 構成された更新競合ハンドラの表示

**参照：**

- 「変更の適用の概要」(4-6 ページ)
- 「適用プロセスの管理」(5-6 ページ)
- 「適用エラーの管理」(5-30 ページ)

## 適用プロセスの情報の表示

Enterprise Manager を使用して、適用プロセスに関する情報を表示できます。この情報には、適用プロセスのステータスと状態、適用プロセスで使用されるルールおよび適用プロセスに関するその他の情報が含まれます。

**Enterprise Manager で適用プロセスに関する情報を表示するには：**

1. Oracle Enterprise Manager で、Oracle Streams 管理者としてハブ・データベースにログインします。
2. 「データベース」 ホームページに移動します。
3. 「データ移動」 をクリックして「データ移動」 サブページを開きます。
4. 「Streams」 セクションの「管理」 をクリックします。  
「Streams」 ページが表示され、「概要」 サブページが表示されます。
5. 「適用」 をクリックして「適用」 サブページを開きます。

Database Instance: database > Logged in As STRMADMIN

**Streams**

Overview Capture Propagation **Apply** Messaging

Last Refresh **May 8, 2008 10:01:51 AM PDT**

View Edit Statistics Start Stop Errors Delete

| Select                           | Name       | Source Database | Positive Rule Set | Negative Rule Set | Queue         | Handlers | Apply Captured | Apply Tag | Status  | State            |
|----------------------------------|------------|-----------------|-------------------|-------------------|---------------|----------|----------------|-----------|---------|------------------|
| <input checked="" type="radio"/> | APPLY_SIMP | II1.EXAMPLE.COM | RULESET\$ 3       | n/a               | STREAMS_QUEUE | n/a      | YES            | 00        | ENABLED | DEQUEUE MESSAGES |

6. 監視する適用プロセスを選択します。
7. 「表示」 をクリックして「適用の詳細表示」 ページを開きます。

Database Instance: database > Streams > Logged in As STRMADMIN

**View Apply Details: APPLY\_SIMP**

**General**

Name **APPLY\_SIMP**  
 Queue **STREAMS\_QUEUE**  
 Queue Schema **STRMADMIN**  
 Status **ENABLED**  
 Positive Rule Set Name [RULESET\\$ 3](#)  
 Positive Rule Set Owner **STRMADMIN**  
 Negative Rule Set Name [n/a](#)  
 Negative Rule Set Owner [n/a](#)  
 Precommit Handler [No](#)  
 DDL Handler [No](#)  
 Message Handler [No](#)  
 Error Number  
 Apply Captured **YES**  
 Apply Tag [00](#)  
 Maximum Applied Message Number

**Objects**

Search

| Name                | Rule Set Type | Change Type | Rule Name | Rule Type | Same Rule Condition |
|---------------------|---------------|-------------|-----------|-----------|---------------------|
| No Search Conducted |               |             |           |           |                     |

「適用の詳細表示」ページには、適用プロセスに関する詳細情報が表示されます。また、適用プロセスが変更を適用するデータベース・オブジェクトを表示できます。ルールでは、適用プロセスでデキューおよび適用されるデータベース変更が制御されます。「オブジェクト」セクションの検索ツールを使用して、適用プロセス・ルールを表示します。

- ポジティブ・ルール・セットのすべてのルールを表示するには、「**ポジティブ・ルール・タイプ**」を選択し、「**実行**」をクリックします。ポジティブ・ルールは、データベース・オブジェクトに対する変更をデキューして適用するよう適用プロセスに指示します。
- ネガティブ・ルール・セットのすべてのルールを表示するには、「**ネガティブ・ルール・タイプ**」を選択し、「**実行**」をクリックします。ネガティブ・ルールは、データベース・オブジェクトに対する変更をデキューしないよう適用プロセスに指示します。

このページの詳細を表示するには、「**ヘルプ**」をクリックします。

---

**注意：** 適用プロセスに関する情報を、ALL\_APPLY データ・ディクショナリ・ビューに問い合わせることもできます。

---

**参照：**

- 「[変更の適用の概要](#)」(4-6 ページ)
- 「[取得、伝播および適用の動作を制御するルールの概要](#)」(4-7 ページ)
- 「[適用プロセスの管理](#)」(5-6 ページ)

## 適用プロセスに関する統計情報の表示

Enterprise Manager を使用して、適用プロセスの統計を表示できます。統計には、適用プロセス・キュー内のメッセージ数、適用プロセスで適用されたメッセージ数、および適用プロセスに関連するその他の統計が含まれます。

### Enterprise Manager で適用プロセスの統計を表示するには：

1. Oracle Enterprise Manager で、Oracle Streams 管理者としてハブ・データベースにログインします。
2. 「データベース」ホームページに移動します。
3. 「**データ移動**」をクリックして「データ移動」サブページを開きます。
4. 「Streams」セクションの「**管理**」をクリックします。  
「Streams」ページが表示され、「概要」サブページが表示されます。
5. 「**適用**」をクリックして「適用」サブページを開きます。

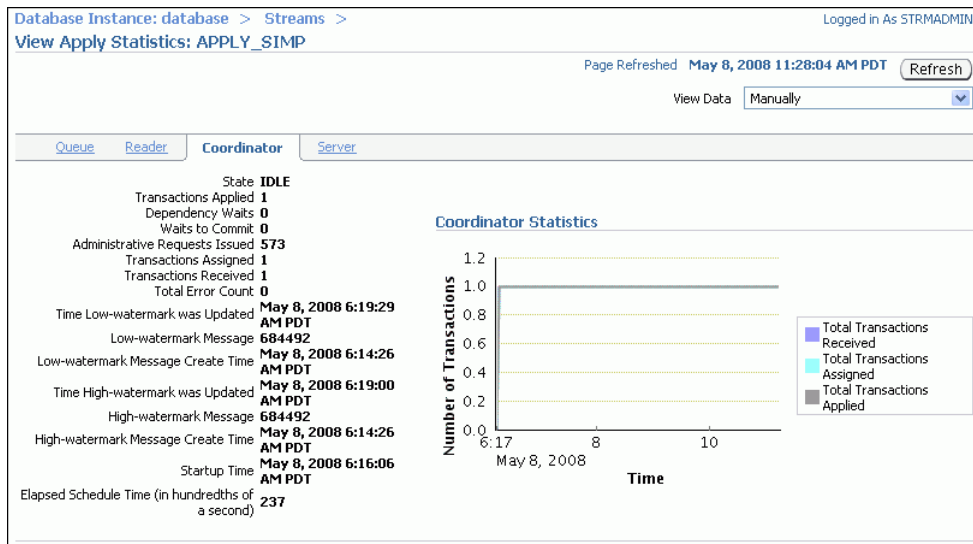
| Database Instance: database >                                                                                                                                                                                |            |                  |                         |                   |                             |          | Logged in As STRMADMIN                          |           |                           |                  |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|------------------|-------------------------|-------------------|-----------------------------|----------|-------------------------------------------------|-----------|---------------------------|------------------|
| Streams                                                                                                                                                                                                      |            |                  |                         |                   |                             |          |                                                 |           |                           |                  |
| <a href="#">Overview</a>                                                                                                                                                                                     |            |                  | <a href="#">Capture</a> |                   | <a href="#">Propagation</a> |          | <a href="#">Apply</a>                           |           | <a href="#">Messaging</a> |                  |
|                                                                                                                                                                                                              |            |                  |                         |                   |                             |          | Last Refresh <b>May 8, 2008 10:01:51 AM PDT</b> |           |                           |                  |
| <div style="display: flex; justify-content: space-between;"> <span>View</span> <span>Edit</span> <span>Statistics</span> <span>Start</span> <span>Stop</span> <span>Errors</span> <span>Delete</span> </div> |            |                  |                         |                   |                             |          |                                                 |           |                           |                  |
| Select                                                                                                                                                                                                       | Name       | Source Database  | Positive Rule Set       | Negative Rule Set | Queue                       | Handlers | Apply Captured                                  | Apply Tag | Status                    | State            |
| <input checked="" type="radio"/>                                                                                                                                                                             | APPLY_SIMP | III1.EXAMPLE.COM | RULESET# 3              | n/a               | STREAMS_QUEUE               | n/a      | YES                                             | 00        | ENABLED                   | DEQUEUE MESSAGES |

6. 監視する適用プロセスを選択します。
7. 「**統計**」をクリックして「適用の統計を表示」ページを開きます。

「適用の統計を表示」ページには、次のサブページがあります。

- 「キュー」サブページは、適用プロセス・キューの永続キュー部分とバッファ・キュー部分両方にあった過去数時間のメッセージ数が表示されます。適用プロセスが、同期取得で取得された変更を適用する場合は、永続キュー統計を分析します。適用プロセスが、取得プロセスで取得された変更を適用する場合は、バッファ・キュー統計を分析します。
- 「リーダー」サブページには、リーダー・サーバーの統計が表示されます。リーダー・サーバーは、適用プロセス・キューからメッセージをデキューし、別々のトランザクションにアSEMBルします。
- 「座標」サブページには、コーディネータ・プロセスの統計が表示されます。コーディネータ・プロセスは、リーダー・サーバーからトランザクションを取得し、適用サーバーに渡します。
- 「サーバー」サブページには、適用サーバーの統計が表示されます。適用サーバーは、データベース・オブジェクトに対する変更を適用するか、適用ハンドラに変更を渡します。特定の適用サーバーの詳細を表示するには、その適用サーバーを選択し、「詳細の表示」をクリックします。

現在のサブページに表示されている統計の詳細は、「ヘルプ」をクリックしてください。



**注意：** バッファ・キュー統計および適用プロセス統計を、次の動的パフォーマンス・ビューに問い合わせることもできます。

- V\$BUFFERED\_QUEUES
- V\$STREAMS\_APPLY\_COORDINATOR
- V\$STREAMS\_APPLY\_READER
- V\$STREAMS\_APPLY\_SERVER

**参照：**

- 「変更の適用の概要」(4-6 ページ)
- 「適用プロセスの管理」(5-6 ページ)



## 構成された更新競合ハンドラの表示

レプリケーション環境で、更新競合ハンドラは、同じ行が2つの異なるデータベースではほぼ同時に更新された場合に発生する競合を自動的に解消します。ALL\_APPLY\_CONFLICT\_COLUMNS データ・ディクショナリ・ビューを使用して、更新競合ハンドラをリストできます。

### 構成された更新競合ハンドラを表示するには：

1. SQL\*Plus を開き、データベースに Oracle Streams 管理者として接続します。

SQL\*Plus の起動の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。

2. 次の問合せを実行します。

```
COLUMN OBJECT_OWNER HEADING 'Table|Owner' FORMAT A5
COLUMN OBJECT_NAME HEADING 'Table Name' FORMAT A12
COLUMN METHOD_NAME HEADING 'Method' FORMAT A12
COLUMN RESOLUTION_COLUMN HEADING 'Resolution|Column' FORMAT A13
COLUMN COLUMN_NAME HEADING 'Column Name' FORMAT A30
```

```
SELECT OBJECT_OWNER,
 OBJECT_NAME,
 METHOD_NAME,
 RESOLUTION_COLUMN,
 COLUMN_NAME
FROM ALL_APPLY_CONFLICT_COLUMNS
ORDER BY OBJECT_OWNER, OBJECT_NAME, RESOLUTION_COLUMN;
```

出力は、次のようになります。

| Table<br>Owner | Table Name  | Method  | Resolution<br>Column | Column Name     |
|----------------|-------------|---------|----------------------|-----------------|
| HR             | COUNTRIES   | MAXIMUM | TIME                 | TIME            |
| HR             | COUNTRIES   | MAXIMUM | TIME                 | REGION_ID       |
| HR             | COUNTRIES   | MAXIMUM | TIME                 | COUNTRY_NAME    |
| HR             | DEPARTMENTS | MAXIMUM | TIME                 | TIME            |
| HR             | DEPARTMENTS | MAXIMUM | TIME                 | MANAGER_ID      |
| HR             | DEPARTMENTS | MAXIMUM | TIME                 | LOCATION_ID     |
| HR             | DEPARTMENTS | MAXIMUM | TIME                 | DEPARTMENT_NAME |

この例の出力は、最新時間の競合解消が hr.countries 表および hr.departments 表に対して構成されていることを示しています。「Column Name」の下にリストされている任意の列で競合が発生した場合は、「TIME」解決列の時間が最大（最新）の変更が使用され、その他の変更は破棄されます。

#### 参照：

- 「競合と競合解消の概要」 (4-9 ページ)
- 「チュートリアル: 表の最新時刻競合解消の構成」 (4-55 ページ)

## 「バッファ・キューの統計」の表示

Enterprise Manager を使用して、レプリケーション環境の取得プロセス、伝播または適用プロセスで使用されるバッファ・キューの統計を表示できます。取得プロセスを使用して変更を取得する Oracle Streams レプリケーション環境では、各取得プロセスは変更をそのキューのバッファ・キュー部分にエンキューします。変更は、あるキューから別のキューに伝播されたときにバッファ・キューに残り、適用プロセスがバッファ・キューから変更をデキューします。

**バッファ・キュー統計を表示するには：**

1. Oracle Enterprise Manager で、Oracle Streams 管理者としてハブ・データベースにログインします。
2. 「データベース」 ホームページに移動します。
3. 「データ移動」 をクリックして「データ移動」 サブページを開きます。
4. 「Streams」 セクションの「管理」 をクリックします。  
「Streams」 ページが表示され、「概要」 サブページが表示されます。
5. 「メッセージ機能」 をクリックして「メッセージ機能」 サブページを開きます。
6. 監視するキューを選択します。
7. 「アクション」 リストの「キュー統計」 を選択します。
8. 「実行」 をクリックして「キュー統計」 ページを開きます。
9. 「バッファ・キューの統計」 をクリックして「バッファ・キューの統計」 サブページを開きます。

Database Instance: database > Streams > Logged in As STRMADMIN

**Queue Statistics: STREAMS\_QUEUE** Page Refreshed May 8, 2008 12:37:35 PM PDT

View Data

[Propagation Statistics](#) [Message Statistics](#) **Buffered Queue Statistics**

Queue Owner **STRMADMIN**  
 Current Messages In Memory **0**  
 Current Total Messages **1**  
 Total Messages Enqueued Into The Buffered Queue **23**  
 Current Spilled Messages **1**  
 Total Messages Spilled From The Buffered Queue **22**

**Publisher Statistics**  
 This table lists the statistics related to the Capture Processes that enqueue data into the buffered queue.

| Publisher    | Sender Queue | Current Messages In Queue | Total Messages Enqueued Into The Buffered Queue | ID of Last Message Enqueued |
|--------------|--------------|---------------------------|-------------------------------------------------|-----------------------------|
| CAPTURE_SIMP |              | 1                         | 23                                              | 23                          |

**Subscriber Statistics**  
 This table lists the statistics related to the Propagation and Apply Processes that dequeue data from the buffered queue.

| Subscriber Name | Subscriber Type | Subscriber Address                          | Current Messages In Queue | Total Messages Enqueued Into The Buffered Queue | Total Messages Spilled From The Buffered Queue | Total Messages Dequeued From The Buffered Queue |
|-----------------|-----------------|---------------------------------------------|---------------------------|-------------------------------------------------|------------------------------------------------|-------------------------------------------------|
| APPLY           | APPLY           |                                             | 0                         | 0                                               | 0                                              | 0                                               |
|                 | PROPAGATE       | "STRMADMIN"."STREAMS_QUEUE"@I12.EXAMPLE.COM | 1                         | 23                                              | 22                                             | 22                                              |

[Propagation Statistics](#) [Message Statistics](#) **Buffered Queue Statistics**

この「バッファ・キューの統計」サブページには、選択したキューのバッファ・キュー部分に関する次の情報が表示されます。

- キュー内の変更の数およびメモリーからディスクにあふれた変更の数に関する情報。
- ローカル取得プロセスによってエンキューされた変更の数。取得プロセスがローカルの場合、「送信者キュー」フィールドは空です。
- リモート取得プロセスによって異なるデータベースで取得され、バッファ・キューに送信された変更の数。取得プロセスがリモートの場合、「送信者キュー」フィールドには、変更の送信元のキューおよびデータベースが含まれます。
- ローカル適用プロセスによってバッファ・キューからデキューされた変更の数。
- 伝播によってバッファ・キューから別のキューに送信された変更の数。

このサブページに表示されている統計の詳細は、「ヘルプ」をクリックしてください。

---

**注意：** バッファ・キュー統計を、次の動的パフォーマンス・ビューに問い合わせることもできます。

- V\$BUFFERED\_QUEUES
  - V\$BUFFERED\_PUBLISHERS
  - V\$BUFFERED\_SUBSCRIBERS
- 

**参照：**

- 「Oracle Streams レプリケーションの概要」 (4-2 ページ)
- 「キューの変更」 (9-25 ページ)

## 取得から適用までの時間の表示

V\$STREAMS\_APPLY\_COORDINATOR 動的パフォーマンス・ビューを問い合わせ、Oracle Streams レプリケーション環境のパフォーマンスを監視できます。具体的には、このビューを問い合わせ、変更がソース・データベースで取得されてから、現在のデータベースで適用プロセスによって適用されるまでの時間を判断できます。問合せは、現在のデータベース内で各適用プロセスによって適用された最近の変更の時間を示します。Streams レプリケーション環境では、取得から適用までの時間が比較的短くなります。

一部の変更は、他の変更よりも取得、伝播および適用に時間がかかります。問合せでは、監視している変更のメッセージ数が表示され、各変更が異なるメッセージ番号を持ちます。したがって、この問合せを複数回実行して、異なる変更について取得から適用までの時間を表示できます。

**最近の変更の取得から適用までの時間を判断するには：**

1. SQL\*Plus を開き、適用プロセスを実行しているデータベースに Oracle Streams 管理者としてログインします。

SQL\*Plus の起動の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。

2. 次の問合せを実行します。

```
COLUMN APPLY_NAME HEADING 'Apply Process Name' FORMAT A30
COLUMN 'Time Between Capture and Apply' FORMAT 999999
COLUMN HWM_MESSAGE_NUMBER HEADING 'Applied Message Number' FORMAT 999999999

SELECT APPLY_NAME,
 (HWM_TIME-HWM_MESSAGE_CREATE_TIME)*86400 "Time in Seconds",
 HWM_MESSAGE_NUMBER
FROM V$STREAMS_APPLY_COORDINATOR;
```

出力は、次のようになります。

| Apply Process Name | Time in Seconds | Applied Message Number |
|--------------------|-----------------|------------------------|
| APPLY_SIMP         | 3               | 1486062                |

**参照：**

- 「Oracle Streams レプリケーションの概要」 (4-2 ページ)
- 「Oracle Streams レプリケーション環境の管理」 (5-2 ページ)

## Oracle Streams レプリケーション環境のトラブルシューティング

この項では、Oracle Streams レプリケーション環境で最も一般的な問題について説明します。これらの問題の解決方法も説明します。

次の各項で、Oracle Streams レプリケーション環境のトラブルシューティングを説明します。

- [Enterprise Manager](#) での自動化されたアラートへの対応
- [適用エラーの管理](#)
- [レプリケーション先が使用できない場合のレプリケーション環境の管理](#)

### 参照：

- [第4章「Oracle Streams を使用したデータのレプリケート」](#)
- [第5章「Oracle Streams レプリケーション環境の管理」](#)
- Oracle Streams のトラブルシューティングの詳細は、『Oracle Streams 概要および管理』を参照してください。
- Oracle Streams レプリケーション環境のトラブルシューティングの詳細は、『Oracle Streams レプリケーション管理者ガイド』を参照してください。

## Enterprise Manager での自動化されたアラートへの対応

アラートとは、潜在的に問題があったり、クリティカルなしきい値を超えた場合に発せられる警告のことです。アラートには2つのタイプがあります。

- **ステートレス**：必ずしもシステムの状態と結び付かない単一イベントを示すアラート。たとえば、取得が特定のエラーで中断したことを示すアラートはステートレス・アラートです。
- **ステートフル**：特定のシステム状態に関連するアラート。ステートフル・アラートは、通常は数値に基づき、警告レベルとクリティカル・レベルでしきい値が定義されます。たとえば、警告レベルが 85% でクリティカル・レベルが 95% の現在の Oracle Streams プール・メモリー使用率に関するアラートは、ステートフル・アラートです。

### データベースのアラートを表示するには：

1. Enterprise Manager に Oracle Streams 管理者などの管理ユーザーとしてログインします。
2. 管理するデータベースの「データベース」ホームページに移動します。
3. 「アラート」セクションで、データベースのアラートを表示します。「アラート」セクションを表示するには下にスクロールすることが必要な場合があります。

Oracle Streams には独自のアラートのセットがあります。表 5-1 で、Oracle Streams に関連したアラートを説明します。

表 5-1 Oracle Streams アラート

| アラート                    | メッセージ                                                                                                  | アラート・タイプ |
|-------------------------|--------------------------------------------------------------------------------------------------------|----------|
| 取得の中断アラート               | STREAMS の取得プロセス <i>capture_name</i> は、ORA-error_number で中断しました                                         | ステートレス   |
| 伝播の中断アラート               | STREAMS の伝播プロセス <i>source_queue</i> 、 <i>destination_queue</i> 、 <i>database_link</i> は、16 の失敗後に中断しました | ステートレス   |
| 適用の中断アラート               | STREAMS の適用プロセス <i>apply_name</i> は、ORA-error_number で中断しました                                           | ステートレス   |
| 適用エラー・アラート              | 適用プロセス <i>apply_name</i> の STREAMS エラー・キューには、ORA-error_number となる新しいトランザクションがあります                      | ステートレス   |
| Oracle Streams プール・アラート | アラート・インフラストラクチャによって自動的に設定されます                                                                          | ステートフル   |

---

---

**注意：** Oracle Streams アラートは情報のみです。Oracle Streams アラートを管理する必要はありません。Oracle Streams 環境を定期的に監視し、問題発生時に対処する場合、Oracle Streams アラートを監視する必要はありません。

---

---

**参照：**

- 「Oracle Streams レプリケーション環境のトラブルシューティング」 (5-26 ページ)
- 『Oracle Streams 概要および管理』
- 『Oracle Streams レプリケーション管理者ガイド』

### 取得の中断アラート

このアラートは、クリティカル・エラーを示します。取得プロセスは停止し、取得プロセスに依存するレプリケーションもすべて停止します。また、取得プロセスは、再起動するまで REDO ログのスキャンをこれ以上実行しません。

#### レスポンス

次のいずれかの方法で正確なエラー・メッセージを取得します。

- Enterprise Manager での方法については、5-12 ページの「取得プロセスの情報の表示」を参照してください。
- SQL\*Plus で、ALL\_CAPTURE ビューを問い合わせます。

エラーに適したアクションを実行します。

適切なアクションを実行した後で、次のいずれかの方法で取得プロセスを再起動します。

- Enterprise Manager での方法については、5-2 ページの「取得プロセスの起動と停止」を参照してください。
- SQL\*Plus で、DBMS\_CAPTURE\_ADM.START\_CAPTURE プロシージャを実行します。

**参照：**

- 「Enterprise Manager での自動化されたアラートへの対応」 (5-26 ページ)
- 「取得プロセスでの変更の取得の概要」 (4-3 ページ)
- 「取得プロセスの管理」 (5-2 ページ)
- 「取得プロセスの監視」 (5-12 ページ)
- 『Oracle Streams 概要および管理』
- 『Oracle Streams レプリケーション管理者ガイド』

### 伝播の中断アラート

このアラートはクリティカル・エラーを示します。伝播が停止し、通常は1つのキューから別のキューに伝播によって送信されるメッセージがソース・キューに残ります。伝播に依存するレプリケーションも停止します。最終的に、ソース・キューが非常に大きくなり、メッセージがディスクにあふれることがあります。メッセージがディスクにあふれると、Oracle Streams のパフォーマンスが低下します。

## レスポンス

次のいずれかの方法で正確なエラー・メッセージを取得します。

- Enterprise Manager での方法については、5-16 ページの「[伝播の情報の表示](#)」を参照してください。
- SQL\*Plus で、DBA\_QUEUE\_SCHEDULES ビューを問い合わせます。

エラーに適したアクションを実行します。

適切なアクションを実行した後で、次のいずれかの方法で伝播を再起動します。

- Enterprise Manager での方法については、5-4 ページの「[伝播の有効化および無効化](#)」を参照してください。
- SQL\*Plus で、DBMS\_PROPAGATION\_ADM.START\_PROPAGATION プロシージャを実行します。

### 参照：

- 「[Enterprise Manager での自動化されたアラートへの対応](#)」 (5-26 ページ)
- 「[データベース間の変更の伝播の概要](#)」 (4-6 ページ)
- 「[伝播の監視](#)」 (5-16 ページ)
- 『Oracle Streams 概要および管理』
- 『Oracle Streams レプリケーション管理者ガイド』

## 適用の中断アラート

このアラートはクリティカル・エラーを示します。適用プロセスが停止し、通常は適用プロセスによってデキューされるメッセージが適用プロセス・キューに残ります。適用プロセスに依存するレプリケーションも停止します。最終的に、適用プロセス・キューが非常に大きくなり、メッセージがディスクにあふれることがあります。適用プロセス・キューにメッセージを送信する他のキューも大きくなり、メッセージがディスクにあふれることがあります。メッセージがディスクにあふれると、Oracle Streams のパフォーマンスが低下します。

## レスポンス

次のいずれかの方法で正確なエラー・メッセージを取得します。

- Enterprise Manager での方法については、5-20 ページの「[適用プロセスの情報の表示](#)」を参照してください。
- SQL\*Plus で、ALL\_APPLY ビューを問い合わせます。

エラーに適したアクションを実行します。エラーが ORA-26714 エラーの場合は、DISABLE\_ON\_ERROR 適用プロセス・パラメータを N に設定して、今後のユーザー・エラーでの中断を回避することを検討してください。方法については、5-7 ページの「[適用プロセスのパラメータの設定](#)」を参照してください。

適切なアクションを実行した後で、次のいずれかの方法で適用プロセスを再起動します。

- Enterprise Manager での方法については、5-6 ページの「[適用プロセスの起動と停止](#)」を参照してください。
- SQL\*Plus で、DBMS\_APPLY\_ADM.START\_APPLY プロシージャを実行します。

### 参照：

- 「[Enterprise Manager での自動化されたアラートへの対応](#)」 (5-26 ページ)
- 『Oracle Streams 概要および管理』
- 『Oracle Streams レプリケーション管理者ガイド』

## 適用エラー・アラート

このアラートは、適用プロセスがトランザクションを適用したときにエラーが発生したことを示します。適用プロセスは、トランザクション内のすべてのメッセージをエラー・キューに移動します。エラー・トランザクションに依存する他のトランザクションも適用エラーになる可能性があり、エラー・キューがすぐに大きくなる可能性があります。したがって、管理者はできるだけ早く適用エラーを解決する必要があります。

### レスポンス

エラー・キューの適用エラーは、次のいずれかの方法で管理します。

- Enterprise Manager での方法については、5-30 ページの「[適用エラーの管理](#)」を参照してください。
- SQL\*Plus で、ALL\_APPLY\_ERROR ビューを問い合わせ、エラーを表示して解決し、DBMS\_APPLY\_ADM パッケージのプロシージャを使用してエラーを実行または削除します。

#### 参照:

- 「[Enterprise Manager での自動化されたアラートへの対応](#)」 (5-26 ページ)
- 「[変更の適用の概要](#)」 (4-6 ページ)
- 「[適用プロセスの管理](#)」 (5-6 ページ)
- 「[適用プロセスの監視](#)」 (5-19 ページ)
- 『Oracle Streams 概要および管理』
- 『Oracle Streams レプリケーション管理者ガイド』

## Oracle Streams プール・アラート

メトリックとは、システムの状態の報告に使用される測定の単位です。Oracle Streams プールのメモリー使用率が STREAMS\_POOL\_USED\_PCT メトリックで指定された割合を超えた場合、このアラートが生成されます。

このアラートは、データベースで自動メモリー管理または自動共有メモリー管理を使用していない場合のみ発生します。具体的には、このアラートは、次の初期化パラメータが 0 に設定されているか、未設定の場合にのみ発生します。

- MEMORY\_TARGET
- SGA\_TARGET

---

**注意:** 自動メモリー管理または自動共有メモリー管理を使用することをお勧めします。

---

### レスポンス

メトリックしきい値は、アラート・インフラストラクチャで自動的に、または Enterprise Manager を使用して設定できます。現在実行中のワークロードが標準的な場合は、Oracle Streams プールのサイズを増やすことを検討してください。「データベース」ホームページのヘッダーにある、関連項目の一部のリンクからメトリックを管理できます。メトリックしきい値の表示および設定の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。

#### 参照:

- 「[Enterprise Manager での自動化されたアラートへの対応](#)」 (5-26 ページ)
- 「[Oracle Streams レプリケーションの準備](#)」 (4-21 ページ)
- 『Oracle Streams 概要および管理』

## 適用エラーの管理

適用エラーは様々な理由で発生します。トランザクションでの変更が原因で適用エラーが発生した場合、適用プロセスによって、その変更だけでなく同じトランザクションの他の変更もすべてエラー・キューに移動されます。適用エラーが発生した場合は、その原因を特定して修正してください。特定のエラー・トランザクションだけ、または1つの適用プロセスのすべてのエラー・トランザクションを再試行できます。

次の各項で、適用エラーの管理について説明します。

- [データベース・オブジェクトでの適用エラーの修正](#)
- [適用エラー・トランザクションの再試行または削除](#)

### 参照：

- [「変更の適用の概要」](#) (4-6 ページ)
- [「適用プロセスの管理」](#) (5-6 ページ)
- [「適用プロセスの監視」](#) (5-19 ページ)
- 『Oracle Streams 概要および管理』

## データベース・オブジェクトでの適用エラーの修正

エラー・トランザクションを再試行する前に、データベース・オブジェクトに対してデータ操作言語 (DML) またはデータ定義言語 (DDL) の変更を実行し、1つ以上の適用エラーの原因となった状況を修正する必要がある場合があります。データベース・オブジェクトに対する変更を行うが、その変更をレプリケートしたくない場合は、変更を行うセッションにセッション・タグを設定します。

### たとえば、hr.employees 表を更新して適用エラーを修正するには：

1. SQL\*Plus を開き、hr.employees 表を更新できるユーザーとして接続します。

SQL\*Plus の起動の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。

2. セッション・タグを設定します。

```
BEGIN
 DBMS_STREAMS.SET_TAG(
 tag => HEXTORAW('1D'));
END;
/
```

取得プロセスと同期取得によって変更が取得されないようにする値をタグに設定してください。

3. hr.employees 表を更新してエラーを解決します。
4. SQL\*Plus セッションを終了します。

1つ以上のエラー・トランザクションの原因となった問題を解決した後で、エラー・トランザクションを再試行または削除できます。方法については、5-31 ページの [「適用エラー・トランザクションの再試行または削除」](#) を参照してください。

特定の適用エラーとその修正方法の詳細は、『Oracle Streams レプリケーション管理者ガイド』を参照してください。



## 適用エラー・トランザクションの再試行または削除

5-30 ページの「データベース・オブジェクトでの適用エラーの修正」では、1つ以上のエラー・トランザクションの原因となった問題を解決する方法を説明します。問題を解決した後で、エラー・トランザクションを再試行または削除できます。

- トランザクションの変更を宛先表に行う必要がある場合は、トランザクションを再試行します。再試行するのは、これらの変更を行うために表のデータをまだ変更していない場合のみです。
- トランザクションのすべての変更を宛先表に行った場合、または表に変更を加えない場合は、トランザクションを削除します。

### 適用エラー・トランザクションを再試行または削除するには：

1. Oracle Enterprise Manager で、Oracle Streams 管理者としてハブ・データベースにログインします。
2. 「データベース」 ホームページに移動します。
3. 「データ移動」 をクリックして「データ移動」 サブページを開きます。
4. 「Streams」 セクションの「管理」 をクリックします。  
「Streams」 ページが表示され、「概要」 サブページが表示されます。
5. 「適用」 をクリックして「適用」 サブページを開きます。

| Select                              | Name       | Source Database | Positive Rule Set | Negative Rule Set | Queue         | Handlers | Apply Captured | Apply Tag | Status  | State            |
|-------------------------------------|------------|-----------------|-------------------|-------------------|---------------|----------|----------------|-----------|---------|------------------|
| <input checked="" type="checkbox"/> | APPLY_SIMP | III.EXAMPLE.COM | RULESET\$ 3       | n/a               | STREAMS_QUEUE | n/a      | YES            | 00        | ENABLED | DEQUEUE MESSAGES |

適用エラーによって、適用プロセスが中断する場合があります。中断すると、適用プロセスのステータスが ABORTED に変更されます。

6. エラー・トランザクションのある適用プロセスを選択します。
7. 「エラー」 をクリックして「適用エラー」 ページを開きます。

| Select                   | Local Transaction ID | Message Number | Message Count | Commit SCN | Source Database | Error Number | Error Message                                                                                                                                 | View Error LCRs |
|--------------------------|----------------------|----------------|---------------|------------|-----------------|--------------|-----------------------------------------------------------------------------------------------------------------------------------------------|-----------------|
| <input type="checkbox"/> | 17.6.477             | 1              | 19            | 949065     | III.EXAMPLE.COM | 26786        | ORA-26786: A row with key ("JOB_ID") = (AD_PRE\$) exists but has conflicting column(s) "MIN_SALARY" in table HR_JOBS ORA-01403: no data found |                 |

エラー・トランザクションの「**エラー LCR の表示**」フィールドのアイコンをクリックして、エラーに関する詳細情報を表示できます。詳細情報は「エラー LCR の表示」ページに表示され、トランザクションの各変更が含まれます。

Database Instance: database > Streams > Apply Errors > Logged in As STRMADMIN

View Error LCRs: APPLY\_SIMP

View LCRs

Compare Values

| Select                           | Message Number | Source Database | Object Name | Owner | Operation Type | LCR Type |
|----------------------------------|----------------|-----------------|-------------|-------|----------------|----------|
| <input checked="" type="radio"/> | 1              | III.EXAMPLE.COM | JOBS        | HR    | UPDATE         | DML      |
| <input type="radio"/>            | 2              | III.EXAMPLE.COM | JOBS        | HR    | UPDATE         | DML      |

さらにドリルダウンするには、行変更を選択して「**値の比較**」をクリックします。「値の比較」ページが表示され、行変更が変更の適用先となる表のデータと比較されます。現行のページの「**ヘルプ**」をクリックすると、ページの詳細が表示されます。

8. 「適用エラー」ページで、次のいずれかのアクションを実行します。
- すべてのエラー・トランザクションを再試行するには、「**すべてのエラーの再試行**」をクリックします。
  - すべてのエラー・トランザクションを削除するには、「**すべて削除**」をクリックします。
  - 特定のエラー・トランザクションを再試行するには、エラー・トランザクションを選択し、「**エラーの再試行**」をクリックします。
  - 特定のエラー・トランザクションを削除するには、エラー・トランザクションを選択し、「**削除**」をクリックします。

---

**注意：** DBMS\_APPLY\_ADM パッケージの次のプロシージャを使用して、エラー・トランザクションを削除または再実行することもできます。

- DELETE\_ALL\_ERRORS
  - DELETE\_ERROR
  - EXECUTE\_ALL\_ERRORS
  - EXECUTE\_ERROR
- 

## レプリケーション先が使用できない場合のレプリケーション環境の管理

Oracle Streams レプリケーション環境の宛先キューが、伝播された変更の受入れを停止することがあります。宛先キューは、たとえばキューを含むデータベースが停止した場合、宛先キューに問題がある場合、キューを含むデータベースを実行しているシステムが停止した場合などに変更の受入れを停止することがあります。

取得プロセスを使用するレプリケーション環境で宛先が使用不能の場合は、宛先キューに送信できない取得済の変更がソース・キューに残ります。ソース・キューのサイズが増大し、最終的に変更がバッファ・キュー・メモリーからディスクにあふれます。この状況が発生した場合は、Oracle Streams レプリケーション環境のパフォーマンスが低下します。

大量の取得済変更がディスクにあふれているかどうかを判断するには、5-14 ページの「**取得プロセスに関する統計情報の表示**」の手順に従います。「**キュー統計**」グラフは、取得プロセス・キュー内の、ディスクにあふれている現在の変更数を示します。

レプリケーション環境で変更を取得するための取得プロセスを使用する場合は、DBMS\_STREAMS\_ADM パッケージの SPLIT\_STREAMS プロシージャおよび MERGE\_STREAMS\_JOB プロシージャを使用して、使用できない宛先を管理できます。

**使用できない宛先を管理するには：**

1. SPLIT\_STREAMS プロシージャを使用して、問題のある宛先のストリームを、取得プロセスから他の宛先に流れている他のすべてのストリームから分離します。

SPLIT\_STREAMS プロシージャは、取得プロセス、キューおよび伝播のクローンを作成し、これらのコンポーネントのクローン・バージョンが、分離されたストリームによって使用されます。変更を伝播できないストリームは分離されますが、他の宛先へのストリームは通常どおりに処理されます。

2. 宛先を使用不能にしている問題を解決します。
3. MERGE\_STREAMS\_JOB プロシージャを使用して、分離されたストリームを元の取得プロセスにマージします。MERGE\_STREAMS\_JOB は、SPLIT\_STREAMS プロシージャで作成されたマージ・ジョブによって自動的に実行できます。マージ操作が完了すると、元のレプリケーション環境がリストアされます。

**参照：**

- 分離およびマージの使用の詳細は、『Oracle Streams レプリケーション管理者ガイド』を参照してください。
- 『Oracle Database PL/SQL パッケージ・プロシージャおよびタイプ・リファレンス』
- [第 4 章「Oracle Streams を使用したデータのレプリケート」](#)



---

## Oracle Streams レプリケーション環境の拡張

この章では、データベース・オブジェクトまたはデータベースを追加することによる Oracle Streams レプリケーション環境の拡張について説明します。

この章は次の項で構成されています。

- [Oracle Streams レプリケーション環境の拡張の概要](#)
- [チュートリアル: レプリケーション環境へのデータベース・オブジェクトの追加](#)
- [チュートリアル: レプリケーション環境へのデータベースの追加](#)

**参照:**

- [第4章「Oracle Streams を使用したデータのレプリケート」](#)
- [第5章「Oracle Streams レプリケーション環境の管理」](#)
- [「Oracle Streams でのデータのレプリケート時期」 \(1-5 ページ\)](#)
- [『Oracle Streams 概要および管理』](#)
- [『Oracle Streams レプリケーション管理者ガイド』](#)

## Oracle Streams レプリケーション環境の拡張の概要

組織でのニーズが変化すると、Oracle Streams レプリケーション環境の拡張が必要な場合があります。Oracle Streams レプリケーション環境は、データベース・オブジェクトまたはデータベースを追加することによって拡張することができます。

Oracle Streams レプリケーション環境の拡張には2つの方法があります。

- [DBMS\\_STREAMS\\_ADM](#) パッケージの単一の構成プロシージャを使用
- 複数の手順で Oracle Streams コンポーネントを個別に追加

### DBMS\_STREAMS\_ADM パッケージの単一の構成プロシージャを使用

Oracle Streams レプリケーション環境を拡張する最も簡単な方法は、[DBMS\\_STREAMS\\_ADM](#) パッケージにある次のいずれかのプロシージャを実行することです。

- [MAINTAIN\\_GLOBAL](#) プロシージャでは、データベースのオブジェクトすべてに変更をレプリケートする環境にデータベースを追加できます。
- [MAINTAIN\\_SCHEMAS](#) プロシージャでは、レプリケーション環境の既存データベースに1つ以上の新規スキーマを追加、または現在レプリケートされているスキーマをレプリケートする新規データベースを追加できます。
- [MAINTAIN\\_SIMPLE\\_TTS](#) プロシージャでは、既存のレプリケーション環境に新規の単純表領域を追加、または現在レプリケートされている単純表領域をレプリケートする新規データベースを追加できます。
- [MAINTAIN\\_TABLES](#) プロシージャでは、レプリケーション環境の既存データベースに1つ以上の新規表を追加するか、現在レプリケートされている表をレプリケートする新規データベースを追加できます。
- [MAINTAIN\\_TTS](#) プロシージャでは、既存のレプリケーション環境に新規の表領域セットを追加、または現在レプリケートされている表領域セットをレプリケートする新規データベースを追加できます。

