

Oracle8*i*

移行ガイド

リリース 8.1

ORACLE®

Oracle8i 移行ガイド リリース 8.1

部品番号 : A62768-1

第 1 版 : 1999 年 5 月 (第 1 刷)

原本名 : Oracle8i Migration, Release 8.1.5

原本部品番号 : A67774-01

原本著者 : Randy Urbano

グラフィック・デザイナー : Valarie Moore

原本協力者 : Nipun Agarwal, Karleen Aghevli, Reema Al-Shaikh, Rick Anderson, Vikas Arora, Neerja Bhatt, Bill Bridge, Thong Bui, Sashi Chandrasekran, Ben Chang, Debashish Chatterjee, Lakshminaray Chidambaran, Maria Chien, Eugene Chong, George Claborn, David Colello, Michael Depledge, Alan Downing, Sandy Dreskin, Sreenivas Gollapudi, Brajesh Goyal, Miranda Gresham, Terry Hart, Michael Hartstein, Jeffrey Hebert, Bhaskar Himatsingka, Thuvan Hoang, Alison Holloway, Chin Hong, Wei Huang, Nancy Ikeda, Pavana Jain, Robert Jenkins, Sanjeev Jhala, Maura Joglekar, Christopher Jones, Sanjay Kaluskar, Dhiraj Kapoor, Vishwanath Karra, Susan Kotsovolos, Viswanathan Krishnamurthy, Muralidhar Krishnaprasad, Janaki Krishnaswamy, Andre Kruglikov, Thomas Kurian, Paul Lane, Gordon Larimer, Lefty Leverenz, Jing Liu, Juan Loaiza, Neil Le, J. Bill Lee, Tracy Lee, Bill Maimone, Ethan Malasky, Raghu Mani, Shailendra Mishra, Ari Mozes, Kannan Muthukkaruppan, Subramanian Muralidhar, Ravi Murthy, Karuna Muthiah, Anil Nori, Peter Ogilvie, Irene Paradisis, Rosanne Park, Joan Pearson, Elizabeth Pitt, Greg Pongracz, Lois Price, Franco Putzolu, Anil Ramdin, N. C. Ramesh, Paul Raveling, Ann Rhee, Mary Rhodes, Anindo Roy, Usha Sangam, Richard Sarwal, Ashok Saxena, Ajay Sethi, Carol Sexton, Franz Spickhoff, James Stamos, Debbie Steiner, Harry Sun, Katia Tarkhanov, Juan Tellez, Alvin To, Alex Tsukerman, Douglas Utzig, Peter Vasterd, Guhan Viswanathan, Rahim Yaseen, Steven Wertheimer, Rick Wessman, Andrew Witkowski, Lik Wong, Aravind Yalamanchi, Qin Yu

Copyright © 1996, 1999, Oracle Corporation. All rights reserved.

Printed in Japan.

制限付権利の説明

プログラムの使用、複製または開示は、オラクル社との契約に記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に関する法律により保護されています。

当ソフトウェア（プログラム）のリバース・エンジニアリング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ことがあります。オラクル社は本ドキュメントの無謬性を保証しません。

* オラクル社とは、Oracle Corporation（米国オラクル）または日本オラクル株式会社（日本オラクル）を指します。

危険な用途への使用について

オラクル社製品は、原子力、航空産業、大量輸送、医療あるいはその他の危険が伴うアプリケーションに用途として開発されておりません。オラクル社製品を上述のようなアプリケーションに使用することについての安全確保は、顧客各位の責任と費用により行ってください。万一かかる用途での使用によりクレームや損害が発生いたしましても、日本オラクル株式会社と開発元である Oracle Corporation（米国オラクル）およびその関連会社は一切責任を負いかねます。当プログラムを米国国防総省の米国政府機関に提供する際には、『Restricted Rights』と共に提供してください。この場合次の Legend が適用されます。

Restricted Rights Legend

Programs delivered subject to the DOD FAR Supplement are "commercial computer software" and use, duplication and disclosure of the Programs shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement. Otherwise, Programs delivered subject to the Federal Acquisition Regulations are "restricted computer software" and use, duplication and disclosure of the Programs shall be subject to the restrictions in FAR 52.227-14, Rights in Data -- General, including Alternate III (June 1987). Oracle Corporation, 500 Oracle Parkway, Redwood City, CA 94065.

このドキュメントに記載されているその他の会社名および製品名は、あくまでその製品および会社を識別する目的にのみ使用されており、それぞれの所有者の商標または登録商標です。

目次

はじめに.....	xi
-----------	----

1 概要

用語.....	1-2
スクリプトの実行.....	1-3
ワード・サイズの変更.....	1-4
最適フレキシブル・アーキテクチャ（OFA）の使用.....	1-5
Oracle Parallel Server のローリング・アップグレード.....	1-5
オプションの削除.....	1-6
移行ステップの概要.....	1-6
ステップ 1: 移行の準備.....	1-8
ステップ 2: 移行プロセスのテスト.....	1-8
ステップ 3: 移行したテスト・データベースのテスト.....	1-8
ステップ 4: ソース・データベースの準備および保存.....	1-9
ステップ 5: 本番データベースの移行.....	1-9
ステップ 6: 新しい本番データベースのチューニングおよび調整.....	1-9
移行におけるデータベース管理者の役割.....	1-10
移行におけるアプリケーション開発者の役割.....	1-10

2 移行の準備

移行の準備.....	2-2
新しいデータベースの機能の理解.....	2-2
移行方法の選択.....	2-3
システム要件と使用可能なリソースの見積り.....	2-15
新しいリリースに対する Oracle ホーム・ディレクトリの選択.....	2-18

共通移行問題の回避.....	2-18
バックアップ計画の準備.....	2-18
テスト計画の作成.....	2-19
移行処理のテスト.....	2-22
移行したテスト・データベースのテスト.....	2-23

3 移行ユーティリティによる移行

移行ユーティリティを使用するためのドキュメント・ロードマップ.....	3-2
移行ユーティリティを使用した移行の概要.....	3-3
移行処理の概要.....	3-3
移行ユーティリティの使用.....	3-5
システムの考慮事項と要件.....	3-6
領域要件.....	3-6
ブロック・サイズの考慮事項.....	3-6
レプリケーション環境のための考慮事項.....	3-7
Oracle Parallel Server をインストールしたシステムの移行.....	3-7
異なるオペレーティング・システムへの移行.....	3-7
キャラクタ・セットに関する考慮事項.....	3-8
移行のための Oracle7 ソース・データベースの準備.....	3-9
リリース 8.1 Oracle ソフトウェアのインストール.....	3-12
移行ユーティリティ・コマンド行オプションの検討.....	3-14
Oracle7 のソース・データベースの移行.....	3-15
移行に対する Oracle7 の環境の準備.....	3-15
Oracle7 の環境での移行ステップ.....	3-16
Oracle7 のソース・データベースの保存.....	3-20
Oracle8i の環境での移行ステップ.....	3-21
移行時のエラーのトラブルシューティング.....	3-30
移行の中止.....	3-30

4 Oracle Data Migration Assistant による移行

Oracle Data Migration Assistant を使用するためのドキュメント・ロードマップ.....	4-2
Oracle Data Migration Assistant を使用した移行の概要.....	4-3
Oracle Data Migration Assistant がサポートする Oracle7 データベースからの開始.....	4-3
ダウングレード.....	4-3
システムの考慮事項と要件.....	4-4

領域要件.....	4-4
ブロック・サイズの考慮事項.....	4-5
レプリケーション環境のための考慮事項.....	4-5
異なるオペレーティング・システムへの移行.....	4-5
キャラクタ・セットに関する考慮事項.....	4-6
移行のための Oracle7 ソース・データベースの準備.....	4-7
Oracle ソフトウェア リリース 8.1 のインストールおよびそのデータベースの移行.....	4-11
Oracle Data Migration Assistant 単独での実行.....	4-16
移行の終了.....	4-17
移行時のトラブルシューティング.....	4-20
移行の中止.....	4-20
5 エクスポート / インポートによる移行	
エクスポート / インポートの基本.....	5-2
エクスポート要件.....	5-2
インポート要件.....	5-2
その他のオプション.....	5-3
エクスポート / インポートを使用したリリース 8.0 以前のソース・データベースの移行.....	5-3
6 データベースの移行後	
移行したデータベースのバックアップ.....	6-2
無効な日付制約のチェック.....	6-2
使用不可のビットマップ索引の再構築.....	6-3
パラレル実行を使用する問題の回避.....	6-3
パーティション・ビューからパーティション表への移行.....	6-4
OUTLN ユーザーのパスワードの変更.....	6-4
Net8 の新しいリリースへの移行またはアップグレード (オプション) 6-4	
ご使用の listener.ora ファイルの変更.....	6-5
データベースのテストと結果の比較.....	6-6
移行されたデータベースのチューニング.....	6-6
新機能の適宜追加.....	6-7
必要な新管理手順の作成.....	6-7
7 Oracle8i の新しいリリースへのアップグレード	
アップグレード・パス.....	7-2

Oracle8i の新しいリリースへのアップグレード	7-3
アップグレードの準備	7-3
データベースの移行	7-6
特定のコンポーネントのアップグレード	7-21
アドバンスト・レプリケーションのアップグレード	7-21
Oracle Parallel Server のアップグレード	7-24
スナップショットのアップグレード	7-25
アドバンスト・キューイング・オプションのアップグレード	7-26
ユーザー定義データ型のアップグレード	7-30
リカバリ・カタログのアップグレード	7-30
無効な PL/SQL モジュールの再コンパイル	7-32
新リリースへのアップグレードの後	7-33
新しい TO_LOB 演算子の使用	7-33
無効な日付制約のチェック	7-33
パラレル実行を使用する問題の回避	7-34
新しいリリース用の INITsid.ORA ファイルの調整	7-34
OUTLN ユーザーのパスワードの変更	7-35
ご使用の listener.ora ファイルの変更	7-35
Java オブジェクトの削除	7-36
現在のリリースのワード・サイズの変更	7-37

8 互換性および相互運用性

互換性の概要	8-2
COMPATIBLE パラメータ	8-2
8.1.0 以降の互換性レベルを必要とする機能	8-9
アプリケーション	8-9
表領域	8-9
スキーマ・オブジェクト	8-10
パーティション化	8-11
組込みデータ型	8-11
ユーザー定義データ型	8-12
Oracle Parallel Server	8-12
データの保護	8-13
分散データベース	8-13
データ・ウェアハウジング	8-13

データ・アクセス.....	8-14
Spatial 情報および Visual 情報.....	8-15
相互運用性の概要.....	8-15
互換性および相互運用性の問題.....	8-16
アプリケーション.....	8-16
起動およびシャットダウン.....	8-22
表領域とデータ・ファイル.....	8-23
データ・ディクショナリ.....	8-24
スキーマ・オブジェクト.....	8-25
データ型.....	8-25
ユーザー定義データ型.....	8-29
SQL と PL/SQL.....	8-30
アドバンスト・キューイング (AQ).....	8-32
プロシージャおよびパッケージ.....	8-33
Oracle オプティマイザ.....	8-34
Oracle Parallel Server.....	8-34
データベース・セキュリティ.....	8-37
データベース・バックアップおよび回復.....	8-39
分散データベース.....	8-44
SQL*Net または Net8.....	8-46
エクスポート / インポート.....	8-48
その他の互換性および相互運用性の問題.....	8-50