これらのプロシージャを使用して Oracle Streams レプリケーション環境を拡張するには、環境が次の条件を満たしている必要があります。

- [DBMS\\_STREAMS\\_ADM](#) パッケージにあるいずれかの構成プロシージャによって構成された、2データベース・レプリケーション環境またはハブアンドスポーク・レプリケーション環境である必要があります。これらのレプリケーション環境については、[4-11](#) ページの「[Oracle Streams レプリケーション環境の一般的なタイプの概要](#)」を参照してください。
- Oracle Streams レプリケーション環境のどのデータベースでも、同期取得は使用できません。同期取得の詳細は、[4-5](#) ページの「[同期取得での変更の取得の概要](#)」を参照してください。
- データベースを環境に追加する場合、変更を取得する各データベースはローカル取得プロセスを使用する必要があります。どのデータベースでもダウンストリーム取得プロセスは使用できません。1つ以上のデータベース・オブジェクトを環境に追加する場合、そのデータベースはローカルかダウンストリームのいずれかの取得プロセスを使用できます。ダウンストリーム取得の詳細は、[4-3](#) ページの「[取得プロセスでの変更の取得の概要](#)」を参照してください。
- レプリケーション環境にデータベース・オブジェクトを追加する場合、そのデータベース・オブジェクトは、任意のデータベースではなく、構成プロシージャの `source_database` パラメータで指定されたデータベースに存在する必要があります。
- レプリケーション環境にデータベースを追加する場合、その新規データベースには、現在の環境でレプリケートされているデータベース・オブジェクトが含まれないようにします。

環境がこれらの条件を満たす場合は、次のいずれかの項の手順を完了して環境を拡張します。

- [チュートリアル: レプリケーション環境へのデータベース・オブジェクトの追加](#)
- [チュートリアル: レプリケーション環境へのデータベースの追加](#)

**複数の手順で Oracle Streams コンポーネントを個別に追加**

DBMS\_STREAMS\_ADM パッケージの構成プロシージャを使用して Oracle Streams レプリケーション環境を拡張できない場合は、構成手順を手動で完了する必要があります。この手順には、必要なルールと Oracle Streams コンポーネントを環境に追加するなど、いくつかの構成手順が含まれます。

Oracle Streams レプリケーション環境を手動で拡張する必要がある場合は、『Oracle Streams レプリケーション管理者ガイド』を参照してください。

**参照:**

- [第 4 章「Oracle Streams を使用したデータのレプリケート」](#)
- [「ハブアンドスポーク・レプリケーション環境の概要」](#) (4-12 ページ)
- DBMS\_STREAMS\_ADM パッケージのプロシージャの詳細は、『Oracle Database PL/SQL パッケージ・プロシージャおよびタイプ・リファレンス』を参照してください。

**チュートリアル: レプリケーション環境へのデータベース・オブジェクトの追加**

この項では、DBMS\_STREAMS\_ADM パッケージの MAINTAIN\_TABLES プロシージャを使用して、既存のハブアンドスポーク・レプリケーション環境に表を追加する例を説明します。この例を完了すると、Oracle Streams レプリケーション環境によって、その環境のデータベースに追加された表に対する変更がレプリケートされます。構成プロシージャの詳細は、4-15 ページの「[Oracle Streams レプリケーションの構成プロシージャの概要](#)」を参照してください。

具体的には、この項の例では、4-39 ページの「[チュートリアル: ローカル取得プロセスでのハブアンドスポーク・レプリケーションの構成](#)」で構成したレプリケーション環境を拡張します。この構成には、次のような特徴があります。

- hr スキーマが、hub.example.com、spoke1.example.com、spoke2.example.com の各データベースでレプリケートされます。
- hub.example.com データベースが、ハブアンドスポーク環境におけるハブ・データベースであり、他のデータベースがスポーク・データベースです。
- スポーク・データベースでは、レプリケートされたスキーマへの変更が可能で、各データベースはそのローカル取得プロセスでこれらの変更を取得します。
- 競合を解消するために、各データベースでレプリケートされた各表に、更新競合ハンドラが構成されます。

この例では、環境に次の表を追加します。

- oe.orders
- oe.order\_items

この例では oe サンプル・スキーマにある表を使用します。oe サンプル・スキーマは Oracle Database とともにデフォルトでインストールされます。

---

**注意:** DBMS\_STREAMS\_ADM パッケージの構成プロシージャを使用して Oracle Streams レプリケーション環境を拡張する前に、6-2 ページの「[Oracle Streams レプリケーション環境の拡張の概要](#)」で説明されている条件を環境が満たしていることを確認してください。

---

**Oracle Streams レプリケーション環境にデータベース・オブジェクトを追加するには:**

- 次のディレクトリ・オブジェクトが存在することを確認して、それらのオブジェクトから、以前の構成に関連するすべてのファイル（データ・ポンプ・エクスポート・ダンプ・ファイルおよびエクスポート・ログ・ファイルを含む）を削除します。
  - hub.example.com データベースの hub\_dir ディレクトリ・オブジェクト。
  - spoke1.example.com データベースの spoke1\_dir ディレクトリ・オブジェクト。
  - spoke2.example.com データベースの spoke2\_dir ディレクトリ・オブジェクト。
- ハブアンドスポーク環境で、ハブ・データベースの取得プロセスを停止します。

この例では、hub.example.com データベースの取得プロセスを停止します。取得プロセスを停止しても、レプリケートされたデータベース・オブジェクトは変更に対してオープンなままにしておくことができます。これらの変更は、取得プロセスが再開されたときに取得されます。

- Oracle Enterprise Manager で、Oracle Streams 管理者としてハブ・データベースにログインします。
- 「データベース」ホームページに移動します。
- 「データ移動」をクリックして「データ移動」サブページを開きます。
- 「Streams」セクションの「管理」をクリックします。  
「Streams」ページが表示され、「概要」サブページが表示されます。
- 「取得」をクリックして「取得」サブページを開きます。

The screenshot shows the Oracle Streams management interface. At the top, it says "Database Instance: database >" and "Logged in As STRMADMIN". Below that is the "Streams" section with tabs for "Overview", "Capture", "Propagation", "Apply", and "Messaging". The "Capture" tab is selected. There are buttons for "View", "Edit", "Statistics", "Start", "Stop", and "Delete". A "Last Refresh" timestamp is shown as "May 8, 2008 7:48:26 AM PDT". Below these buttons is a table with the following columns: Select, Name, Source Database, Positive Rule Set, Negative Rule Set, Queue, State, Status, First SCN, Start SCN, Capture Type, and Error. One row is visible with the following data: CAPTURE\_SIMP, III.EXAMPLE.COM, RULESETS, n/a, STREAMS\_QUEUE, CAPTURING CHANGES, ENABLED, 683625, 683625, LOCAL, and an empty error cell.

| Select                           | Name         | Source Database | Positive Rule Set | Negative Rule Set | Queue         | State             | Status  | First SCN | Start SCN | Capture Type | Error |
|----------------------------------|--------------|-----------------|-------------------|-------------------|---------------|-------------------|---------|-----------|-----------|--------------|-------|
| <input checked="" type="radio"/> | CAPTURE_SIMP | III.EXAMPLE.COM | RULESETS          | n/a               | STREAMS_QUEUE | CAPTURING CHANGES | ENABLED | 683625    | 683625    | LOCAL        |       |

- 停止する取得プロセスを選択します。
- 「停止」をクリックします。
- 確認ページで「はい」をクリックして、取得プロセスを定位します。

**注意:** DBMS\_CAPTURE\_ADM.STOP\_CAPTURE プロシージャを使用して、取得プロセスを停止することもできます。

- SQL\*Plus を使用して、ハブ・データベースで DBMS\_STREAMS\_ADM パッケージ内の適切な構成プロシージャを実行し、各スポーク・データベースに新しいデータベース・オブジェクトを追加します。

環境に複数のスポーク・データベースがある場合は、プロシージャを複数回実行する必要があります。この例では、次の手順を完了します。

- SQL\*Plus を開き、hub.example.com データベースに Oracle Streams 管理者として接続します。

SQL\*Plus の起動の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。



- b. MAINTAIN\_TABLES プロシージャを実行して、hub.example.com と spoke1.example.com の間のレプリケーションに oe.orders 表および oe.order\_items 表を追加します。

```

DECLARE
 tables DBMS_UTILITY.UNCL_ARRAY;
BEGIN
 tables(1) := 'oe.orders';
 tables(2) := 'oe.order_items';
 DBMS_STREAMS_ADM.MAINTAIN_TABLES(
 table_names => tables,
 source_directory_object => 'hub_dir',
 destination_directory_object => 'spoke1_dir',
 source_database => 'hub.example.com',
 destination_database => 'spoke1.example.com',
 capture_name => 'capture_hns',
 capture_queue_table => 'source_hns_qt',
 capture_queue_name => 'source_hns',
 propagation_name => 'propagation_spoke1',
 apply_name => 'apply_spoke1',
 apply_queue_table => 'destination_spoke1_qt',
 apply_queue_name => 'destination_spoke1',
 bi_directional => TRUE);
END;
/

```

MAINTAIN\_TABLES プロシージャは、多くの構成タスクを実行するため時間がかかることがあります。プロシージャの実行中は、宛先データベースの指定された表にデータ操作言語 (DML) またはデータ定義言語 (DDL) 変更が行われないようにしてください。プロシージャが完了すると、新しいデータベース・オブジェクトが環境に追加され、手順 2 で停止した取得プロセスが再開します。構成プロシージャの詳細は、4-15 ページの「[Oracle Streams レプリケーションの構成プロシージャの概要](#)」を参照してください。

構成プロシージャを実行すると、進捗情報が DBA\_RECOVERABLE\_SCRIPT、DBA\_RECOVERABLE\_SCRIPT\_PARAMS、DBA\_RECOVERABLE\_SCRIPT\_BLOCKS および DBA\_RECOVERABLE\_SCRIPT\_ERRORS データ・ディクショナリ・ビューに記録されます。プロシージャがエラーによって停止した場合は、DBMS\_STREAMS\_ADM パッケージ内の RECOVER\_OPERATION プロシージャの使用方法について、『[Oracle Streams レプリケーション管理者ガイド](#)』を参照してください。

Oracle Streams コンポーネント名を指定するパラメータ値は、DBMS\_STREAMS\_ADM パッケージ内にある、レプリケーション環境を構成した構成プロシージャで指定された値と同じである必要があります。指定された Oracle Streams コンポーネント名は、取得プロセス名、キュー名、キュー表名、伝播名、適用プロセス名などです。この例では、Oracle Streams コンポーネント名は 4-39 ページの「[チュートリアル: ローカル取得プロセスでのハブアンドスポーク・レプリケーションの構成](#)」で指定した名前と一致します。

- c. MAINTAIN\_TABLES プロシージャを実行して、hub.example.com と spoke2.example.com の間のレプリケーションに oe.orders 表および oe.order\_items 表を追加します。

```

DECLARE
 tables DBMS_UTILITY.UNCL_ARRAY;
BEGIN
 tables(1) := 'oe.orders';
 tables(2) := 'oe.order_items';
 DBMS_STREAMS_ADM.MAINTAIN_TABLES(
 table_names => tables,
 source_directory_object => 'hub_dir',
 destination_directory_object => 'spoke2_dir',

```

```

source_database => 'hub.example.com',
destination_database => 'spoke2.example.com',
capture_name => 'capture_hns',
capture_queue_table => 'source_hns_qt',
capture_queue_name => 'source_hns',
propagation_name => 'propagation_spoke2',
apply_name => 'apply_spoke2',
apply_queue_table => 'destination_spoke2_qt',
apply_queue_name => 'destination_spoke2',
bi_directional => TRUE);
END;
/

```

4. スポーク・データベースで、レプリケートされた表のインスタンス化 SCN を設定します。

---

**注意:** この例でこの手順が必要なのは、MAINTAIN\_TABLES プロシージャの実行前に、レプリケートされた表がスポーク・データベースに存在していたためです。MAINTAIN\_TABLES プロシージャの実行前に、レプリケートされた表がスポーク・データベースに存在していなかった場合は、プロシージャによって、レプリケートされた表にインスタンス化 SCN が設定されるため、この手順は必要ありません。

---

- a. SQL\*Plus で、hub.example.com データベースに Oracle Streams 管理者として接続します。

SQL\*Plus でのデータベースへの接続の詳細は、『Oracle Database 管理者ガイド』を参照してください。

- b. spoke1.example.com データベースで oe.orders 表のインスタンス化 SCN を設定します。

```

DECLARE
 iscn NUMBER; -- Variable to hold instantiation SCN value
BEGIN
 iscn := DBMS_FLASHBACK.GET_SYSTEM_CHANGE_NUMBER();
 DBMS_APPLY_ADM.SET_TABLE_INSTANTIATION_SCN@spoke1.example.com(
 source_object_name => 'oe.orders',
 source_database_name => 'hub.example.com',
 instantiation_scn => iscn);
END;
/

```

- c. spoke1.example.com データベースで oe.order\_items 表のインスタンス化 SCN を設定します。

```

DECLARE
 iscn NUMBER; -- Variable to hold instantiation SCN value
BEGIN
 iscn := DBMS_FLASHBACK.GET_SYSTEM_CHANGE_NUMBER();
 DBMS_APPLY_ADM.SET_TABLE_INSTANTIATION_SCN@spoke1.example.com(
 source_object_name => 'oe.order_items',
 source_database_name => 'hub.example.com',
 instantiation_scn => iscn);
END;
/

```

- d. spoke2.example.com データベースで oe.orders 表のインスタンス化 SCN を設定します。

```
DECLARE
 iscn NUMBER; -- Variable to hold instantiation SCN value
BEGIN
 iscn := DBMS_FLASHBACK.GET_SYSTEM_CHANGE_NUMBER();
 DBMS_APPLY_ADM.SET_TABLE_INSTANTIATION_SCN@spoke2.example.com(
 source_object_name => 'oe.orders',
 source_database_name => 'hub.example.com',
 instantiation_scn => iscn);
END;
/
```

- e. spoke2.example.com データベースで oe.order\_items 表のインスタンス化 SCN を設定します。

```
DECLARE
 iscn NUMBER; -- Variable to hold instantiation SCN value
BEGIN
 iscn := DBMS_FLASHBACK.GET_SYSTEM_CHANGE_NUMBER();
 DBMS_APPLY_ADM.SET_TABLE_INSTANTIATION_SCN@spoke2.example.com(
 source_object_name => 'oe.order_items',
 source_database_name => 'hub.example.com',
 instantiation_scn => iscn);
END;
/
```

5. hub.example.com、spoke1.example.com および spoke2.example.com の各データベースの oe スキーマにある orders 表および order\_items 表に対して最新時刻競合解消を構成します。方法については、4-55 ページの「チュートリアル: 表の最新時刻競合解消の構成」を参照してください。

**参照:**

- 第 4 章「Oracle Streams を使用したデータのレプリケート」

## チュートリアル: レプリケーション環境へのデータベースの追加

この項では、DBMS\_STREAMS\_ADM パッケージの MAINTAIN\_SCHEMAS プロシージャを使用して、既存のハブアンドスポーク・レプリケーション環境に新しいスポーク・データベースを追加する例を説明します。この例を完了すると、Oracle Streams レプリケーション環境によって、新しいデータベースのスキーマに対する変更がレプリケートされます。構成プロシージャの詳細は、4-15 ページの「Oracle Streams レプリケーションの構成プロシージャの概要」を参照してください。

具体的には、この項の例では、4-39 ページの「チュートリアル: ローカル取得プロセスでのハブアンドスポーク・レプリケーションの構成」で構成したレプリケーション環境を拡張します。この構成には、次のような特徴があります。

- hr スキーマが、hub.example.com、spoke1.example.com、spoke2.example.com の各データベースでレプリケートされます。
- hub.example.com データベースが、ハブアンドスポーク環境におけるハブ・データベースであり、他のデータベースがスポーク・データベースです。
- スポーク・データベースでは、レプリケートされたスキーマへの変更が可能で、各データベースはそのローカル取得プロセスでこれらの変更を取得します。

この例では、レプリケーション環境に spoke3.example.com データベースを追加します。

---

---

**注意:** DBMS\_STREAMS\_ADM パッケージの構成プロシージャを使用して Oracle Streams レプリケーション環境を拡張する前に、6-2 ページの「[Oracle Streams レプリケーション環境の拡張の概要](#)」で説明されている条件を環境が満たしていることを確認してください。

---

---

#### 既存の Oracle Streams レプリケーション環境にデータベースを追加するには:

- 次のタスクを完了して、新しいデータベースの環境を準備します。
  - ハブ・データベースが新しいスポーク・データベースと通信できるようにネットワーク接続を構成します。この例では、hub.example.com データベースと spoke3.example.com データベースが相互に通信できるようにネットワーク接続を構成します。

データベース間のネットワーク接続の構成の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。
  - 新しいスポーク・データベースで Oracle Streams 管理者を構成します。この例では、spoke3.example.com データベースで Oracle Streams 管理者を構成します。方法については、2-3 ページの「[チュートリアル: Oracle Streams 管理者の作成](#)」を参照してください。この例では、Oracle Streams 管理者が strmadmin であると想定します。
  - ハブ・データベースから新しいスポーク・データベースおよび新しいスポーク・データベースからハブ・データベースへのデータベース・リンクを作成します。この例では、次のデータベース・リンクを作成します。
    - hub.example.com データベースから spoke3.example.com データベースへ。データベース・リンクの名前とサービス名は、どちらも spoke3.example.com である必要があります。
    - spoke3.example.com データベースから hub.example.com データベースへ。データベース・リンクの名前とサービス名は、どちらも hub.example.com である必要があります。各データベース・リンクは、Oracle Streams 管理者のスキーマに作成する必要があります。また、各データベース・リンクは宛先データベースで Oracle Streams 管理者に接続する必要があります。方法については、2-8 ページの「[チュートリアル: データベース・リンクの作成](#)」を参照してください。
  - 新しいスポーク・データベースで初期化パラメータを正しく設定します。この例では、spoke3.example.com データベースで初期化パラメータを正しく設定します。方法については、4-21 ページの「[Oracle Streams レプリケーションの準備](#)」を参照してください。
  - ARCHIVELOG モードで実行するように新しいスポーク・データベースを構成します。取得プロセスがソース・データベースで生成された変更を取得するためには、ソース・データベースを ARCHIVELOG モードで実行する必要があります。この例では、spoke3.example.com データベースを ARCHIVELOG モードで実行する必要があります。ARCHIVELOG モードで実行するためのデータベースの構成の詳細は、『Oracle Database 管理者ガイド』を参照してください。
  - hub.example.com データベースに hub\_dir ディレクトリ・オブジェクトが存在することを確認して、そのオブジェクトから、以前の構成に関連するすべてのファイル（データ・ポンプ・エクスポート・ダンプ・ファイルおよびエクスポート・ログ・ファイルを含む）を削除します。
- SQL\*Plus を開き、spoke3.example.com データベースに Oracle Streams 管理者として接続します。

SQL\*Plus の起動の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。

3. MAINTAIN\_SCHEMAS プロシージャで生成されるファイル（インスタンス化に使用するデータ・ポンプ・エクスポート・ダンプ・ファイルを含む）を保持するディレクトリ・オブジェクトを作成します。ディレクトリ・オブジェクトは、コンピュータ・システム上の任意のアクセス可能ディレクトリを指すことができます。たとえば、次の文は、`/usr/spoke3_log_files` ディレクトリを指す `spoke3_dir` という名前のディレクトリ・オブジェクトを作成します。

```
CREATE DIRECTORY spoke3_dir AS '/usr/spoke3_log_files';
```

4. ハブアンドスポーク環境で、ハブ・データベースの取得プロセスを停止します。

この例では、`hub.example.com` データベースの取得プロセスを停止します。取得プロセスを停止しても、レプリケートされたデータベース・オブジェクトは変更に対してオープンなままにしておくことができます。これらの変更は、取得プロセスが再開されたときに取得されます。

- Oracle Enterprise Manager で、Oracle Streams 管理者としてハブ・データベースにログインします。
- 「データベース」ホームページに移動します。
- 「データ移動」をクリックして「データ移動」サブページを開きます。
- 「Streams」セクションの「管理」をクリックします。  
「Streams」ページが表示され、「概要」サブページが表示されます。
- 「取得」をクリックして「取得」サブページを開きます。

| Select                           | Name         | Source Database | Positive Rule Set | Negative Rule Set | Queue         | State             | Status  | First SCN | Start SCN | Capture Type | Error |
|----------------------------------|--------------|-----------------|-------------------|-------------------|---------------|-------------------|---------|-----------|-----------|--------------|-------|
| <input checked="" type="radio"/> | CAPTURE_SIMP | III.EXAMPLE.COM | RULESET\$ 6       | n/a               | STREAMS_QUEUE | CAPTURING CHANGES | ENABLED | 683625    | 683625    | LOCAL        |       |

- 停止する取得プロセスを選択します。
- 「停止」をクリックします。
- 確認ページで「はい」をクリックして、取得プロセスを定位します。

---

**注意：** `DBMS_CAPTURE_ADM.STOP_CAPTURE` プロシージャを使用して、取得プロセスを停止することもできます。

---

5. SQL\*Plus を使用して、ハブ・データベースで `DBMS_STREAMS_ADM` パッケージ内の適切な構成プロシージャを実行し、新しいスポーク・データベースを追加します。

この例では、次の手順を完了します。

- SQL\*Plus を開き、`hub.example.com` データベースに Oracle Streams 管理者として接続します。
- MAINTAIN\_SCHEMAS プロシージャを実行して、`spoke3.example.com` データベースを Oracle Streams レプリケーション環境に追加します。

```
BEGIN
 DBMS_STREAMS_ADM.MAINTAIN_SCHEMAS (
 schema_names => 'hr',
 source_directory_object => 'hub_dir',
 destination_directory_object => 'spoke3_dir',
 source_database => 'hub.example.com',
 destination_database => 'spoke3.example.com',
```

```

capture_name => 'capture_hns',
capture_queue_table => 'source_hns_qt',
capture_queue_name => 'source_hns',
propagation_name => 'propagation_spoke3',
apply_name => 'apply_spoke3',
apply_queue_table => 'destination_spoke3_qt',
apply_queue_name => 'destination_spoke3',
bi_directional => TRUE);
END;
/

```

MAINTAIN\_SCHEMAS プロシージャは、多くの構成タスクを実行するため時間がかかることがあります。プロシージャの実行中は、宛先データベースの指定されたスキーマのデータベース・オブジェクトにデータ操作言語 (DML) またはデータ定義言語 (DDL) 変更が行われないようにしてください。プロシージャが完了すると、新しいデータベース・オブジェクトが環境に追加され、手順 4 で停止した取得プロセスが再開します。構成プロシージャの詳細は、4-15 ページの「[Oracle Streams レプリケーションの構成プロシージャの概要](#)」を参照してください。

capture\_name、capture\_queue\_table および capture\_queue\_name で指定するパラメータ値は、DBMS\_STREAMS\_ADM パッケージ内にある、レプリケーション環境を構成した構成プロシージャで指定された値と同じである必要があります。この例では、これらのパラメータ値は 4-39 ページの「[チュートリアル: ローカル取得プロセスでのハブアンドスポーク・レプリケーションの構成](#)」で指定した値と一致します。

構成プロシージャを実行すると、進捗情報が DBA\_RECOVERABLE\_SCRIPT、DBA\_RECOVERABLE\_SCRIPT\_PARAMS、DBA\_RECOVERABLE\_SCRIPT\_BLOCKS および DBA\_RECOVERABLE\_SCRIPT\_ERRORS データ・ディクショナリ・ビューに記録されます。プロシージャがエラーによって停止した場合は、DBMS\_STREAMS\_ADM パッケージ内の RECOVER\_OPERATION プロシージャの使用方法について、『[Oracle Streams レプリケーション管理者ガイド](#)』を参照してください。

- spoke3.example.com データベースで hr スキーマ内のすべての表に対して最新時刻競合解消を構成します。このスキーマには、countries、departments、employees、jobs、job\_history、locations および regions の各表が含まれます。方法については、4-55 ページの「[チュートリアル: 表の最新時刻競合解消の構成](#)」を参照してください。

#### 参照:

- 第 4 章「[Oracle Streams を使用したデータのレプリケート](#)」

---

# マテリアライズド・ビューを使用したデータのレプリケート

この章では、マテリアライズド・ビューを使用したレプリケーションに関する概念情報を示し、データベース間のデータの連続的なレプリケート方法を説明します。

この章は次の項で構成されています。

- マテリアライズド・ビュー・レプリケーションの概要
- マテリアライズド・ビュー・レプリケーションの準備
- マテリアライズド・ビュー・サイトの構成
- マスター・サイトでのマテリアライズド・ビュー・ログの構成
- マテリアライズド・ビューを使用した読取り専用データのレプリケート
- マテリアライズド・ビューを使用した読取り / 書込みデータのレプリケート
- リフレッシュ・グループの構成

**参照：**

- 第8章「マテリアライズド・ビュー・レプリケーション環境の管理」
- 第4章「Oracle Streams を使用したデータのレプリケート」
- 第1章「データ・レプリケーションと統合の概要」

## マテリアライズド・ビュー・レプリケーションの概要

レプリケーションは、複数のデータベースでデータベース・オブジェクトとデータを共有するプロセスです。レプリケートされたデータベース・オブジェクトとデータを複数のデータベースでメンテナンスするために、あるデータベースでのこれらのデータベース・オブジェクトの1つに対する変更が他のデータベースと共有されます。このようにして、レプリケーション環境内のすべてのデータベースでデータベース・オブジェクトとデータの同期が維持されます。

外交販売など多くの環境では、データベースがネットワークに接続されていない期間があります。このような環境では、オンデマンドで、または夜間などにスケジュールした一定間隔でレプリカがデータを同期する必要があります。これらのデータベースは、ネットワークに再接続されたときに、同じデータを共有している他のデータベースと同期されます。

これらのニーズに対処するために、Oracle Database には、マテリアライズド・ビュー・レプリケーションというレプリケーション・タイプが用意されています。マテリアライズド・ビューには、ある単一の時点の表の完全コピーまたは部分コピーが含まれます。マテリアライズド・ビューは、読取り専用または更新可能にすることができます。

- **読取り専用マテリアライズド・ビュー**は、表のコピーへの読取り専用アクセスを可能にします。読取り専用マテリアライズド・ビューを使用して、アプリケーションおよびユーザーは、リモートの場所に存在する表のローカル・コピーにアクセスできます。読取り専用マテリアライズド・ビューは、データへのローカル・アクセスを可能にし、複数の場所にある同じデータに対する問合せを許可することにより、任意の単一の場所で必要なリソースを削減します。たとえば、読取り専用マテリアライズド・ビューは、通常はレポートの目的で使用されます。
- **更新可能マテリアライズド・ビュー**は、表のコピーへの読取り / 書込みアクセスを提供します。更新可能マテリアライズド・ビューを使用して、アプリケーションおよびユーザーは、表と表のコピーの両方を変更でき、これらの変更をある時点で同期できます。たとえば、更新可能マテリアライズド・ビューは、通常は地域のオフィスに製品カタログを定期的に配布するため、また営業員が顧客サイトから注文を行えるようにするために使用されます。

簡単な一括デプロイメントと切断されたコンピューティングがサポートされるため、読取り専用マテリアライズド・ビューと更新可能マテリアライズド・ビューはどちらもモバイル・コンピューティング・アプリケーションに特に適しています。

次の各項で、マテリアライズド・ビュー・レプリケーションの詳細を説明します。

- [マスター・サイト、マスター表およびマテリアライズド・ビュー・サイトの概要](#)
- [マテリアライズド・ビューのリフレッシュの概要](#)
- [リフレッシュ・グループの概要](#)

### 参照：

- 「マテリアライズド・ビューでのデータのレプリケート時期」 (1-6 ページ)
- 「マテリアライズド・ビューを使用した読取り専用データのレプリケート」 (7-8 ページ)
- 「マテリアライズド・ビューを使用した読取り / 書込みデータのレプリケート」 (7-12 ページ)



## マスター・サイト、マスター表およびマテリアライズド・ビュー・サイトの概要

レプリケーション環境では、マテリアライズド・ビューはデータを**マスター・サイト**という別のデータベース内の表と共有します。マスター・サイトのマテリアライズド・ビューに関連付けられている表は**マスター表**と呼ばれます。マテリアライズド・ビューには、ある単一の時点のマスター表の完全コピーまたは部分コピーが含まれます。マテリアライズド・ビューが存在するデータベースは**マテリアライズド・ビュー・サイト**と呼ばれます。

### 参照:

- 「マテリアライズド・ビュー・サイトの構成」(7-4 ページ)

## マテリアライズド・ビューのリフレッシュの概要

マテリアライズド・ビューをリフレッシュすると、マテリアライズド・ビューのデータがマスター表のデータと同期されます。マテリアライズド・ビューは、固定スケジュールまたはオンデマンドでのリフレッシュの実行時期を制御します。リフレッシュの際には、マスター表の行データがプル・ダウンされ、マテリアライズド・ビュー・サイトに適用されます。Oracle Database には、次のリフレッシュ方式があります。

- **高速リフレッシュ**は、最後のリフレッシュ以降に変更された行のみプル・ダウンします。
- **完全リフレッシュ**では、マテリアライズド・ビュー全体が更新されます。
- **強制リフレッシュ**では、可能な場合に高速リフレッシュが実行されます。高速リフレッシュが不可能な場合、強制リフレッシュでは、完全リフレッシュが実行されます。

高速リフレッシュは、完全リフレッシュよりも効率的です。マテリアライズド・ビューの高速リフレッシュは、マスター表に**マテリアライズド・ビュー・ログ**がある場合にのみ可能です。マテリアライズド・ビュー・ログは、マスター表に対して実行されたすべての挿入、更新および削除操作を記録するマスター・サイトの表です。

マテリアライズド・ビュー・ログは、単一のマスター表に関連付けられ、各マスター表には、マスター表からリフレッシュするマテリアライズド・ビューの数に関係なく1つのマテリアライズド・ビュー・ログしかありません。マテリアライズド・ビューに高速リフレッシュが実行されると、マテリアライズド・ビューが最後にリフレッシュされた後に追加されたマテリアライズド・ビュー・ログ内のエントリがマテリアライズド・ビューに適用されます。

### 参照:

- 「マテリアライズド・ビューのリフレッシュ」(8-2 ページ)
- 「リフレッシュ・グループの概要」(7-3 ページ)
- 「現在リフレッシュされているマテリアライズド・ビューの判断」(8-14 ページ)

## リフレッシュ・グループの概要

トランザクションで2つ以上のマテリアライズド・ビュー間に一貫性が求められる場合、マテリアライズド・ビューを**リフレッシュ・グループ**に編成できます。リフレッシュ・グループをリフレッシュすることにより、リフレッシュ・グループのマテリアライズド・ビューすべてのデータがトランザクションにおいて同じ一貫した時点に対応付けられます。リフレッシュ・グループ内のマテリアライズド・ビューを個別にリフレッシュすることもできます。ただし、マテリアライズド・ビューを個別にリフレッシュした場合、リフレッシュ・グループ内の残りのマテリアライズド・ビューはリフレッシュされないため、リフレッシュ・グループのメリットがなくなります。

### 参照:

- 「マテリアライズド・ビューのリフレッシュの概要」(7-3 ページ)
- 「リフレッシュ・グループのリフレッシュ」(8-3 ページ)

## マテリアライズド・ビュー・レプリケーションの準備

マテリアライズド・ビュー・レプリケーションを構成する前に、レプリケーション環境に参加するデータベースを準備します。

### マテリアライズド・ビュー・レプリケーションを準備するには：

1. 可能な場合は、レプリケートされた各表に主キーがあることを確認します。主キーが可能でない場合は、レプリケートされた各表に、表の各行の一意識別子として使用できる列または列のセットが必要です。レプリケーション環境で使用することを計画している表に主キーまたは一意列のセットがない場合は、これらの表を適宜変更します。
2. マテリアライズド・ビュー・レプリケーションを構成する前に、レプリケーション環境の各データベースで初期化パラメータを正しく設定します。
  - **グローバル名**：レプリケーション環境に参加する各データベースで GLOBAL\_NAMES 初期化パラメータを TRUE に設定します。2-2 ページの「GLOBAL\_NAMES 初期化パラメータを TRUE に設定」を参照してください。
  - **システム・グローバル領域 (SGA) および共有プール**：共有プールが、レプリケーション環境に対して作成されるコンポーネントを十分に収容できる大きさであることを確認します。共有プールは、SGA の一部です。MEMORY\_TARGET 初期化パラメータ (自動メモリー管理)、SGA\_TARGET 初期化パラメータ (自動共有メモリー管理) または SHARED\_POOL\_SIZE 初期化パラメータを設定することにより、共有プールを管理できます。通常、非レプリケーション環境よりもレプリケーション環境の方が Oracle Database の共有プールが大きくなります。

初期化パラメータの変更の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。

3. ネットワーク接続を構成して、レプリケーション環境内のデータベースが相互に通信できるようにします。データベース間のネットワーク接続の構成の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。

### 参照：

- 「マテリアライズド・ビューを使用した読取り専用データのレプリケート」 (7-8 ページ)
- 「マテリアライズド・ビューを使用した読取り / 書込みデータのレプリケート」 (7-12 ページ)
- レプリケーション環境での初期化パラメータの詳細は、『Oracle Database アドバンスト・レプリケーション』を参照してください。

## マテリアライズド・ビュー・サイトの構成

マテリアライズド・ビューを構成する前に、レプリケーション環境のマテリアライズド・ビュー・サイトとして機能するデータベースを構成する必要があります。この項の手順を完了すると、マテリアライズド・ビュー環境に必要なユーザーとデータベース・リンクが構成されます。また、更新可能なマテリアライズド・ビューを構成すると、これらの手順でも更新可能なマテリアライズド・ビューへの変更に対してスケジュールされたリンクおよびページを構成されます。

### Enterprise Manager でマテリアライズド・ビュー・サイトを構成するには：

1. 7-4 ページの「マテリアライズド・ビュー・レプリケーションの準備」で説明するアクションを完了します。
2. Oracle Enterprise Manager では、SYSTEM などの管理ユーザーとしてマテリアライズド・ビュー・サイトにログインします。
3. 「データベース」ホームページに移動します。

4. 「**データ移動**」をクリックして「データ移動」サブページを開きます。
5. 「アドバンスド・レプリケーション」セクションの「**設定**」をクリックします。
6. 「アドバンスド・レプリケーション：設定」ページで、「**更新可能なマテリアライズド・ビュー・レプリケーション**」を拡張します。
7. 「**レプリケーション用のマテリアライズド・ビュー・サイトの構成**」を選択します。
8. 「**続行**」をクリックして、レプリケーション用のマテリアライズド・ビュー・サイトを構成するウィザードを開きます。

**Configure Materialized View Sites for Replication: Add Replication Sites**

Enter master site information for setup. The SYSTEM account password is required at each of these sites. Cancel Step 1 of 7 Next

**Master Site**

Enter the database to be the Master Site. The SYSTEM account password is used to set up master sites and is not stored permanently.

| Host | Port | SID | Username | Password |
|------|------|-----|----------|----------|
|      |      |     | SYSTEM   |          |

**Materialized View Sites**

Enter the databases to be the Materialized View sites. The SYSTEM account password is used to set up the sites and is not stored permanently. Same site cannot be both the Materialized View Site and Master Site for a Master group.

| Host | Port | SID | Username | Password |
|------|------|-----|----------|----------|
|      |      |     | SYSTEM   |          |
|      |      |     | SYSTEM   |          |
|      |      |     | SYSTEM   |          |
|      |      |     | SYSTEM   |          |
|      |      |     | SYSTEM   |          |

Add 5 More Rows

Cancel Step 1 of 7 Next

ウィザードのページに入力して、マテリアライズド・ビュー・サイトを構成します。「ユーザーの作成」ページで、マテリアライズド・ビュー管理者の適切なパスワードを入力します。パスワードの選択については、『Oracle Database 2 日でセキュリティ・ガイド』を参照してください。ウィザードの使用については、ウィザードの各ページにある「ヘルプ」をクリックしてください。

**ヒント：** ウィザードの「スキーマの作成」ページでは、レプリケートするデータベース・オブジェクトを含んだスキーマが選択されていることを確認してください。

ウィザードを終了すると、マテリアライズド・ビュー・サイトを構成するように Enterprise Manager ジョブがスケジュールされます。ジョブが正常に実行されると、マテリアライズド・ビュー・サイトは構成され、それぞれにマテリアライズド・ビュー管理者ができます。デフォルトでは、マテリアライズド・ビュー管理者の名前は mvadmin です。ウィザードが完了すると、このユーザーに対してパスワードを指定します。各マテリアライズド・ビューには、ウィザードの「スキーマの作成」ページで指定した各スキーマが含まれています。

9. 各マテリアライズド・ビュー・サイトで、レプリケート・スキーマに必要なデータベース・リンクを作成します。

マテリアライズド・ビュー・サイトのスキーマにマテリアライズド・ビューを作成するには、スキーマは、データベース・リンクを通じて、マスター・サイト内の対応するスキーマに接続する必要があります。

データベース・リンクを作成するには：

- a. コマンドラインで SQL\*Plus を開き、レプリケート・スキーマの所有者としてマテリアライズド・ビュー・データベースに接続します。

たとえば、ii2.example.com データベース内の hr スキーマにマテリアライズド・ビューを作成することを計画している場合、次を入力します。

```
sqlplus hr@ii2.example.com
Enter password: password
```

SQL\*Plus の起動の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。

- b. マスター・データベースで、対応するレプリケート・スキーマに接続するデータベース・リンクを作成します。

たとえば、ii1.example.com データベース内の hr スキーマに接続するデータベース・リンクを作成して、次を入力します。

```
CREATE DATABASE LINK ii1.example.com CONNECT TO hr
IDENTIFIED BY password USING 'ii1.example.com';
```

- c. マテリアライズド・ビュー・データベースの各レプリケート・スキーマに対し、手順 a～b を繰り返します。
- d. 各マテリアライズド・ビュー・データベースに対して手順 a～c を繰り返します。

**参照：**

- [「マテリアライズド・ビューでのデータのレプリケート時期」](#) (1-6 ページ)
- [「マスター・サイト、マスター表およびマテリアライズド・ビュー・サイトの概要」](#) (7-3 ページ)

## マスター・サイトでのマテリアライズド・ビュー・ログの構成

マテリアライズド・ビューを高速リフレッシュするには、マスター・サイトのマスター表に対してマテリアライズド・ビュー・ログを構成します。高速リフレッシュは、前回のリフレッシュ後に変更された行のみを更新するので、完全リフレッシュより効率的です。詳細は、7-3 ページの「[マテリアライズド・ビューのリフレッシュの概要](#)」を参照してください。

**マテリアライズド・ビュー・ログを構成するには：**

1. Oracle Enterprise Manager で、SYSTEM などの管理ユーザーとしてマスター・サイトにログインします。
2. 「データベース」ホームページに移動します。
3. 「**データ移動**」をクリックして「データ移動」サブページを開きます。
4. 「アドバンスト・レプリケーション」セクションの「**設定**」をクリックします。
5. 「アドバンスト・レプリケーション：設定」ページで、「**更新可能なマテリアライズド・ビュー・レプリケーション**」を拡張します。
6. 「**高速リフレッシュ用にソース・データベース表にマテリアライズド・ビュー・ログを作成**」を選択します。

7. 「**続行**」をクリックして「マテリアライズド・ビュー・ログの作成」ページを開きます。

8. **Schema.Table** フィールドに *schema.table* の形式でスキーマ名とマスター表の表名を入力するか、懐中電灯アイコンをクリックして表を検索します。
9. 「**列の移入**」をクリックします。
10. 作成しているマテリアライズド・ビュー・ログの特定の表領域がある場合は、「**表領域**」フィールドに表領域名を入力します。それ以外の場合は、フィールドを <Default> のままにします。ページの詳細は「**ヘルプ**」を参照してください。
11. 「リフレッシュ・タイプ」で「**主キー**」が選択されていることを確認してください。
12. 「**OK**」をクリックしてマテリアライズド・ビュー・ログを作成します。
13. 各マスター表で手順 5～12 を繰り返します。

---

**注意：** SQL 文 CREATE MATERIALIZED VIEW LOG を使用して、マテリアライズド・ビュー・ログを作成することもできます。

---

**参照：**

- 「マテリアライズド・ビューのリフレッシュの概要」 (7-3 ページ)
- 「マスター・サイトのマテリアライズド・ビュー・ログの表示」 (8-20 ページ)
- 「マテリアライズド・ビュー・ログが大きくなりすぎるのを防止」 (8-23 ページ)
- 「チュートリアル: マスター・サイトでのマテリアライズド・ビュー・サポートのクリーン・アップ」 (8-8 ページ)

## マテリアライズド・ビューを使用した読取り専用データのレプリケート

表の読取り専用レプリカは、レプリケートされたデータの変更をアプリケーションに許可することなくレプリケートされたデータへのアクセスを可能にするデータベースに適しています。

次の各項で、マテリアライズド・ビューを使用した読取り専用データのレプリケートを説明します。

- [マテリアライズド・ビューを使用した読取り専用データのレプリケートの概要](#)
- [チュートリアル: マテリアライズド・ビューを使用した読取り専用データ・レプリケーションの構成](#)

**参照:**

- [「マテリアライズド・ビューでのデータのレプリケート時期」](#) (1-6 ページ)
- [「マテリアライズド・ビューを使用した読取り / 書込みデータのレプリケート」](#) (7-12 ページ)
- [「マテリアライズド・ビュー・レプリケーション環境の管理」](#) (8-1 ページ)

### マテリアライズド・ビューを使用した読取り専用データのレプリケートの概要

読取り専用マテリアライズド・ビューは、マスター・サイトで発生した表データへの読取り専用アクセスを可能にします。ネットワークの可用性にかかわらず、アプリケーションは読取り専用マテリアライズド・ビューにデータを問い合わせ、マスター・サイトへのネットワーク・アクセスを回避できます。ただし、システム全体のアプリケーションは、挿入、更新および削除操作を実行するためにマスター・サイトのデータにアクセスする必要があります。

図 7-1 に、読取り専用マテリアライズド・ビューの動作を示します。

図 7-1 読取り専用マテリアライズド・ビュー

クライアント・アプリケーション

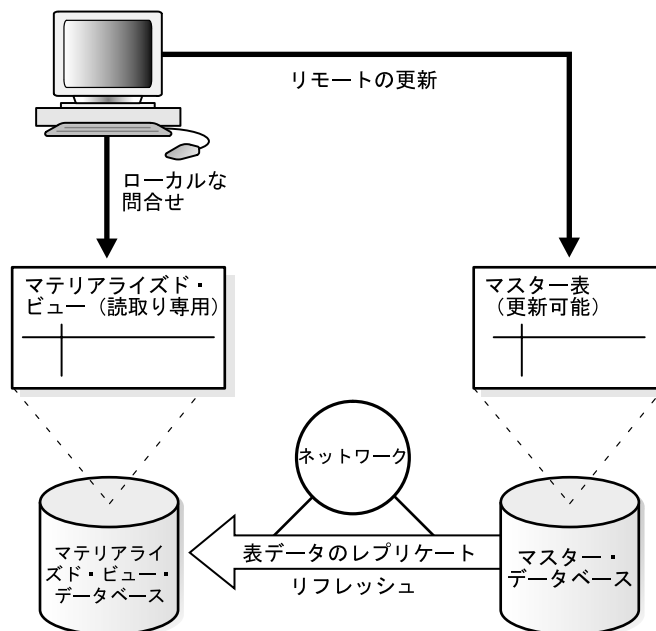


図 7-1 は、クライアント・アプリケーションがマテリアライズド・ビュー・サイトの読取り専用マテリアライズド・ビューに対してローカル問合せを実行できることを示しています。これらのアプリケーションはリモート・マスターのデータを更新できます。マテリアライズド・ビューは、マテリアライズド・ビューがネットワーク上でリフレッシュされる際にマスターでの変更で更新されます。

## チュートリアル：マテリアライズド・ビューを使用した読取り専用データ・レプリケーションの構成

この項の例では、マテリアライズド・ビューを使用した読取り専用データ・レプリケーションを構成します。具体的には、この例ではマテリアライズド・ビュー・サイトに次の読取り専用マテリアライズド・ビューを作成します。

- マスター・サイトの `hr.employees` 表に基づく `mvadmin.employees_mvr` マテリアライズド・ビュー。
- マスター・サイトの `hr.departments` 表に基づく `mvadmin.departments_mvr` マテリアライズド・ビュー。

これらの読取り専用マテリアライズド・ビューを構成するには：

1. 次の項で説明するアクションを完了します。
  - 「マテリアライズド・ビュー・レプリケーションの準備」 (7-4 ページ)
  - 「マテリアライズド・ビュー・サイトの構成」 (7-4 ページ)
  - マテリアライズド・ビューの高速リフレッシュを実行する場合は、7-6 ページの「マスター・サイトでのマテリアライズド・ビュー・ログの構成」。この例では、`hr.departments` 表と `hr.employees` 表のマテリアライズド・ビュー・ログを作成します。
2. Enterprise Manager で、マテリアライズド・ビュー・サイトにマテリアライズド・ビュー管理者としてログインします。マテリアライズド・ビュー・サイトは、マテリアライズド・ビューを作成するデータベースです。マテリアライズド・ビュー管理者のデフォルトのユーザー名は `mvadmin` です。
3. 「データベース」 ホームページに移動します。
4. 「データ移動」 をクリックして「データ移動」 サブページを開きます。
5. 「アドバンスド・レプリケーション」 セクションの「設定」 をクリックします。
6. 「アドバンスド・レプリケーション：設定」 ページで、「読取り専用マテリアライズド・ビュー」 を拡張します。
7. 「マテリアライズド・ビューの作成」 を選択します。

8. 「**続行**」をクリックして「マテリアライズド・ビューの作成」ページを開きます。


Database Instance: database > Materialized Views > Logged in As MVADMIN


### Create Materialized View

After you specify attributes for the materialized view, you can use Get Recommendation to get optimized SQL for materialized view and supporting structures. Get Recommendation Show SQL Schedule Job Cancel OK

**General** Refresh Storage Index Storage Options

\* Name

\* Schema  

Tablespace  

#### Definition

Enable the materialized view for query rewrite (Used in Data Warehousing only)

Make the materialized view updatable (Used in Advanced Replication only)

Build From Existing Table  
A table can be registered as a pre-initialized materialized view with the following restrictions: the table and the materialized view must have the same name, same column definition and belong to the same schema.

\* Materialized View Query Explain

Analyze the materialized view after it is created (Used in Data Warehousing only)

9. 次のアクションを完了して、hr.employees\_mvr マテリアライズド・ビューを作成します。
- a. 「名前」フィールドに employees\_mvr と入力します。
  - b. 「スキーマ」フィールドに hr と入力します。
  - c. 作成しているマテリアライズド・ビュー・ログの特定の表領域がある場合は、「表領域」フィールドに表領域名を入力します。それ以外の場合は、フィールドを <Default> のままにします。
  - d. 「マテリアライズド・ビューをクエリー・リライトに対して有効化」を選択解除します。
  - e. 次の項目が選択されていないことを確認します。
    - マテリアライズド・ビューを更新可能にする
    - 既存の表からの作成
  - f. 「マテリアライズド・ビュー問合せ」フィールドに次のように入力します。

```
SELECT employee_id,
 first_name,
 last_name,
 email,
 phone_number,
 hire_date,
 job_id,
 salary,
 commission_pct,
 manager_id,
 department_id
FROM hr.employees@master_site
```

master\_site を、マスター表を含むマスター・サイトのグローバル名で置換します。



- g. 「マテリアライズド・ビューを作成後に分析」を選択解除します。
- h. 必要に応じて、他のサブページでマテリアライズド・ビューのオプションを設定します。サブページの詳細は、選択したサブページで「ヘルプ」をクリックしてください。
- i. 「OK」をクリックしてマテリアライズド・ビューを作成します。

hr.employees\_mvr マテリアライズド・ビューは、マスター hr.employees 表の完全なコピーです。データのサブセットを含むマテリアライズド・ビューを作成するには、SELECT 文に WHERE 句を含めます。

- 10. 「アドバンスド・レプリケーション:設定」 ページで、「読取り専用マテリアライズド・ビュー」を拡張します。
- 11. 「マテリアライズド・ビューの作成」を選択します。
- 12. 「続行」をクリックして「マテリアライズド・ビューの作成」 ページを開きます。
- 13. 次のアクションを完了して、hr.departments\_mvr マテリアライズド・ビューを作成します。
  - a. 「名前」フィールドに departments\_mvr と入力します。
  - b. 「スキーマ」フィールドに hr と入力します。
  - c. 作成しているマテリアライズド・ビュー・ログの特定の表領域がある場合は、「表領域」フィールドに表領域名を入力します。それ以外の場合は、フィールドを <Default> のままにします。
  - d. 「マテリアライズド・ビューをクエリー・リライトに対して有効化」を選択解除します。
  - e. 次の項目が選択されていないことを確認します。
    - マテリアライズド・ビューを更新可能にする
    - 既存の表からの作成
  - f. 「マテリアライズド・ビュー問合せ」フィールドに次のように入力します。

```
SELECT department_id,
 department_name,
 manager_id,
 location_id
FROM hr.departments@master_site
```

master\_site を、マスター表を含むマスター・サイトで置換します。

- g. 「マテリアライズド・ビューを作成後に分析」を選択解除します。
- h. 必要に応じて、他のサブページでマテリアライズド・ビューのオプションを設定します。サブページの詳細は、選択したサブページで「ヘルプ」をクリックしてください。
- i. 「OK」をクリックしてマテリアライズド・ビューを作成します。

14. リフレッシュ・グループを作成し、それにマテリアライズド・ビューを追加して、リフレッシュされる、ある単一の時点でこれらが一貫するようにします。この例では、`hr.employees_mvr` および `hr.departments_mvr` マテリアライズド・ビューをリフレッシュ・グループに追加します。7-23 ページの「リフレッシュ・グループの構成」を参照してください。

---

**注意：** SQL 文 `CREATE MATERIALIZED VIEW` を使用して、マテリアライズド・ビューを作成することもできます。

---

**参照：**

- 「マテリアライズド・ビューでのデータのレプリケート時期」 (1-6 ページ)
- 「マテリアライズド・ビューを使用した読取り専用データのレプリケートの概要」 (7-8 ページ)
- 「マテリアライズド・ビュー・レプリケーション環境の管理」 (8-1 ページ)

## マテリアライズド・ビューを使用した読取り / 書込みデータのレプリケート

表の読取り / 書込みレプリカは、レプリケートされたデータの変更をアプリケーションに許可するデータベースに適しています。

次の各項で、マテリアライズド・ビューを使用した読取り / 書込みデータのレプリケートを説明します。

- マテリアライズド・ビューを使用した読取り / 書込みデータのレプリケートの概要
- マテリアライズド・ビューを使用した読取り / 書込みデータのレプリケーションの構成

**参照：**

- 「マテリアライズド・ビューでのデータのレプリケート時期」 (1-6 ページ)
- 「マテリアライズド・ビューを使用した読取り専用データのレプリケート」 (7-8 ページ)
- 第 8 章「マテリアライズド・ビュー・レプリケーション環境の管理」

## マテリアライズド・ビューを使用した読取り / 書き込みデータのレプリケートの概要

更新可能マテリアライズド・ビューでは、ユーザーがマテリアライズド・ビューの行を挿入、更新および削除できます。更新可能マテリアライズド・ビューがリフレッシュされると、これらの変更はマスター・サイトにプッシュされ、マスター表に適用されます。

図 7-2 に、更新可能マテリアライズド・ビューの動作を示します。

図 7-2 更新可能マテリアライズド・ビュー

クライアント・アプリケーション

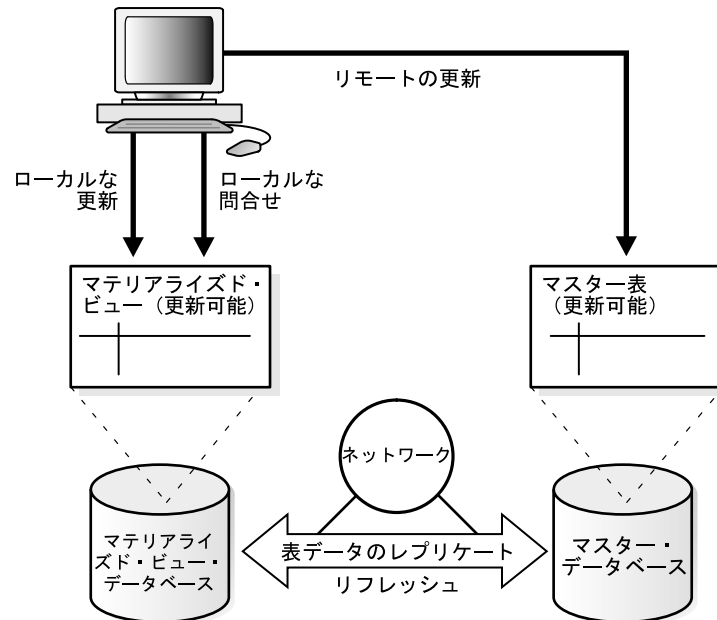


図 7-2 は、クライアント・アプリケーションがマテリアライズド・ビュー・サイトの更新可能マテリアライズド・ビューを問い合わせで更新できることを示しています。また、これらのアプリケーションはリモート・マスター・サイトのデータを更新することもできます。マテリアライズド・ビューによってネットワーク上のリフレッシュが初期化される時、リフレッシュには主に 2 つの段階があります。まず、マテリアライズド・ビュー上で実行されたトランザクションがマスター・サイトにプッシュされ、適切に適用されます。次に、更新の競合があればそれを解消した後で、変更された行データをマテリアライズド・ビューがマスター・サイトからプルして適用します。

次の各項で、マテリアライズド・ビューを使用したレプリケーションの詳細を説明します。

- レプリケーション・グループと更新可能マテリアライズド・ビューの概要
- スケジュールされたリンクと遅延トランザクションの概要
- 競合と更新可能マテリアライズド・ビューの概要

### 参照:

- 「マテリアライズド・ビューを使用した読取り / 書き込みデータのレプリケーションの構成」 (7-16 ページ)
- 第 8 章「マテリアライズド・ビュー・レプリケーション環境の管理」

## レプリケーション・グループと更新可能マテリアライズド・ビューの概要

レプリケーション・グループは、論理的に関連するレプリケーション・オブジェクトのコレクションです。関連するデータベース・オブジェクトをレプリケーション・グループ内で編成することにより、多くのオブジェクトをまとめて管理するのが簡単になります。マスター・サイトでは、レプリケーション・グループは**マスター・グループ**と呼ばれます。マテリアライズド・ビュー・サイトでは、レプリケーション・グループは**マテリアライズド・ビュー・グループ**と呼ばれます。

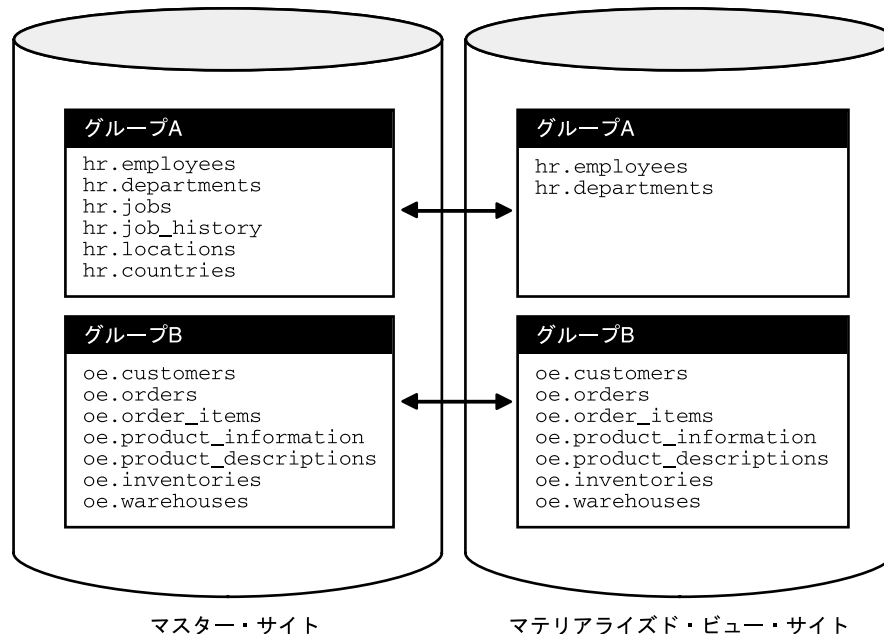
更新可能マテリアライズド・ビューに対する変更をリフレッシュ中にマスターにプッシュするには、次の条件が満たされている必要があります。

- 更新可能マテリアライズド・ビューは、マテリアライズド・ビュー・グループに属する必要があります。
- マテリアライズド・ビュー・グループは、マスター・サイトのマスター・グループに基づいている必要があります。

マテリアライズド・ビュー・グループには、マスター・グループ内のすべての表または表のサブセットのマテリアライズド・ビューを含めることができます。

図 7-3 は、2つのマテリアライズド・ビュー・グループを示しています。マテリアライズド・ビュー・グループ A には、マスター・グループの表のサブセットに対応するマテリアライズド・ビューが含まれ、マテリアライズド・ビュー・グループ B には、マスター・グループの各表のマテリアライズド・ビューが含まれます。

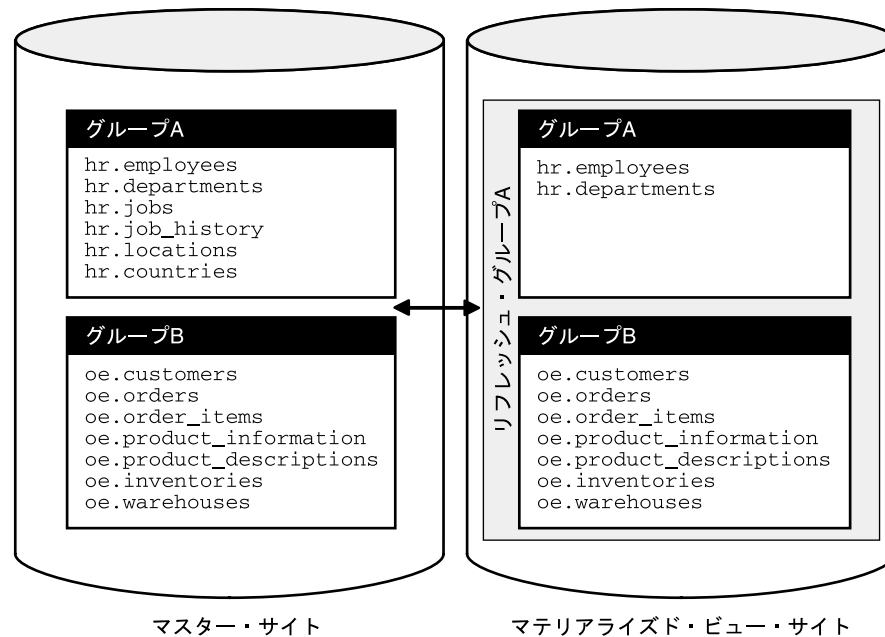
図 7-3 マスター・グループに対応するマテリアライズド・ビュー・グループ



マテリアライズド・ビュー・グループとリフレッシュ・グループを同じサイトで使用している場合、単一リフレッシュ・グループには複数のマテリアライズド・ビュー・グループを含めることができます。マテリアライズド・ビュー・グループはリフレッシュ・グループと同じではありませんが、同じマテリアライズド・ビューを含む場合があります。

図 7-4 に、2つのマテリアライズド・ビュー・グループを含むリフレッシュ・グループを示します。

図 7-4 リフレッシュ・グループは複数のマテリアライズド・ビュー・グループのオブジェクトを含むことができる



**参照:**

- 「リフレッシュ・グループの概要」 (7-3 ページ)
- 「マテリアライズド・ビューを使用した読取り / 書き込みデータのレプリケーションの構成」 (7-16 ページ)

### スケジュールされたリンクと遅延トランザクションの概要

マテリアライズド・ビュー・サイトでは、**遅延トランザクション**には更新可能マテリアライズド・ビューに対する変更が含まれます。遅延トランザクションは、マテリアライズド・ビュー・サイトに格納されるため、マスター・サイトに送信でき、マスター表に適用できます。

更新可能マテリアライズド・ビューのリフレッシュは、最初にマテリアライズド・ビュー・サイトの遅延トランザクションをマスター・サイトにプッシュします。次に、マスター・サイトのデータがプルされて、マテリアライズド・ビューに適用されます。

オプションとして、リフレッシュとは関係のない定期的な間隔で遅延トランザクションをプッシュすることを選択できます。**スケジュールされたリンク**は、遅延トランザクションをプッシュするためのユーザー定義スケジュールがあるデータベース・リンクです。スケジュールされたリンクは、マテリアライズド・ビュー・サイトが遅延トランザクション・キューをそのマスター・サイトに送信する方法を決定します。スケジュールされたリンクを作成する場合、**Oracle Database** はローカル・ジョブ・キューにジョブを作成して、マスター・サイトに遅延トランザクション・キューをプッシュします。

マテリアライズド・ビュー・サイトにマスター・サイトへの常時接続がある場合は、オプションとして、スケジュールされたリンクを使用して遅延トランザクションを定期的な間隔でマスター・サイトにプッシュできます。マテリアライズド・ビュー・サイトが長時間マスター・サイトから切断されている場合、通常は遅延トランザクションをスケジュールどおりにプッシュしないことをお勧めします。この場合は、オンデマンドでリフレッシュするのが最適です。これにより、変更がマスター・サイトにもプッシュされます。

更新可能マテリアライズド・ビューに対する変更がマスター・サイトにプッシュされた後は、マテリアライズド・ビュー・サイトに格納する必要がなくなります。遅延トランザクション・キューのサイズを維持するために、正常に完了した遅延トランザクションをページする必要があります。これらの遅延トランザクションは、**ページ・スケジュール**に従って一定の間隔でページできます。

### 参照：

- 「マテリアライズド・ビューを使用した読取り / 書込みデータのレプリケートの概要」 (7-13 ページ)
- 「マスター・サイト、マスター表およびマテリアライズド・ビュー・サイトの概要」 (7-3 ページ)
- 「マテリアライズド・ビューのリフレッシュの概要」 (7-3 ページ)
- 「マテリアライズド・ビュー・サイトの構成」 (7-4 ページ)
- 「更新可能マテリアライズド・ビューの遅延トランザクションに関する情報の表示」 (8-16 ページ)

### 競合と更新可能マテリアライズド・ビューの概要

更新可能マテリアライズド・ビューを使用する場合は、**競合**が発生する可能性があります。マスター表とマテリアライズド・ビューの同じ行に変更がほぼ同時に加えられた場合は、競合が発生します。リフレッシュ時に、前回のリフレッシュ後にマスター表および対応するマテリアライズド・ビューで同じ行が変更された場合、マスター・サイトは競合を検出します。

デフォルトでは、Oracle Database は競合更新を必ず検出し、ログに記録します。**競合解消**を使用して、各競合がビジネス・ルールに従って解消されるようにし、すべてのサイトでデータが正しく収束するようにできます。Oracle Database には、マスター・サイトで構成できる組込みの競合解消方法があります。

### 参照：

- 「マテリアライズド・ビューを使用した読取り / 書込みデータのレプリケーションの構成」 (7-16 ページ)
- 『Oracle Database アドバンスド・レプリケーション』
- 「マテリアライズド・ビューを使用した読取り / 書込みデータのレプリケートの概要」 (7-13 ページ)
- 「レプリケーション・グループと更新可能マテリアライズド・ビューの概要」 (7-14 ページ)

## マテリアライズド・ビューを使用した読取り / 書込みデータのレプリケーションの構成

この項では、マテリアライズド・ビューを使用した読取り / 書込みデータ・レプリケーションを構成する方法を示します。具体的には、この例では次のものを構成します。

- マスター表を含むマスター・サイトのマスター・グループ。7-14 ページの「**レプリケーション・グループと更新可能マテリアライズド・ビューの概要**」を参照してください。
- マスター・サイトのマスター表に基づく更新可能マテリアライズド・ビューを含むマテリアライズド・ビュー・サイトにあるマテリアライズド・ビュー・グループ。マテリアライズド・ビューを更新可能にするには、そのマテリアライズド・ビューがマテリアライズド・ビュー・サイトのマテリアライズド・ビュー・グループ内にあり、マテリアライズド・ビュー・グループがマスター・サイトのマスター・グループに対応している必要があります。7-14 ページの「**レプリケーション・グループと更新可能マテリアライズド・ビューの概要**」を参照してください。

- マスター表のマスター・サイトでの最新のタイムスタンプ競合解消。競合解消により、マスター表および対応する更新可能マテリアライズド・ビュー内の同じ行がほぼ同時に変更された場合に、マスター表とマテリアライズド・ビューの一貫性が保たれます。7-16 ページの「[競合と更新可能マテリアライズド・ビューの概要](#)」を参照してください。

競合が発生した場合、最新のタイムスタンプ競合解消とは、最新の変更は保持され、古い変更は破棄されることを意味します。ある時点で行がマテリアライズド・ビュー内で更新されると、後でマスター表の行も更新され、マテリアライズド・ビューがリフレッシュされると、マスター表の行は自動的にマテリアライズド・ビューの行に置換されます。マテリアライズド・ビューの行がマスター表の行より先に更新されている場合、マテリアライズド・ビューの行は、リフレッシュ中にマスター表の行を置換します。

#### これらの更新可能マテリアライズド・ビューを構成するには：

1. 次の項で説明するアクションを完了します。
  - 「[マテリアライズド・ビュー・レプリケーションの準備](#)」 (7-4 ページ)
  - 「[マテリアライズド・ビュー・サイトの構成](#)」 (7-4 ページ)
  - マテリアライズド・ビューの高速リフレッシュを実行する場合は、7-6 ページの「[マスター・サイトでのマテリアライズド・ビュー・ログの構成](#)」を参照してください。マスター表になる各マスター表のマテリアライズド・ビュー・ログを構成します。
2. マスター・サイトの構成をします。マスター・サイトには、更新可能なマテリアライズド・ビューがベースとなっているマスター表を含みます。
  - a. Oracle Enterprise Manager で、SYSTEM などの管理ユーザーとしてマスター・サイトになるデータベースにログインします。
  - b. 「データベース」 ホームページに移動します。
  - c. 「**データ移動**」 をクリックして「データ移動」 サブページを開きます。
  - d. 「アドバンスド・レプリケーション」 セクションの「**設定**」 をクリックします。
  - e. 「アドバンスド・レプリケーション：設定」 ページで、「**更新可能なマテリアライズド・ビュー・レプリケーション**」 を拡張します。
  - f. 「**レプリケーション用のマスター・サイトの構成**」 を選択します。
  - g. 「**続行**」 をクリックして、レプリケーション用のマスター・サイトを構成するウィザードを開きます。

| Host                 | Port                 | SID                  | Username | Password             |
|----------------------|----------------------|----------------------|----------|----------------------|
| <input type="text"/> | <input type="text"/> | <input type="text"/> | SYSTEM   | <input type="text"/> |
| <input type="text"/> | <input type="text"/> | <input type="text"/> | SYSTEM   | <input type="text"/> |
| <input type="text"/> | <input type="text"/> | <input type="text"/> | SYSTEM   | <input type="text"/> |
| <input type="text"/> | <input type="text"/> | <input type="text"/> | SYSTEM   | <input type="text"/> |
| <input type="text"/> | <input type="text"/> | <input type="text"/> | SYSTEM   | <input type="text"/> |

ウィザードのページに入力して、マスター・サイトを構成します。ウィザードの使用  
方法の詳細は、ウィザードの各ページの「ヘルプ」をクリックしてください。

ウィザードを終了すると、マスター・サイトを構成するように Enterprise Manager  
ジョブがスケジュールされます。ジョブが正常に実行されると、マスター・サイトは  
構成され、レプリケーション管理者ができます。デフォルトでは、レプリケーション  
管理者のユーザー名は、repadmin です。ウィザードが完了したら、このユーザーの  
パスワードを指定します。

3. 最新のタイムスタンプ競合解消の各マスター表に time 列を追加します。
  - a. SQL\*Plus で、SYSTEM などの管理ユーザーとしてデータベースに接続します。別の方  
法として、時間列が追加される表を所有しているユーザーとして接続することもでき  
ます。  
  
SQL\*Plus の起動の詳細は、『Oracle Database 2 日でデータベース管理者』を参照して  
ください。
  - b. SQL 文 ALTER TABLE を使用して、表に time 列を追加します。たとえば、次の SQL  
文は、time 列を hr.departments 表に追加します。

```
ALTER TABLE hr.departments ADD (time TIMESTAMP WITH TIME ZONE);
```

4. トリガーを作成して変更が発生した時刻で各マスター表の time 列を更新します。

**ヒント：** トリガーを使用して time 列を更新するかわりに、アプリケー  
ションで表に対して列を修正または挿入するたびに、time 列を移入でき  
ます。

- a. Oracle Enterprise Manager で、SYSTEM などの管理者ユーザーとしてマスター表を含  
むデータベースにログインします。  
  
手順 3 の完了後に Enterprise Manager からログアウトしなければ、管理ユーザーとし  
てマスター・サイトにログインした状態を保持できます。
- b. 「データベース」ホームページに移動します。
- c. 「スキーマ」をクリックして「スキーマ」サブページを開きます。
- d. 「プログラム」セクションで「トリガー」をクリックします。
- e. 「トリガー」ページで、「作成」をクリックします。  
  
「トリガーの作成」ページが表示され、「一般」サブページが表示されます。

Database Instance: database > Triggers > Create Trigger

Logged in As SYSTEM

Show SQL Cancel OK

General Event Advanced

\* Name

\* Schema SYSTEM

Replace If Exists  Enable

\* Trigger Body



- f. 「名前」フィールドに、トリガーの名前を入力します。
- g. 「スキーマ」フィールドに管理者ユーザー名を保持します。
- h. 「トリガー本体」フィールドに次のように入力します。

```
BEGIN
-- The IF/THEN statement ensures that the trigger does not fire during
-- materialized view refresh.
IF (DBMS_REPUTIL.FROM_REMOTE = FALSE AND
 DBMS_SNAPSHOT.I_AM_A_REFRESH = FALSE)
THEN
 :NEW.TIME := SYSTIMESTAMP;
END IF;
END;
```

- i. 「イベント」をクリックして「イベント」サブページを開きます。
- j. 「対象オブジェクト」リストで「表」が選択されていることを確認します。
- k. 「表 (Schema.Table)」フィールドに *schema.table* の形式で表名を入力するか、懐中電灯アイコンを使用して、データベース・オブジェクトを検索します。
- l. 「起動タイミング」に「前」が選択されていることを確認してください。
- m. 「イベント」に「選択」および「列の更新」を選択します。  
表内の列が表示されます。
- n. 新規の time 列を除く表のすべての列を選択します。
- o. 「詳細」をクリックします。
- p. 「行レベル・トリガー」を選択します。
- q. 「OK」をクリックしてトリガーを作成します。
- r. 各マスター表で手順 e ~ q を繰り返し、トリガーを作成します。
- s. Enterprise Manager からログアウトします。

---

**注意：** SQL 文 CREATE TRIGGER を使用して、トリガーを作成することもできます。

---

- 5. マスター・グループを作成します。
  - a. Enterprise Manager で、レプリケーション管理者としてマスター・サイトにログインします。デフォルトでは、レプリケーション管理者のユーザー名は repadmin です。
  - b. 「データベース」ホームページに移動します。
  - c. 「データ移動」をクリックして「データ移動」サブページを開きます。
  - d. 「アドバンスド・レプリケーション」セクションの「設定」をクリックします。  
「アドバンスド・レプリケーション:設定」ページが表示されます。
  - e. 「マルチマスター・レプリケーション」を拡張します。
  - f. 「マスター・グループの作成」を選択します。

- g. 「**続行**」をクリックして「マスター・グループの作成」ウィザードを開きます。

ウィザードのページに入力して、マスター・グループを構成します。ウィザードの使用  
方法の詳細は、ウィザードの各ページの「**ヘルプ**」をクリックしてください。

**ヒント:**

- 「オブジェクトの追加」ページで、マテリアライズド・ビューのマス  
ター表になる表を追加します。
- 「マスター・サイトの追加」ページで、追加のマスター・サイトを追  
加しないでください。

ウィザードを終了すると、マスター・グループを構成するように Enterprise Manager  
ジョブがスケジュールされます。

6. 各マスター表の競合解消を構成するには:
- a. マスター・サイトにレプリケーション管理者としてログインしながら、新規マス  
ター・グループを含むデータベースの「データベース」ホームページに移動します。
  - b. 「**データ移動**」をクリックして「データ移動」サブページを開きます。
  - c. 「アドバンスド・レプリケーション」セクションの「**管理**」をクリックします。  
「アドバンスド・レプリケーション:管理」ページが表示され、「概要」サブページが  
表示されます。
  - d. 「マルチマスター・レプリケーション」セクションで、「**マスター・グループ**」に関連  
付けられている数字をクリックします。
  - e. 「マスター・グループ」ページで、マスター・グループ・ステータスが NORMAL の場  
合、「**停止**」をクリックします。ステータスが QUIESCED の場合、次の手順に移動し  
ます。
  - f. コマンドラインで SQL\*Plus を開き、レプリケーション管理者としてマスター・サイ  
トに接続します。

たとえば、レプリケーション管理者が repadmin で、マスター・サイトが  
i11.example.com の場合、次を入力します。

```
sqlplus repadmin@i11.example.com
Enter password: password
```

SQL\*Plus の起動の詳細は、『Oracle Database 2 日でデータベース管理者』を参照して  
ください。

- g. SQL\*Plus で、レプリケート表の列グループを作成して、DBMS\_REPCAT パッケージで MAKE\_COLUMN\_GROUP プロシージャを実行します。

たとえば、hr.departments 表の列グループを作成して、次のプロシージャを実行します。

```
BEGIN
 DBMS_REPCAT.MAKE_COLUMN_GROUP (
 sname => 'hr',
 oname => 'departments',
 column_group => 'dep_time_cg',
 list_of_column_names => 'department_id,
 department_name,
 manager_id,
 location_id,
 time');

END;
/
```

list\_of\_columns パラメータのすべての表列を含みます。

MAKE\_COLUMN\_GROUP プロシージャの詳細は、『Oracle Database アドバンスド・レプリケーション・マネージメント API リファレンス』を参照してください。

- h. SQL\*Plus で、DBMS\_REPCAT パッケージで ADD\_UPDATE\_RESOLUTION プロシージャを実行して、表の LATEST TIME 競合解消を指定します。

たとえば、次のプロシージャを実行して、hr.departments 表の手順 g で作成した列グループを使用して LATESTTIME 競合解消を指定します。

```
BEGIN
 DBMS_REPCAT.ADD_UPDATE_RESOLUTION (
 sname => 'hr',
 oname => 'departments',
 column_group => 'dep_time_cg',
 sequence_no => 1,
 method => 'LATEST TIMESTAMP',
 parameter_column_name => 'time');

END;
/
```

column\_group パラメータの手順 g で作成した列グループを指定し、parameter\_column\_name 列の時間列を指定します。

ADD\_UPDATE\_RESOLUTION プロシージャの詳細は、『Oracle Database アドバンスド・レプリケーション・マネージメント API リファレンス』を参照してください。

- i. マスター・グループの各表で手順 g ~ h を繰り返します。
- j. Enterprise Manager で、レプリケーション管理者としてマスター・サイトにログインします。デフォルトでは、レプリケーション管理者のユーザー名は repadmin です。
- k. 「データベース」ホームページに移動します。
- l. 「データ移動」をクリックして「データ移動」サブページを開きます。
- m. 「アドバンスド・レプリケーション」セクションの「管理」をクリックします。  
「アドバンスド・レプリケーション:管理」ページが表示され、「概要」サブページが表示されます。
- n. 「マルチマスター・レプリケーション」セクションで、「マスター・グループ」に関連付けられている数字をクリックします。
- o. マスター・グループを選択します。

- p. 「**編集**」をクリックしてマスター・グループを編集するページを開きます。
  - q. 「**オブジェクト**」をクリックして「オブジェクト」サブページを開きます。
  - r. 競合解消に追加した各オブジェクトに関する「**レプリケーション・サポートの作成**」を選択していることを確認します。
  - s. 「**適用**」をクリックすると変更が保存されます。
  - t. ページ上部の「**マスター・グループ**」をクリックして「マスター・グループ」ページに戻ります。
  - u. マスター・グループの「**再開**」をクリックします。
7. マテリアライズド・ビュー・グループを作成します。
- a. Enterprise Manager で、マテリアライズド・ビュー・サイトにマテリアライズド・ビュー管理者としてログインします。マテリアライズド・ビュー・サイトは、マテリアライズド・ビューを作成するデータベースです。マテリアライズド・ビュー管理者のデフォルトのユーザー名は mvadmin です。
  - b. 「データベース」ホームページに移動します。
  - c. 「**データ移動**」をクリックして「データ移動」サブページを開きます。
  - d. 「アドバンスド・レプリケーション」セクションの「**設定**」をクリックします。
  - e. 「アドバンスド・レプリケーション:設定」ページで、「**更新可能なマテリアライズド・ビュー・レプリケーション**」を拡張します。
  - f. 「**マテリアライズド・ビュー・グループの作成**」を選択します。
  - g. 「**続行**」をクリックして「マテリアライズド・ビュー・グループの作成」ウィザードを開きます。

ウィザードのページに入力して、マテリアライズド・グループを構成します。ウィザードの使用の詳細は、ウィザードの各ページの「**ヘルプ**」をクリックしてください。

**ヒント:** 「マテリアライズド・ビューのカスタマイズ」ページの各マテリアライズド・ビューについて:

- マスター・サイトで各表が使用する競合解消の「**最小通信**」を選択します。このオプションは、競合解消メカニズムのサポートに必要なデータ量を削減します。
- 更新可能にする各マテリアライズド・ビューで「**更新可能**」を選択します。
- 高速リフレッシュを実行する各マテリアライズド・ビューで「**高速リフレッシュ**」を選択します。

ウィザードを終了すると、マテリアライズド・ビュー・グループを構成するように Enterprise Manager ジョブがスケジュールされます。

8. 変更が発生したときに最新の時刻を持つ各マテリアライズド・ビューの time 列を更新するトリガーを作成します。
  - a. Enterprise Manager で、マテリアライズド・ビュー・サイトにマテリアライズド・ビュー管理者としてログインします。マテリアライズド・ビュー・サイトは、マテリアライズド・ビューを作成するデータベースです。マテリアライズド・ビュー管理者のデフォルトのユーザー名は mvadmin です。
  - b. 各マテリアライズド・ビューに対して手順 4 の b ~ 4 の q を完了します。
9. マテリアライズド・ビュー・グループの構成中に既存のリフレッシュ・グループを指定しなかった場合は、リフレッシュ・グループを作成し、マテリアライズド・ビューをそのグループに追加して、リフレッシュされたときにこれらがある時点で一貫するようにします。7-23 ページの「リフレッシュ・グループの構成」を参照してください。

**参照：**

- 「マテリアライズド・ビューでのデータのレプリケート時期」 (1-6 ページ)
- 「マテリアライズド・ビューを使用した読取り / 書き込みデータのレプリケートの概要」 (7-13 ページ)
- 「レプリケーション・グループと更新可能マテリアライズド・ビューの概要」 (7-14 ページ)
- 「競合と更新可能マテリアライズド・ビューの概要」 (7-16 ページ)
- 第 8 章「マテリアライズド・ビュー・レプリケーション環境の管理」

## リフレッシュ・グループの構成

2 つ以上のマテリアライズド・ビューをある時点で一致させる必要がある場合は、マテリアライズド・ビューが同じリフレッシュ・グループに属している必要があります。

**リフレッシュ・グループを作成し、これにマテリアライズド・ビューを追加するには：**

1. Enterprise Manager で、マテリアライズド・ビュー・サイトにマテリアライズド・ビュー管理者としてログインします。マテリアライズド・ビュー管理者のデフォルトのユーザー名は mvadmin です。
2. 「データベース」 ホームページに移動します。
3. 「データ移動」 をクリックして「データ移動」 サブページを開きます。
4. 「アドバンスド・レプリケーション」 セクションの「管理」 をクリックします。  
「アドバンスド・レプリケーション：管理」 ページが表示され、「概要」 サブページが表示されます。
5. 「マテリアライズド・ビュー・サイト」 セクションで、「リフレッシュ・グループ」 に関連付けられている数字をクリックします。

6. 「リフレッシュ・グループ」 ページで、「作成」 をクリックします。  
「リフレッシュ・グループの作成」 ページが表示され、「一般」 サブページが表示されます。

Database Instance: database > Refresh Groups > Logged in As MVADMIN

**Create Refresh Group** Show SQL Cancel OK

**General** [Materialized Views](#)

\* Name

\* Schema

**Refresh**

\* Next Date  Change...