9 アプリケーションのアップグレード

Oracle8i へのアプリケーションのアップグレードの概要.....	9-2
OCI アプリケーションおよびプリコンパイラ・アプリケーションのアップグレード.....	9-2
OCI アプリケーションのアップグレード.....	9-3
プリコンパイラ・アプリケーションのアップグレード.....	9-4
SQL*Plus スクリプトのアップグレード.....	9-6
Oracle7 の Forms または Developer/2000 アプリケーションのアップグレード.....	9-6
LONG から LOB へのコピー.....	9-7

10 Server Manager から SQL*Plus への移行

起動の違い.....	10-2
Server Manager の起動.....	10-2

SQL*Plus の起動.....	10-2
コマンド.....	10-3
新 SQL*Plus リリース 8.1 のコマンド.....	10-3
Server Manager と SQL*Plus に共通のコマンド.....	10-5
Server Manager コマンドに等しい SQL*Plus コマンド.....	10-6
SET TIMING コマンド機能が異なる場合	10-7
SQL*Plus で使用不可な Server Manager コマンド.....	10-7
構文の違い.....	10-7
コメント.....	10-7
ブランク行.....	10-10
ハイフン継続文字.....	10-10
アンパサンド.....	10-12
CREATE TYPE および CREATE LIBRARY コマンド	10-13
COMMIT コマンド	10-14

11 物理 ROWID の移行について

アプリケーションおよびデータの移行.....	11-2
DBMS_ROWID パッケージ	11-3
ROWID 変換型.....	11-3
ROWID 変換関数.....	11-4
変換プロシージャの例.....	11-5
例 1.....	11-5
例 2.....	11-6
例 3.....	11-6
例 4.....	11-6
例 5.....	11-6
スナップショット・リフレッシュ.....	11-6
バージョン 8 以前のクライアントとの互換性の問題.....	11-7
ROWID 関連の移行に関する質問および回答	11-7

12 古いバージョン 8 リリースへのダウングレード

全オフライン・バックアップの実行.....	12-2
非互換性の削除.....	12-2
表領域.....	12-5
スキーマ・オブジェクト.....	12-6

パーティション化.....	12-14
データ型.....	12-17
ユーザー定義データ型.....	12-21
SQL および PL/SQL.....	12-25
Java.....	12-26
アドバンスト・キューイング (AQ).....	12-27
プロシージャおよびパッケージ.....	12-33
制約およびトリガー.....	12-34
セキュリティ.....	12-36
データベース・バックアップおよび回復.....	12-37
分散データベース.....	12-38
Net8.....	12-41
データベース互換性のリセット.....	12-42
データベースのダウングレード.....	12-43
アドバンスト・レプリケーション・サポートの再生成.....	12-51
リリース 8.0.4 での UTL_REF パッケージの再インストール.....	12-51
リリース 8.0.3 でのリカバリ・マネージャ・パッケージの再インストール.....	12-52

13 Oracle7 へのダウングレード

Oracle8i から Oracle7 へのダウングレードの概要.....	13-2
新規または変更データを含まないデータベースのダウングレード.....	13-2
新規または変更データを含むデータベースのダウングレード.....	13-3
ダウングレードの代替方法.....	13-4

A 移行問題のトラブルシューティング

移行ユーティリティおよび Oracle Data Migration Assistant の使用に関する問題.....	A-2
一般的な移行問題.....	A-2
移行ユーティリティのエラー.....	A-5
ALTER DATABASE CONVERT コマンドにおける問題.....	A-16
Oracle7 制御ファイルの存在.....	A-16
NOMOUNT 以外のモードで起動するデータベース.....	A-17
変換ファイルが見付からない.....	A-17
REMOTE_LOGIN_PASSWORDFILE から EXCLUSIVE へ初期化パラメータの設定.....	A-18
データベース名の不合致.....	A-19
ALTER DATABASE CONVERT コマンドの再実行.....	A-19

データ・ファイルのバージョン整合性の問題.....	A-20
---------------------------	------

B 初期化パラメータの変更

バージョン 8 で追加された初期化パラメータ.....	B-2
リリース 8.0 で追加された初期化パラメータ.....	B-2
リリース 8.1 で追加された初期化パラメータ.....	B-3
バージョン 8 で改名された初期化パラメータ.....	B-4
リリース 8.0 で改名された初期化パラメータ.....	B-4
リリース 8.1.4 で改名された初期化パラメータ.....	B-5
リリース 8.1.5 で改名された初期化パラメータ.....	B-5
バージョン 8 で廃止された初期化パラメータ.....	B-6
リリース 8.0 で廃止された初期化パラメータ.....	B-6
リリース 8.1 で廃止された初期化パラメータ.....	B-6
初期化パラメータの互換性の問題.....	B-8
LOG_CHECKPOINT_TIMEOUT の新規デフォルト値	B-8
データ・ディクショナリの保護.....	B-8
DML_LOCKS パラメータ	B-8
DB_DOMAIN パラメータ	B-9
大規模プールから割り当てられたパラレル実行.....	B-9
アーカイブ・ログの宛先のパラメータ	B-13

C 静的データ・ディクショナリ・ビューの変更

バージョン 8 で追加された静的データ・ディクショナリ・ビュー.....	C-2
リリース 8.0 で追加された静的データ・ディクショナリ・ビュー.....	C-2
リリース 8.1 で追加された静的データ・ディクショナリ・ビュー.....	C-4
バージョン 8 で追加された列と静的データ・ディクショナリ・ビュー.....	C-7
リリース 8.0 で追加された列と静的データ・ディクショナリ・ビュー.....	C-7
リリース 8.1 で追加された列と静的データ・ディクショナリ・ビュー.....	C-9
バージョン 8 で削除された列と静的データ・ディクショナリ・ビュー.....	C-11
リリース 8.0 で削除された列と静的データ・ディクショナリ・ビュー.....	C-11
リリース 8.1 で削除された列と静的データ・ディクショナリ・ビュー.....	C-12
バージョン 8 で改名された列と静的データ・ディクショナリ・ビュー.....	C-13
リリース 8.0 で改名された列と静的データ・ディクショナリ・ビュー.....	C-13
NULL を戻す可能性のある列と静的データ・ディクショナリ・ビュー	C-14
バージョン 8 で廃止された静的データ・ディクショナリ・ビュー.....	C-15

リリース 8.0 で廃止された静的データ・ディクショナリ・ビュー	C-16
リリース 8.1 で廃止された静的データ・ディクショナリ・ビュー	C-16

D 動的パフォーマンス・ビューの変更

バージョン 8 で追加された動的パフォーマンス・ビュー	D-2
リリース 8.0 で追加された動的パフォーマンス・ビュー	D-2
リリース 8.1 で追加された動的パフォーマンス・ビュー	D-5
バージョン 8 で改名された動的パフォーマンス・ビュー	D-7
リリース 8.1 で改名された動的パフォーマンス・ビュー	D-7
バージョン 8 で追加された列と動的パフォーマンス・ビュー	D-8
リリース 8.0 で追加された列と動的パフォーマンス・ビュー	D-8
リリース 8.1 で追加された列と動的パフォーマンス・ビュー	D-9
リリース 8.1 で削除された列と動的パフォーマンス・ビュー	D-10
リリース 8.1 で削除された列と動的パフォーマンス・ビュー	D-10
バージョン 8 で廃止された動的パフォーマンス・ビュー	D-11
リリース 8.1 で廃止された動的パフォーマンス・ビュー	D-11
動的パフォーマンス・ビューの日付列	D-11

E 新しい内部データ型と SQL 関数

新しいリリースで追加された内部データ型	E-2
リリース 8.0 で追加された内部データ型	E-2
リリース 8.1 で追加された内部データ型	E-2
新しいリリースで追加された SQL 関数	E-3
リリース 8.0 で追加された SQL 関数	E-3
リリース 8.1 で追加された SQL 関数	E-4

索引

はじめに

このマニュアルは、Oracle データベース・システムの移行、アップグレード、ダウングレードの計画と実行の手引きを示しています。ここでは、移行ユーティリティ、Oracle Data Migration Assistant、エクスポート / インポートおよびデータのコピーを使用した移行について説明します。

また、このマニュアルでは互換性、現行リリースへのアプリケーションのアップグレードおよび初期化パラメータの変更やデータ・ディクショナリの変更などの Oracle の現行リリース内の重要な変更について説明します。

この章では、次のトピックについて一般的な説明を行います。

- [対象読者と前提条件](#)
- [Oracle8i 移行ガイドの構成](#)
- [このマニュアルで使用する表記規則](#)

Oracle8i 移行ガイドでは、Oracle8i および Oracle8i Enterprise Edition 製品のさまざまな機能について説明します。Oracle8i および Oracle8i Enterprise Edition の基本機能は同じです。ただし、いくつかの拡張機能は Oracle8 Enterprise Edition でのみ使用でき、その一部はオプションです。たとえば、パーティション化を使用するには、Oracle8 Enterprise Edition とパーティション化・オプションが必要です。

対象読者と前提条件

このマニュアルは、Oracle ソフトウェアの移行またはアップグレード、ダウングレード操作の計画と実行を担当する、データベース管理者（DBA）、アプリケーション開発者、セキュリティ管理者、システム・オペレータを対象に作成されています。このマニュアルのユーザーが Oracle Server の現行リリースとそのオペレーティング・システム環境を理解していることが前提です。ユーザーが Oracle データベース管理システム（DBMS）の概要を理解していることも前提としています。Oracle8i 概要の最初の章では、このマニュアルで使用する概念と用語について広範囲に紹介しています。

Oracle8i 移行ガイドの構成

このマニュアルは、次の章と付録で構成されています。

第 1 章: 概要

このマニュアル全体で使用される用語の定義について説明します。移行の手順とデータベース管理者およびアプリケーション開発者の役割について概説します。

第 2 章: 移行の準備

データベースを移行する前に行う手順について説明します。

第 3 章: 移行ユーティリティによる移行

移行ユーティリティを使用して Oracle7 データベースを Oracle8i に移行する方法について説明します。

第 4 章: Oracle Data Migration Assistant による移行

Oracle Data Migration Assistant を使用して、Oracle7 データベースを Oracle8i に移行する方法について説明します。

第 5 章: エクスポート/インポートによる移行

エクスポート/インポート・ユーティリティを使用して、Oracle7 またはバージョン 6 のデータベースを Oracle8i に移行する方法について説明します。

第 6 章: データベースの移行後

データベースを Oracle8i へ移行した後で行う手順について説明します。

第7章: Oracle8i の新しいリリースへのアップグレード

次の手順について段階的に説明します。

- 8.0 リリースまたは前の 8.1 リリースから Oracle の新規 8.1 リリースへのアップグレード。
- Oracle ソフトウェアの固有のコンポーネントの現行リリースへのアップグレード。
- ユーザーのデータベースのワード・サイズの変更 (32 ビットと 64 ビットの切換え)。