\* Interval  Change...

Push changes from materialized views to master before refresh

Continue to refresh despite errors

Delete group when last member deleted

**Rollback Segments**

Use default rollback segment

Segment

7. 次のアクションを完了します。
- リフレッシュ・グループの名前を「名前」フィールドに入力します。
  - 「スキーマ」フィールドにリフレッシュ・グループの所有者を入力します。通常は、マテリアライズド・ビュー管理者がリフレッシュ・グループを所有します。
  - 必要に応じて、「次の日付」および「間隔」設定を調整します。「次の日付」設定は、リフレッシュ・グループの次のリフレッシュが発生する時期を決定します。「間隔」設定は、リフレッシュ・グループが自動的にリフレッシュされる頻度を決定します。「変更」をクリックして、各設定を調整する新しいページを開きます。詳細は、「ヘルプ」をクリックしてください。
- マテリアライズド・ビュー・サイトとマスター・サイト間に制限された接続がある場合は、オンデマンドでリフレッシュできます。この場合は、「間隔」フィールドをクリックします。
- 「リフレッシュ前にマテリアライズド・ビューから変更をマスターに送信」を選択します。このオプションでは、リフレッシュ時に、マテリアライズド・ビュー・サイトがマテリアライズド・ビューのリフレッシュ前に、マスター表の更新可能なマテリアライズド・ビューに対する変更のプッシュを指定します。
8. 「マテリアライズド・ビュー」サブページを開いて、「マテリアライズド・ビュー」をクリックします。

Database Instance: database > Refresh Groups > Logged in As MVADMIN

**Create Refresh Group** Show SQL Cancel OK

[General](#) **Materialized Views**

Add

| Select Schema | Name | Status | Next Date | Job ID |
|---------------|------|--------|-----------|--------|
|               |      |        |           |        |

[General](#) **Materialized Views**

Show SQL Cancel OK

9. 「追加」をクリックして「検索と選択:マテリアライズド・ビュー」ページを開きます。
10. 検索ツールを使用して、リフレッシュ・グループに追加するマテリアライズド・ビューをリストします。
11. リフレッシュ・グループに追加するマテリアライズド・ビューを選択します。
12. 「OK」をクリックして「マテリアライズド・ビュー」サブページに戻ります。選択したマテリアライズド・ビューがリフレッシュ・グループにリストされます。
13. 「OK」をクリックして、リフレッシュ・グループを作成します。

---

---

**注意:** DBMS\_REFRESH パッケージの次のプロシージャを使用して、リフレッシュ・グループを作成し、これにオブジェクトを追加することもできます。

- MAKE
  - ADD
- 
- 

**参照:**

- 「リフレッシュ・グループの概要」 (7-3 ページ)
- 「リフレッシュ・グループのリフレッシュ」 (8-3 ページ)
- 「リフレッシュ・グループへのマテリアライズド・ビューの追加」 (8-5 ページ)
- 「リフレッシュ・グループに関する情報の表示」 (8-18 ページ)





---

# マテリアライズド・ビュー・レプリケーション 環境の管理

この章では、マテリアライズド・ビュー・レプリケーション環境の管理、監視およびトラブルシューティング方法を説明します。

この章は次の項で構成されています。

- [マテリアライズド・ビュー・レプリケーション環境の管理](#)
- [マテリアライズド・ビュー・レプリケーション環境の監視](#)
- [マテリアライズド・ビュー・レプリケーション環境のトラブルシューティング](#)

**参照：**

- [第7章「マテリアライズド・ビューを使用したデータのレプリケート」](#)
- [第1章「データ・レプリケーションと統合の概要」](#)

## マテリアライズド・ビュー・レプリケーション環境の管理

マテリアライズド・ビューが構成された後で、それらをリフレッシュしてマスター表と同期できます。この項では、マテリアライズド・ビューのリフレッシュおよびリフレッシュ・グループへのマテリアライズド・ビューの追加の方法を説明します。

マテリアライズド・ビュー・サイトでマテリアライズド・ビューが不要になることがあります。この項では、マテリアライズド・ビューの削除およびマスター・サイトでマテリアライズド・ビュー・サポートのクリーン・アップに関する情報も示します。

次の各項で、マテリアライズド・ビュー・レプリケーション環境の管理を説明します。

- [マテリアライズド・ビューのリフレッシュ](#)
- [リフレッシュ・グループへのマテリアライズド・ビューの追加](#)
- [マテリアライズド・ビューの削除](#)
- [チュートリアル: マスター・サイトでマテリアライズド・ビュー・サポートのクリーン・アップ](#)

### 参照:

- [「マテリアライズド・ビュー・レプリケーション環境の監視」](#) (8-10 ページ)
- [「マテリアライズド・ビュー・レプリケーション環境のトラブルシューティング」](#) (8-22 ページ)
- [第7章「マテリアライズド・ビューを使用したデータのレプリケート」](#)

## マテリアライズド・ビューのリフレッシュ

マテリアライズド・ビューをリフレッシュすると、マスター表と同期します。マテリアライズド・ビューを最後にリフレッシュした後でマスター表に加えられた変更は、リフレッシュ時にマテリアライズド・ビューに適用されます。マテリアライズド・ビューが更新可能な場合は、最後のリフレッシュ後にマテリアライズド・ビューに加えられた変更も、リフレッシュ時にマスター表に適用されます。

マテリアライズド・ビューのグループを1つのアプリケーションなどの特定の目的に使用している場合、通常は、これらのマテリアライズド・ビューを1つのリフレッシュ・グループに配置することが最適です。リフレッシュ・グループをリフレッシュすると、リフレッシュ・グループ内のマテリアライズド・ビューは、ある一致した時点でリフレッシュされます。マテリアライズド・ビューを個別にリフレッシュすることもできますが、その場合、マテリアライズド・ビューはある時点で一致しません。

次の各項で、マテリアライズド・ビューのリフレッシュについて説明します。

- [リフレッシュ・グループのリフレッシュ](#)
- [マテリアライズド・ビューのリフレッシュ](#)

### 参照:

- [「マテリアライズド・ビューのリフレッシュの概要」](#) (7-3 ページ)
- [「リフレッシュ・グループの概要」](#) (7-3 ページ)
- [「レプリケーション・グループと更新可能マテリアライズド・ビューの概要」](#) (7-14 ページ)

## リフレッシュ・グループのリフレッシュ

リフレッシュ・グループをリフレッシュすると、リフレッシュ・グループ内のすべてのマテリアライズド・ビューが、ある一致した時点でリフレッシュされます。

### リフレッシュ・グループをリフレッシュするには:

1. Enterprise Manager で、マテリアライズド・ビュー・サイトにマテリアライズド・ビュー管理者としてログインします。マテリアライズド・ビュー管理者のデフォルトのユーザー名は mvadmin です。
2. 「データベース」 ホームページに移動します。
3. 「データ移動」 をクリックして「データ移動」 サブページを開きます。
4. 「アドバンスド・レプリケーション」 セクションの「管理」 をクリックします。  
「アドバンスド・レプリケーション:管理」 ページが表示され、「概要」 サブページが表示されます。
5. 「マテリアライズド・ビュー・サイト」 セクションで、「リフレッシュ・グループ」に関連付けられている数字をクリックします。
6. 「リフレッシュ・グループ」 ページで、検索ツールを使用して、リフレッシュするリフレッシュ・グループをリストします。
7. リストでリフレッシュ・グループを選択します。
8. 「編集」 をクリックします。  
「リフレッシュ・グループの編集」 ページが表示され、「一般」 サブページが表示されます。

The screenshot shows the 'Edit Refresh Group' page for 'MVADMIN.MVR'. The page is divided into several sections:

- General:** Name: MVR, Schema: MVADMIN, Status: Normal, Job ID: 62.
- Refresh:**
  - \* Next Date: 9/29/06 1:45:41 PM PDT (Change...)
  - \* Interval: /\*24:hr\*/sysdate + 24/24 (Change...)
  - Push changes from materialized views to master before refresh
  - Continue to refresh despite errors
  - Delete group when last member deleted (Refresh Now)
- Rollback Segments:**
  - Use default rollback segment
  - Segment: SYSTEM

At the bottom, there are tabs for 'General' and 'Materialized Views', and an 'Actions' section with buttons for 'Create Like', 'Go', 'Show SQL', 'Revert', and 'Apply'.

9. 「リフレッシュ」 セクションの「今すぐリフレッシュ」 をクリックします。

---

**注意:** DBMS\_REFRESH.REFRESH プロシージャを使用して、リフレッシュ・グループをリフレッシュすることもできます。

---

**参照：**

- 「マテリアライズド・ビューのリフレッシュの概要」 (7-3 ページ)
- 「リフレッシュ・グループの概要」 (7-3 ページ)
- 「リフレッシュ・グループの構成」 (7-23 ページ)
- 「リフレッシュ・グループに関する情報の表示」 (8-18 ページ)

**マテリアライズド・ビューのリフレッシュ**

マテリアライズド・ビューをリフレッシュすると、マスター表と一致します。マテリアライズド・ビューがリフレッシュ・グループの一部である場合、通常は個別のマテリアライズド・ビューではなくリフレッシュ・グループをリフレッシュするのが最適です。リフレッシュ・グループをリフレッシュすると、リフレッシュ・グループ内のすべてのマテリアライズド・ビューがある時点で一致します。

**マテリアライズド・ビューをリフレッシュするには：**

1. Enterprise Manager で、マテリアライズド・ビュー・サイトにマテリアライズド・ビュー管理者としてログインします。マテリアライズド・ビュー管理者のデフォルトのユーザー名は mvadmin です。
2. 「データベース」 ホームページに移動します。
3. 「データ移動」 をクリックして「データ移動」 サブページを開きます。
4. 「アドバンスド・レプリケーション」 セクションの「管理」 をクリックします。  
「アドバンスド・レプリケーション：管理」 ページが表示され、「概要」 サブページが表示されます。
5. 「マテリアライズド・ビュー・サイト」 セクションで、「マテリアライズド・ビュー」 に関連付けられている数字をクリックします。
6. 「マテリアライズド・ビュー」 ページで、検索ツールを使用して、リフレッシュするマテリアライズド・ビューをリストします。  
すべてのマテリアライズド・ビューをリストするには、「スキーマ」 フィールドと「オブジェクト名」 フィールドを空白にしたまま「実行」 をクリックします。
7. リストでマテリアライズド・ビューを選択します。

Database Instance: database > Logged in As MVIEWADMIN

**Materialized Views** Object Type: Materialized View

**Search**  
Enter a schema name and an object name to filter the data that is displayed in your results set.

Schema:

Object Name:

By default, the search returns all uppercase matches beginning with the string you entered. To run an exact or case-sensitive match, double quote the search string. You can use the wildcard symbol (%) in a double quoted string.

Selection Mode:

Actions:

| Select                           | Schema | Materialized View(Snapshot) | Master Owner | Master Table | Master Link      | Type | Updatable | Can Use Log | Last Refresh               |
|----------------------------------|--------|-----------------------------|--------------|--------------|------------------|------|-----------|-------------|----------------------------|
| <input checked="" type="radio"/> | HR     | EMPLOYEES_MV1               | HR           | EMPLOYEES    | @III.EXAMPLE.COM | FAST | YES       | YES         | May 8, 2008 2:28:31 PM PDT |
| <input type="radio"/>            | HR     | LOCATIONS_MV1               | HR           | LOCATIONS    | @III.EXAMPLE.COM | FAST | YES       | YES         | May 8, 2008 2:28:33 PM PDT |
| <input type="radio"/>            | HR     | JOBS_MV1                    | HR           | JOBS         | @III.EXAMPLE.COM | FAST | YES       | YES         | May 8, 2008 2:28:32 PM PDT |
| <input type="radio"/>            | HR     | COUNTRIES_MV1               | HR           | COUNTRIES    | @III.EXAMPLE.COM | FAST | YES       | YES         | May 8, 2008 2:28:20 PM PDT |
| <input type="radio"/>            | HR     | DEPARTMENTS_MV1             | HR           | DEPARTMENTS  | @III.EXAMPLE.COM | FAST | YES       | YES         | May 8, 2008 2:28:31 PM PDT |
| <input type="radio"/>            | HR     | REGIONS_MV1                 | HR           | REGIONS      | @III.EXAMPLE.COM | FAST | YES       | YES         | May 8, 2008 2:28:33 PM PDT |
| <input type="radio"/>            | HR     | JOB_HISTORY_MV1             | HR           | JOB_HISTORY  | @III.EXAMPLE.COM | FAST | YES       | YES         | May 8, 2008 2:28:32 PM PDT |

8. 「アクション」リストで「リフレッシュ」を選択します。
9. 「実行」をクリックして、選択したマテリアライズド・ビューの「マテリアライズド・ビューのリフレッシュ」ページを開きます。

Database Instance: database > Materialized Views > Refresh Materialized View: HR.JOBS\_MV1 Logged in As MVADMIN

Consider the materialized view fresh  
 Refresh the materialized view with data from master tables

**Refresh Type**

FORCE - Incremental refresh if possible or complete refresh if incremental refresh is not possible  
 FAST - Incremental refresh  
 COMPLETE - Complete refresh

**Rollback Segment**

Use default rollback segment  
 Segment: <Default>

**Other Options**

Continue to refresh despite errors  
 Push Changes from materialied view to maser before refresh  
 Analyze the materialized view after it is created (Used in dataware housing only)

10. 必要に応じてオプションを調整します。  
通常は、デフォルト値を使用する必要があります。このページの詳細は、「ヘルプ」をクリックしてください。
11. 「OK」をクリックします。
12. 確認ページで、「はい」をクリックしてマテリアライズド・ビューをリフレッシュします。

---

**注意：** DBMS\_MVIEW.REFRESH プロシージャを使用して、マテリアライズド・ビューをリフレッシュすることもできます。

---

**参照：**

- 「マテリアライズド・ビューのリフレッシュの概要」 (7-3 ページ)
- 「リフレッシュ・グループの概要」 (7-3 ページ)

## リフレッシュ・グループへのマテリアライズド・ビューの追加

2つ以上のマテリアライズド・ビューをある時点で一致させる必要がある場合は、マテリアライズド・ビューが同じリフレッシュ・グループに属している必要があります。

**マテリアライズド・ビューをリフレッシュ・グループに追加するには：**

1. Enterprise Manager で、マテリアライズド・ビュー・サイトにマテリアライズド・ビュー管理者としてログインします。マテリアライズド・ビュー管理者のデフォルトのユーザー名は mvadmin です。
2. 「データベース」 ホームページに移動します。

3. 「**データ移動**」をクリックして「データ移動」サブページを開きます。
4. 「アドバンスド・レプリケーション」セクションの「**管理**」をクリックします。  
「アドバンスド・レプリケーション:管理」ページが表示され、「概要」サブページが表示されます。
5. 「マテリアライズド・ビュー・サイト」セクションで、「**リフレッシュ・グループ**」に関連付けられている数字をクリックします。
6. 「グループのリフレッシュ」ページで、検索ツールを使用して、変更するリフレッシュ・グループをリストします。
7. リフレッシュ・グループを選択します。
8. 「**編集**」をクリックします。
9. 「リフレッシュ・グループの編集」ページで、「**マテリアライズド・ビュー**」をクリックします。
10. 「マテリアライズド・ビュー」サブページで、「**追加**」をクリックします。
11. 「検索と選択:マテリアライズド・ビュー」ページで、検索ツールを使用して、リフレッシュ・グループに追加するマテリアライズド・ビューをリストします。

Database Instance: database > Refresh Groups > Logged in As MWADMIN

**Search and Select: Materialized View** Cancel OK

**Search**  
To find your item, optionally enter a schema name and object name in the text field, then hit "Go" button. To see a list of all items, clear the text field and hit "Go" button

Schema  Materialized View  Go

**Results**  
[Select All](#) | [Select None](#)

| Select                   | Schema | Materialized View |
|--------------------------|--------|-------------------|
| <input type="checkbox"/> | HR     | DEPARTMENTS_MV2   |
| <input type="checkbox"/> | HR     | DEPARTMENTS_MVR   |
| <input type="checkbox"/> | HR     | EMPLOYEES_MV2     |
| <input type="checkbox"/> | HR     | EMPLOYEES_MVR     |
| <input type="checkbox"/> | HR     | JOBS_MV1          |
| <input type="checkbox"/> | HR     | JOBS_MV2          |
| <input type="checkbox"/> | HR     | JOB_HISTORY_MV1   |
| <input type="checkbox"/> | HR     | JOB_HISTORY_MV2   |
| <input type="checkbox"/> | HR     | LOCATIONS_MV1     |
| <input type="checkbox"/> | HR     | LOCATIONS_MV2     |
| <input type="checkbox"/> | HR     | REGIONS_MV1       |
| <input type="checkbox"/> | HR     | REGIONS_MV2       |

Cancel OK

12. リフレッシュ・グループに追加するマテリアライズド・ビューを選択します。
13. 「**OK**」をクリックします。

14. 「リフレッシュ・グループの編集」 ページで、「適用」をクリックしてリフレッシュ・グループへのマテリアライズド・ビューの追加を終了します。

---

**注意：** DBMS\_REFRESH.ADD プロシージャを使用して、リフレッシュ・グループにマテリアライズド・ビューを追加することもできます。

---

**参照：**

- 「マテリアライズド・ビューのリフレッシュの概要」 (7-3 ページ)
- 「リフレッシュ・グループの概要」 (7-3 ページ)
- 「リフレッシュ・グループのリフレッシュ」 (8-3 ページ)
- 「リフレッシュ・グループの構成」 (7-23 ページ)
- 「リフレッシュ・グループに関する情報の表示」 (8-18 ページ)

## マテリアライズド・ビューの削除

マテリアライズド・ビューがマテリアライズド・ビュー・サイトで不要になった場合は、マテリアライズド・ビューを削除できます。可能な場合は、マテリアライズド・ビュー・サイトがマスター・サイトに接続できるときにマテリアライズド・ビューを削除する必要があります。マテリアライズド・ビューに関する情報は、ネットワーク接続が確立されたときにマスター・サイトから自動的に削除されます。

マテリアライズド・ビュー・サイトがマスター・サイトに接続できないときにマテリアライズド・ビューを削除する必要がある場合は、マテリアライズド・ビューに関する情報をマスター・サイトから手動で削除する必要があります。マテリアライズド・ビューを削除した後でこの情報を削除する方法は、8-8 ページの「チュートリアル: マスター・サイトでのマテリアライズド・ビュー・サポートのクリーン・アップ」を参照してください。

**マテリアライズド・ビューを削除するには：**

1. Enterprise Manager で、マテリアライズド・ビュー・サイトにマテリアライズド・ビュー管理者としてログインします。マテリアライズド・ビュー管理者のデフォルトのユーザー名は mvadmin です。
2. 「データベース」 ホームページに移動します。
3. 「データ移動」をクリックして「データ移動」サブページを開きます。
4. 「アドバンスド・レプリケーション」セクションの「管理」をクリックします。  
「アドバンスド・レプリケーション: 管理」ページが表示され、「概要」サブページが表示されます。
5. 「マテリアライズド・ビュー・サイト」セクションで、「マテリアライズド・ビュー」に関連付けられている数字をクリックします。
6. 「マテリアライズド・ビュー」 ページで、検索ツールを使用して、削除するマテリアライズド・ビューをリストします。

すべてのマテリアライズド・ビューをリストするには、「スキーマ」フィールドと「オブジェクト名」フィールドを空白にしたまま「実行」をクリックします。

7. リストでマテリアライズド・ビューを選択します。

Database Instance: database > Logged in As MVIEWADMIN

**Materialized Views** Object Type: Materialized View

**Search**  
Enter a schema name and an object name to filter the data that is displayed in your results set.

Schema:

Object Name:

By default, the search returns all uppercase matches beginning with the string you entered. To run an exact or case-sensitive match, double quote the search string. You can use the wildcard symbol (%) in a double quoted string.

Selection Mode:

| Select                           | Schema | Materialized View(Snapshot) | Master Owner | Master Table | Master Link      | Type | Updatable | Can Use Log | Last Refresh               |
|----------------------------------|--------|-----------------------------|--------------|--------------|------------------|------|-----------|-------------|----------------------------|
| <input checked="" type="radio"/> | HR     | EMPLOYEES_MV1               | HR           | EMPLOYEES    | @III.EXAMPLE.COM | FAST | YES       | YES         | May 8, 2008 2:28:31 PM PDT |
| <input type="radio"/>            | HR     | LOCATIONS_MV1               | HR           | LOCATIONS    | @III.EXAMPLE.COM | FAST | YES       | YES         | May 8, 2008 2:28:33 PM PDT |
| <input type="radio"/>            | HR     | JOBS_MV1                    | HR           | JOBS         | @III.EXAMPLE.COM | FAST | YES       | YES         | May 8, 2008 2:28:32 PM PDT |
| <input type="radio"/>            | HR     | COUNTRIES_MV1               | HR           | COUNTRIES    | @III.EXAMPLE.COM | FAST | YES       | YES         | May 8, 2008 2:28:20 PM PDT |
| <input type="radio"/>            | HR     | DEPARTMENTS_MV1             | HR           | DEPARTMENTS  | @III.EXAMPLE.COM | FAST | YES       | YES         | May 8, 2008 2:28:31 PM PDT |
| <input type="radio"/>            | HR     | REGIONS_MV1                 | HR           | REGIONS      | @III.EXAMPLE.COM | FAST | YES       | YES         | May 8, 2008 2:28:33 PM PDT |
| <input type="radio"/>            | HR     | JOB_HISTORY_MV1             | HR           | JOB_HISTORY  | @III.EXAMPLE.COM | FAST | YES       | YES         | May 8, 2008 2:28:32 PM PDT |

8. 「削除」をクリックします。
9. 確認ページで、「はい」をクリックしてマテリアライズド・ビューを削除します。

**注意：** SQL 文 DROP MATERIALIZED VIEW を使用して、マテリアライズド・ビューを削除することもできます。

**参照：**

- 「マテリアライズド・ビュー・レプリケーションの概要」 (7-2 ページ)

## チュートリアル: マスター・サイトでのマテリアライズド・ビュー・サポートのクリーン・アップ

マテリアライズド・ビュー・サイトがネットワーク上でマスター・サイトに接続していなかったときにマテリアライズド・ビューを削除した場合は、マスター・サイトでマテリアライズド・ビュー・サポートをクリーン・アップする必要があります。マテリアライズド・ビュー・サポートのクリーン・アップには、次のアクションが含まれます。

- マテリアライズド・ビューの登録解除  
マテリアライズド・ビューを登録解除すると、マスター・サイトのデータ・ディクショナリ内のマテリアライズド・ビューに関する情報が削除されます。
- マテリアライズド・ビュー・ログのページまたはマテリアライズド・ビュー・ログの削除  
マテリアライズド・ビューのマスター表が他のマテリアライズド・ビューのマスター表である場合は、削除したマテリアライズド・ビューの情報のマテリアライズド・ビュー・ログをページする必要があります。マテリアライズド・ビューのマスター表が他のマテリアライズド・ビューのマスター表でない場合は、マテリアライズド・ビュー・ログを削除できます。

Oracle Database は、マテリアライズド・ビュー・ログのどの行がマテリアライズド・ビューのリフレッシュ中に使用されたかを自動的に追跡し、これらの行をログからページして、ログ・サイズが無限に増大しないようにします。複数のマテリアライズド・ビューが同じマテリアライズド・ビュー・ログを使用できるため、あるマテリアライズド・ビューのリフレッシュにすでに使用された行も、別のマテリアライズド・ビューをリフレッシュするのに必要です。Oracle Database は、すべてのマテリアライズド・ビューが行を使用するまで、その行をログから削除しません。マスター・サイトをクリーン・アップしないでマテリアライズド・ビューを削除した場合、そのマテリアライズド・ビューのマテリアライズド・ビュー・ログが非常に大きくなる可能性があります。



この項の例では、次のマテリアライズド・ビュー・サポートをクリーン・アップします。

- マスター・サイトは `ii1.example.com` です。
- マテリアライズド・ビュー・サイトは `ii2.example.com` です。
- マテリアライズド・ビューの名前は `employees_mvr` です。
- マテリアライズド・ビューの所有者は `hr` です。

#### マスター・サイトでこのマテリアライズド・ビューのサポートをクリーン・アップするには：

1. コマンドラインで SQL\*Plus を開き、マスター・サイト `ii1.example.com` にレプリケーション管理者や `SYSTEM` などの管理ユーザーとして接続します。デフォルトでは、レプリケーション管理者のユーザー名は `repadmin` です。

SQL\*Plus の起動の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。

2. マテリアライズド・ビューのマスター表のマテリアライズド・ビュー・ログをパージまたは削除します。

- マスター表が、削除されたマテリアライズド・ビュー以外のマテリアライズド・ビューのマスター表でもある場合は、削除されたマテリアライズド・ビューに関する情報のマテリアライズド・ビュー・ログをパージします。

```
BEGIN
 DBMS_MVIEW.PURGE_MVIEW_FROM_LOG (
 mviewowner => 'hr',
 mviewname => 'employees_mvr',
 mviewsite => 'ii2.example.com!');
END;
/
```

- マスター表が、削除されたマテリアライズド・ビュー以外のマテリアライズド・ビューのマスター表でない場合は、マスター表のマテリアライズド・ビュー・ログを削除します。

```
DROP MATERIALIZED VIEW LOG ON hr.employees;
```

3. マテリアライズド・ビューを登録解除します。

```
BEGIN
 DBMS_MVIEW.UNREGISTER_MVIEW (
 mviewowner => 'hr',
 mviewname => 'employees_mvr',
 mviewsite => 'ii2.example.com!');
END;
/
```

マテリアライズド・ビューの所有者、名前またはマテリアライズド・ビュー・サイトが不明な場合は、`ALL_REGISTERED_MVIEWS` データ・ディクショナリ・ビューを問い合わせることでこの情報を取得できます。

#### 参照：

- 「マテリアライズド・ビューの削除」 (8-7 ページ)
- 「マテリアライズド・ビューのリフレッシュの概要」 (7-3 ページ)
- 「マテリアライズド・ビュー・レプリケーションの概要」 (7-2 ページ)

## マテリアライズド・ビュー・レプリケーション環境の監視

この項では、Enterprise Manager と SQL\*Plus を使用してマテリアライズド・ビュー・レプリケーション環境に関する情報を表示する方法を説明します。マテリアライズド・ビュー・サイトとマスター・サイトの両方でこの情報を表示します。

次の各項で、マテリアライズド・ビュー・レプリケーション環境の監視を説明します。

- データベースのレプリケーション・コンポーネントの概要の表示
- マテリアライズド・ビューに関する情報の表示
- 現在リフレッシュされているマテリアライズド・ビューの判断
- マテリアライズド・ビュー・グループに関する情報の表示
- 更新可能マテリアライズド・ビューの遅延トランザクションに関する情報の表示
- リフレッシュ・グループに関する情報の表示
- マスター・サイトのマテリアライズド・ビュー・ログの表示
- マスター・サイトのマテリアライズド・ビューの表示

### 参照:

- 第7章「マテリアライズド・ビューを使用したデータのレプリケート」
- 「マテリアライズド・ビュー・レプリケーション環境の管理」(8-2 ページ)
- 「マテリアライズド・ビュー・レプリケーション環境のトラブルシューティング」(8-22 ページ)

## データベースのレプリケーション・コンポーネントの概要の表示

Enterprise Manager の「アドバンスド・レプリケーション:管理」ページには、データベースのレプリケーション・コンポーネントの概要が表示されます。

### データベースのレプリケーション・コンポーネントの概要を表示するには:

1. Enterprise Manager で、マテリアライズド・ビュー・サイトにマテリアライズド・ビュー管理者としてログインします。マテリアライズド・ビュー管理者のデフォルトのユーザー名は mvadmin です。
2. 「データベース」ホームページに移動します。
3. 「データ移動」をクリックして「データ移動」サブページを開きます。

## 4. 「アドバンスド・レプリケーション」セクションの「管理」をクリックします。

「アドバンスド・レプリケーション:管理」ページが表示され、「概要」サブページが表示されます。

The screenshot displays the Oracle Advanced Replication Administration interface. At the top, it indicates the database instance is 'database' and the user is logged in as 'MVADMIN'. The page title is 'Advanced Replication: Administration'. There are buttons for 'Page Refreshed', 'October 4, 2006 7:06:15 AM PDT', 'Refresh', 'View Data', and 'Manual Refresh'. The main content area is divided into three sections: 'General', 'Multimaster Replication', and 'Materialized View Replication'. Each section contains a list of items with associated counts and links. A right-hand sidebar provides an 'Overview' of the replication types.

各レプリケーション・コンポーネントは、現在のデータベースのコンポーネントの数量を示す数値に関連付けられています。数値をクリックすると、これらのコンポーネントに関する詳細情報が表示され、これらのコンポーネントを管理できます。

マテリアライズド・ビュー・サイトを表示している場合は、次のコンポーネントを監視できます。

- 「一般」セクションの「スケジュール済リンク」は、マテリアライズド・ビュー・サイトのスケジュールされたリンク数を示します。スケジュールされたリンクを使用して、遅延トランザクションを自動的にプッシュします。7-15 ページの「スケジュールされたリンクと遅延トランザクションの概要」を参照してください。
- 「マテリアライズド・ビュー・サイト」セクションの「マテリアライズド・ビュー・グループ」は、マテリアライズド・ビュー・サイトのマテリアライズド・ビュー・グループの数を示します。マテリアライズド・ビューをグループに編成すると、管理が簡単になります。また、マテリアライズド・ビューを更新可能にするには、マテリアライズド・ビュー・グループに入れる必要があります。マテリアライズド・ビュー・グループの詳細は、7-14 ページの「レプリケーション・グループと更新可能マテリアライズド・ビューの概要」を参照してください。
- 「マテリアライズド・ビュー・サイト」セクションの「マテリアライズド・ビュー」は、マテリアライズド・ビュー・サイトのマテリアライズド・ビューの数を示します。マテリアライズド・ビューの詳細は、7-2 ページの「マテリアライズド・ビュー・レプリケーションの概要」を参照してください。
- 「マテリアライズド・ビュー・サイト」セクションの「リフレッシュ・グループ」は、マテリアライズド・ビュー・サイトのリフレッシュ・グループの数を示します。リフレッシュ・グループがリフレッシュされると、リフレッシュ・グループ内のマテリアライズド・ビューが同じ時点で一致します。7-3 ページの「リフレッシュ・グループの概要」を参照してください。

1つ以上のマテリアライズド・ビュー・サイトのマスター・サイトを表示している場合は、次のコンポーネントを監視できます。

- 「マルチマスター・レプリケーション」セクションの「**マスター・グループ**」は、マスター・サイトのマスター・グループの数を示します。マテリアライズド・ビュー・サイトのマテリアライズド・ビュー・グループは、マスター・グループに基づくことができます。マスター・グループとマテリアライズド・ビュー・グループの詳細は、7-14 ページの「**レプリケーション・グループと更新可能マテリアライズド・ビューの概要**」を参照してください。
- 「マスター・サイト」セクションの「**マテリアライズド・ビュー・ログ**」は、マスター・サイトのマテリアライズド・ビュー・ログの数を示します。マテリアライズド・ビュー・ログにより、マテリアライズド・ビューをリフレッシュできます。マテリアライズド・ビュー・ログの詳細は、7-3 ページの「**マテリアライズド・ビューのリフレッシュの概要**」を参照してください。
- 「一般」セクションの「**エラー・トランザクション**」は、マスター・サイトのエラー・トランザクションの数を示します。エラー・トランザクションは、更新可能マテリアライズド・ビューがリフレッシュされ、解消できない競合が発生した場合に発生することがあります。7-16 ページの「**競合と更新可能マテリアライズド・ビューの概要**」を参照してください。

「アドバンスド・レプリケーション:管理」ページの「概要」サブページにリストされるその他のレプリケーション・コンポーネントについては、「**ヘルプ**」をクリックしてください。

**参照:**

- 第7章「マテリアライズド・ビューを使用したデータのレプリケート」
- 「マテリアライズド・ビュー・レプリケーション環境の管理」(8-2 ページ)

## マテリアライズド・ビューに関する情報の表示

マテリアライズド・ビューには、ある単一の時点での表の完全コピーまたは部分コピーが含まれます。レプリケーション環境では、通常、マテリアライズド・ビューはベースとなっている表とは異なるデータベースにあります。

**マテリアライズド・ビューに関する情報を表示するには:**

1. Enterprise Manager で、マテリアライズド・ビュー・サイトにマテリアライズド・ビュー管理者としてログインします。マテリアライズド・ビュー管理者のデフォルトのユーザー名は mvadmin です。
2. 「データベース」ホームページに移動します。
3. 「**データ移動**」をクリックして「データ移動」サブページを開きます。
4. 「アドバンスド・レプリケーション」セクションの「**管理**」をクリックします。  
「アドバンスド・レプリケーション:管理」ページが表示され、「概要」サブページが表示されます。
5. 「マテリアライズド・ビュー・サイト」セクションの「**マテリアライズド・ビュー**」に関連付けられている数字をクリックして、「マテリアライズド・ビュー」ページを開きます。

- 検索ツールを使用して、表示するマテリアライズド・ビューをリストします。すべてのマテリアライズド・ビューをリストするには、「スキーマ」フィールドと「オブジェクト名」フィールドを空白のままにして「実行」をクリックします。

Database Instance: database > Logged in As MVIEWADMIN

**Materialized Views** Object Type: Materialized View

**Search**  
Enter a schema name and an object name to filter the data that is displayed in your results set.

Schema

Object Name

By default, the search returns all uppercase matches beginning with the string you entered. To run an exact or case-sensitive match, double quote the search string. You can use the wildcard symbol (%) in a double quoted string.