第8章: 互換性および相互運用性

Oracle の異なるリリース間の互換性および相互運用性について、COMPATIBLE 初期化パラメータの詳細も含めて説明します。この章では、リリース 8.1.0 以上の互換性レベルを必要とするリリース 8.1 の機能も説明します。また、互換性および相互運用性に関連する固有の問題も説明します。

第9章: アプリケーションのアップグレード

Oracle7 アプリケーションおよび Tools を Oracle8i で使用できるようにアップグレードする方法の概要を説明します。

第10章: Server Manager から SQL*Plus への移行

SQL*PLUS で使用できるように Server Manager 行モードのスクリプトを変更する方法を説明します。Server Manager は、今後のリリースでは使用されなくなる予定です。

第11章: 物理 ROWID の移行について

バージョン 8 の ROWID に関連する問題について、ROWID を含む列のバージョン 8 への移行など固有の情報も含めて説明します。

第12章: 古いバージョン 8 リリースへのダウングレード

新しい 8.1 リリースから前回の 8.1 リリースまたは 8.0 リリースへのダウングレードの方法を説明します。この章では、ダウングレード先のリリースとの非互換性の解消、およびデータベースの互換性レベルの再設定についても説明します。

第13章: Oracle7 へのダウングレード

Oracle8i から Oracle7 へのデータベースのダウングレードについて説明します。

付録 A: 移行問題のトラブルシューティング

移行に共通の問題およびこれら問題の訂正に必要な処置について説明します。また、移行ユーティリティおよび Oracle Data Migration Assistant によって表示されるメッセージをリストで示し、各メッセージについても説明します。エラー・メッセージの場合、考えられる原因および訂正のための処置について説明します。

付録 B: 初期化パラメータの変更

移行に重要な Oracle 初期化パラメータをリストで示します。特に、この付録ではバージョン 8 で追加、改名または使用されなくなった初期化パラメータについて説明します。また、固有の初期化パラメータに関連する互換性の問題も説明します。

付録 C: 静的データ・ディクショナリ・ビューの変更

追加または使用されなくなった静的データ・ディクショナリ・ビューをリストで示します。また、列が追加、削除および改名された静的データ・ディクショナリ・ビューもリストで示します。リリース 8.1 から NULL を戻すようになった静的データ・ディクショナリ・ビューの列もリストで示します。

付録 D: 動的パフォーマンス・ビューの変更

追加されたまたは使用されなくなった動的パフォーマンス・ビュー（V\$ ビュー）をリストで示します。追加されたおよび削除された動的パフォーマンス・ビューもリストで示します。

付録 E: 新しい内部データ型と SQL 関数

新規の内部データ型および SQL 関数をリストで示します。

このマニュアルで使用する表記規則

このマニュアルでは、以下に示す表記上の規則に従います。

大文字 アルファベットの大文字は、コマンドのキーワード、オブジェクト名、初期化パラメータ、静的データ・ディクショナリ・ビュー名および動的データ・ディクショナリ・ビュー名を示します。次にその例を示します。

「COMPATIBLE 初期化パラメータを使用して、データベースの互換性が制御できます。」

イタリック文字 イタリック文字は変数を示します。次にその例を示します。

「アップグレードする前にインストールされていたリリースを *old_release* が参照する場合、*uold_release.sql* を実行してください。」

コーディング例 SQL および Server Manager 行モード、SQL*Plus のコマンドと文は、次に示す例のように通常のテキストとは別に固定幅のクーリエ・フォントで表示されます。

```
SELECT username FROM dba_users WHERE username = 'MIGRATE';
```

サンプル文には、コンマまたは引用符などの句読点が含まれている場合があります。サンプル文にある句読点は、すべて必須構文です。セミコロンまたはその他の終了文字が文を終了するために必須であるかどうかは、アプリケーションによって異なります。また、文を終了するために次行にスラッシュ（/）が必要なアプリケーションもあります。

サンプル文の大文字は、Oracle SQL 内のキーワードを示しています。ただし、文を発行するとき、キーワードの大文字と小文字は区別されません。

サンプル文の小文字は、その単語が単なる例として使用されていることを示しています。たとえば、小文字の単語は、場合によって表、列、ファイルなどの名前を示します。

固定幅のクーリエ・フォントが段落内にある場合、それは次の例にあるようにファイル名、スクリプト名、実行ファイル名またはディレクトリ・パスを意味します。

リリース 8.1 の互換性レベルでユーザー定義型を識別するには、リリース 8.1 に付属する *utlincmpt.sql* スクリプトを実行します。

この章では、以前にリリースされた 8.0 データベース（Oracle7 またはバージョン 6 など）を Oracle8i へ移行するために必要な手順を概説します。これらの移行の手順によって、既存のバージョン 8 以前のデータベース・システム（対応付けられたアプリケーションを含む）を Oracle8i のデータベース・システムへ変換します。Oracle8i は、Oracle の以前のバージョンおよびリリースのすべてと互換性があります。したがって、このマニュアルで説明する移行手順によって変換されたデータベースは、以前のバージョンの場合と同様に機能し、Oracle8i の新機能を任意に活用できます。

現行の本番データベースを移行する前に、いくつかの準備処理を実行する必要があります。データベースを移行した後、移行をテストするために、さらにテスト・ステップをいくつか実行します。他の複数の手順によって、既存のリリース 8.0 以前のアプリケーションに Oracle8i の新機能を追加できます。

この章では、このマニュアルで使用される用語の定義とデータベースのワード・サイズの変更にに関する情報が含まれます。

この章で説明するトピックは、次のとおりです。

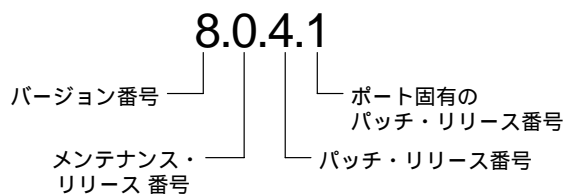
- [用語](#)
- [スクリプトの実行](#)
- [ワード・サイズの変更](#)
- [最適フレキシブル・アーキテクチャ（OFA）の使用](#)
- [オプションの削除](#)
- [移行ステップの概要](#)
- [移行におけるデータベース管理者の役割](#)
- [移行におけるアプリケーション開発者の役割](#)

用語

次の用語（イタリックで表記）は、このマニュアル全体で使用されます。

このマニュアルに記載されている指示は、Oracle の異なるバージョンおよびリリース間での移行について説明しています。図 1-1 には、リリース番号の各部分が何を表しているのかが説明されています。

図 1-1 Oracle リリース番号の例



参照： Oracle リリース番号の詳細は、『Oracle8i 管理者ガイド』を参照してください。

このマニュアルで言及される Oracle のバージョンは、そのバージョンのすべてのリリースに適用されます。バージョン 8 についての説明は、リリース 8.0 と 8.1 中のすべてのリリースに適用されます。Oracle7 についての説明は、7.0、7.1、7.2 および 7.3 を含むバージョン 7 のすべてのリリースに適用されます。

同様に、このマニュアルで言及されるメンテナン・ス・リリースは、そのメンテナン・ス・リリース内のすべての本番パッチ・リリースおよびポート指定パッチ・リリースに適用されます。したがって、バージョン 8.0 についての説明は、リリース 8.0.3、8.0.4、8.0.5 および 8.0.6 を含むリリース 8.0 内のすべての本番リリースに適用されますが、ベータ版である 8.0.2 には適用されません。

同じことが、パッチ・リリースにも適用されます。このマニュアルで言及されるパッチ・リリースは、そのパッチ・リリース内のすべてのポート指定パッチ・リリースに適用されます。したがって、リリース 8.0.4 についての説明は、8.0.4.0、8.0.4.1 およびリリース 8.0.4 内のその他すべてのポート指定パッチ・リリースに適用されます。

移行は、Oracle データベースのインストール済みのバージョンを新しいバージョンに変換する処理です。たとえば、バージョン 7 のデータベースを Oracle8i に変換することが、データベース・システムの移行です。

ソース・データベースは Oracle8i へ移行されるデータベースであり、移行中は、バージョン 6 や Oracle7 などの古いバージョンの Oracle ソフトウェアを使用します。ターゲット・データベースはソース・データベースを移行するデータベースであり、移行中は、新しい Oracle8i のソフトウェアを使用します。

アップグレードは、Oracle データベースをインストールされているリリースから同じバージョンの新しいリリースに変換する処理です。たとえば、パッチ・リリース 8.0.3 をパッチ・リリース 8.0.4 に変換することが、アップグレードです。また、8.0 のメンテナンス・リリースを 8.1 のメンテナンス・リリースへ変換することもアップグレードです。

参照： アップグレードの詳細は、[第 7 章「Oracle8i の新しいリリースへのアップグレード」](#)を参照してください。

ダウングレードとは Oracle データベースのインストール済みのバージョンを、新しいリリースから以前のリリースに変換する処理です。たとえば、Oracle データベースをリリース 8.1.5 からリリース 8.0.5 に戻すことがダウングレードであり、Oracle8i からバージョン 7 に戻すこともダウングレードです。

参照： リリース 8.1 データベースを前回の 8.1 リリースまたは 8.0 リリースへダウングレードするには、[第 12 章「古いバージョン 8 リリースへのダウングレード」](#)を参照してください。Oracle7 へのダウングレードの詳細は、[第 13 章「Oracle7 へのダウングレード」](#)を参照してください。

スクリプトの実行

移行、アップグレードおよびダウングレード処理を実行する場合、さまざまなスクリプトを実行する必要があります。スクリプトを実行するとき、スクリプトが「ORA-」エラーを報告する場合があります。一般的に、領域が不足していることを警告するエラーまたはスクリプトが実行しないことを警告するエラーは、調査する必要があります。これらのタイプのエラーが発生する場合、スクリプト操作は成功していない場合があります。ただし、存在しないオブジェクトの変更または削除の障害についてのエラーは、通常無視できます。

ワード・サイズの変更

移行、アップグレード、ダウングレード操作の際、Oracle データベース・サーバーのワード・サイズを変更することができます。ワード・サイズの変更には次のような状況が考えられます。

- 64 ビット・ハードウェアにインストールされている 32 ビット Oracle ソフトウェアを 64 ビット Oracle ソフトウェアへ変更する。
- 64 ビット・ハードウェアにインストールされている 64 ビット Oracle ソフトウェアを 32 ビット Oracle ソフトウェアへ変更する。

移行、アップグレードまたはダウングレード操作中にワード・サイズを変更する場合、特別な処理はありません。ワード・サイズはこれらの操作中に、自動的に変更されます。ただし、同じリリース間でワード・サイズを変更する場合、7-37 ページの「[現在のリリースのワード・サイズの変更](#)」に記載されている指示に従ってください。たとえば、Oracle リリース 8.1.5 の 32 ビット・バージョンを Oracle リリース 8.1.5 の 64 ビット・バージョンへ変更する場合、この手順を完了する必要があります。