Selection Mode:

Actions:

| Select                           | Schema | Materialized View(Snapshot)     | Master Owner | Master Table | Master Link      | Type | Updatable | Can Use Log | Last Refresh               |
|----------------------------------|--------|---------------------------------|--------------|--------------|------------------|------|-----------|-------------|----------------------------|
| <input checked="" type="radio"/> | HR     | <a href="#">EMPLOYEES_MV1</a>   | HR           | EMPLOYEES    | @III.EXAMPLE.COM | FAST | YES       | YES         | May 8, 2008 2:28:31 PM PDT |
| <input type="radio"/>            | HR     | <a href="#">LOCATIONS_MV1</a>   | HR           | LOCATIONS    | @III.EXAMPLE.COM | FAST | YES       | YES         | May 8, 2008 2:28:33 PM PDT |
| <input type="radio"/>            | HR     | <a href="#">JOBS_MV1</a>        | HR           | JOBS         | @III.EXAMPLE.COM | FAST | YES       | YES         | May 8, 2008 2:28:32 PM PDT |
| <input type="radio"/>            | HR     | <a href="#">COUNTRIES_MV1</a>   | HR           | COUNTRIES    | @III.EXAMPLE.COM | FAST | YES       | YES         | May 8, 2008 2:28:20 PM PDT |
| <input type="radio"/>            | HR     | <a href="#">DEPARTMENTS_MV1</a> | HR           | DEPARTMENTS  | @III.EXAMPLE.COM | FAST | YES       | YES         | May 8, 2008 2:28:31 PM PDT |
| <input type="radio"/>            | HR     | <a href="#">REGIONS_MV1</a>     | HR           | REGIONS      | @III.EXAMPLE.COM | FAST | YES       | YES         | May 8, 2008 2:28:33 PM PDT |
| <input type="radio"/>            | HR     | <a href="#">JOB_HISTORY_MV1</a> | HR           | JOB_HISTORY  | @III.EXAMPLE.COM | FAST | YES       | YES         | May 8, 2008 2:28:32 PM PDT |

「マテリアライズド・ビュー」ページには、マスター表、マスター・サイト（「**マスター・リンク**」列内）、および最終リフレッシュ時刻を含め、リストされている各マテリアライズド・ビューに関する重要情報が表示されます。このページのすべての列の詳細は、「ヘルプ」をクリックしてください。

- マテリアライズド・ビューの詳細は、そのマテリアライズド・ビューを選択し、「表示」をクリックしてマテリアライズド・ビューを表示するページを開きます。

Database Instance: database > Materialized Views > Logged in As MVIEWADMIN

**View Materialized View: HR.JOBS\_MV1** Actions:

**General**

Name **JOBS\_MV1**  
 Schema **HR**  
 Tablespace **SYSTEM**  
 Last Analyze Date **5/9/08 6:35:11 AM PDT**  
 Refresh State **UNDEFINED**  
 Compile State **VALID**  
 Updatable **true**  
 Built From Table **false**  
 Number of Indexes **1**

Materialized View Query

```
SELECT "JOBS"."JOB_ID" "JOB_ID","JOBS"."JOB_TITLE"
"JOB_TITLE","JOBS"."MIN_SALARY" "MIN_SALARY","JOBS"."MAX_SALARY"
"MAX_SALARY" FROM "HR"."JOBS"@III.EXAMPLE.COM "JOBS"
```

**Refresh**

Refresh Type **Fast**  
 Refresh Method **Primary Key**  
 Refresh Interval **On Demand**  
 Master Rollback Segment  
 Local Rollback Segment **<Default>**  
 Last Refresh Date **5/8/08 2:28:32 PM PDT**  
 Refresh Group

「マテリアライズド・ビューの表示:」ページには、マテリアライズド・ビューに関する詳細情報が表示されます。情報は、「一般」、「リフレッシュ」、「記憶域」、「索引記憶域」、「オプション」および「マスター情報」の各セクションに編成されています。このページのすべてのセクションの詳細は、「ヘルプ」をクリックしてください。

---

**注意：** マテリアライズド・ビューに関する情報を、ALL MVIEWS データ・ディクショナリ・ビューに問い合わせることもできます。

---

**参照：**

- 「マテリアライズド・ビュー・レプリケーションの概要」 (7-2 ページ)
- 「マテリアライズド・ビュー・レプリケーション環境の管理」 (8-2 ページ)

## 現在リフレッシュされているマテリアライズド・ビューの判断

マテリアライズド・ビューをリフレッシュすると、マテリアライズド・ビューのデータがそのマスター表のデータと同期します。V\$MVREFRESH 動的パフォーマンス・ビューを問い合わせ、どのマテリアライズド・ビューが現在リフレッシュされているかを判断できます。

マテリアライズド・ビューを含むデータベースを停止する前に、いずれかのマテリアライズド・ビューが現在リフレッシュされているかどうかを確認するのが最適です。いずれかのマテリアライズド・ビューがリフレッシュされている場合は、すべてのリフレッシュが完了するまで待つから、データベースを停止します。

### マテリアライズド・ビュー・サイトでどのマテリアライズド・ビューが現在リフレッシュされているかを判断するには：

1. SQL\*Plus を開き、マテリアライズド・ビュー管理者としてマテリアライズド・ビュー・サイトに接続します。

SQL\*Plus の起動の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。

2. 次の問合せを実行します。

```
COLUMN CURRMVOWNER HEADING 'Owner' FORMAT A15
COLUMN CURRMVNAME HEADING 'Materialized|View' FORMAT A25
```

```
SELECT CURRMVOWNER, CURRMVNAME FROM V$MVREFRESH;
```

出力は、次のようになります。

| Owner | Materialized<br>View |
|-------|----------------------|
| HR    | COUNTRIES_MV         |
| HR    | EMPLOYEES_MV         |

マテリアライズド・ビューの遅延トランザクションがマスターにプッシュされている場合、V\$MVREFRESH 動的パフォーマンス・ビューに、更新可能なマテリアライズド・ビューに関する情報は含まれません。

**参照：**

- 「マテリアライズド・ビューのリフレッシュの概要」 (7-3 ページ)
- 「マテリアライズド・ビューのリフレッシュ」 (8-2 ページ)

## マテリアライズド・ビュー・グループに関する情報の表示

マテリアライズド・ビュー・グループを使用すると、論理的に関連するマテリアライズド・ビューを簡単に管理できます。たとえば、スキーマ内のすべてのマテリアライズド・ビューまたはアプリケーションによって使用されているすべてのマテリアライズド・ビューをマテリアライズド・ビュー・グループに追加できます。また、マテリアライズド・ビューを更新可能にするには、それらがマテリアライズド・ビュー・グループに属している必要があります。

### マテリアライズド・ビュー・サイトのマテリアライズド・ビュー・グループに関する情報を表示するには：

1. Enterprise Manager で、マテリアライズド・ビュー・サイトにマテリアライズド・ビュー管理者としてログインします。マテリアライズド・ビュー管理者のデフォルトのユーザー名は mvadmin です。
2. 「データベース」 ホームページに移動します。
3. 「**データ移動**」 をクリックして「データ移動」 サブページを開きます。
4. 「アドバンスド・レプリケーション」 セクションの「**管理**」 をクリックします。  
「アドバンスド・レプリケーション：管理」 ページが表示され、「概要」 サブページが表示されます。
5. 「マテリアライズド・ビュー・サイト」 セクションで、「**マテリアライズド・ビュー・グループ**」 に関連付けられている数字をクリックします。  
「マテリアライズド・ビュー・グループ」 ページが表示され、マテリアライズド・ビュー・サイトのマテリアライズド・ビュー・グループの名前がリストされます。
6. マテリアライズド・ビュー・グループに関する詳細情報を表示するには、マテリアライズド・ビュー・グループの名前をクリックして「マテリアライズド・ビュー・グループの表示」 ページを開きます。

Database Instance: database > Advanced Replication: Administration > Advanced Replication: Materialized View Groups > Logged in As MVIEWADMIN

**View Materialized View Group: HR\_REPG** Edit

**General**

Link To Master **III1.EXAMPLE.COM**

Propagation Mode **Asynchronous**

Description

**Objects**

| Name            | Schema | Type     | Status | Generation Status |
|-----------------|--------|----------|--------|-------------------|
| COUNTRIES_MV1   | HR     | SNAPSHOT | VALID  |                   |
| DEPARTMENTS_MV1 | HR     | SNAPSHOT | VALID  |                   |
| EMPLOYEES_MV1   | HR     | SNAPSHOT | VALID  |                   |
| REGIONS_MV1     | HR     | SNAPSHOT | VALID  |                   |
| JOB_HISTORY_MV1 | HR     | SNAPSHOT | VALID  |                   |
| LOCATIONS_MV1   | HR     | SNAPSHOT | VALID  |                   |
| JOBS_MV1        | HR     | SNAPSHOT | VALID  |                   |

「マテリアライズド・ビュー・グループの表示」 ページには、次の情報が含まれます。

- 「**マスターへのリンク**」 フィールドには、データベース・リンクの名前。マスター・サイトには、マテリアライズド・ビュー・グループ内のマテリアライズド・ビューのマスター表が含まれます。
- 伝播により伝播スケジュールに応じたメッセージが送信されるため、「**伝播モード**」 フィールドには「**非同期**」 と表示されます。
- 「**説明**」 フィールドには、グループに説明がある場合のマテリアライズド・ビュー・グループの説明。
- 「**オブジェクト**」 セクションには、マテリアライズド・ビュー・グループ内のデータベース・オブジェクトのリストと各データベース・オブジェクトに関する情報

このページの詳細を表示するには、「**ヘルプ**」 をクリックします。

**注意：** マテリアライズド・ビュー・グループに関する情報を、次のデータ・ディクショナリ・ビューに問い合わせることもできます。

- ALL\_REPGROUP
- ALL\_REPSITES

**参照：**

- 「レプリケーション・グループと更新可能マテリアライズド・ビューの概要」 (7-14 ページ)
- 「マスター・サイト、マスター表およびマテリアライズド・ビュー・サイトの概要」 (7-3 ページ)
- 「マテリアライズド・ビューを使用した読取り / 書込みデータのレプリケート」 (7-12 ページ)

## 更新可能マテリアライズド・ビューの遅延トランザクションに関する情報の表示

マテリアライズド・ビュー・サイトでは、遅延トランザクションには更新可能マテリアライズド・ビューに対する変更が含まれます。遅延トランザクションは、マテリアライズド・ビュー・サイトに格納されるため、マスター・サイトに送信でき、マスター表に適用できます。マテリアライズド・ビューの遅延トランザクションは、マテリアライズド・ビューがリフレッシュされると必ずプッシュされます。スケジュールされたリンクは、遅延トランザクションがリフレッシュ・プロセスに依存せずにマスター・サイトにいつプッシュされるかを決定します。

**更新可能マテリアライズド・ビューの遅延トランザクションに関する情報を表示するには：**

1. Enterprise Manager で、マテリアライズド・ビュー・サイトにマテリアライズド・ビュー管理者としてログインします。マテリアライズド・ビュー管理者のデフォルトのユーザー名は mvadmin です。
2. 「データベース」 ホームページに移動します。
3. 「データ移動」 をクリックして「データ移動」 サブページを開きます。
4. 「アドバンスド・レプリケーション」 セクションの「管理」 をクリックします。  
「アドバンスド・レプリケーション：管理」 ページが表示され、「概要」 サブページが表示されます。
5. 「一般」 セクションの「遅延トランザクション」 に関連付けられている数字をクリックして、「管理：遅延トランザクション」 ページを開きます。

| Database Instance: database > Advanced Replication: Administration > Logged in As MVIEWADMIN |                           |                 |                            |                 |
|----------------------------------------------------------------------------------------------|---------------------------|-----------------|----------------------------|-----------------|
| Administration: Deferred Transactions                                                        |                           |                 |                            |                 |
| Select                                                                                       | Transactions ID           | Destination     | Start Time                 | Number Of Calls |
|                                                                                              | <a href="#">16.27.619</a> | III.EXAMPLE.COM | 2008-05-09 08:11:44.451526 | 19              |
|                                                                                              | <a href="#">16.12.619</a> | III.EXAMPLE.COM | 2008-05-09 08:10:54.666266 | 1               |



「管理: 遅延トランザクション」ページには、トランザクション ID と、各遅延トランザクションに関する次の情報が表示されます。

- 「宛先」 フィールドは、遅延トランザクションがプッシュされるマスター・サイトを示します。
- 「開始時間」 フィールドは、マテリアライズド・ビュー・サイトの遅延トランザクション・キューに遅延トランザクションがいつ入れられたかを示します。
- 「コール数」 フィールドは、通常はトランザクションで変更された個々の行数を示します。

このページの詳細を表示するには、「ヘルプ」をクリックします。

6. 「トランザクション ID」列でトランザクション ID をクリックして、「遅延トランザクション」ページを開き、トランザクションに関する詳細情報を表示します。

Database Instance: database > Advanced Replication: Administration > Administration: Deferred Transactions > Logged in As MVIEWADMIN  
Deferred Transaction: 16.27.619

Source **II1.EXAMPLE.COM**  
Start Time **2008-05-09 08:11:44.451526**

**Remote Procedure Calls**

| Select Table                          | Schema | Operation  |
|---------------------------------------|--------|------------|
| <input checked="" type="radio"/> JOBS | HR     | REP_UPDATE |
| <input type="radio"/> JOBS            | HR     | REP_UPDATE |
| <input type="radio"/> JOBS            | HR     | REP_UPDATE |

「遅延トランザクション」ページには、マテリアライズド・ビューと、そのマテリアライズド・ビューを変更した操作のタイプが表示されます。たとえば、REP\_UPDATE は更新操作を表します。

7. 遅延トランザクション内の特定の変更に関する詳細情報を表示するには、変更を選択し、「詳細」をクリックします。

Database Instance: database > Advanced Replication: Administration > Administration: Deferred Transactions > Deferred Transaction: 5.4.978 > Logged in As MVADMIN  
Table: ORDER\_ITEMS

You can view the columns in the table and the datatype for each column. The original value and/or new values may be displayed for each column depending on the operation. INSERT operation will display the new value, DELETE operation will display the original value and UPDATE operation will display both the original and new values.

Operation **REP\_UPDATE**

| Column       | Data Type | Original Value | New Value |
|--------------|-----------|----------------|-----------|
| LINE_ITEM_ID | NUMBER    | 1              |           |
| ORDER_ID     | NUMBER    | 2361           |           |
| PRODUCT_ID   | NUMBER    | 2289           |           |
| QUANTITY     | NUMBER    | 180            | 210       |
| UNIT_PRICE   | NUMBER    | 46             |           |

このページには、マテリアライズド・ビュー内の各列が、元の値と、該当する場合には新しい値とともに表示されます。列に新しい値がある場合は、変更された列値を元の値から新しい値に変更します。

**参照：**

- 「スケジュールされたリンクと遅延トランザクションの概要」 (7-15 ページ)
- 「レプリケーション・グループと更新可能マテリアライズド・ビューの概要」 (7-14 ページ)
- 「マテリアライズド・ビューを使用した読取り / 書込みデータのレプリケート」 (7-12 ページ)
- 遅延トランザクションに関連するデータ・ディクショナリ・ビューの詳細は、『Oracle Database アドバンスド・レプリケーション・マネージメント API リファレンス』を参照してください。

**リフレッシュ・グループに関する情報の表示**

リフレッシュ・グループをリフレッシュすると、リフレッシュ・グループ内のすべてのマテリアライズド・ビューが、ある一致した時点でリフレッシュされます。

**マテリアライズド・ビュー・サイトのリフレッシュ・グループに関する情報を表示するには：**

1. Enterprise Manager で、マテリアライズド・ビュー・サイトにマテリアライズド・ビュー管理者としてログインします。マテリアライズド・ビュー管理者のデフォルトのユーザー名は mvadmin です。
2. 「データベース」 ホームページに移動します。
3. 「データ移動」 をクリックして「データ移動」 サブページを開きます。
4. 「アドバンスド・レプリケーション」 セクションの「管理」 をクリックします。  
「アドバンスド・レプリケーション：管理」 ページが表示され、「概要」 サブページが表示されます。
5. 「マテリアライズド・ビュー・サイト」 セクションで、「リフレッシュ・グループ」 に関連付けられている数字をクリックします。  
「リフレッシュ・グループ」 ページが表示されます。
6. 検索ツールを使用して、表示するリフレッシュ・グループをリストします。すべてのリフレッシュ・グループをリストするには、「スキーマ」 フィールドと「オブジェクト名」 フィールドを必ず空白のままにして「実行」 をクリックします。

Database Instance: database > Logged in As MVADMIN

**Refresh Groups** Object Type Refresh Group

**Search**  
Select an object type and optionally enter a schema name and an object name to filter the data that is displayed in your results set.

Schema

Object Name

By default, the search returns all uppercase matches beginning with the string you entered. To run an exact or case-sensitive match, double quote the search string. You can use the wildcard symbol (%) in a double quoted string.

Selection Mode

Actions

| Select                              | Schema  | Name | Broken | Next Date                  | Interval                 |
|-------------------------------------|---------|------|--------|----------------------------|--------------------------|
| <input checked="" type="checkbox"/> | MVADMIN | MVR  | NO     | Oct 4, 2006 1:45:49 PM PDT | /*24:hr*/sysdate + 24/24 |

「グループのリフレッシュ」ページには、マテリアライズド・ビュー・サイトのリフレッシュ・ビュー・グループの名前をリストします。これには、各リフレッシュ・グループの次の情報が含まれます。

- 「**中断**」列は、リフレッシュ・ジョブが正しく動作しているかどうかを示します。「**いいえ**」は、ジョブが正しく動作していることを意味します。「**はい**」は、ジョブが中断していることを意味します。リフレッシュ・グループのリフレッシュ試行が 16 回失敗すると、リフレッシュ・ジョブは中断しているとみなされます。リフレッシュ・ジョブが中断している場合は、8-22 ページの「マテリアライズド・ビュー・リフレッシュの問題の修正」を参照してください。
  - 「**次の日付**」列は、リフレッシュ・グループが次回自動的にリフレッシュされる日時を示します。
  - 「**間隔**」列は、リフレッシュ・グループの自動リフレッシュ間隔を示します。
7. リフレッシュ・グループの詳細は、そのリフレッシュ・グループを選択し、「**表示**」をクリックして「リフレッシュ・グループの表示」ページを開きます。

Database Instance: database > Refresh Groups > Logged in As MWADMIN

View Refresh Group: MVADMIN.MVR

Actions:

---

**General**

Name **MVR**  
 Schema **MVADMIN**  
 Status **Normal**  
 Job ID **62**  
 Rollback Segment

---

**Refresh**

Next Date **10/4/06 1:45:49 PM PDT**  
 Interval **/\*24:hr\*/sysdate + 24/24**

---

**Materialized Views**

| Schema | Name            | Status | Next Date              | Job ID |
|--------|-----------------|--------|------------------------|--------|
| HR     | DEPARTMENTS_MVR | Normal | 10/4/06 1:45:49 PM PDT | 62     |
| HR     | EMPLOYEES_MVR   | Normal | 10/4/06 1:45:49 PM PDT | 62     |

Actions:

「リフレッシュ・グループの表示」ページには、リフレッシュ・グループに関する詳細情報が表示されます。情報は、「一般」、「リフレッシュ」および「マテリアライズド・ビュー」の各セクションに編成されています。このページのすべてのセクションの詳細は、「**ヘルプ**」をクリックしてください。

---

**注意：** リフレッシュ・グループに関する情報を、ALL\_REFRESH データ・ディクショナリ・ビューに問い合わせることもできます。

---

**参照：**

- 「マテリアライズド・ビューのリフレッシュの概要」 (7-3 ページ)
- 「リフレッシュ・グループの概要」 (7-3 ページ)
- 「リフレッシュ・グループの構成」 (7-23 ページ)
- 「リフレッシュ・グループのリフレッシュ」 (8-3 ページ)

## マスター・サイトのマテリアライズド・ビュー・ログの表示

マテリアライズド・ビュー・ログは、マテリアライズド・ビューの高速リフレッシュが可能になるようにマスター表に対する変更を追跡します。

### マスター・サイトのマテリアライズド・ビュー・ログに関する情報を表示するには：

- Enterprise Manager では、レプリケーション管理者または SYSTEM などの管理ユーザーとしてマスター・サイトにログインします。デフォルトでは、レプリケーション管理者のユーザー名は repadmin です。
- 「データ移動」をクリックして「データ移動」サブページを開きます。
- 「アドバンスド・レプリケーション」セクションの「管理」をクリックします。  
「アドバンスド・レプリケーション：管理」ページが表示され、「概要」サブページが表示されます。
- 「マスター・サイト」セクションの「マテリアライズド・ビュー・ログ」に関連付けられている数字をクリックします。  
「マテリアライズド・ビュー・ログ」ページが表示されます。
- 検索ツールを使用して、表示するマテリアライズド・ビューをリストします。  
すべてのマテリアライズド・ビュー・ログをリストするには、「スキーマ」フィールドと「オブジェクト名」フィールドを空白にしたまま「実行」をクリックします。  
「マテリアライズド・ビュー・ログ」ページには、リストされている各マテリアライズド・ビュー・ログのスキーマ、基礎となるログ表およびマスター表がリストされます。
- マテリアライズド・ビュー・ログの詳細は、そのマテリアライズド・ビュー・ログを選択し、「表示」をクリックして「マテリアライズド・ビュー・ログの表示」ページを開きます。

Database Instance: database > Materialized View Logs > Logged in As MVADMIN  
View Materialized View Log: OE.MLOG\$\_ORDER\_ITEMS

Actions:

**General**

Log Table **MLOG\$\_ORDER\_ITEMS**  
Schema **OE**  
Master Table **ORDER\_ITEMS**  
Tablespace **SYSTEM**  
Refresh Types **Primary Key**  
Filter Columns

**Options**

Parallel Degree **Disabled**  
Cache **false**

**Storage**

**Tablespace**

|                    |                   |
|--------------------|-------------------|
| Name               | <b>SYSTEM</b>     |
| Extent Management  | <b>Dictionary</b> |
| Segment Management | <b>Manual</b>     |
| Allocation Type    | <b>USER</b>       |
| Logging            | <b>Yes</b>        |

**Extents**

|                    |             |
|--------------------|-------------|
| Initial Size       | <b>16KB</b> |
| Next Size          | <b>16KB</b> |
| Increment Size (%) | <b>50</b>   |
| Minimum Number     | <b>1</b>    |
| Maximum Number     | <b>505</b>  |

**Space Usage**

|                         |           |
|-------------------------|-----------|
| Free Space (PCTFREE)(%) | <b>60</b> |
| Used Space (PCTUSED)(%) | <b>30</b> |

**Number of Transactions**

|         |            |
|---------|------------|
| Initial | <b>1</b>   |
| Maximum | <b>255</b> |

**Free Lists**

|                  |          |
|------------------|----------|
| Free Lists       | <b>1</b> |
| Free List Groups | <b>1</b> |

**Buffer Pool**

|             |                |
|-------------|----------------|
| Buffer Pool | <b>DEFAULT</b> |
|-------------|----------------|

Actions:

「マテリアライズド・ビュー・ログの表示」ページには、マテリアライズド・ビュー・ログに関する詳細情報が表示されます。情報は、「一般」、「オプション」および「記憶域」セクションに編成されています。このページのセクションの詳細は、「ヘルプ」をクリックしてください。

---

**注意：** マテリアライズド・ビュー・ログに関する情報を、ALL\_MVIEW\_LOGS データ・ディクショナリ・ビューに問い合わせることができます。

---

**参照：**

- 「マテリアライズド・ビューのリフレッシュの概要」 (7-3 ページ)
- 「マスター・サイトでのマテリアライズド・ビュー・ログの構成」 (7-6 ページ)
- 「マテリアライズド・ビュー・ログが大きくなりすぎるのを防止」 (8-23 ページ)
- 「チュートリアル: マスター・サイトでのマテリアライズド・ビュー・サポートのクリーン・アップ」 (8-8 ページ)

## マスター・サイトのマテリアライズド・ビューの表示

マテリアライズド・ビュー・サイトでマテリアライズド・ビューを構成すると、マテリアライズド・ビューがマスター・サイトに登録されます。ALL\_REGISTERED\_MVIEWS データ・ディクショナリ・ビューを問い合わせ、マスター・サイトに登録されている各マテリアライズド・ビューに関する次の情報を表示できます。

- マテリアライズド・ビューの所有者
- マテリアライズド・ビューの名前
- マテリアライズド・ビューを含むマテリアライズド・ビュー・サイト
- マテリアライズド・ビューが更新可能か読取り専用か

**この情報を表示するには：**

1. SQL\*Plus を開き、レプリケーション管理者や SYSTEM などの管理ユーザーとしてマスター・サイトに接続します。デフォルトでは、レプリケーション管理者のユーザー名は repadmin です。

SQL\*Plus の起動の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。

2. 次の問合せを実行します。

```
COLUMN OWNER HEADING 'Owner' FORMAT A20
COLUMN NAME HEADING 'Name' FORMAT A20
COLUMN MVIEW_SITE HEADING 'Materialized|View Site' FORMAT A15
COLUMN UPDATABLE HEADING 'Updatable?' FORMAT A10
```

```
SELECT OWNER, NAME, MVIEW_SITE, UPDATABLE
FROM ALL_REGISTERED_MVIEWS;
```

出力は、次のようになります。

| Owner | Name           | Materialize     |            |
|-------|----------------|-----------------|------------|
|       |                | View Site       | Updatable? |
| HR    | COUNTRIES_MV   | II2.EXAMPLE.COM | YES        |
| HR    | DEPARTMENTS_MV | II2.EXAMPLE.COM | YES        |
| HR    | EMPLOYEES_MV   | II2.EXAMPLE.COM | YES        |
| HR    | JOBS_MV        | II2.EXAMPLE.COM | YES        |
| HR    | JOB_HISTORY_MV | II2.EXAMPLE.COM | YES        |
| HR    | LOCATIONS_MV   | II2.EXAMPLE.COM | YES        |
| HR    | REGIONS_MV     | II2.EXAMPLE.COM | YES        |

**参照：**

- 「マスター・サイト、マスター表およびマテリアライズド・ビュー・サイトの概要」 (7-3 ページ)
- 「レプリケーション・グループと更新可能マテリアライズド・ビューの概要」 (7-14 ページ)

## マテリアライズド・ビュー・レプリケーション環境のトラブルシューティング

この項では、マテリアライズド・ビュー・レプリケーション環境で最も一般的な問題について説明します。これらの問題の解決方法も説明します。

次の各項で、マテリアライズド・ビュー・レプリケーション環境のトラブルシューティングを説明します。

- マテリアライズド・ビュー・リフレッシュの問題の修正
- マテリアライズド・ビュー・ログが大きくなりすぎるのを防止

**参照：**

- 『Oracle Database アドバンスド・レプリケーション』
- 第7章「マテリアライズド・ビューを使用したデータのレプリケート」
- 「マテリアライズド・ビュー・レプリケーション環境の管理」 (8-2 ページ)
- 「マテリアライズド・ビュー・レプリケーション環境の監視」 (8-10 ページ)

## マテリアライズド・ビュー・リフレッシュの問題の修正

マテリアライズド・ビュー・サイトで1つ以上のマテリアライズド・ビューをリフレッシュできない場合の一般的な原因と解決方法を次に示します。

- ネットワークの問題により、1つ以上のマテリアライズド・ビューをリフレッシュできません。複数のリフレッシュ・グループと複数のマテリアライズド・ビューがマテリアライズド・ビュー・サイトにあり、いずれもリフレッシュできない場合は、ネットワークの問題が最も可能性の高い原因です。この場合は、ネットワークの問題を解決し、関連するマテリアライズド・ビューをリフレッシュします。データベース間のネットワーク接続の構成の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。
- マテリアライズド・ビューのマスター・サイトが停止しています。マスター・サイトが開いていない場合は、マテリアライズド・ビューをリフレッシュできません。複数のマスター・サイトを使用するマテリアライズド・ビューのあるマテリアライズド・ビュー・サイトがあり、特定のマスター・サイトのマテリアライズド・ビューのみリフレッシュできない場合は、マスター・サイトの問題が最も可能性の高い原因です。この場合は、マスター・サイトでデータベースをチェックします。停止している場合は起動し、関連するマテリアライズド・ビューをリフレッシュします。データベースの開始の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。
- データベース・リンクが中断しています。マテリアライズド・ビュー・グループ内のすべてのマテリアライズド・ビューをリフレッシュできない場合は、マテリアライズド・ビューのリフレッシュに使用されているデータベース・リンクが中断している可能性があります。データベース・リンクが中断している場合は、データベース・リンクを削除してから再び作成することにより、問題を解決できます。

**中断しているデータベース・リンクを再作成するには：**

1. Enterprise Manager に SYSTEM などの管理ユーザーとしてログインします。
2. 「データベース」 ホームページに移動します。
3. 「スキーマ」 をクリックして 「スキーマ」 サブページを開きます。
4. 「データベース・オブジェクト」 セクションの 「データベース・リンク」 をクリックします。  
「データベース・リンク」 ページが表示されます。
5. 検索ツールを使用して、マテリアライズド・ビューで使用されるデータベース・リンクをリストします。  
データベース・リンク名は、マテリアライズド・ビューのマスター・サイトのグローバル名と同じである必要があります。
6. データベース・リンクを選択します。
7. 「表示」 をクリックします。
8. データベース・リンクを表示するページで 「テスト」 をクリックして、データベース・リンクがアクティブかどうかを確認します。アクティブな場合、マテリアライズド・ビューのリフレッシュの問題は、別の問題が原因です。データベース・リンクがアクティブでない場合は、手順 9～13 に従って削除し、再作成します。
9. 「OK」 をクリックして 「データベース・リンク」 ページに戻ります。
10. データベース・リンクを選択します。
11. 「削除」 をクリックします。
12. 確認ページで 「はい」 をクリックしてデータベース・リンクを削除します。
13. 2-8 ページの 「チュートリアル: データベース・リンクの作成」 の手順に従って、このデータベース・リンクを再作成します。

**参照：**

- 「マテリアライズド・ビューのリフレッシュの概要」 (7-3 ページ)
- 「マテリアライズド・ビューのリフレッシュ」 (8-2 ページ)
- 『Oracle Database アドバンスド・レプリケーション』

## マテリアライズド・ビュー・ログが大きくなりすぎるのを防止

マテリアライズド・ビュー・ログは、マスター・サイトのマスター表に対する変更を追跡します。マテリアライズド・ビューに高速リフレッシュが実行される場合、マテリアライズド・ビュー・ログ内の変更はマテリアライズド・ビューに適用されます。マテリアライズド・ビュー・ログを使用するすべてのマテリアライズド・ビューに変更が適用されると、変更がマテリアライズド・ビュー・ログから削除されます。

マテリアライズド・ビュー・ログは、マテリアライズド・ビュー・ログを使用する 1 つ以上のマテリアライズド・ビューが比較的長期間リフレッシュされない場合に非常に大きくなる場合があります。この状況により、他のマテリアライズド・ビューのリフレッシュ・パフォーマンスが低下することがあります。次に、一般的な原因と解決方法を示します。

- ネットワークの問題により、マテリアライズド・ビュー・ログを使用する 1 つ以上のマテリアライズド・ビューをリフレッシュできません。この場合は、ネットワークの問題を解決し、関連するマテリアライズド・ビューをリフレッシュします。データベース間のネットワーク接続の構成の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。

- マテリアライズド・ビュー・サイトがネットワークに接続されていないときに、マテリアライズド・ビュー・サイトでマテリアライズド・ビューが削除されました。マテリアライズド・ビューが削除され、ネットワーク接続が使用可能な場合は、マテリアライズド・ビューに関する情報がマスター・サイトから自動的に削除されます。一方、ネットワーク接続が使用可能でない場合は、マテリアライズド・ビューに関する情報がマスター・サイトに残り、手動でクリーン・アップする必要があります。8-8 ページの「[チュートリアル: マスター・サイトでのマテリアライズド・ビュー・サポートのクリーン・アップ](#)」を参照してください。
- マテリアライズド・ビューが十分な頻度でリフレッシュされていません。マテリアライズド・ビューは、手動でリフレッシュするか、リフレッシュ・グループ内にある場合は自動的にリフレッシュできます。

マテリアライズド・ビューを手動でリフレッシュする場合に、マテリアライズド・ビュー・ログが非常に大きくなる場合は、より頻繁にリフレッシュする必要があります。8-4 ページの「[マテリアライズド・ビューのリフレッシュ](#)」を参照してください。

マテリアライズド・ビューが、十分な頻度でリフレッシュされていないリフレッシュ・グループにある場合は、リフレッシュ・グループのリフレッシュ間隔を短くできます。

**リフレッシュ・グループのリフレッシュ間隔を変更するには:**

1. Enterprise Manager で、マテリアライズド・ビュー・サイトにマテリアライズド・ビュー管理者としてログインします。マテリアライズド・ビュー管理者のデフォルトのユーザー名は mvadmin です。
2. 「データベース」 ホームページに移動します。
3. 「**データ移動**」 をクリックして「データ移動」 サブページを開きます。
4. 「アドバンスド・レプリケーション」 セクションの「**管理**」 をクリックします。  
「アドバンスド・レプリケーション: 管理」 ページが表示され、「概要」 サブページが表示されます。
5. 「マテリアライズド・ビュー・サイト」 セクションで、「**リフレッシュ・グループ**」に関連付けられている数字をクリックします。
6. 「グループのリフレッシュ」 ページで、検索ツールを使用して、変更するリフレッシュ・グループをリストします。
7. リフレッシュ・グループを選択します。



## 8. 「編集」をクリックします。

「リフレッシュ・グループの編集」ページが表示され、「一般」サブページが表示されます。

The screenshot shows the 'Edit Refresh Group' page for 'MVADMIN.MVR'. The page is titled 'Database Instance: database > Refresh Groups > Edit Refresh Group: MVADMIN.MVR' and is logged in as 'MVADMIN'. The 'General' tab is selected, showing the following details:

- Name: MVR
- Schema: MVADMIN
- Status: Normal
- Job ID: 62

The 'Refresh' section contains the following options:

- \*Next Date: 9/29/06 1:45:41 PM PDT (Change...)
- \*Interval: /\*24:hr\*/sysdate + 24/24 (Change...)
- Push changes from materialized views to master before refresh
- Continue to refresh despite errors
- Delete group when last member deleted (Refresh Now)

The 'Rollback Segments' section contains the following options:

- Use default rollback segment
- Segment: SYSTEM

At the bottom of the page, there are two sets of action buttons: 'Create Like', 'Go', 'Show SQL', 'Revert', and 'Apply'.

## 9. 「リフレッシュ」セクションで、「間隔」フィールドの「変更」をクリックします。

## 10. 「毎」フィールドとリストを使用して間隔を短くします。

## 11. 「OK」をクリックします。

## 12. 「リフレッシュ・グループの編集」ページで、「適用」をクリックして変更を保存します。

---

**注意：** DBMS\_REFRESH.CHANGE プロシージャを使用して、リフレッシュ・グループのリフレッシュ間隔を変更することもできます。

---

**参照：**

- 「マテリアライズド・ビューのリフレッシュの概要」 (7-3 ページ)
- 「リフレッシュ・グループの概要」 (7-3 ページ)
- 「マスター・サイト、マスター表およびマテリアライズド・ビュー・サイトの概要」 (7-3 ページ)
- 「マスター・サイトでのマテリアライズド・ビュー・ログの構成」 (7-6 ページ)
- 『Oracle Database アドバンスド・レプリケーション』



---

## Oracle Streams Advanced Queuing を 使用したメッセージ送信

この章では、データベースおよびアプリケーション間でメッセージを送信する Oracle Streams Advanced Queuing (AQ) 環境の構成方法を説明します。また、構成後のメッセージング環境の管理、監視およびトラブルシューティングについても説明します。

この章は次の項で構成されています。

- [メッセージ機能の概要](#)
- [メッセージ機能の準備](#)
- [チュートリアル: Oracle Database 間でのメッセージの送信](#)
- [チュートリアル: メッセージ通知の構成](#)
- [キューの変更](#)
- [キュー表の変更](#)
- [伝播の変更](#)
- [メッセージング環境の監視](#)
- [メッセージング環境のトラブルシューティング](#)

## メッセージ機能の概要

メッセージング環境は、情報をキューに格納します。メッセージをキューに置くプロセスが**エンキュー**、メッセージをキューから取得するプロセスが**デキュー**です。

キュー内の情報は、タスクを完了するために使用されることも、アプリケーションによって処理されることもあります。メッセージング環境では、アプリケーションの非同期な相互通信が可能で、つまり、アプリケーションは他のアプリケーションが特定のタスクを完了するまで待機する必要がありません。非同期通信の重要性は、メッセージング・システムがそれを利用するアプリケーションの機能に最小の影響しか与えないという点にあります。

たとえば、他のアプリケーションとの通信が必要な場合、アプリケーションはメッセージをキューに置くことができます。メッセージは、他方のアプリケーションによって取得されるまでキュー内に格納されます。実際、メッセージのエンキュー時に他方のアプリケーションが実行されていなくても、メッセージは後で処理されます。メッセージは、取得側アプリケーションにアクションの実行を指示する場合も、あるいは取得側アプリケーションによる処理が必要な情報を含んでいる場合もあります。

組織に、相互通信が必要な複数の異なるシステムがある場合は、メッセージング環境が適切なソリューションになります。様々なシステムが異なる場所にある場合があり、一部は他のシステムより古く、一部は異なるプラットフォームで稼働していることがあります。メッセージングは、これらのシステム間で重要な情報を転送する標準的な信頼性のある方法を提供します。

Oracle Database のメッセージング機能は、Oracle Streams Advanced Queuing (AQ) と呼ばれます。Oracle Streams AQ はメッセージ機能の利点を提供するとともに、メッセージング・システムと Oracle Database を統合します。したがって、Oracle Streams AQ は Oracle Database の信頼性、スケーラビリティ、セキュリティ、管理性をもたらします。

Oracle Streams AQ を使用して、単一データベース内のキュー間または異なるデータベースのキュー間でメッセージを送信するメッセージング環境を構成できます。キュー間でメッセージを送信するメッセージング環境には、次のコンポーネントが含まれます。

- **キュー**: メッセージをメッセージング環境に格納する抽象ストレージ・ユニット
- **プロデューサ**: メッセージをエンキューするユーザーまたはアプリケーション
- **伝播**: 設定されたスケジュールに従って、あるキューから別のキューにメッセージをコピーする Oracle Scheduler ジョブ
- **コンシューマ**: メッセージをデキューするユーザーまたはアプリケーション

シングル・ユーザーまたはアプリケーションは、プロデューサおよびコンシューマとして動作できます。メッセージング環境では、**サブスクライバ**および**メッセージ・クライアント**は、コンシューマです。サブスクライバおよびメッセージ・クライアントは、キューからメッセージをデキューすることを許可されたメカニズムで、ルールを使用してどのメッセージをデキューするかを判断できます。

Oracle Streams AQ は多くの機能を提供し、その一部はこのマニュアルの対象外です。次の各項で、Oracle Streams AQ のいくつかの機能の概要を示します。

- [メッセージの順序付けの概要](#)
- [メッセージ・モードの概要](#)
- [メッセージ通知の概要](#)
- [伝播の概要](#)
- [Oracle Messaging Gateway の概要](#)

**参照:**

- 「データベース間のメッセージの送信時期」(1-6 ページ)
- メッセージ機能クライアントの詳細は、『Oracle Streams 概要および管理』を参照してください。
- サブスクライバおよび Oracle Streams AQ のすべての他の機能の詳細は、『Oracle Streams アドバンスド・キューイング・ユーザーズ・ガイド』を参照してください。

## メッセージの順序付けの概要

メッセージの順序付けでは、メッセージがデキューされる順序を決定します。Oracle Streams Advanced Queuing (AQ) には、メッセージの順序付けに次のオプションが用意されています。

- **エンキュー時間:** メッセージは、エンキューされたのと同じ順序でデキューされます。このオプションは、先入れ先出し (FIFO) の順序付けと呼ばれることがあります。たとえば、銀行アプリケーションは、多くの場合にこの順序付けを使用して、財務トランザクションが常に発生した時刻に従って順序付けされるようにします。
- **優先度:** 優先度は、エンキュー中にメッセージに対して指定され、メッセージは優先度に基づいてデキューされます。たとえば、輸送会社では、一部の荷物の優先度が他よりも高いため、この順序付けを使用することがあります。
- **コミット時間:** メッセージは、メッセージをエンキューしたトランザクションがコミットされた時刻に基づいてデキューされます。通常、コミット時間順序付けは、メッセージによってトランザクション依存関係のあるデータベース変更が行われる場合に使用されます。たとえば、会社の売上げを処理するアプリケーションは、メッセージによって依存性を持つデータベース表の数を変更される場合にコミット時間順序付けを使用できます。

## メッセージ・モードの概要

Oracle Streams Advanced Queuing (AQ) では、次の**メッセージ・モード**がサポートされています。

- **永続メッセージ機能:** メッセージは、ディスク上の**キュー表**と呼ばれるデータベース表に常に格納されます。この記憶域のタイプは、**永続キュー**記憶域と呼ばれることがあります。
- **バッファ・メッセージ機能:** メッセージは、メモリーに格納されますが、一定の条件でキュー表にあふれることがあります。この記憶域のタイプは、**バッファ・キュー**記憶域と呼ばれることがあります。

バッファ・メッセージ機能は、より高いパフォーマンスを提供しますが、メッセージ保持などの一部のメッセージ機能をサポートしません。メッセージ保持では、デキュー後にメッセージがキュー表に保持される時間を指定できます。また、データベースが予期せず停止した場合、永続メッセージはディスクに保持されますが、バッファ・メッセージはメモリーに格納されているため失われることがあります。

メッセージ保留は、一部の組織が必要な場合がある監査証跡を提供します。たとえば、メッセージがキュー表に格納されている間でも、ユーザーおよびアプリケーションは問合せを実行してそのメッセージに関する情報を収集することができます。その例として、販売アプリケーションでは販売情報を収集して注文に関するレポートを生成できます。

**参照:**

- 永続メッセージ機能とバッファ・メッセージ機能の相違点については、『Oracle Streams アドバンスド・キューイング・ユーザーズ・ガイド』を参照してください。

## メッセージ通知の概要

Oracle Streams Advanced Queuing (AQ) には、対象のメッセージがエンキューされたときにアプリケーションおよびユーザーに通知する機能があります。通知によって、アプリケーションおよびユーザーは、作業中にキューで新しいメッセージを頻繁にチェックする必要がなくなります。アプリケーションへの通知の送信には、PL/SQL、Java Message Service (JMS) または Oracle Call Interface (OCI) のコールバック機能を使用できます。通知は、指定した電子メール・アドレスや HTTP ポストに送信することもできます。また、通知は RAW データ型形式でも XML 形式でも表示することができます。通知を受信する際、アプリケーションおよびユーザーはデータベースに接続している必要はありません。対象のメッセージがキューに出現したことを通知されてからデータベースに接続し、キューをチェックできます。

### 参照：

- 「メッセージ機能の準備」(9-5 ページ)
- 「チュートリアル:メッセージ通知の構成」(9-16 ページ)

## 伝播の概要

伝播は、あるキューから別のキューにメッセージを送信するプロセスです。これらのキューは同じデータベースにあっても異なるデータベースにあってもかまいません。伝播はメッセージング・システムに必須の要素ではありませんが、伝播により、アプリケーションは同じデータベースまたは同じキューに接続されていなくても相互に通信できます。また、複数の伝播を使用して、あるキューから他の複数のキューに同じメッセージを送信できます。

伝播では、データベース・リンクを使用して、異なるデータベースのキュー間でメッセージを送信します。**データベース・リンク**は、Oracle Database から別のデータベースへの 1 方向通信パスを定義するポインタです。伝播は、**伝播ジョブ**と呼ばれる Oracle Scheduler ジョブによって実行されます。伝播スケジュールは、伝播がメッセージを送信する頻度を決定します。また、各伝播ジョブのスケジュールを制御できます。