ハードウェアを 32 ビットから 64 ビットへアップグレードしたり、64 ビットから 32 ビットへダウングレードしたりする場合は、次の情報が適用されます。

- ハードウェアをアップグレードする場合、32 ビット・ハードウェアから 64 ビット・ハードウェアへ変換した後でも、問題なく既存の 32 ビット Oracle ソフトウェアを使用できる。
- ハードウェアを 64 ビットから 32 ビットへダウングレードする場合は、ハードウェアのダウングレードの前に、まず Oracle ソフトウェアを 32 ビット・ソフトウェアへダウングレードしなければならない。

データベース・データ、再試行、取消しのディスク上の形式は、Oracle の 32 ビットおよび 64 ビット・インストールとも同じです。32 ビットと 64 ビット Oracle のインストールで異なるのは、次の内部ストラクチャのみです。

- PL/SQL のコンパイルされた形式。PL/SQL の再コンパイルの方法とタイミングに関する情報は、このマニュアルの該当する章で説明します。
- オブジェクトの記憶形式。これは、データベースを作成した Oracle のリリースによって決定されます。既存の記憶形式は、必要に応じて、その他のフォーマットへ透過的に変換されます。

最適フレキシブル・アーキテクチャ（OFA）の使用

バージョン 7 データベースからの移行またはバージョン 8 データベースからのアップグレードのいずれかの場合、オラクル社は Oracle8i のインストール用に最適フレキシブル・アーキテクチャ（OFA）規格をお薦めします。OFA 規格とは、ほとんどメンテナンスの手間の要らない効率的で、信頼性のある Oracle データベース用の構成ガイドラインの集合です。

OFA には、次の利点があります。

- ディスク上の大量の複雑なソフトウェアおよびデータをデバイスのボトルネックまたはパフォーマンスの低下を避けながら構成する。
- ソフトウェアまたはデータ・バックアップ機能などのデータ破損の危険性のある日常業務を簡単にする。
- 複数の Oracle データベース間での切替えを軽減する。
- データベース成長を処理および管理できる。
- データ・ディクショナリにおける空き領域の断片化を除去し、その他の断片化を見極め、リソース競合を最小限に抑える。

現在 OFA 規格を使用していない場合は、OFA 規格への切替えには、ディレクトリ構造の修正とデータベース・ファイルの再配置が伴います。

参照： OFA の詳細は、Oracle オペレーティング・システム固有のドキュメントを参照してください。また、データベース・ファイルの再配置の詳細は、『Oracle8i 管理者ガイド』を参照してください。

Oracle Parallel Server のローリング・アップグレード

ローリング・アップグレードはサポートされていません。ローリング・アップグレードとは、Oracle Parallel Server 内の同じデータベースの異なるインスタンスが、一度に Oracle の新しいバージョンやリリースへアップグレードされることです。したがって、ローリング・アップグレードは、アップグレード・プロセス中に同時に異なるリリースが実行される結果となります。

注意： ローリング・アップグレードという用語は、Oracle7 から Oracle8i への移行およびリリース 8.0 から 8.1 へのアップグレードの両方を意味します。ローリング・アップグレードはいずれの場合もサポートされていません。

オプションの削除

新しい Oracle のリリースに移行またはアップグレードする場合に古いオプションを削除するときは、インストーラで削除します。アップグレードまたは移行の前後に削除できますが、削除する項目に対応したインストーラのバージョンを使用する必要があります。たとえば、Oracle Parallel Server をインストールした Oracle のリリース 8.0 を実行しており、このオプションがリリース 8.1 へアップグレードしたときに必要なくなると判断した場合は、次の方法のいずれかで Oracle Parallel Server を削除する必要があります。

- リリース 8.1 へアップグレードする前に、ご使用のリリース 8.0 のインストーラを使用して、Oracle Parallel Server を削除します。その後、リリース 8.1 をインストールするとき、Oracle Parallel Server をインストールしないでください。
- リリース 8.1 へアップグレードするとき、Oracle Parallel Server をインストールおよびアップグレードします。次にリリース 8.1 のインストーラで Oracle Parallel Server を削除します。

注意： オプションを削除した後、付加的なデータ・ディクショナリが残る場合があります。

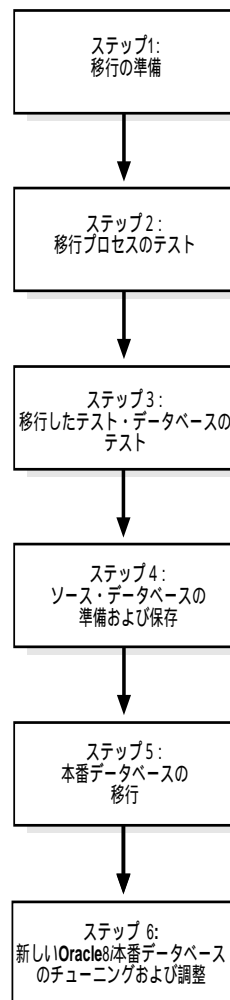
参照： Oracle Universal Installer の使用方法は、ご使用のオペレーティング・システム固有の Oracle インストレーション・ドキュメントを参照してください。

移行ステップの概要

データベースの移行を行う前に、移行処理の主なステップについて理解しておく必要があります。これらのステップはすべてのオペレーティング・システムに適用されますが、例外として一部のオペレーティング・システム固有の詳細項目があり、それについてはオペレーティング・システム固有の Oracle ドキュメントに示されています。