Oracle Streams Advanced Queuing (AQ) は、キューとキューの間 (キュー・ツー・キュー) でも、キューとデータベース・リンクの間 (キュー・ツー・dblink) でも伝播をサポートしています。**キュー・ツー・キュー伝播**では、常に独自の排他的な伝播ジョブを使用してソース・キューから宛先キューにメッセージを伝播します。そのため、キュー・ツー・キューの各伝播スケジュールは個別の管理が可能です。キュー・ツー・キューの複数の伝播で、1 つのデータベース・リンクを使用できます。1 つのキューからメッセージを送信する複数の伝播をファイナリティンク制御する際に、キュー・ツー・キュー伝播を使用し、各伝播が異なるスケジュールを使用する必要があります。

一方、**キュー・ツー dblink 伝播**の場合は、同じデータベース・リンクを使用する同じソース・キューからのキュー・ツー dblink 伝播の間で、伝播ジョブが共有されます。複数の伝播によって、1 つのソース・キューからリモート・データベースの複数のキューにメッセージが送信される場合、同じ伝播スケジュールが共有されます。伝播スケジュールに変更があれば、データベース・リンクを使用する同じソース・キューからのキュー・ツー dblink 伝播のすべてに影響します。1 つのキューからメッセージを送信する複数の伝播を一括処理する際、キュー・ツー dblink 伝播を使用し、すべての伝播が同じスケジュールを使用する必要があります。

### 参照：

- 「メッセージ機能の準備」(9-5 ページ)
- 「チュートリアル: Oracle Database 間でのメッセージの送信」(9-6 ページ)
- 「伝播の変更」(9-28 ページ)
- 「チュートリアル: データベース・リンクの作成」(2-8 ページ)
- 『Oracle Streams 概要および管理』

## Oracle Messaging Gateway の概要

Oracle Messaging Gateway は、Oracle Streams Advanced Queuing (AQ) メッセージング・システムを、IBM Websphere MQ (以前は MQSeries と呼ばれていました) や TIBCO Rendezvous などの他のメッセージング・システムと統合します。この統合により、Oracle Streams AQ を使用する Oracle Database アプリケーションは、Oracle 以外のメッセージング・システムを使用する他のアプリケーションと、双方向でシームレスに通信できます。

Messaging Gateway では、ゲートウェイ・エージェントを使用して、Oracle Streams AQ キューから Oracle 以外のメッセージング・システムのキューにメッセージを送信します。また、ゲートウェイ・エージェントによって Oracle Streams AQ キューが Oracle 以外のメッセージング・システムからメッセージを受信できるようになります。

Messaging Gateway は、Oracle 以外のメッセージング・システムに直接マッピングできる Oracle Streams AQ タイプの自動タイプ変換をサポートします。直接マッピングできないタイプの場合は、カスタム変換を定義してタイプをマッピングできます。一度定義されると、カスタム変換は伝播の間実行されます。

Messaging Gateway は、メッセージ・システムのネイティブ・メッセージの書式をサポートします。Oracle Streams AQ メッセージには、RAW または Oracle Streams AQ にサポートされる Oracle のオブジェクト型ペイロードが含まれます。IBM Websphere MQ メッセージは、テキストまたはバイト形式のメッセージです。TIBCO Rendezvous メッセージは、TIBCO Rendezvous ワイヤ書式のデータ型です。

Oracle Streams AQ キューには、インターネットを通じて安全にアクセスできます。Messaging Gateway は、安全なインターネット対応メッセージングを Oracle Streams AQ キューと Oracle 以外のメッセージング・システムのキューの間に提供します。

### 参照：

- Oracle Messaging Gateway とその使用方法の詳細は、『Oracle Streams アドバンスド・キューイング・ユーザーズ・ガイド』を参照してください。

## メッセージ機能の準備

この項では、メッセージング環境用のデータベースの準備に必要なアクションについて説明します。

### メッセージング環境用にデータベースを準備するには：

1. メッセージング環境に関係する各データベースで GLOBAL\_NAMES 初期化パラメータを TRUE に設定します。方法については、2-2 ページの「GLOBAL\_NAMES 初期化パラメータを TRUE に設定」を参照してください。
2. ネットワーク接続を構成して、メッセージを送信するデータベースがメッセージを受信するデータベースと通信できるようにします。データベース間のネットワーク接続の構成の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。
3. メッセージング環境に関係する各データベースで Oracle Streams 管理者を構成します。方法については、2-3 ページの「チュートリアル: Oracle Streams 管理者の作成」を参照してください。
4. Oracle Streams プールが、メッセージング環境に対して作成されるキューを収容するのに十分な大きさであることを確認します。各キューには少なくとも 10MB のメモリが必要です。Oracle Streams プールはシステム・グローバル領域 (SGA) の一部です。MEMORY\_TARGET 初期化パラメータ (自動メモリ管理)、SGA\_TARGET 初期化パラメータ (自動共有メモリ管理) または STREAMS\_POOL\_SIZE 初期化パラメータを設定することで Oracle Stream プールを管理できます。Oracle Streams プールの詳細は、『Oracle Streams 概要および管理』を参照してください。

**参照 :**

- 「メッセージ機能の概要」 (9-2 ページ)
- 「チュートリアル : Oracle Database 間でのメッセージの送信」 (9-6 ページ)
- 「チュートリアル : メッセージ通知の構成」 (9-16 ページ)

## チュートリアル : Oracle Database 間でのメッセージの送信

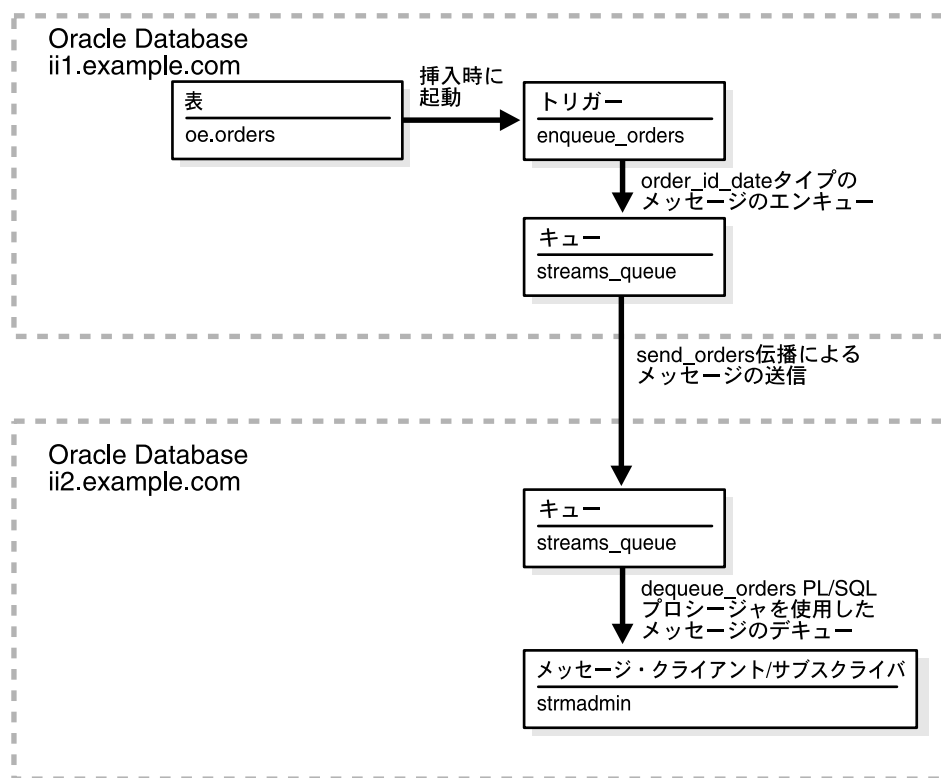
この項の例には、本社の Oracle Database に注文を入力する企業が関係します。注文が入力されると、企業はメッセージング・システムを使用して、注文 ID と注文日を異なる場所の倉庫にある Oracle Database に送信します。これらのメッセージは、倉庫の従業員に注文を通知し、従業員が注文を履行して出荷できるようにします。倉庫の従業員は、本社のデータにアクセスして、特定の注文に関する詳細情報を検索できます。

この例では、注文入力スキーマとして oe サンプル・スキーマを使用します。oe サンプル・スキーマは Oracle Database とともにデフォルトでインストールされます。

この例では、本社のデータベースのグローバル名は ii1.example.com で、倉庫のデータベースのグローバル名は ii2.example.com です。ただし、使用している環境の任意の 2 つのデータベースをかわりに使用して例を完了することができます。

図 9-1 に、この例で作成するメッセージング環境の概要を示します。

図 9-1 データベース間でメッセージを送信するメッセージング環境の例



この例を開始する前に、9-5 ページの「メッセージ機能の準備」で説明されているタスクを完了します。



注文に関するメッセージを ii1.example.com データベースから ii2.example.com データベースに送信するメッセージング・システムを構成するには：

- タスク 1: 各データベースでのメッセージ・タイプの作成
- タスク 2: キューの構成とキュー間の伝播の構成
- タスク 3: メッセージ・エンキュー・メカニズムの構成
- タスク 4: メッセージをデキューするためのメッセージ・クライアントの構成
- タスク 5: メッセージのエンキュー
- タスク 6: メッセージのデキュー

参照：

- 「データベース間のメッセージの送信時期」 (1-6 ページ)

## タスク 1: 各データベースでのメッセージ・タイプの作成

注文について送信する情報を定義するユーザー定義タイプを作成します。この例では、注文 ID と注文日を含むメッセージ用に `strmadmin.order_id_date` タイプを作成します。

各データベースに `strmadmin.order_id_date` タイプを作成するには：

1. まだ準備していない場合は、メッセージ機能用に環境を準備します。9-5 ページの「メッセージ機能の準備」を参照してください。
2. Oracle Enterprise Manager で、Oracle Streams 管理者として `ii1.example.com` データベースにログインします。
3. 「データベース」 ホームページに移動します。
4. 「スキーマ」をクリックして「スキーマ」サブページを開きます。
5. 「ユーザー定義タイプ」セクションの「オブジェクト・タイプ」をクリックします。
6. 「オブジェクト・タイプ」ページで、「作成」をクリックして「オブジェクト・タイプの作成」ページを開きます。

Database Instance: database > Object Types > Logged in As STRMADMIN

Create Object Type Show SQL Cancel OK

**General**

\* Name

\* Schema

**Attributes** Datatype: Predefined type Add

| Select Name | Schema | Object type | Length | Precision | Ref |
|-------------|--------|-------------|--------|-----------|-----|
|             |        |             |        |           |     |

**Methods** Add

| Select Name | Method type | Number of Parameters |
|-------------|-------------|----------------------|
|             |             |                      |

Show SQL Cancel OK

7. 「名前」フィールドに `order_id_date` と入力します。
8. 「スキーマ」フィールドで `strmadmin` が選択されていることを確認します。
9. 「属性」セクションで「データ型」に対して「事前定義済タイプ」が選択されていることを確認します。
10. 「属性」セクションで「追加」をクリックして、「事前定義済タイプ属性の追加」ページを開きます。

Logged in As STRMADMIN

**Add Predefined Type Attribute**

\* Name

Type

Length

Only applies to CHAR, VARCHAR, NUM, FLOAT, and RAW

Precision

Only applies to NUMBER

Cancel OK

Cancel OK

11. 属性を追加するには：
  - a. 「名前」フィールドに `order_id` と入力します。
  - b. 「タイプ」では `NUMBER` を選択します。
  - c. 「長さ」フィールドに `12` と入力します。
  - d. 「OK」をクリックします。
12. 「オブジェクト・タイプの作成」ページの「属性」セクションで「事前定義済タイプ」が選択されていることを確認します。
13. 「属性」セクションで「追加」をクリックして、「事前定義済タイプ属性の追加」ページを開きます。
14. 属性を追加するには：
  - a. 「名前」フィールドに `order_date` と入力します。
  - b. 「タイプ」では `VARCHAR2` を選択します。
  - c. 「長さ」フィールドに `100` と入力します。
  - d. 「OK」をクリックします。
15. 「オブジェクト・タイプの作成」ページで、「OK」をクリックしてタイプを作成します。
16. Enterprise Manager の Oracle Streams 管理者として `ii2.example.com` データベースにログインします。
17. `ii2.example.com` データベースで手順 3～15 を完了して、両方のデータベースが `strmadmin.order_id_date` タイプを持つようにします。
18. この拡張例を続行するには、「タスク 2: キューの構成とキュー間の伝播の構成」の手順に進んでください。

---

**注意：** SQL 文 `CREATE TYPE` を使用して、タイプを作成することもできます。

---

## タスク 2: キューの構成とキュー間の伝播の構成

各データベースにキューを作成し、注文に関するメッセージを `ii1.example.com` データベースのキューから `ii2.example.com` データベースのキューに送信する伝播を作成します。

伝播を作成するには:

1. `streams_queue` という名前のキューを `ii1.example.com` データベースと `ii2.example.com` データベース両方の Oracle Streams 管理者のスキーマに作成します。方法については、2-7 ページの「[ANYDATA キューの作成](#)」を参照してください。
2. `ii2.example.com` という名前のデータベース・リンクを `ii1.example.com` の Oracle Streams 管理者スキーマに作成します。Oracle Streams 管理者スキーマに接続するデータベース・リンクを `ii2.example.com` に構成します。データベース・リンクのサービス名は、`ii2.example.com` である必要があります。方法については、2-8 ページの「[チュートリアル: データベース・リンクの作成](#)」を参照してください。
3. Oracle Enterprise Manager で、Oracle Streams 管理者として `ii1.example.com` データベースにログインします。
4. 「データベース」 ホームページに移動します。
5. 「データ移動」 をクリックして「データ移動」 サブページを開きます。
6. 「Streams」 セクションの「管理」 をクリックします。
7. 「Streams」 ページで、「伝播」 をクリックして「伝播」 サブページを開きます。
8. 「伝播の設定」 をクリックして「伝播の設定」 ページを開きます。

9. 「伝播名」 フィールドに `send_orders` と入力します。
10. 「ソース・キュー」 フィールドに `strmadmin.streams_queue` と入力します。このキューは、注文に関するメッセージがエンキューされる `ii1.example.com` データベースのキューです。
11. 手順 2 で作成したデータベース・リンクの名前を「接続先データベース・リンク」フィールドに入力します。この例では、データベース・リンク名は `ii2.example.com` です。
12. 「宛先キュー」 フィールドで、`strmadmin.streams_queue` を入力します。これは注文に関するメッセージが送信される `ii2.example.com` データベースのキューです。
13. 「ポジティブ・ルール・セット」 フィールドと「ネガティブ・ルール・セット」 フィールドは空のままにします。伝播にルール・セットがない場合は、ソース・キュー内のすべてのメッセージが宛先キューに送信されます。
14. 「OK」 をクリックして伝播を作成します。
15. この拡張例を続行するには、「[タスク 3: メッセージ・エンキュー・メカニズムの構成](#)」の手順に進んでください。

**注意:** `DBMS_PROPAGATION_ADM.CREATE_PROPAGATION` プロシージャを使用して、伝播を作成することもできます。

## タスク 3: メッセージ・エンキュー・メカニズムの構成

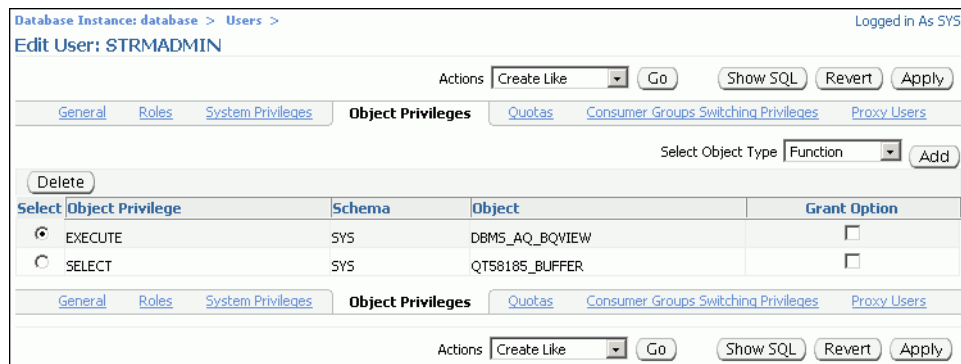
メッセージング・システム内のメッセージをエンキューするメカニズムを構成します。通常、アプリケーションは、別のアプリケーションによってデキューされ、処理されるメッセージを作成し、エンキューします。簡略化のために、この例では `enqueue_orders` というトリガーを作成して、注文の注文 ID と注文日を含むメッセージをエンキューします。トリガーは、注文が `oe.orders` 表に挿入されたときに起動します。

### メッセージ・エンキュー・メカニズムを構成するには：

1. Oracle Streams 管理者に、`DBMS_STREAMS_MESSAGING` パッケージに対する `EXECUTE` 権限を付与します。

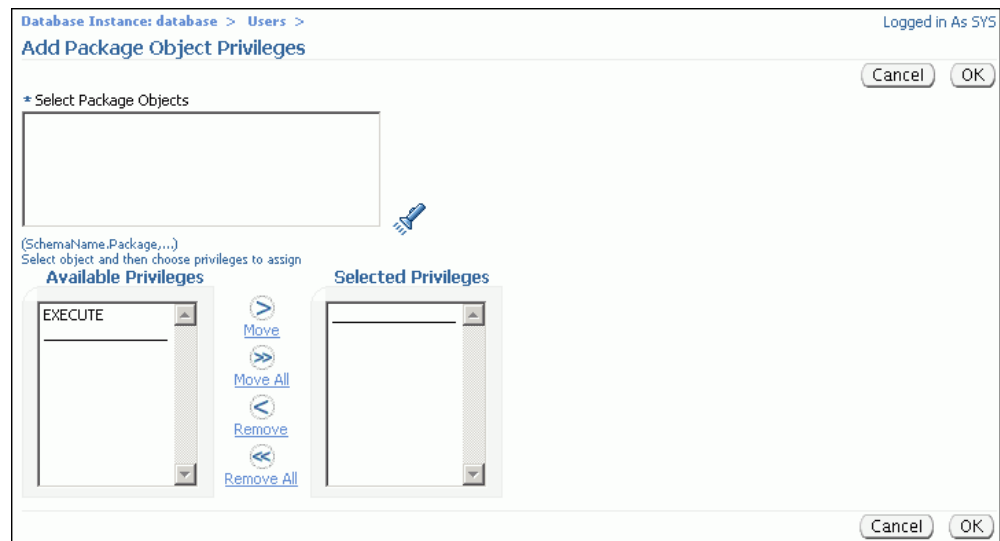
この例では、`DBMS_STREAMS_MESSAGING` パッケージの `ENQUEUE` プロシージャを実行するトリガーを構成します。トリガー内のこのプロシージャを実行するユーザーには、プロシージャを含むパッケージに対する明示的な `EXECUTE` 権限が必要です。ロールを通じて権限を付与することはできません。したがって、Oracle Streams 管理者にはパッケージに対する明示的な `EXECUTE` 権限を付与する必要があります。

- a. `strmadmin` ユーザーに権限を付与できる管理ユーザーとして **Enterprise Manager** にログインします。たとえば、`SYSDBA` 権限を持つユーザーとしてログインできます。この例では、`ii1.example.com` データベースにログインします。
- b. 「データベース」ホームページに移動します。
- c. 「サーバー」をクリックして「サーバー」サブページを開きます。
- d. 「セキュリティ」セクションで「ユーザー」をクリックします。  
「ユーザー」ページが表示されます。
- e. `STRMADMIN` ユーザーを選択します。
- f. 「編集」をクリックします。  
「ユーザーの編集」ページが表示され、「一般」サブページが表示されます。
- g. 「オブジェクト権限」をクリックして「オブジェクト権限」サブページを開きます。



- h. 「オブジェクト・タイプの選択」リストの「パッケージ」を選択します。

- i. 「追加」をクリックしてパッケージ・オブジェクト権限の追加ページを開きます。



- j. パッケージ・オブジェクトの選択フィールドに `SYS.DBMS_STREAMS_MESSAGING` と入力します。
- k. 「使用可能な権限」リストから「選択した権限」リストに **EXECUTE** を移動します。
- l. 「OK」をクリックして権限を追加します。
- m. 「適用」をクリックして権限を付与します。
- n. Enterprise Manager からログアウトします。

---

**注意:** SQL 文 GRANT を使用して、ユーザーに権限を付与することもできます。

---

2. `oe.orders` 表に変更があったときメッセージを自動的にエンキューするトリガーを、`ii1.example.com` データベースに作成します。
- a. Oracle Streams 管理者として Enterprise Manager の `ii1.example.com` データベースにログインします。
- b. 「データベース」ホームページに移動します。
- c. 「スキーマ」をクリックして「スキーマ」サブページを開きます。
- d. 「プログラム」セクションで「トリガー」をクリックします。

- e. 「トリガー」 ページで、「作成」 をクリックします。  
「トリガーの作成」 ページが表示され、「一般」 サブページが表示されます。

- f. 「名前」 フィールドに enqueue\_orders と入力します。
- g. 「スキーマ」 フィールドで strmadmin が入力されていることを確認します。
- h. 「トリガー本体」 フィールドに次のように入力します。
- ```

DECLARE
  message  strmadmin.order_id_date;
BEGIN
  message := strmadmin.order_id_date(
    order_id  => :NEW.order_id,
    order_date => TO_CHAR(:NEW.order_date));
  DBMS_STREAMS_MESSAGING.ENQUEUE (
    queue_name => 'strmadmin.streams_queue',
    payload    => ANYDATA.CONVERTOBJECT(message));
END;

```
- i. 「イベント」 をクリックして「イベント」 サブページを開きます。
- j. 「対象オブジェクト」 リストで「表」 が選択されていることを確認します。
- k. 「表 (Schema.Table)」 フィールドに oe.orders と入力します。
- l. 「起動タイミング」 では「後」 を選択します。
- m. 「イベント」 では「挿入」 を選択します。この例では、メッセージが既存の注文に対する変更ではなく新しい注文を追跡するため、「挿入」 のみが選択されている必要があります。
- n. 「詳細」 をクリックします。
- o. 「行レベル・トリガー」 を選択します。
- p. 「参照」 セクションの「旧値 (OLD AS)」 フィールドに OLD を入力します。

- q. 「参照」セクションの「**新規値 (NEW AS)**」フィールドに NEW を入力します。
- r. 「OK」をクリックしてトリガーを作成します。

注意: SQL 文 CREATE TRIGGER を使用して、トリガーを作成することもできます。

- 3. この拡張例を続行するには、「[タスク 4: メッセージをデキューするためのメッセージ・クライアントの構成](#)」の手順に進んでください。

タスク 4: メッセージをデキューするためのメッセージ・クライアントの構成

メッセージング・システム内のメッセージをデキューするメカニズムを構成します。通常、アプリケーションは別のアプリケーションによって作成されたメッセージをデキューして処理します。簡略化のために、この例では dequeue_orders という PL/SQL プロシージャを作成して、注文の注文 ID および注文日を含むメッセージをデキューします。この例ではプロシージャをコールしてメッセージをデキューしますが、アプリケーションはこのようなプロシージャを定期的に行うことができます。

メッセージをデキューするためのメッセージ・クライアントを構成するには:

- 1. ii2.example.com データベースの Oracle Streams 管理者に、SYS.DBMS_STREAMS_MESSAGING パッケージに対する EXECUTE 権限を付与します。ii1.example.com データベースの Oracle Streams 管理者にこの権限を付与する例は、9-10 ページの「[タスク 3: メッセージ・エンキュー・メカニズムの構成](#)」の手順 1 を参照してください。

- 2. コマンドラインで SQL*Plus を開き、Oracle Streams 管理者として ii2.example.com データベースに接続します。

SQL*Plus の起動の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。

- 3. メッセージ・クライアントを作成して、Oracle Streams 管理者が streams_queue キューからメッセージをデキューできるようにします。

```
BEGIN
  DBMS_STREAMS_ADM.ADD_MESSAGE_RULE (
    message_type => 'strmadmin.order_id_date',
    rule_condition => ':MSG.ORDER_ID > 0',
    streams_type => 'DEQUEUE',
    streams_name => 'strmadmin',
    queue_name => 'strmadmin.streams_queue');
END;
/
```

Oracle Streams 管理者のユーザー名は、streams_name パラメータで指定する必要があります。この例では、Oracle Streams 管理者のユーザー名は strmadmin です。

メッセージ・クライアントは、ルールを使用して、どのメッセージをデキューするかを判断します。この例では、メッセージ・クライアントのルールは、order_id が 0 より大きいすべてのメッセージをデキューする必要があることを指定します。このため、このルールを使用すると、メッセージ・クライアントは、strmadmin.streams_queue キューに出現する strmadmin.order_id_date タイプの新しいメッセージをすべてデキューします。

- 4. Oracle Enterprise Manager で、Oracle Streams 管理者として ii2.example.com データベースにログインします。

5. 「データベース」 ホームページに移動します。
6. 「スキーマ」 をクリックして 「スキーマ」 サブページを開きます。
7. 「プログラム」 セクションで 「プロシージャ」 をクリックします。
8. 「プロシージャ」 ページで 「作成」 をクリックします。
9. 「プロシージャの作成」 ページで、 「名前」 フィールドに dequeue_orders と入力します。
10. 「スキーマ」 フィールドで strmadmin が入力されていることを確認します。
11. 「ソース」 フィールドのサンプル・テキストを削除します。
12. 「ソース」 フィールドに次のように入力します。

```
AS
msg          ANYDATA;
user_msg     strmadmin.order_id_date;
num_var      PLS_INTEGER;
more_messages BOOLEAN := TRUE;
navigation   VARCHAR2(30);
BEGIN
navigation := 'FIRST MESSAGE';
WHILE (more_messages) LOOP
  BEGIN
    DBMS_STREAMS_MESSAGING.DEQUEUE (
      queue_name => 'strmadmin.streams_queue',
      streams_name => 'strmadmin',
      payload     => msg,
      navigation  => navigation,
      wait        => DBMS_STREAMS_MESSAGING.NO_WAIT);
    IF msg.GETTYPENAME() = 'STRMADMIN.ORDER_ID_DATE' THEN
      num_var := msg.GETOBJECT(user_msg);
      DBMS_OUTPUT.PUT_LINE('Order ID: ' || user_msg.order_id);
      DBMS_OUTPUT.PUT_LINE('Order Date: ' || user_msg.order_date);
    END IF;
    navigation := 'NEXT MESSAGE';
    COMMIT;
  EXCEPTION WHEN SYS.DBMS_STREAMS_MESSAGING.ENDOFCURTRANS THEN
    navigation := 'NEXT TRANSACTION';
  WHEN DBMS_STREAMS_MESSAGING.NOMOREMSGS THEN
    more_messages := FALSE;
    DBMS_OUTPUT.PUT_LINE('No more messages. ');
  WHEN OTHERS THEN
    RAISE;
  END;
END LOOP;
END;
```

13. 「OK」 をクリックします。
14. この拡張例を続行するには、「タスク 5: メッセージのエンキュー」 の手順に進んでください。

注意： SQL 文 CREATE PROCEDURE を使用して、プロシージャを作成することもできます。

タスク 5: メッセージのエンキュー

9-10 ページの「[タスク 3: メッセージ・エンキュー・メカニズムの構成](#)」の例では、トリガーを作成しました。トリガーは、oe.orders 表に挿入された注文に関する情報とともにメッセージを strmadmin.streams_queue キューにエンキューします。このため、oe.orders 表に行を挿入することにより、メッセージを strmadmin.streams_queue キューにエンキューできます。

行を oe.orders 表に挿入するには:

1. コマンドラインで SQL*Plus を開き、oe ユーザーとして ii1.example.com データベースに接続します。

SQL*Plus の起動の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。

2. SQL*Plus で、行を oe.orders 表に挿入します。

```
INSERT INTO oe.orders VALUES (3000, SYSDATE, 'direct', 116, 0, 4000.00, 153, NULL);
```

```
INSERT INTO oe.orders VALUES (3001, SYSDATE, 'direct', 117, 5, 5000.00, 163, NULL);
```

```
INSERT INTO oe.orders VALUES (3002, SYSDATE, 'direct', 118, 7, 6000.00, 159, NULL);
```

3. SQL*Plus で、変更をコミットします。

```
COMMIT;
```

4. この拡張例を続行するには、「[タスク 6: メッセージのデキュー](#)」の手順に進んでください。

タスク 6: メッセージのデキュー

9-15 ページの「[タスク 5: メッセージのエンキュー](#)」の例では、ii1.example.com データベースの strmadmin.streams_queue キューにメッセージをエンキューします。メッセージがエンキューされた後で、9-9 ページの「[タスク 2: キューの構成とキュー間の伝播の構成](#)」で構成した伝播により、メッセージが ii2.example.com データベースの strmadmin.streams_queue キューに送信されます。

この例では、9-13 ページの「[タスク 4: メッセージをデキューするためのメッセージ・クライアントの構成](#)」で作成した dequeue_orders プロシージャを使用して、ii2.example.com データベースのキューからメッセージをデキューします。

メッセージをデキューするには:

1. オプションとして、メッセージのデキューを試行する前に、ii2.example.com の strmadmin.streams_queue キューにメッセージが届いているかどうかをチェックします。伝播によってメッセージが ii1.example.com のキューから ii2.example.com のキューに送信されるまでには時間がかかることがあります。方法については、9-31 ページの「[キュー内のメッセージの表示](#)」を参照してください。

「メッセージ」ページには、3 つのメッセージが表示されます。各メッセージには、9-15 ページの「[タスク 5: メッセージのエンキュー](#)」で oe.orders 表に挿入した注文に関する情報が含まれます。

2. コマンドラインで SQL*Plus を開き、Oracle Streams 管理者として ii2.example.com データベースに接続します。

SQL*Plus の起動の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。

3. SQL*Plus で、`dequeue_orders` プロシージャを実行します。

```
SET SERVEROUTPUT ON  
exec dequeue_orders;
```

出力は、次のようになります。

```
Order ID: 3000  
Order Date: 01-FEB-07 01.47.48.000000000 PM  
Order ID: 3001  
Order Date: 01-FEB-07 01.47.57.000000000 PM  
Order ID: 3002  
Order Date: 01-FEB-07 01.48.04.000000000 PM  
No more messages.
```

```
PL/SQL procedure successfully completed.
```

チュートリアル:メッセージ通知の構成

対象のメッセージがエンキューされたときにアプリケーションまたはユーザーに警告するメッセージ通知を構成することができます。

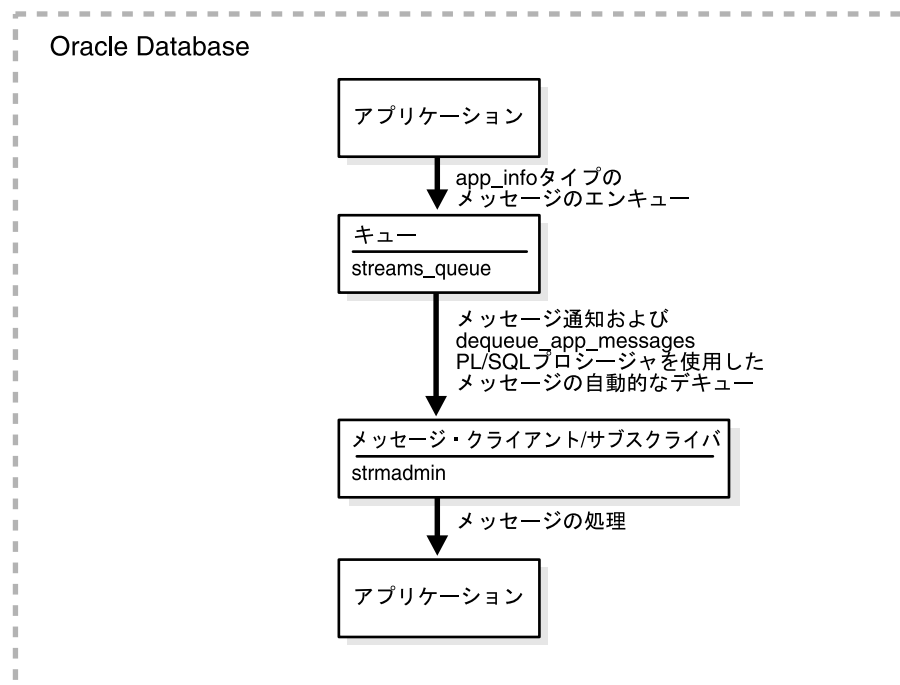
この項の例では、メッセージ通知を使用して、2つのアプリケーションが相互に通信できるようにする方法を示します。この例では、アプリケーションは次の方法で通信します。

1. 最初のアプリケーションは、2番目のアプリケーションで設定する必要がある様々なパラメータと、それらのパラメータの値を判断します。
2. 最初のアプリケーションは、次の属性を含むメッセージをエンキューします。
 - `parameter`: 2番目のアプリケーションで設定するパラメータを指定する
 - `value`: パラメータ値を指定する
3. メッセージ通知は、キュー内に新しいメッセージがあることを2番目のアプリケーションに警告します。
4. 2番目のアプリケーションは、メッセージをデキューし、パラメータをメッセージ内の値に設定します。

簡略化のために、この例では2つのアプリケーションを作成しません。かわりに、メッセージ通知の構成方法、および対象のメッセージがエンキューされたときにメッセージ通知を使用してそのメッセージを自動的にデキューする方法を示します。

図 9-2 に、この例で作成するメッセージング環境の概要を示します。

図 9-2 メッセージ通知を使用するメッセージング環境の例



注意: データベース間でメッセージを送信する環境でメッセージ通知を構成できます。データベース間でメッセージを送信する環境の例の詳細は、9-6 ページの「チュートリアル: Oracle Database 間でのメッセージの送信」を参照してください。

メッセージ通知を構成するには:

- タスク 1: メッセージ・タイプの作成
- タスク 2: キューおよびメッセージ・クライアントの構成
- タスク 3: メッセージをデキューするメカニズムの構成
- タスク 4: メッセージ通知の構成
- タスク 5: メッセージのエンキューとメッセージ通知のチェック

参照:

- 「メッセージ通知の概要」 (9-4 ページ)
- メッセージ通知機能の詳細は、『Oracle Streams アドバンスド・キューイング・ユーザーズ・ガイド』を参照してください。
- 電子メール・メッセージ通知を構成する例は、『Oracle Streams 概要および管理』を参照してください。

タスク 1: メッセージ・タイプの作成

ユーザー定義型を作成して、アプリケーションについて追跡する情報を定義します。この例では、パラメータと値を含むメッセージに対して `strmadmin.app_info` 型を作成します。

strmadmin.app_info 型を作成するには:

1. まだ準備していない場合は、メッセージ機能用に環境を準備します。9-5 ページの「[メッセージ機能の準備](#)」を参照してください。
2. Oracle Enterprise Manager で、Oracle Streams 管理者としてハブ・データベースにログインします。
3. 「データベース」ホームページに移動します。
4. 「スキーマ」をクリックして「スキーマ」サブページを開きます。
5. 「ユーザー定義タイプ」セクションの「オブジェクト・タイプ」をクリックします。
6. 「オブジェクト・タイプ」ページで、「作成」をクリックして「オブジェクト・タイプの作成」ページを開きます。

Database Instance: database > Object Types > Create Object Type

Logged in As STRMADMIN

General

* Name

* Schema STRMADMIN

Attributes

Datatype: Predefined type

Select Name	Schema	Object type	Length	Precision	Ref

Methods

Select Name	Method type	Number of Parameters

7. 「名前」フィールドに `app_info` と入力します。
8. 「スキーマ」フィールドで `strmadmin` が選択されていることを確認します。
9. 「属性」セクションで「データ型」に対して「事前定義済タイプ」が選択されていることを確認します。
10. 「属性」セクションで「追加」をクリックして、「事前定義済タイプ属性の追加」ページを開きます。

Logged in As STRMADMIN

Add Predefined Type Attribute

* Name

Type CHAR

Length

Only applies to CHAR, VARCHAR, NUM, FLOAT, and RAW

Precision

Only applies to NUMBER

11. app_info 型に 1 番目の属性を追加するには:
 - a. 「名前」フィールドに parameter と入力します。
 - b. 「タイプ」では VARCHAR2 を選択します。
 - c. 「長さ」フィールドに 20 と入力します。
 - d. 「OK」をクリックします。
12. 「オブジェクト・タイプの作成」ページの「属性」セクションで「データ型」に対して「事前定義済タイプ」が選択されていることを確認します。
13. 「属性」セクションで「追加」をクリックして、「事前定義済タイプ属性の追加」ページを開きます。
14. app_info 型に次の属性を追加するには:
 - a. 「名前」フィールドに value と入力します。
 - b. 「タイプ」では NUMBER を選択します。
 - c. 「長さ」フィールドに 12 と入力します。
 - d. 「OK」をクリックします。
15. 「オブジェクト・タイプの作成」ページで、「OK」をクリックしてタイプを作成します。
16. この拡張例を続行するには、「[タスク 2: キューおよびメッセージ・クライアントの構成](#)」の手順に進んでください。

注意: SQL 文 CREATE TYPE を使用して、タイプを作成することもできます。

参照:

- 「[チュートリアル:メッセージ通知の構成](#)」(9-16 ページ)
- 「[メッセージ機能の概要](#)」(9-2 ページ)

タスク 2: キューおよびメッセージ・クライアントの構成

キュー表とキューを作成して、最初のアプリケーションが 2 番目のアプリケーションに対して生成するメッセージを格納します。キューの作成後に、メッセージ通知はメッセージをキューからデキューできるコンシューマを必要とします。この例では、メッセージ・クライアントが、streams_queue キューからメッセージをデキューできるコンシューマです。

キューおよびメッセージ・クライアントを構成するには:

1. streams_queue という名前のキューを Oracle Streams 管理者のスキーマに作成します。方法については、2-7 ページの「[ANYDATA キューの作成](#)」を参照してください。
2. コマンドラインで SQL*Plus を開き、データベースに Oracle Streams 管理者として接続します。

SQL*Plus の起動の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。

3. メッセージ・クライアントを作成して、Oracle Streams 管理者が streams_queue キューからメッセージをデキューできるようにします。

```
BEGIN
  DBMS_STREAMS_ADM.ADD_MESSAGE_RULE (
    message_type => 'strmadmin.app_info',
    rule_condition => ':MSG.VALUE >= 0',
    streams_type => 'DEQUEUE',
    streams_name => 'strmadmin',
    queue_name => 'strmadmin.streams_queue');
END;
/
```

Oracle Streams 管理者のユーザー名は、streams_name パラメータで指定する必要があります。この例では、Oracle Streams 管理者のユーザー名は strmadmin です。新しいメッセージ・クライアントの名前も strmadmin です。

メッセージ・クライアントは、ルールを使用して、どのメッセージをデキューするかを判断します。この例では、メッセージ・クライアントのルールは、value が 0 よりも大きいすべてのメッセージをデキューする必要があることを指定します。このため、このルールを使用すると、メッセージ・クライアントは、すべてのパラメータ値が 0 以上であるため、strmadmin.streams_queue に出現する strmadmin.app_info 型の新しいメッセージをすべてデキューします。

4. この拡張例を続行するには、「[タスク 3: メッセージをデキューするメカニズムの構成](#)」の手順に進んでください。

参照:

- 「[チュートリアル:メッセージ通知の構成](#)」(9-16 ページ)
- 「[メッセージ機能の概要](#)」(9-2 ページ)

タスク 3: メッセージをデキューするメカニズムの構成

この拡張例では、2 番目のアプリケーションが streams_queue キューからメッセージをデキューする必要があります。簡略化のために、この例では dequeue_app_messages という PL/SQL プロシージャを作成して、strmadmin.app_info 型のメッセージをデキューします。このプロシージャは、9-19 ページの「[タスク 2: キューおよびメッセージ・クライアントの構成](#)」で作成したメッセージ・クライアントを使用してメッセージをデキューします。

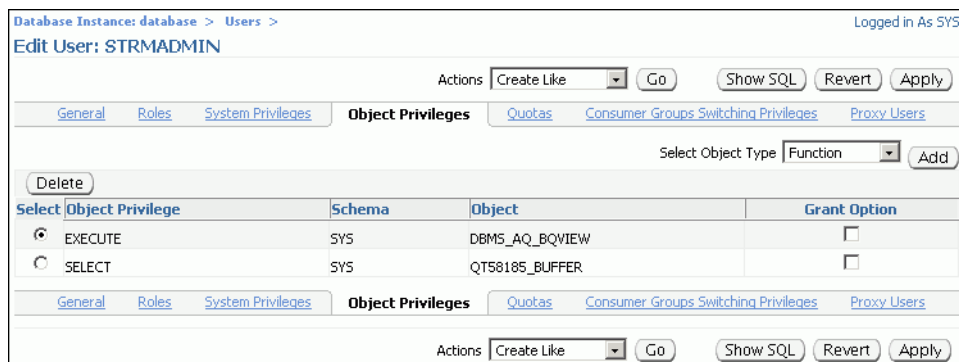
メッセージをデキューするメカニズムを構成するには:

1. Oracle Streams 管理者に、DBMS_AQ パッケージに対する EXECUTE 権限を付与します。

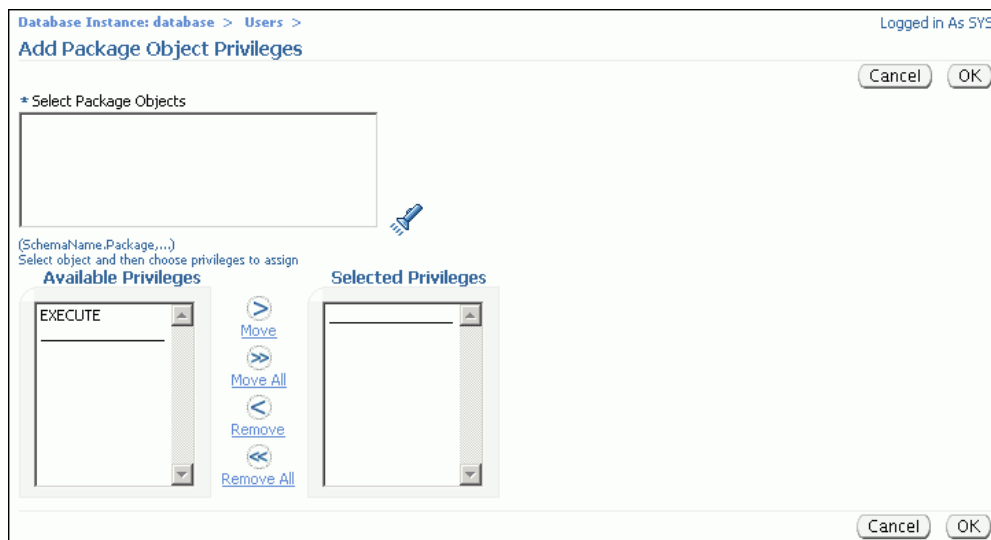
この例では、DBMS_AQ パッケージの DEQUEUE プロシージャを実行するプロシージャを構成します。このプロシージャを実行するユーザーには、プロシージャを含むパッケージに対する明示的な EXECUTE 権限が必要です。ロールを通じて権限を付与することはできません。したがって、Oracle Streams 管理者にはパッケージに対する明示的な EXECUTE 権限を付与する必要があります。

- a. strmadmin ユーザーに権限を付与できる管理ユーザーとして、Enterprise Manager にログインします。
- b. 「サーバー」サブページに進みます。
- c. 「セキュリティ」セクションで「ユーザー」をクリックします。
「ユーザー」ページが表示されます。
- d. STRMADMIN ユーザーを選択します。

- e. 「編集」をクリックします。
「ユーザーの編集」ページの「一般」サブページが表示されます。
- f. 「オブジェクト権限」をクリックして「オブジェクト権限」サブページを開きます。



- g. 「オブジェクト・タイプを選択」リストの「パッケージ」を選択します。
- h. 「追加」をクリックしてパッケージ・オブジェクト権限の追加ページを開きます。



- i. パッケージ・オブジェクトの選択フィールドに SYS.DBMS_AQ と入力します。
- j. 「使用可能な権限」リストから「選択した権限」リストに EXECUTE を移動します。
- k. 「OK」をクリックして権限を追加します。
- l. 「適用」をクリックして権限を付与します。

注意: SQL 文 GRANT を使用して、ユーザーに権限を付与することもできます。

2. メッセージをデキューする PL/SQL プロシージャを作成します。
 - a. Oracle Enterprise Manager で、Oracle Streams 管理者としてハブ・データベースにログインします。
 - b. 「データベース」 ホームページに移動します。
 - c. 「スキーマ」 をクリックして「スキーマ」 サブページを開きます。
 - d. 「プログラム」 セクションで「プロシージャ」 をクリックします。
 - e. 「プロシージャ」 ページで「作成」 をクリックします。
 - f. 「プロシージャの作成」 ページで、「名前」 フィールドに `dequeue_app_messages` と入力します。
 - g. 「スキーマ」 フィールドで `strmadmin` が入力されていることを確認します。
 - h. 「ソース」 フィールドのサンプル・テキストを削除します。
 - i. 「ソース」 フィールドに次のように入力します。

```
(
  context ANYDATA,
  reginfo SYS.AQ$ REG_INFO,
  descr   SYS.AQ$ DESCRIPTOR)
AS
  dequeue_options DBMS_AQ.DEQUEUE_OPTIONS_T;
  message_properties DBMS_AQ.MESSAGE_PROPERTIES_T;
  message_handle RAW(16);
  message ANYDATA;
  app_message strmadmin.app_info;
  rc PLS_INTEGER;
BEGIN
  -- Get the message identifier and consumer name from the descriptor
  dequeue_options.msgid := descr.msg_id;
  dequeue_options.consumer_name := descr.consumer_name;
  -- Dequeue the message
  DBMS_AQ.DEQUEUE(
    queue_name => descr.queue_name,
    dequeue_options => dequeue_options,
    message_properties => message_properties,
    payload => message,
    msgid => message_handle);
  rc := message.getobject(app_message);
  COMMIT;
END;
```

メッセージ通知 PL/SQL プロシージャには次のシグネチャが必要です。

```
PROCEDURE procedure_name(
  context IN ANYDATA,
  reginfo IN SYS.AQ$ REG_INFO,
  descr IN SYS.AQ$ DESCRIPTOR);
```

ここで、`procedure_name` はプロシージャの名前を意味します。プロシージャは、メッセージ通知で起動されるユーザー定義 PL/SQL プロシージャを指定する PLSQLCALLBACK データ構造体です。

この例のプロシージャは、通知によって送信されたメッセージ識別子とコンシューマ名を使用して `strmadmin.app_info` 型のメッセージをデキューする単純な通知プロシージャです。

- j. 「OK」 をクリックしてプロシージャを作成します。

注意: SQL 文 CREATE PROCEDURE を使用して、プロシージャを作成することもできます。

3. この拡張例を続行するには、「[タスク 4: メッセージ通知の構成](#)」の手順に進んでください。

参照:

- 「[チュートリアル:メッセージ通知の構成](#)」 (9-16 ページ)
- 「[メッセージ機能の概要](#)」 (9-2 ページ)

タスク 4: メッセージ通知の構成

この拡張例では、新しいメッセージが `strmadmin.streams_queue` キューにエンキューされたときにメッセージ通知が `strmadmin.dequeue_app_messages` プロシージャを起動します。このプロシージャは、新しいメッセージをデキューします。デキュー後に、アプリケーションはカスタマイズされた任意の方法でメッセージを処理できます。この例では、2 番目のアプリケーションはメッセージ内の情報を使用してアプリケーション・パラメータ値を設定します。簡略化のために、この例では 2 番目のアプリケーションを構成しません。

プロシージャを起動するメッセージ通知を構成するには:

1. コマンドラインで SQL*Plus を開き、データベースに Oracle Streams 管理者として接続します。

SQL*Plus の起動の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。

2. メッセージがエンキューされ、`strmadmin` メッセージ・クライアントがデキューできるときに、`strmadmin.dequeue_app_messages` プロシージャを起動するメッセージ通知を設定します。

```
BEGIN
  DBMS_STREAMS_ADM.SET_MESSAGE_NOTIFICATION (
    streams_name      => 'strmadmin',
    notification_action => 'strmadmin.dequeue_app_messages',
    notification_type  => 'PROCEDURE',
    include_notification => TRUE,
    queue_name        => 'strmadmin.streams_queue');
END;
```

3. この拡張例を続行するには、「[タスク 5: メッセージのエンキューとメッセージ通知のチェック](#)」の手順に進んでください。

参照:

- 「[チュートリアル:メッセージ通知の構成](#)」 (9-16 ページ)
- 「[メッセージ機能の概要](#)」 (9-2 ページ)

タスク 5: メッセージのエンキューとメッセージ通知のチェック

この拡張例では、1 つのアプリケーションが、2 番目のアプリケーションによって処理されるメッセージをエンキューします。簡略化のために、この例ではメッセージを手動でエンキューします。メッセージがエンキューされた後で、メッセージ通知を通じて自動的にデキューされたことを確認できます。

メッセージをエンキューし、メッセージ通知をチェックするには:

1. コマンドラインで SQL*Plus を開き、データベースに Oracle Streams 管理者として接続します。

SQL*Plus の起動の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。

2. メッセージを `strmadmin.streams_queue` キューにエンキューします。

```
DECLARE
    msg    strmadmin.app_info;
BEGIN
    msg := strmadmin.app_info(
        parameter => 'THRESHOLD',
        value     => 1000);
    DBMS_STREAMS_MESSAGING.ENQUEUE (
        queue_name => 'strmadmin.streams_queue',
        payload    => ANYDATA.CONVERTOBJECT(msg));
    COMMIT;
END;
/
```

```
DECLARE
    msg    strmadmin.app_info;
BEGIN
    msg := strmadmin.app_info(
        parameter => 'MIN_WAIT',
        value     => 1);
    DBMS_STREAMS_MESSAGING.ENQUEUE (
        queue_name => 'strmadmin.streams_queue',
        payload    => ANYDATA.CONVERTOBJECT(msg));
    COMMIT;
END;
/
```

```
DECLARE
    msg    strmadmin.app_info;
BEGIN
    msg := strmadmin.app_info(
        parameter => 'BUFFER_SIZE',
        value     => 30);
    DBMS_STREAMS_MESSAGING.ENQUEUE (
        queue_name => 'strmadmin.streams_queue',
        payload    => ANYDATA.CONVERTOBJECT(msg));
    COMMIT;
END;
/
```

3. オプションとして、`streams_queue` キュー内のメッセージを表示し、これらがメッセージ通知を使用して自動的にデキューされたかどうかを確認します。方法については、9-31 ページの「[キュー内のメッセージの表示](#)」を参照してください。

この例で構成したメッセージ通知が正常に動作している場合は、「メッセージ」ページに次のいずれかが表示されます。

- 1 つ以上のエンキューされたメッセージが、「**状態の異なるコンシューマ**」フィールドに「処理済」と表示されることがあります。このフィールドの「処理済」値は、メッセージがすべてのコンシューマによってデキューされたことを意味します。
- キューにメッセージは表示されません。最終的に、処理されたメッセージはキューから自動的に削除されます。メッセージが表示されない場合は、エンキューされたメッセージがメッセージ通知によってデキューされ、自動的に削除されました。

参照:

- 「チュートリアル:メッセージ通知の構成」(9-16 ページ)
- 「メッセージ機能の概要」(9-2 ページ)

キューの変更

Enterprise Manager を使用して、次のプロパティを含め、既存のキューのプロパティの一部を変更できます。

- **再試行最大数:**メッセージが例外キューに移動されるまでに許可されるデキュー試行の数。メッセージング環境が接続不良の場合、メッセージのデキュー処理中に、接続が切断される可能性があります。このような環境では、メッセージが例外キューに移動されるまでアプリケーションが何度もメッセージをデキューするよう、最大再試行数を増やす必要があります。
- **再試行の遅延:**アプリケーションのロールバック後、スケジュールされたメッセージの再処理が行われるまでの時間量。メッセージング環境が接続不良の場合、メッセージのデキューに失敗した後、アプリケーションを一定の時間待機させる必要があります。このような環境では、待機時間を指定して再試行の遅延設定を増やす必要があります。
- **保有時間:**メッセージがすべてのコンシューマにデキューされた後、メッセージがキュー内に保持される時間。消費されたメッセージを追跡する場合、キュー内に格納されたメッセージに関する情報を記録できる保存期間を設定できます。

これらのプロパティは、キューからメッセージをデキューするアプリケーションで最適な値に設定できます。

キューを変更するには:

1. Oracle Enterprise Manager で、Oracle Streams 管理者としてハブ・データベースにログインします。
2. 「データベース」ホームページに移動します。
3. 「データ移動」をクリックして「データ移動」サブページを開きます。
4. 「Streams」セクションの「管理」をクリックします。
5. 「Streams」ページで、「メッセージ機能」をクリックして「メッセージ機能」サブページを開きます。

Select	Queue Name	Queue Table Name	Schema	Type	Enqueue Enabled	Dequeue Enabled	Propagation Errors	Description
<input type="radio"/>	ALERT_QUE	ALERT_QT	SYS	Normal Queue	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Server Generated Alert Queue
<input checked="" type="radio"/>	APP_MESSAGES	APP_MESSAGES_QT	STRMADMIN	Normal Queue	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
<input type="radio"/>	AQ\$ ALERT_QT_E	ALERT_QT	SYS	Exception Queue		<input checked="" type="checkbox"/>		exception queue
<input type="radio"/>	AQ\$ APP_MESSAGES_QT_E	APP_MESSAGES_QT	STRMADMIN	Exception Queue				exception queue
<input type="radio"/>	AQ\$ AQ\$ MEM_MC_E	AQ\$ MEM_MC	SYS	Exception Queue				exception queue

6. 変更するキューを選択します。
7. 「編集」をクリックして「キューの編集」ページを開きます。

Database Instance: database > Streams > Logged in As STRMADMIN

Edit Queue: APP_MESSAGES Show SQL Revert Apply

Schema	STRMADMIN
Queue Table	APP_MESSAGES_QT
Type	Normal Queue
	<input checked="" type="checkbox"/> Enqueue Enabled
	<input checked="" type="checkbox"/> Dequeue Enabled
Maximum Retries	5 <small>Maximum number of retries to dequeue a message in the REMOVE mode after which it is moved to the Exception queue</small>
Retry Delay (seconds)	0 <small>Delay between each retry attempt. 0 means message can be retried as soon as possible</small>
Retention Time (seconds)	0 <small>Time for which a message is retained in the queue table after being dequeued from the queue. 0 means no retention</small>
Description	<input type="text"/>

Show SQL Revert Apply

8. 1つ以上のキュー・プロパティを変更します。このページの詳細は、「ヘルプ」をクリックしてください。
9. 「適用」をクリックすると変更が保存されます。

注意： DBMS_AQADM.ALTER_QUEUE プロシージャを使用して、キューを変更することもできます。

参照：

- 「メッセージ機能の概要」 (9-2 ページ)
- 「チュートリアル: Oracle Database 間でのメッセージの送信」 (9-6 ページ)
- 「チュートリアル: メッセージ通知の構成」 (9-16 ページ)
- 「キュー表の変更」 (9-26 ページ)
- 「メッセージング環境の監視」 (9-30 ページ)

キュー表の変更

Enterprise Manager を使用して、既存のキュー表の記憶域オプションの一部を変更できます。キュー表の記憶域オプションを変更すると、キュー表を使用するキューのパフォーマンスが向上します。

キュー表を変更するには：

1. Oracle Enterprise Manager で、Oracle Streams 管理者としてハブ・データベースにログインします。
2. 「データベース」 ホームページに移動します。
3. 「データ移動」をクリックして「データ移動」サブページを開きます。

4. 「管理」をクリックします。
「Streams」ページが表示され、「概要」サブページが表示されます。

Database Instance: database > Streams

Logged in As STRMADMIN

Overview | Capture | Propagation | Apply | Messaging

Page Refreshed February 5, 2007 9:57:31 AM PST Refresh

View Data Manual Refresh

Capture

Capture Processes 1
Capture Processes Having Errors ✓ 0

Apply

Apply Processes 1
Apply Processes Having Errors ✓ 0

Propagation

Propagation Jobs 1
Propagation Errors ✓ 0

Messaging

Queue Tables 21
Queues 41
Total Propagation Errors ✓ 0

Overview

Oracle Streams enables information sharing. Oracle Streams can share database changes and other information in a stream, which can propagate events within a database or from one database to another. The specified information is routed to specified destinations. The result is a feature that provides greater functionality and flexibility than traditional solutions for capturing and managing information, and sharing the information with other databases and applications.

- A capture process is an Oracle background process that scans the database redo log to capture DML and DDL changes made to database objects. It formats these changes into events called logical change records (LCRs) and enqueues them into a queue.
- Propagations send events from one queue to another, and these queues can be in the same database or in different databases.
- An apply process is an Oracle background process that dequeues events from a queue and applies each event directly to a database object or sends events to apply handlers for custom processing.
- Oracle Streams Messaging, also called as Oracle Streams Advanced Queuing, provides database-integrated message queuing functionality.

5. 「メッセージ機能」セクションで、「キュー表」に関連付けられている数字をクリックして、「キュー表」ページを開きます。

Database Instance: database > Streams > Queue Tables

Logged in As STRMADMIN

Search

Schema

Queue Table

Go

By default, the search returns all uppercase matches beginning with the string you entered. To run an exact or case-sensitive match, double quote the search string. You can use the wildcard symbol (%) in a double quoted string.

Messages Edit Delete Create

Select	Queue Table	Schema	Number Of Queues	Type	Compatible	Description
<input checked="" type="radio"/>	MGMT_NOTIFY_QTABLE	SYSMAN	2	OBJECT	10.0	
<input type="radio"/>	DEF\$ AQCALL	SYSTEM	2	VARIANT	8.0	

6. 変更するキュー表を選択します。
7. 「編集」をクリックします。
「キュー表の編集」ページが表示され、「一般」サブページが表示されます。

8. 「記憶域」をクリックして「記憶域」サブページを開くか、「LOB 記憶域」をクリックして「LOB 記憶域」サブページを開きます。

9. 1つ以上のキュー・プロパティを変更します。サブページの詳細は、「ヘルプ」をクリックしてください。
10. 「適用」をクリックすると変更が保存されます。

注意： DBMS_AQADM.ALTER_QUEUE_TABLE プロシージャを使用して、キュー表を変更することもできます。

参照：

- 「メッセージ機能の概要」(9-2 ページ)
- 「チュートリアル: Oracle Database 間でのメッセージの送信」(9-6 ページ)
- 「チュートリアル: メッセージ通知の構成」(9-16 ページ)
- 「キューの変更」(9-25 ページ)
- 「メッセージング環境の監視」(9-30 ページ)

伝播の変更

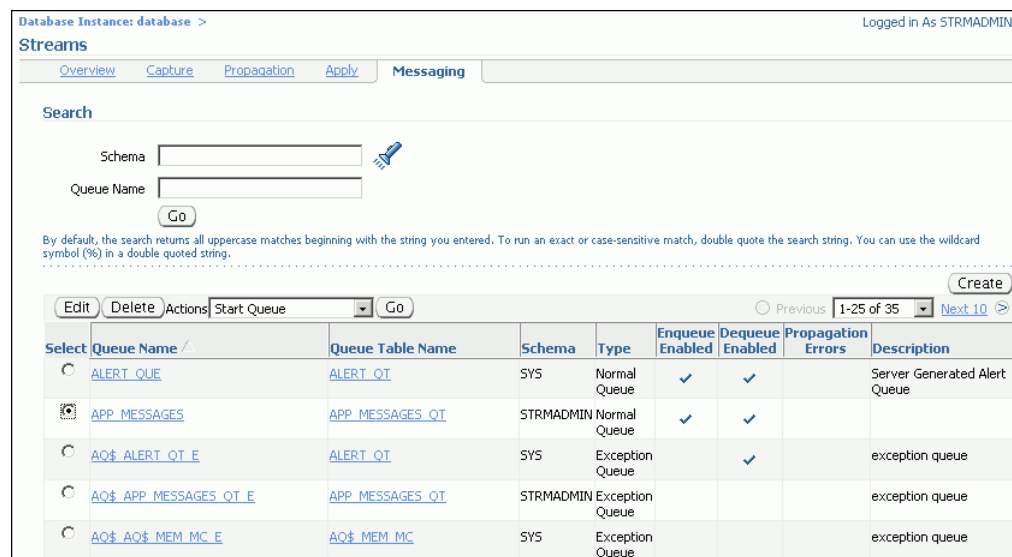
Enterprise Manager を使用して、既存の伝播のスケジュールを変更できます。伝播スケジュールによって、伝播があるキューから別のキューに送信される時期と頻度、および各伝播が終了するまでの時間が決定されます。次のスケジュール・オプションを変更できます。

- **待機時間：**完全に伝播されたキュー内の新しいメッセージが伝播される前に待機する時間
- **伝播の期間：**個々の伝播が終了するまでの時間
- **次回：**個々の伝播間の時間

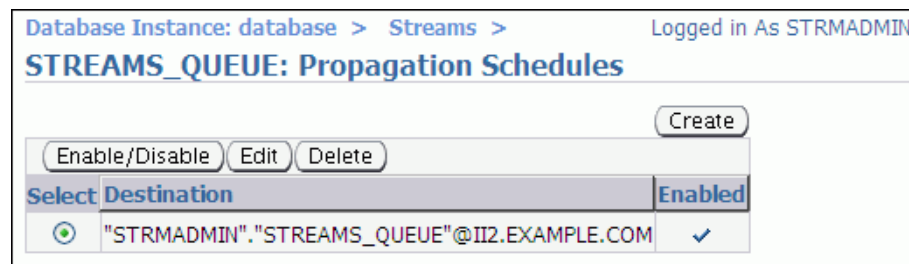
一般的に、伝播スケジュールを変更して、メッセージング環境での伝播のパフォーマンスを向上します。これらのオプションはデフォルト設定では、通常最適な設定になっています。ただし、これらのオプションに対して異なる設定をして、パフォーマンスを最適化できます。

伝播を変更するには：

1. Oracle Enterprise Manager で、Oracle Streams 管理者としてハブ・データベースにログインします。
2. 「データベース」 ホームページに移動します。
3. 「データ移動」 をクリックして「データ移動」 サブページを開きます。
4. 「Streams」 セクションの「管理」 をクリックします。
「Streams」 ページが表示され、「概要」 サブページが表示されます。
5. 「メッセージ機能」 をクリックして「メッセージ機能」 サブページを開きます。



6. 変更する伝播のソース・キューを選択します。ソース・キューは、伝播がメッセージを宛先キューに送信する元のキューです。
7. 「アクション」 リストの「伝播スケジュール」 を選択します。
8. 「実行」 をクリックして「伝播スケジュール」 ページを開きます。



9. 変更する伝播スケジュールを選択します。

10. 「編集」をクリックして「伝播スケジュールの編集」ページを開きます。

11. 1つ以上のスケジュール・オプションを変更します。現在のページの詳細は、「ヘルプ」をクリックしてください。
12. 「適用」をクリックすると変更が保存されます。

注意： DBMS_AQADM.ALTER_PROPAGATION_SCHEDULE プロシージャを使用して、伝播スケジュールを変更することもできます。

参照：

- 「伝播の概要」 (9-4 ページ)
- 「チュートリアル : Oracle Database 間でのメッセージの送信」 (9-6 ページ)

メッセージング環境の監視

この項では、Enterprise Manager を使用してメッセージング環境に関する情報を表示する方法を説明します。この項では、キュー内のメッセージ、キュー統計およびキュー・コンシューマを表示する方法も示します。

次の各項で、メッセージング環境の監視について説明します。

- キュー内のメッセージの表示
- 「永続キューの統計」の表示
- メッセージをデキューできるコンシューマの表示

参照：

- 「メッセージ機能の概要」 (9-2 ページ)
- 「メッセージ機能の準備」 (9-5 ページ)
- 「チュートリアル: Oracle Database 間でのメッセージの送信」 (9-6 ページ)
- 「チュートリアル: メッセージ通知の構成」 (9-16 ページ)
- 「メッセージング環境のトラブルシューティング」 (9-34 ページ)

キュー内のメッセージの表示

キュー内のメッセージを表示して、アプリケーションがメッセージを正しくエンキューしていることを確認したり、どのメッセージが現在デキューまたは別のキューに伝播される準備ができているかを参照したりできます。

キューに現在格納されているメッセージを表示するには：

1. Oracle Enterprise Manager で、Oracle Streams 管理者としてハブ・データベースにログインします。
2. 「データベース」 ホームページに移動します。
3. 「データ移動」 をクリックして「データ移動」 サブページを開きます。
4. 「Streams」 セクションの「管理」 をクリックします。
「Streams」 ページが表示され、「概要」 サブページが表示されます。
5. 「メッセージ機能」 をクリックして「メッセージ機能」 サブページを開きます。
6. メッセージを含むキューを選択します。
7. 「アクション」 リストの「メッセージ」 を選択します。
8. 「実行」 をクリックします。
9. 「メッセージ」 ページで、「実行」 をクリックします。メッセージがリストされます。

Database Instance: database > Streams > Logged in As STRMADMIN
Messages: STREAMS_QUEUE

Search

State: All
Priority: is
Go

Enqueue Time: All
Dequeue Time: All
Consumer:
Specify the consumer in the format name:address

ID	Priority	Consumers In Different States	Enqueue Time	Enqueue User ID	Enqueue Transaction ID	Delay	Expiration (in seconds)	Exception Queue
28722EE8DB64D12DE040578CB10B335E	1	Ready	February 1, 2007 1:48:10 PM PST	STRMADMIN	1			Default
28722EE8DB65D12DE040578CB10B335E	1	Ready	February 1, 2007 1:48:10 PM PST	STRMADMIN	1			Default
28722EE8DB66D12DE040578CB10B335E	1	Ready	February 1, 2007 1:48:10 PM PST	STRMADMIN	1			Default

10. 「状態の異なるコンシューマ」 フィールドの番号リンクをクリックして、メッセージのコンシューマを表示します。コンシューマは、メッセージを別のキューに送信する伝播またはメッセージをデキューするアプリケーションです。

ページの詳細は、ページの「ヘルプ」 をクリックしてください。

参照：

- 「キューの変更」 (9-25 ページ)

「永続キューの統計」の表示

永続メッセージング・モードを使用している場合、メッセージはハード・ディスクのキュー表に格納されます。この項では、永続モードで永続キューにエンキューされたメッセージの統計の表示について説明します。

永続キュー統計を表示するには：

1. Oracle Enterprise Manager で、Oracle Streams 管理者としてハブ・データベースにログインします。
2. 「データベース」 ホームページに移動します。
3. 「データ移動」 をクリックして「データ移動」 サブページを開きます。
4. 「Streams」 セクションの「管理」 をクリックします。
「Streams」 ページが表示され、「概要」 サブページが表示されます。
5. 「メッセージ機能」 をクリックして「メッセージ機能」 サブページを開きます。
6. メッセージを含むキューを選択します。
7. 「アクション」 リストの「キュー統計」 を選択します。
8. 「実行」 をクリックします。
9. 「キュー統計」 ページで、「メッセージの統計」 をクリックしてキューの永続キュー統計を表示します。

Database Instance: database > Streams > Queue Statistics: STREAMS_QUEUE

Page Refreshed October 24, 2006 10:08:41 AM PDT Refresh

View Data Manual Refresh

Propagation Statistics Message Statistics Buffered Queue Statistics

The following list gives the number of messages in different states in the queue.

Messages in state WAIT	0
Messages in state READY	3
Messages in state EXPIRED	0
Total Wait Time for READY messages in the Queue (seconds)	1449
Average Wait Time for READY messages in the Queue (seconds)	483

Propagation Statistics Message Statistics Buffered Queue Statistics

メッセージ統計は、永続キュー内の各状態のメッセージ数を示します。これらは、メッセージがデキューされる合計待機時間と平均待機時間も示します。

注意：

- メッセージは、バッファ・モードでエンキューすることもできます。この場合、メッセージはメモリー内のバッファ・キューに格納されます。これらのメッセージの統計の表示の詳細は、5-23 ページの「[「バッファ・キューの統計」の表示](#)」を参照してください。
- キュー統計を、V\$AQ 動的パフォーマンス・ビューに問い合わせることもできます。

参照：

- [「キューの変更」](#) (9-25 ページ)

メッセージをデキューできるコンシューマの表示

コンシューマは、特定のキューからメッセージをデキューするように構成されます。メッセージング環境では、コンシューマはキューへのサブスクライバによって表現され、複数のコンシューマが単一のサブスクライバを使用できます。

コンシューマは次の処理を行うことができます。

- メッセージをデキューして処理するアプリケーションまたはユーザー
- あるキューから別のキューにメッセージを送信する伝播
- メッセージング環境のカスタム・プロセスのメッセージのデキュー、またはレプリケーション環境のデータベース・オブジェクトに対する変更のデキューを行えるプロセスの適用

特定のキューからメッセージをデキューできるコンシューマを表示するには：

1. Oracle Enterprise Manager で、Oracle Streams 管理者としてハブ・データベースにログインします。
2. 「データベース」 ホームページに移動します。
3. 「データ移動」 をクリックして「データ移動」 サブページを開きます。
4. 「Streams」 セクションの「管理」 をクリックします。
「Streams」 ページが表示され、「概要」 サブページが表示されます。
5. 「メッセージ機能」 をクリックして「メッセージ機能」 サブページを開きます。
6. キューを選択します。
7. 「アクション」 リストの「サブスクライバ」 を選択します。
8. 「実行」 をクリックして「サブスクライバ」 ページを開きます。

Database Instance: database > Streams > Logged in As STRMADMIN
 STREAMS_QUEUE: Subscribers

Search

Name
 Address
 Database Link

By default, the search returns all uppercase matches beginning with the string you entered. To run an exact or case-sensitive match, double quote the search string. You can use the wildcard symbol (%) in a double quoted string.

Select	Name	Address	Database Link	Rule	Transformation
<input checked="" type="radio"/>	APPLY_SIMP				
<input type="radio"/>		"STRMADMIN", "STREAMS_QUEUE"	II1.EXAMPLE.COM		

「サブスクライバ」 ページには、キューに対する各サブスクライバに関する次の情報が含まれます。

- 「名前」 フィールドには、サブスクライバの名前が含まれます。
- 「アドレス」 フィールドは、通常はサブスクライバが別のキューにメッセージを送信する場合に移入されます。「アドレス」 フィールドは、メッセージを受信するキューの名前を示します。
- 「データベース・リンク」 フィールドは、サブスクライバがリモート・データベースにいる場合にリモート・データベースへのデータベース・リンクを示します。

- 「**ルール**」フィールドは、サブスクライバによって使用されるルールを示します。ルールによって、サブスクライバがデキューするメッセージを決定できます。一部のサブスクライバはルールを使用しません。
- 「**変換**」フィールドは、サブスクライバによって使用される変換を示します。変換は、デキュー中にメッセージを変更します。一部のサブスクライバは変換を使用しません。

このページの詳細を表示するには、「ヘルプ」をクリックします。

注意： ALL_QUEUE_SUBSCRIBERS データ・ディクショナリを問い合わせ、メッセージをデキューできるコンシューマを表示することもできます。

参照：

- 「[メッセージ機能の概要](#)」 (9-2 ページ)
- 「[変更の適用の概要](#)」 (4-6 ページ)

メッセージング環境のトラブルシューティング

この項では、メッセージング環境で最も一般的な問題について説明します。これらの問題の修正方法も説明します。

次の各項で、メッセージング環境のトラブルシューティングについて説明します。

- [メッセージのエンキューまたはデキュー中の ORA-01031 エラーの修正](#)
- [メッセージのエンキュー中の ORA-24033 エラーの修正](#)
- [伝播の ORA-02019 エラーの修正](#)
- [デキューしたメッセージがキュー内に残る理由の理解](#)

参照：

- 『Oracle Streams アドバンスド・キューイング・ユーザーズ・ガイド』
- 「[メッセージ機能の概要](#)」 (9-2 ページ)

メッセージのエンキューまたはデキュー中の ORA-01031 エラーの修正

必要な権限のないユーザーがメッセージをエンキューまたはデキューしようとする場合、Oracle Database から次のエラーが返されます。

ORA-01031: insufficient privileges

Oracle Database がこのメッセージを返す場合、エンキューまたはデキュー操作は失敗します。問題を修正するには、キューでの ENQUEUE または DEQUEUE 権限をユーザーに付与します。

ユーザーに ENQUEUE または DEQUEUE 権限を付与するには：

1. コマンドラインで SQL*Plus を開き、Oracle Streams 管理者や SYSTEM などの管理ユーザーとしてデータベースに接続します。

SQL*Plus の起動の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。

2. DBMS_AQADM パッケージで GRANT_QUEUE_PRIVILEGE プロシージャを実行し、必要なキュー権限をユーザーに付与します。

たとえば、strmadmin.streams_queue キューで、ENQUEUE 権限を hr ユーザーに付与し、次のプロシージャを実行します。

```
BEGIN
  DBMS_AQADM.GRANT_QUEUE_PRIVILEGE (
    privilege => 'ENQUEUE',
    queue_name => 'strmadmin.streams_queue',
    grantee    => 'hr');
END;
/
```

strmadmin.streams_queue キューで、DEQUEUE 権限を hr ユーザーに付与するには、次のプロシージャを実行します。

```
BEGIN
  DBMS_AQADM.GRANT_QUEUE_PRIVILEGE (
    privilege => 'DEQUEUE',
    queue_name => 'strmadmin.streams_queue',
    grantee    => 'hr');
END;
/
```

参照：

- 『Oracle Streams アドバンスド・キューイング・ユーザーズ・ガイド』
- 「[メッセージ機能の概要](#)」 (9-2 ページ)

メッセージのエンキュー中の ORA-24033 エラーの修正

ユーザーまたはアプリケーションがメッセージをエンキューしようとするとき、メッセージのコンシューマがない場合、Oracle Database から次のエラーが返されます。

ORA-24033: no recipients for message

Oracle Database がこのメッセージを返す場合、メッセージはエンキューされません。

問題を解決するには：

1. メッセージに対して少なくとも 1 つのコンシューマを構成します。
2. メッセージをエンキューします。

コンシューマは次の処理を行うことができます。

- メッセージをデキューして処理するアプリケーションまたはユーザー
- あるキューから別のキューにメッセージを送信する伝播
- メッセージング環境のカスタム・プロセスのメッセージのデキュー、またはレプリケーション環境のデータベース・オブジェクトに対する変更のデキューを行えるプロセスの適用
- メッセージをデキューし、アプリケーションに渡すメッセージ・クライアント

参照：

- 『Oracle Streams アドバンスド・キューイング・ユーザーズ・ガイド』
- 「[メッセージ機能の概要](#)」 (9-2 ページ)

伝播の ORA-02019 エラーの修正

メッセージの送信時、伝播ではソース・キューを所有しているデータベース・ユーザーの権限が使用されます。ソース・キューの所有者が、伝播で使用するデータベース・リンクにアクセスできない場合、Oracle Database から次のエラーが返されます。

ORA-02019: connection description for remote database not found

Oracle Database がこのメッセージを返す場合、メッセージは伝播されません。この問題を修正する最も簡単な方法は、データベース・リンクを削除し、再作成して、ソース・キューの所有者がデータベースにアクセスできるようにします。

データベース・リンクを削除し、再作成するには：

1. Oracle Enterprise Manager で、SYSTEM または Oracle Streams 管理者などの管理者ユーザーとしてデータベースにログインします。
2. 「データベース」ホームページに移動します。
3. 「スキーマ」をクリックして「スキーマ」サブページを開きます。
4. 「データベース・オブジェクト」セクションの「データベース・リンク」をクリックします。
「データベース・リンク」ページが表示されます。
5. 検索ツールを使用して、削除および再作成するデータベース・リンクを表示します。
6. データベース・リンクを選択します。
7. 「削除」をクリックします。
8. 確認ページで「はい」をクリックしてデータベース・リンクを削除します。
9. 2-8 ページの「チュートリアル: データベース・リンクの作成」の手順に従って、このデータベース・リンクを再作成します。

伝播のソース・キューの所有者にデータベース・リンクへのアクセス権があることを確認するには、データベース・リンクを作成したときに、「接続モード」セクションで固定ユーザーとしてこのユーザーを指定します。

参照：

- 「伝播の概要」(9-4 ページ)
- データベース・リンクの詳細は、『Oracle Database 管理者ガイド』を参照してください。

デキューしたメッセージがキュー内に残る理由の理解

メッセージがデキューされた後、次の理由によりメッセージはキュー内に残ります。

- メッセージの 1 つ以上のコンシューマがメッセージをデキューしていません。すべてのコンシューマがメッセージをデキューするまで、キューからメッセージは削除されません。
- メッセージはバックグラウンド・プロセスでクリーン・アップされていません。すべてのメッセージのコンシューマがメッセージをデキューした後、バックグラウンド・プロセスでメッセージが自動的に削除されるまで、キュー内にメッセージが残ります。

キュー内でコンシューマがデキューしたメッセージが検出された場合、Enterprise Manager を使用して、メッセージのコンシューマを確認し、すべてのメッセージのコンシューマがメッセージをデキューしたかどうかを判断します。

キュー内でメッセージを確認するには：

1. 9-33 ページの「[メッセージをデキューできるコンシューマの表示](#)」の手順に従い、メッセージをデキューできるコンシューマを表示します。メッセージに対して、複数のコンシューマがある場合、一部のコンシューマはまだメッセージをデキューできない場合があります。
2. 9-31 ページの「[キュー内のメッセージの表示](#)」の手順に従います。「**状態の異なるコンシューマ**」フィールドで、メッセージに対して PROCESSED が表示された場合、すべてのメッセージのコンシューマはメッセージをデキューします。この場合、バックグラウンド・プロセスでメッセージが自動的に削除されるまで、メッセージはキュー内に残ります。

参照：

- 『Oracle Streams アドバンスド・キューイング・ユーザーズ・ガイド』
- 「[メッセージ機能の概要](#)」 (9-2 ページ)

データの比較および収束

この章では、2つの異なるデータベースにある共有データベース・オブジェクト内のデータを比較する方法を説明します。共有データベース・オブジェクト内の相違データを収束する方法も説明します。

この章は次の項で構成されています。

- [異なるデータベース内のデータの比較および収束の概要](#)
- [チュートリアル: データの比較および収束の準備](#)
- [チュートリアル: 2つの異なるデータベース内のデータの比較](#)
- [チュートリアル: 相違データの収束](#)

参照:

- [第7章「マテリアライズド・ビューを使用したデータのレプリケート」](#)
- [第4章「Oracle Streams を使用したデータのレプリケート」](#)
- 『Oracle Database PL/SQL パッケージ・プロシージャおよびタイプ・リファレンス』
- 『Oracle Streams レプリケーション管理者ガイド』

異なるデータベース内のデータの比較および収束の概要

2つ以上のデータベースでデータベース・オブジェクトを共有できます。同じデータベース・オブジェクトのコピーが複数のデータベースに存在している場合、データベース・オブジェクトは**共有データベース・オブジェクト**です。共有データベース・オブジェクトは、データ・レプリケーションによってメンテナンスできます。たとえば、マテリアライズド・ビューまたは Oracle Streams コンポーネントは、データベース・オブジェクトをレプリケートし、複数のデータベースでそれらをメンテナンスできます。カスタム・アプリケーションも、共有データベース・オブジェクトをメンテナンスできます。通常、レプリケーション環境では、データを含むオブジェクト（表など）と、その他のタイプのデータベース・オブジェクト（索引など）を共有します。

1つのデータベースで共有データベース・オブジェクトに変更が加えられた場合、変更が転送され、データベース・オブジェクトを共有する他の各データベースに加えられます。こうして、レプリケーション環境は各データベースの共有データベース・オブジェクトを同期します。

データを含む共有データベース・オブジェクトは、異なるデータベースで一貫性がなくなることがあります。つまり、データは共有データベース・オブジェクトの異なる・インスタンスで拡散します。たとえば、データベース・オブジェクトが表の場合は、表の1つのインスタンスが表の別のインスタンスよりも多くの行を持つこと、または表の2つのインスタンスが同じ行に異なるデータを持つことがあります。

Oracle Streams レプリケーション環境で共有データベース・オブジェクトが拡散する場合、通常は次のいずれかの理由です。

- データ変更が1つ以上のデータベースで取得されていません。
- データ変更は取得されていますが、1つのデータベースから別のデータベースに転送されていません。
- データ変更が取得され、1つのデータベースから別のデータベースに転送されましたが、他のデータベースの共有データベース・オブジェクトに対してその変更が加えられていません。

データ拡散の一般的な原因は、ネットワークの問題、不正な構成またはユーザー・エラーです。共有データベース・オブジェクトが、マテリアライズド・ビューを使用するレプリケーション環境で拡散する場合は、マテリアライズド・ビューのリフレッシュに問題がある可能性があります。

DBMS_COMPARISON パッケージを使用して、異なるデータベースのデータベース・オブジェクトを比較し、差異を識別できます。このパッケージでは、異なるデータベースで一貫するようにデータベース・オブジェクトを収束させることもできます。DBMS_COMPARISON パッケージは、Oracle Database とともに必ずインストールされる Oracle 提供の PL/SQL パッケージです。

DBMS_COMPARISON パッケージは、次のタイプのデータベース・オブジェクトの比較および収束を行うことができます。

- 表
- 単一表ビュー
- マテリアライズド・ビュー
- 表、単一表ビューおよびマテリアライズド・ビューのシノニム

異なるデータベースにある異なるタイプのデータベース・オブジェクトの比較および収束を行うことができます。たとえば、あるデータベースにある表と、別のデータベースにあるマテリアライズド・ビューの比較および収束を行うことができます。

このマニュアルの例では、共有データベース・オブジェクトが2つのデータベースで同じ名前を持ち、データベース・オブジェクト全体が比較および収束されています。ただし、DBMS_COMPARISON パッケージは、異なるデータベースにある共有データベース・オブジェクトの差異について柔軟性を提供します。比較されるデータベース・オブジェクトは、必ずしも同じ名前を持つ必要はありません。また、対応する列が同じデータ型であるかぎり、データベース・オブジェクト内で列名が異なってもかまいません。共有データベース・オブジェクト全体、または列および行のサブセットの比較および収束を行うことができます。

比較を作成するには、DBMS_COMPARISON パッケージの CREATE_COMPARISON プロシージャを使用します。このプロシージャは、共有データベース・オブジェクトの 1 つ以上の索引列を識別します。DBMS_COMPARISON パッケージは、索引列として使用できる少なくとも 1 つの列を識別する必要があります。指定されたデータベース・オブジェクトに、索引列として使用できる列がない場合、CREATE_COMPARISON プロシージャはデータベース・オブジェクトの比較を作成できません。

注意： 環境に、拡散した共有データベース・オブジェクトがある場合は、問題の原因を調査して解決する必要があります。DBMS_COMPARISON パッケージは共有データベース・オブジェクトの比較および収束を行うことができますが、これらのデータベース・オブジェクトは拡散しないことをお勧めします。

参照：

- DBMS_COMPARISON パッケージの詳細は、『Oracle Database PL/SQL パッケージ・プロシージャおよびタイプ・リファレンス』を参照してください。
- DBMS_COMPARISON パッケージの高度な機能の使用の詳細は、『Oracle Streams レプリケーション管理者ガイド』を参照してください。

チュートリアル：データの比較および収束の準備

2 つのデータベースで hr.departments 表を共有するとします。この表を異なるデータベースで比較して、データが一貫しているかどうかを確認します。2 つのデータベースで表が拡散している場合は、これらを収束させて一貫させます。

このチュートリアルを実行する前に、次の前提条件を満たします。

- ネットワーク接続を構成して、2 つのデータベースが相互に通信できるようにします。データベース間のネットワーク接続の構成の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。
- hr サンプル・スキーマが両方のデータベースにインストールされていることを確認します。hr サンプル・スキーマは、Oracle Database とともにデフォルトでインストールされます。

この例では、データベースのグローバル名は ii1.example.com および ii2.example.com ですが、使用している環境で、前提条件を満たす任意の 2 つのデータベースをかわりに使用できます。

ii1.example.com および ii2.example.com データベースの hr.departments 表の比較および収束を準備するには：

1. この例の目的を満たすために、2 つのデータベースで hr.departments 表を拡散させます。
 - a. コマンドラインで SQL*Plus を開き、ii2.example.com データベースに hr ユーザーとして接続してください。

SQL*Plus の起動の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。
 - b. hr.departments 表で、department_id が 270 と等しい部門を削除します。

```
DELETE FROM hr.departments WHERE department_id=270;
COMMIT;
```

- c. hr.departments 表で、ある行のデータを変更します。

```
UPDATE hr.departments SET manager_id=114 WHERE department_id=10;
COMMIT;
```

- d. hr.departments 表に行を挿入します。

```
INSERT INTO hr.departments VALUES(280, 'Bean Counters', 108, 2700);
COMMIT;
```

- e. SQL*Plus を終了します。

```
EXIT;
```

注意: 通常、手順1は不要です。この例では、2つのデータベースで hr.departments 表を拡散させるために含まれています。

2. ii1.example.com データベースから ii2.example.com データベースへのデータベース・リンクを作成します。

データベース・リンクは、ii1.example.com の管理ユーザーのスキーマから ii2.example.com の管理ユーザーのスキーマに接続する必要があります。両方のデータベースの管理ユーザーは、hr.departments 表にアクセスして変更するために必要な権限と、DBMS_COMPARISON パッケージのサブプログラムを実行するために必要な権限を持っている必要があります。どのユーザーがこれらの権限を持っているかが不明な場合は、SYSTEM ユーザーを使用します。また、データベース・リンクの名前とサービス名は、どちらも ii2.example.com である必要があります。方法については、2-8 ページの「チュートリアル: データベース・リンクの作成」を参照してください。

チュートリアル: 2つの異なるデータベース内のデータの比較

この例は、10-3 ページの「チュートリアル: データの比較および収束の準備」で説明したシナリオの続きです。先に進む前に、その項の手順を完了してください。

DBMS_COMPARISON パッケージの CREATE_COMPARISON プロシージャを使用して、2つの異なるデータベースにある共有データベース・オブジェクトの比較を定義できます。比較が定義された後で、このパッケージの COMPARE ファンクションを使用して、比較で指定したデータベース・オブジェクトを現在の時点で比較できます。COMPARE ファンクションは、特定の比較に対して複数回実行できます。ファンクションを実行するたびに、データベース・オブジェクトのスキャンが1回以上行われ、各スキャンが独自のスキャン ID を持ちます。

ii1.example.com および ii2.example.com データベースの hr.departments 表全体を比較するには:

1. コマンドラインで SQL*Plus を開き、10-3 ページの「チュートリアル: データの比較および収束の準備」で作成したデータベース・リンクを所有する管理ユーザーとして ii1.example.com データベースに接続します。たとえば、SYSTEM ユーザーがデータベース・リンクを所有する場合は、SYSTEM ユーザーとして接続します。

```
sqlplus system@ii1.example.com
Enter password: password
```

SQL*Plus の起動の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。

2. CREATE_COMPARISON プロシージャを実行して、hr.departments 表の比較を作成します。

```
BEGIN
  DBMS_COMPARISON.CREATE_COMPARISON(
    comparison_name => 'compare_departments',
    schema_name     => 'hr',
    object_name     => 'departments',
    dblink_name     => 'ii2.example.com');
END;
/
```

新しい比較の名前は compare_departments であることに注意してください。この比較は、CREATE_COMPARISON プロシージャを実行するユーザーが所有します。

3. COMPARE ファンクションを実行して、2つのデータベースの hr.departments 表を比較します。

```
SET SERVEROUTPUT ON
DECLARE
  consistent  BOOLEAN;
  scan_info  DBMS_COMPARISON.COMPARISON_TYPE;
BEGIN
  consistent := DBMS_COMPARISON.COMPARE(
    comparison_name => 'compare_departments',
    scan_info       => scan_info,
    perform_row_dif => TRUE);
  DBMS_OUTPUT.PUT_LINE('Scan ID: ' || scan_info.scan_id);
  IF consistent=TRUE THEN
    DBMS_OUTPUT.PUT_LINE('No differences were found.');
```

```
ELSE
  DBMS_OUTPUT.PUT_LINE('Differences were found.');
```

```
END IF;
END;
/
```

```
Scan ID: 1
Differences were found.
```

```
PL/SQL procedure successfully completed.
```

手順2で作成した比較の名前を comparison_name パラメータに指定します。

ファンクションは、比較のスキャン ID を出力します。比較に関する情報をデータ・ディクショナリ・ビューに問い合わせる場合、およびデータベース・オブジェクトを収束させる場合は、スキャン ID が重要です。

ファンクションは、2つのデータベースで表に差異が見つかったかどうかでも出力します。

- ファンクションが「違いが見つかりません」と出力する場合、表は2つのデータベースで一貫しています。
 - ファンクションが「違いが見つかりました」と出力する場合は、表が2つのデータベースで拡散しています。
4. 前の手順でファンクションから返されたスキャン ID を記録しておきます。この例では、スキャン ID が 1 であると想定します。

5. 手順3で差異が見つかった場合は、次の問合せを実行して、見つかった差異の数を表示します。

```

COLUMN OWNER HEADING 'Comparison Owner' FORMAT A16
COLUMN COMPARISON_NAME HEADING 'Comparison Name' FORMAT A20
COLUMN SCHEMA_NAME HEADING 'Schema Name' FORMAT A11
COLUMN OBJECT_NAME HEADING 'Object Name' FORMAT A11
COLUMN CURRENT_DIF_COUNT HEADING 'Differences' FORMAT 9999999

SELECT c.OWNER,
       c.COMPARISON_NAME,
       c.SCHEMA_NAME,
       c.OBJECT_NAME,
       s.CURRENT_DIF_COUNT
FROM DBA_COMPARISON c, DBA_COMPARISON_SCAN_SUMMARY s
WHERE c.COMPARISON_NAME = s.COMPARISON_NAME AND
      c.OWNER            = s.OWNER AND
      s.SCAN_ID          = 1;

```

問合せの WHERE 句に、手順4で記録したスキャン ID を指定します。

出力は、次のようになります。

Comparison Owner	Comparison Name	Schema Name	Object Name	Differences
SYSTEM	COMPARE_DEPARTMENTS	HR	DEPARTMENTS	3

6. 比較されているデータベース・オブジェクトのどの行が異なるかを確認するには、次の問合せを実行します。

```

COLUMN COLUMN_NAME HEADING 'Index Column' FORMAT A15
COLUMN INDEX_VALUE HEADING 'Index Value' FORMAT A15
COLUMN LOCAL_ROWID HEADING 'Local Row Exists?' FORMAT A20
COLUMN REMOTE_ROWID HEADING 'Remote Row Exists?' FORMAT A20

SELECT c.COLUMN_NAME,
       r.INDEX_VALUE,
       DECODE(r.LOCAL_ROWID,
              NULL, 'No',
              'Yes') LOCAL_ROWID,
       DECODE(r.REMOTE_ROWID,
              NULL, 'No',
              'Yes') REMOTE_ROWID
FROM DBA_COMPARISON_COLUMNS c,
     DBA_COMPARISON_ROW_DIF r,
     DBA_COMPARISON_SCAN s
WHERE c.COMPARISON_NAME = 'COMPARE_DEPARTMENTS' AND
      r.SCAN_ID          = s.SCAN_ID AND
      s.PARENT_SCAN_ID  = 1 AND
      r.STATUS           = 'DIF' AND
      c.INDEX_COLUMN     = 'Y' AND
      c.COMPARISON_NAME = r.COMPARISON_NAME AND
      c.OWNER            = r.OWNER

ORDER BY r.INDEX_VALUE;

```

WHERE 句には、比較の名前と比較のスキャン ID を指定します。この例では、比較の名前は compare_departments で、スキャン ID は 1 です。

出力は、次のようになります。

Index Column	Index Value	Local Row Exists?	Remote Row Exists?
DEPARTMENT_ID	10	Yes	Yes
DEPARTMENT_ID	270	Yes	No
DEPARTMENT_ID	280	No	Yes

この出力は、比較される表の索引列と、共有データベース・オブジェクト内で異なっている各行の索引値を示します。この例では、索引列は hr.departments 表の主キー列 (department_id) です。出力は、各行の差異のタイプも示します。

- Local Row Exists? と Remote Row Exists? の両方が、ある行に対して Yes の場合、その行はデータベース・オブジェクトの両方のインスタンスに存在しますが、行のデータは異なります。
- ある行について Local Row Exists? が Yes で、Remote Row Exists? が No の場合、その行はローカル・データベース・オブジェクトに存在しますが、リモート・データベース・オブジェクトには存在しません。
- ある行について Local Row Exists? が No で、Remote Row Exists? が Yes の場合、その行はリモート・データベース・オブジェクトに存在しますが、ローカル・データベース・オブジェクトには存在しません。

チュートリアル: 相違データの収束

この例は、10-4 ページの「チュートリアル: 2つの異なるデータベース内のデータの比較」で説明したシナリオの続きです。先に進む前に、その項の手順を完了してください。

共有データベース・オブジェクトが2つの異なるデータベースで拡散した場合は、DBMS_COMPARISON パッケージの CONVERGE プロシージャを使用して、データベース・オブジェクトの2つのインスタンスを収束させることができます。CONVERGE プロシージャを正常に実行すると、共有データベース・オブジェクトが2つのデータベースで一貫します。CONVERGE プロシージャを実行するには、次の情報を指定する必要があります。

- DBMS_COMPARISON パッケージの CREATE_COMPARISON プロシージャを使用して作成した既存の比較の名前
- 収束させる比較のスキャン ID

スキャン ID には、収束させる差異に関する情報が含まれます。この例では、比較の名前は compare_departments で、スキャン ID は 1 です。

また、CONVERGE プロシージャを実行する際には、共有データベース・オブジェクトが収束するときに「優先」されるデータベースを指定する必要があります。ローカル・データベースが優先されることを指定した場合は、データが異なるときに、ローカル・データベースのデータベース・オブジェクト内のデータでリモート・データベースのデータベース・オブジェクト内のデータが置換されます。リモート・データベースが優先されることを指定した場合は、データが異なるときに、リモート・データベースのデータベース・オブジェクト内のデータでローカル・データベースのデータベース・オブジェクト内のデータが置換されます。この例では、ローカル・データベース ii1.example.com が優先されます。

ii1.example.com および ii2.example.com データベースの hr.departments 表の相違データを収束させるには:

1. コマンドラインで SQL*Plus を開き、10-3 ページの「チュートリアル: データの比較および収束の準備」で作成したデータベース・リンクを所有する管理ユーザーとして ii1.example.com データベースに接続します。たとえば、SYSTEM ユーザーがデータベース・リンクを所有する場合は、SYSTEM ユーザーとして接続します。

```
sqlplus system@ii1.example.com
Enter password: password
```

SQL*Plus の起動の詳細は、『Oracle Database 2 日でデータベース管理者』を参照してください。

2. CONVERGE プロシージャを実行して、2つのデータベースの hr.departments 表を収束させます。

```

SET SERVEROUTPUT ON
DECLARE
    scan_info    DBMS_COMPARISON.COMPARISON_TYPE;
BEGIN
    DBMS_COMPARISON.CONVERGE(
        comparison_name => 'compare_departments',
        scan_id          => 1,
        scan_info        => scan_info,
        converge_options => DBMS_COMPARISON.CMP_CONVERGE_LOCAL_WINS);
    DBMS_OUTPUT.PUT_LINE('Local Rows Merged: ' || scan_info.loc_rows_merged);
    DBMS_OUTPUT.PUT_LINE('Remote Rows Merged: ' || scan_info.rmt_rows_merged);
    DBMS_OUTPUT.PUT_LINE('Local Rows Deleted: ' || scan_info.loc_rows_deleted);
    DBMS_OUTPUT.PUT_LINE('Remote Rows Deleted: ' || scan_info.rmt_rows_deleted);
END;
/

```

```

Local Rows Merged: 0
Remote Rows Merged: 2
Local Rows Deleted: 0
Remote Rows Deleted: 1

```

PL/SQL procedure successfully completed.

CONVERGE プロシージャは、指定されたスキャンで比較されるデータベース・オブジェクトの部分を同期し、行われた変更に関する情報を返します。一部のスキャンは、データベース・オブジェクトのサブセットを比較することがあります。この例では、指定されたスキャンは表全体を比較しました。このため、比較スキャンの完了後に新しい差異が追加されていないと想定すると、表全体が同期されています。

この例では、プロシージャで `converge_options` パラメータが `DBMS_COMPARISON.CMP_CONVERGE_LOCAL_WINS` に設定されているためローカル表が優先されます。つまり、2つのデータベースで異なる行については、ローカル・データベースの行によって、リモート・データベースの対応する行が置換されます。一部の行がリモート・データベースに存在し、ローカル・データベースには存在しない場合は、リモート・データベースの余分な行が削除されます。リモート・データベースを優先する場合は、プロシージャで `converge_options` パラメータを `DBMS_COMPARISON.CMP_CONVERGE_REMOTE_WINS` に設定します。

また、Oracle Streams レプリケーション環境の一部である共有データベース・オブジェクトに対して CONVERGE プロシージャを実行する場合は、プロシージャによって行われた変更を他のデータベースにレプリケートしたくないことがあります。この場合は、CONVERGE プロシージャで、次のパラメータを変更がレプリケートされない値に設定できます。

- `local_converge_tag`
- `remote_converge_tag`

これらのパラメータのいずれかが NULL 以外の値に設定されている場合、収束中に変更を行うセッションでタグが設定されます。 `local_converge_tag` パラメータはセッション中にローカル・データベースにタグを設定しますが、 `remote_converge_tag` パラメータはセッション中にリモート・データベースにタグを設定します。 CONVERGE プロシージャによって行われた変更をレプリケートしない場合は、これらのパラメータを、Oracle Streams 取得プロセスおよび同期取得が変更を取得しないようにする値に設定します。

参照:

- 「変更の循環を回避するためのタグの概要」 (4-10 ページ)

数字

- 2 データベース・レプリケーション, 4-11
 - 構成
 - ダウンストリーム取得プロセス, 4-29
 - 同期取得, 4-45
 - ローカル取得プロセス, 4-24
- 2 フェーズ・コミット
 - 定義, 3-2

A

- ADD_MESSAGE_RULE プロシージャ, 9-13, 9-20
- ADD_TABLE_PROPAGATION_RULES プロシージャ, 4-49
- ADD_TABLE_RULES プロシージャ, 4-47, 4-50
- ADD_UPDATE_RESOLUTION プロシージャ, 7-21
- ALL_APPLY_CONFLICT_COLUMNS ビュー, 4-20
- ALL_APPLY_ERROR ビュー, 4-20
- ALL_APPLY ビュー, 4-20
- ALL_CAPTURE ビュー, 4-20
- ALL_PROPAGATION ビュー, 4-20
- ALL_STREAMS_COLUMNS ビュー, 4-20
- ALL_STREAMS_RULES ビュー, 4-20
- ALL_STREAMS_UNSUPPORTED ビュー, 4-20
- ALL_SYNC_CAPTURE ビュー, 4-20
- ALTER_APPLY プロシージャ, 4-47
- ANYDATA キュー, 2-7

C

- COMPARE ファンクション, 10-4
- COMPATIBLE パラメータ, 4-21
- CONVERGE プロシージャ, 10-7
- CREATE_APPLY プロシージャ, 4-47
- CREATE_COMPARISON プロシージャ, 10-4
- CREATE_RULE_SET プロシージャ, 4-47

D

- DBA_APPLY_CONFLICT_COLUMNS ビュー, 4-20, 5-23
- DBA_APPLY_ERROR ビュー, 4-20
- DBA_APPLY ビュー, 4-20
- DBA_CAPTURE ビュー, 4-20
- DBA_COMPARISON_COLUMNS ビュー, 10-6
- DBA_COMPARISON_ROW_DIF ビュー, 10-6
- DBA_COMPARISON_SCAN ビュー, 10-6

- DBA_COMPARISON ビュー, 10-6
- DBA_PROPAGATION ビュー, 4-20
- DBA_REGISTERED_MVIEWS ビュー, 8-21
- DBA_STREAMS_COLUMNS ビュー, 4-20
- DBA_STREAMS_RULES ビュー, 4-20
- DBA_STREAMS_UNSUPPORTED ビュー, 4-20
- DBA_SYNC_CAPTURE ビュー, 4-20
- DB_DOMAIN パラメータ, 2-2
- DBMS_APPLY_ADM パッケージ, 4-15, 4-19, 4-45, 4-55, 4-57
- DBMS_AQ パッケージ, 9-20
- DBMS_CAPTURE_ADM パッケージ, 4-15, 4-19, 4-45
- DBMS_COMPARISON パッケージ, 10-2, 10-4, 10-7
- DBMS_MVIEW パッケージ, 8-9
- DBMS_PROPAGATION_ADM パッケージ, 4-19
- DBMS_REPCAT パッケージ, 7-21
- DBMS_STREAMS_ADM パッケージ, 4-15, 4-19, 4-24, 4-29, 4-39, 4-45, 6-2, 6-3, 6-7
- DBMS_STREAMS_ADVISOR_ADM パッケージ, 5-11
- DBMS_STREAMS_MESSAGING パッケージ, 9-10, 9-13, 9-23
- DBMS_STREAMS パッケージ, 5-30
- DB_NAME パラメータ, 2-2
- DDL LCR
 - 定義, 4-2
- DEQUEUE プロシージャ, 9-14, 9-22
- DROP MATERIALIZED VIEW LOG 文, 8-9

E

- ENQUEUE プロシージャ, 9-12, 9-24

G

- GLOBAL_NAMES パラメータ, 2-2, 2-8, 4-21, 7-4, 9-5

H

- HS_FDS_FETCH_ROWS パラメータ, 3-10
- HS_LANGUAGE パラメータ, 3-10
- HS_RPC_FETCH_SIZE パラメータ, 3-10

M

- MAINTAIN_GLOBAL プロシージャ, 4-15, 6-2
- MAINTAIN_SCHEMAS プロシージャ, 4-15, 4-24, 4-29, 4-39, 6-2, 6-7

MAINTAIN_SIMPLE_TTS プロシージャ, 4-15, 6-2
MAINTAIN_TABLES プロシージャ, 4-15, 6-2, 6-3
MAINTAIN_TTS プロシージャ, 4-15, 6-2
MAKE_COLUMN_GROUP プロシージャ, 7-21
MEMORY_TARGET パラメータ, 4-21, 7-4, 9-5
MERGE_STREAMS_JOB プロシージャ, 5-32

N

n-way レプリケーション, 4-14

O

ORA-01031 エラー, 9-34
ORA-02019 エラー, 9-36
ORA-24033 エラー, 9-35
Oracle Database Gateway, 3-3, 3-8
構成, 3-9
ベスト・プラクティス, 3-9
Oracle Messaging Gateway, 9-5
Oracle Streams
Advanced Queuing, 9-1
アラート, 5-26
データ・ディクショナリ・ビュー, 4-19
パッケージ, 4-19
レプリケーション, 4-2
構成プロシージャ, 4-15
Oracle Streams Advanced Queuing, 「メッセージ機能」
を参照
Oracle Streams トポロジ, 5-10
Oracle Streams パフォーマンス・アドバイザー, 5-11
Oracle Warehouse Builder, 1-4
Oracle 以外のデータベース, 3-3
大 / 小文字の差異, 3-10
データのアクセスと変更, 3-8
ベスト・プラクティス, 3-9

P

PROCESSES パラメータ, 4-21
PURGE_MVIEW_FROM_LOG プロシージャ, 8-9

R

REDO 転送サービス
構成, 4-32
RPC, 「リモート・プロシージャ・コール」を参照

S

SESSIONS パラメータ, 4-21
SET_MESSAGE_NOTIFICATION プロシージャ, 9-23
SET_TABLE_INSTANTIATION_SCN プロシージャ,
4-52
SET_TAG プロシージャ, 5-30
SET_UPDATE_CONFLICT_HANDLER プロシージャ,
4-57
SGA_TARGET パラメータ, 4-21, 7-4, 9-5
SHARED_POOL_SIZE パラメータ, 7-4, 9-5
SPLIT_STREAMS プロシージャ, 5-32
START_APPLY プロシージャ, 4-53
STREAMS_POOL パラメータ, 4-21

Streams 管理者
作成, 2-3
表領域の作成, 2-3
Streams クライアント
定義, 4-7
Streams トポロジ
DBMS_STREAMS_ADVISOR_ADM パッケージ,
5-11

U

UNREGISTER_MVIEW プロシージャ, 8-9

V

V\$BUFFERED_QUEUES ビュー, 4-20
V\$MVREFRESH ビュー, 8-14
V\$PROPAGATION_RECEIVER ビュー, 4-20
V\$PROPAGATION_SENDER ビュー, 4-20
V\$STREAMS_APPLY_COORDINATOR ビュー, 4-20,
5-25
V\$STREAMS_APPLY_READER ビュー, 4-20
V\$STREAMS_APPLY_SERVER ビュー, 4-20
V\$STREAMS_CAPTURE ビュー, 4-20
V\$STREAMS_TRANSACTION ビュー, 4-20

W

Warehouse Builder, 1-4

あ

宛先キュー
定義, 4-6
アラート
Oracle Streams, 5-26

い

位置の透過性, 3-2
インスタンス化 SCN
設定, 4-52
定義, 4-7

え

永続キュー
定義, 9-3
統計, 9-32
永続メッセージ機能
定義, 9-3
エラー・キュー
適用エラー, 5-30

お

オブジェクト・タイプ
作成, 9-7, 9-18

か

カスタム・ルールベース変換
定義, 4-8

監視

Oracle Streams レプリケーション, 5-9

競合解消, 5-23

トポロジ, 5-10

パフォーマンス, 5-25

取得プロセス, 5-12

統計, 5-14

適用プロセス, 5-19

統計, 5-21

伝播, 5-16, 5-18

統計, 5-18

マテリアライズド・ビュー, 8-10, 8-12

マテリアライズド・ビュー・グループ, 8-14

マテリアライズド・ビュー・レプリケーション, 8-10

マテリアライズド・ビュー・ログ, 8-20

メッセージ機能, 9-30

リフレッシュ・グループ, 8-18

完全リフレッシュ

定義, 7-3

管理者

Streams

作成, 2-3

き

キュー

ANYDATA タイプ

作成, 2-7

永続, 9-3

権限

付与, 9-34

コンシューマ, 9-36

表示, 9-33

サブスクライバ

表示, 9-33

定義, 9-2

バッファ, 4-3

監視, 5-23

変更, 9-25

メッセージ

削除, 9-36

統計, 9-32

表示, 9-31

キュー・ツー dblink 伝播

定義, 9-4

キュー・ツー・キュー伝播

定義, 9-4

キュー表

作成, 2-7

変更, 9-26

行 LCR

定義, 4-2

競合

定義, 4-9, 7-16

競合解消

監視, 5-23

組込み更新競合ハンドラ, 4-9

構成, 7-20

最新時刻

構成, 4-55, 7-16, 7-21

定義, 4-9, 7-16

トリガーの作成, 4-55, 7-18

列の追加, 4-55, 7-18

強制リフレッシュ

定義, 7-3

共有データベース・オブジェクト

定義, 10-2

く

組込み更新競合ハンドラ

定義, 4-9

グローバル・コーディネータ

定義, 3-2

グローバル・データベース名, 2-8

定義, 2-2

け

権限

付与, 9-10, 9-20, 9-34

こ

更新可能なマテリアライズド・ビュー, 7-13

構成, 7-16

スケジュール済リンク, 7-15

遅延トランザクション, 7-15

監視, 8-16

定義, 7-2

ページ・スケジュール, 7-15

高速リフレッシュ

定義, 7-3

コンシューマ, 9-36

構成, 9-13, 9-19

定義, 9-2

表示, 9-33

さ

最新時刻競合解消

構成, 4-55, 7-16

サブスクライバ

表示, 9-33

サブリメンタル・ロギング

定義, 4-9

し

事後処理

分散データベース

削減, 3-10

シノニム

作成, 3-5

定義, 3-2

分散データベース, 3-2

取得, 4-3

取得プロセス, 4-3

同期取得, 4-5

取得プロセス

監視, 5-12

管理, 5-2

起動, 5-2

サブリメンタル・ロギング, 4-9

定義, 4-3

停止, 5-2

統計, 5-14
パラメータの設定, 5-3
ルール, 4-8
順序付け
メッセージ, 9-3
情報の統合, 1-3
オプション, 1-4
初期化パラメータ
Oracle Streams レプリケーション, 4-21
設定, 2-2
マテリアライズド・ビュー, 7-4
メッセージ機能, 9-5

す

スケジュール済リンク
定義, 7-15
ステートフル・アラート
定義, 5-26
ステートレス・アラート
定義, 5-26
ストアド・ファンクション
定義, 3-3
ストアド・プロシージャ
実行, 3-7
定義, 3-3
ストリームの分離とマージ, 5-32

せ

宣言ルールベース変換
定義, 4-8

そ

ソース・キュー
定義, 4-6

た

大 / 小文字の差異
分散データベース, 3-10
ダウンストリーム取得プロセス
定義, 4-4
タグ
定義, 4-10
データの収束, 10-8
変更の循環の回避, 4-28, 4-44, 4-54

ち

遅延トランザクション
監視, 8-16
定義, 7-15
チューニング
分散データベース, 3-10

つ

通知
構成, 9-23
定義, 9-4
例, 9-16

て

ディレクトリ・オブジェクト
作成, 4-26, 4-36, 4-42, 6-9
データ
収束, 10-1, 10-7
タグ, 10-8
比較, 10-1, 10-4
変更, 3-6
データの収束, 10-1
データの比較, 10-1
データベース・リンク
再作成, 8-23
作成, 2-8, 7-5
定義, 2-8, 9-4
データ・レプリケーションと統合, 1-3
オプション, 1-4
適用, 4-7
適用エラー
管理, 5-30
データベース・オブジェクトの修正, 5-30
トランザクションの再試行, 5-31
適用ハンドラ
定義, 4-7
適用プロセス
インスタンス化 SCN, 4-7
監視, 5-19
統計, 5-21
管理, 5-6
起動, 5-6
定義, 4-6
停止, 5-6
適用エラー, 5-30
適用ハンドラ, 4-7
パラメータの設定, 5-7
ルール, 4-8
伝播
監視, 5-16, 5-18
統計, 5-18
構成, 9-9
定義, 4-6, 9-4
トラブルシューティング, 9-36
変更, 9-28
無効化, 5-4
有効化, 5-4
ルール, 4-8
伝播ジョブ
定義, 9-4

と

問合せ
複数のデータベース, 3-4
同期取得
構成, 4-45
定義, 4-5
ルール, 4-8
統計
永続キュー, 9-32
取得プロセス, 5-14
適用プロセス, 5-21
伝播, 5-18
バッファ・キュー, 5-23

トポロジ, 5-10
DBMS_STREAMS_ADVISOR_ADM パッケージ,
5-11
トラブルシューティング
Oracle Streams レプリケーション, 5-26
使用できないレプリケーション先, 5-32
適用エラー, 5-30
アラート, 5-26
伝播, 9-36
マテリアライズド・ビュー, 8-22
マテリアライズド・ビュー・ログ, 8-23
リフレッシュ, 8-22
メッセージ機能, 9-34
メッセージのエンキュー, 9-34, 9-35
メッセージのデキュー, 9-34
トリガー
作成, 4-55, 7-18, 9-11

ね

ネガティブ・ルール・セット
定義, 4-7

は

ページ・スケジュール
定義, 7-15
バッファ・キュー
監視
統計, 5-23
定義, 4-3, 9-3
バッファ・メッセージ機能
定義, 9-3
ハブアンドスポーク・レプリケーション, 4-12
構成, 4-39
データベース・オブジェクトの追加, 6-3
データベースの追加, 6-7
パフォーマンス
Oracle Streams パフォーマンス・アドバイザ, 5-11
Oracle Streams レプリケーション
監視, 5-25

ひ

表
列の追加, 4-55, 7-18
表領域
Streams 管理者用に作成, 2-3

ふ

ファンクション
定義, 3-3
フェデレーション, 3-2
定義, 1-3
プロシージャ
定義, 3-3
プロデューサ
定義, 9-2
分散 SQL
定義, 3-2
分散データベース
2 フェーズ・コミット, 3-2

Oracle 以外のデータベース, 3-3, 3-8
ベスト・プラクティス, 3-9
構成, 3-9
事後処理
削減, 3-10
シノニム, 3-2
準備, 3-4
使用する時期, 1-4
初期化パラメータ, 3-10
ストアド・プロシージャ, 3-3, 3-7
大 / 小文字の差異, 3-10
チューニング, 3-10
データの変更, 3-6
データベース・リンク, 2-8
問合せ, 3-4
フェデレーション, 3-2
分散 SQL, 3-2
メモリー要件, 3-10
分散問合せ
定義, 3-4
分散トランザクション
定義, 3-6

へ

ベスト・プラクティス
Oracle Streams レプリケーション, 4-21
Oracle 以外のデータベースの処理, 3-9
変換
ルールベース, 4-8
変更の循環
定義, 4-10

ほ

ポジティブ・ルール・セット
定義, 4-7

ま

マスター・グループ
作成, 7-19
定義, 7-14
停止中, 7-20
レプリケーション・サポートの生成, 7-21
マスター・サイト
監視
登録済マテリアライズド・ビュー, 8-21
マテリアライズド・ビュー・ログ, 8-20
構成, 7-4, 7-17
定義, 7-3
マテリアライズド・ビュー・サポートのクリーン・
アップ, 8-8
マスター表
定義, 7-3
マテリアライズド・ビュー, 7-1
監視, 8-10
管理, 8-1
更新可能, 7-13
競合, 7-16
競合解消, 7-20
構成, 7-16
定義, 7-2

- 削除, 8-7
- 準備, 7-4
- 初期化パラメータ, 7-4
- トラブルシューティング, 8-22
 - リフレッシュ, 8-22
- マスター・グループ
 - 作成, 7-19
 - 定義, 7-14
 - 停止中, 7-20
 - レプリケーション・サポートの生成, 7-21
- マスター・サイト
 - 構成, 7-17
 - 定義, 7-3
- マスター・サイトでの登録解除, 8-8
- マスター表
 - 定義, 7-3
- マテリアライズド・ビュー・グループ
 - 監視, 8-14
 - 作成, 7-22
 - 定義, 7-14
 - リフレッシュ・グループとの関係, 7-14
- マテリアライズド・ビュー・サイト
 - 構成, 7-4
 - 定義, 7-3
- マテリアライズド・ビュー・ログ
 - 監視, 8-20
 - 構成, 7-6
 - 定義, 7-3
 - トラブルシューティング, 8-23
- 読取り専用, 7-8
 - 構成, 7-9
 - 定義, 7-2
- リフレッシュ, 8-2
 - 監視, 8-14
 - 定義, 7-3
- リフレッシュ・グループ
 - 監視, 8-18
 - 構成, 7-23
 - 定義, 7-3
 - マテリアライズド・ビュー・グループとの関係, 7-14
 - マテリアライズド・ビューの追加, 8-5
 - リフレッシュ, 8-3
 - レプリケーション, 7-2, 8-1
- マテリアライズド・ビュー・グループ
 - 監視, 8-14
 - 作成, 7-22
 - 定義, 7-14
 - リフレッシュ・グループとの関係, 7-14
- マテリアライズド・ビュー・サイト
 - 構成, 7-4
 - 定義, 7-3
- マテリアライズド・ビュー・ログ
 - 監視, 8-20
 - 構成, 7-6
 - 定義, 7-3
 - トラブルシューティング, 8-23
 - ページ, 8-8

め

- メッセージ
 - 削除, 9-36

- 表示, 9-31
- メッセージ機能, 9-1
 - Oracle Messaging Gateway, 9-5
 - Oracle メッセージと Oracle 以外のメッセージの統合, 9-5
 - 永続, 9-3
 - 監視, 9-30
 - キュー
 - 定義, 9-2
 - 変更, 9-25
 - キュー表
 - 変更, 9-26
 - コンシューマ, 9-2
 - 準備, 9-5
 - 使用する時期, 1-6
 - 初期化パラメータ, 9-5
 - 通知
 - 構成, 9-23
 - 定義, 9-4
 - 例, 9-16
 - 伝播
 - 定義, 9-4
 - 変更, 9-28
 - 統計, 9-32
 - トラブルシューティング, 9-34
 - エンキュー, 9-34, 9-35
 - デキュー, 9-34
 - 伝播, 9-36
 - バッファ, 9-3
 - プロデューサ, 9-2
 - メッセージ
 - 表示, 9-31
 - メッセージ・クライアント
 - 構成, 9-13, 9-19
 - メッセージのエンキュー, 9-2, 9-10, 9-15, 9-23
 - メッセージの順序付け, 9-3
 - メッセージのタイプ
 - 作成, 9-7, 9-18
 - メッセージのデキュー, 9-2, 9-13, 9-15, 9-20
 - メッセージの表示, 9-23
 - メッセージ・モード, 9-3
 - 例, 9-6
 - メッセージ・クライアント
 - 構成, 9-13, 9-19
 - メッセージのエンキュー, 9-2, 9-15, 9-23
 - メカニズムの構成, 9-10
 - メッセージの順序付け
 - 定義, 9-3
 - メッセージのタイプ
 - 作成, 9-7, 9-18
 - メッセージのデキュー, 9-2, 9-15
 - メッセージの統計, 9-32

ゆ

- ユーザー
 - 作成, 2-5

よ

- 読取り専用マテリアライズド・ビュー, 7-8
 - 構成, 7-9
 - 定義, 7-2

り

リフレッシュ

- 監視, 8-14
- 定義, 7-3
- トラブルシューティング, 8-22
- マテリアライズド・ビュー, 8-2
- メソッド, 7-3

リフレッシュ・グループ

- 監視, 8-18
- 構成, 7-23
- 定義, 7-3
- マテリアライズド・ビュー・グループとの関係, 7-14
- マテリアライズド・ビューの追加, 8-5
- リフレッシュ, 8-3

リモート・プロシージャ・コール (RPC)

- 定義, 3-7

使用する時期, 1-6

データの読取り / 書込み, 7-12

トラブルシューティング, 8-22

読取り専用データ, 7-8

連続, 4-2

レプリケーション・グループ

定義, 7-14

レプリケーション・サポートの生成, 7-21

連続レプリケーション, 4-2

ろ

ローカル取得プロセス

定義, 4-4

論理変更レコード (LCR)

定義, 4-2

る

ルール

- 取得プロセス, 4-8
- 定義, 4-7
- 適用プロセス, 4-8
- 伝播, 4-8
- 同期取得, 4-8

ルール・セット

定義, 4-7

ルールベースの変換

定義, 4-8

れ

レプリケーション

Oracle Streams, 4-2

2 データベース, 4-11, 4-24, 4-29, 4-45

n-way, 4-14

Streams クライアント, 4-7

拡張, 6-1

監視, 5-9

管理, 5-1, 5-2

競合解消, 4-9, 4-55, 5-23

構成, 4-24, 4-29, 4-39, 4-45

構成プロシージャ, 4-15

サブリメンタル・ロギング, 4-9

取得プロセス, 4-3, 5-2

準備, 4-21

使用する時期, 1-5

タグ, 4-10

適用プロセス, 4-6, 5-6

伝播, 4-6

トポロジ, 5-10

トラブルシューティング, 5-26

ハブアンドスポーク, 4-12, 4-39, 6-3, 6-7

ベスト・プラクティス, 4-21

変更の取得, 4-3

ルール, 4-7

ルールベースの変換, 4-8

定義, 4-2, 7-2

マテリアライズド・ビュー, 7-1, 7-2

監視, 8-10

管理, 8-2

準備, 7-4