注意： この章の残りは、移行について説明します。アップグレードやダウングレードなどの移行以外の操作を実行する場合、その操作に該当する章へ進んでください。

図 1-2 主な移行ステップ



Oracle8i Tools を注意深く計画して使用すると、Oracle8i へのデータベースの移行プロセスを容易に行えます。Oracle Data Migration Assistant を使用することが、データベース全体を移行する最も簡単な方法です。一方、移行ユーティリティはより複雑ですが、データベース全体の移行プロセスをより制御できます。エクスポート / インポート・ユーティリティおよび SQL コピー・ユーティリティを使用すると、データベース各部を部分的に移行できます。

次の項では、[図 1-2](#) に示される主なステップの概要について説明します。これらの説明は、移行プロセスの主なステップを理解することを目的としています。詳細は、このマニュアルの該当する章および項を参照してください。

ステップ 1: 移行の準備

- Oracle8i のデータベースの機能を理解します。これらの機能の概要は、『Oracle8i の理解』を参照してください。
- 現行の本番データベースおよび移行の目的、使用可能な移行方法の動作と機能に関する考慮事項に基づいて、使用する移行方法を決定します。
- 移行に必要なシステム・リソースを見積もり、確保します。
- Oracle8i のテスト・データベースでの移行をテストする計画と、移行した Oracle8i の本番データベースをテストする計画を作成します。
- 予期しない問題や遅れが生じた場合にすぐに回復できるように、バックアップ計画を準備します。

ステップ 2: 移行プロセスのテスト

- Oracle7 のテスト・データベースを使用して、移行をテストします。テスト移行は、移行をテストするために作成した環境で行う必要があり、実際の Oracle7 の本番データベースに干渉してはなりません。

ステップ 3: 移行したテスト・データベースのテスト

- 移行前の Oracle7 のテスト・データベースと、Oracle8i へ移行された Oracle7 のテスト・データベースに対して、ステップ 1 で計画したテストを実行します。
- 結果を比較して、移行前のバージョン 7 のテスト・データベースに対するテスト結果と、移行された Oracle8i のテスト・データベースに対するテスト結果の間の相違点を記録します。
- 見つけた相違点を訂正する方法を調べて、その訂正をインプリメントします。
- テスト移行が問題なく完了し必要なアプリケーションで正常に動作するまで、必要に応じてステップ 1、ステップ 2 およびステップ 3 の最初の部分を繰り返します。

[第 2 章「移行の準備」](#)では、ステップ 1 からステップ 3 までを詳しく説明します。

ステップ 4: ソース・データベースの準備および保存

- Oracle8i への移行が正しく行われるように、現行の本番データベースを準備します。
- リリース 8.0 以前の本番データベースのバックアップ処理と Oracle8i への移行処理に必要なダウン時間をスケジュールリングします。
- 現行の本番データベースの全体バックアップを作成します。このステップは、移行方法として Oracle Data Migration Assistant または Oracle 移行ユーティリティのいずれかを使用する場合のみ必要です。

ステップ 5: 本番データベースの移行

- リリース 8.0 以前の本番データベースを Oracle8i へ移行します。
- 移行の後、本番データベースの全体バックアップおよびその他の移行後の処理を行います。

第 3 章では、ステップ 4 およびステップ 5 での移行ユーティリティの使用方法について説明します。第 4 章では、ステップ 4 およびステップ 5 での Oracle Data Migration Assistant の使用方法について説明します。第 5 章では、ステップ 4 およびステップ 5 でのエクスポート / インポート・ユーティリティの使用方法について説明します。第 6 章では、移行後のバックアップ手順およびその他の移行後の処理について説明します。

参照： アドバンスド・レプリケーションをインストールしたデータベース・システムを移行する場合は、『Oracle8i レプリケーション・ガイド』の付録 B「移行および互換性」を参照してください。

ステップ 6: 新しい本番データベースのチューニングおよび調整

- 新しい Oracle8i の本番データベースをチューニングします。Oracle8i の本番データベースのパフォーマンスは、移行前の Oracle データベースと同等以上になる必要があります。第 6 章では、これらのチューニング調整について説明します。
- Oracle8i データベースのどの新機能をデータに適用すれば適切かを決定し、それに応じてアプリケーションを更新します。
- 必要に応じ、新しいデータベース管理の手順を作成します。
- すべてのアプリケーションがテストされ、適切に動作するまでは、本番ユーザーを Oracle8i のデータベースに移行しないでください。第 9 章では、アプリケーションを更新する際の考慮事項について説明します。

移行の際、マルチバージョンングは便利な機能です。これによって、同じデータベースの複数のコピーを1台のコンピュータ・システムに保管できます。新しいバージョンをテストする間、移行前のバージョンを本番環境として使用できます。

移行におけるデータベース管理者の役割

通常、移行処理を確実に成功させる責任は、データベース管理者（DBA）が負います。DBAは、通常、移行処理の各ステップに関係しますが、移行したデータベースでのアプリケーションのテストに関連するステップにはかかりません。

データベース管理者に固有の任務は、次のとおりです。

- 移行処理に関連するすべてのスタッフと会合を持ち、移行処理における各スタッフの役割を明確に決定する
- テスト移行を実行する
- テストおよび本番移行処理をスケジュールリングする
- 移行前の Oracle7 の本番データベースのバックアップを行う
- 本番データベースの移行を完了する
- 新規に移行した Oracle8i の本番データベースのバックアップを作成する

すべてのアプリケーションがテストされ、適切に動作するまでは、移行された Oracle8i のデータベースに対する運用を開始しないでください。

移行におけるアプリケーション開発者の役割

アプリケーション開発者は、移行前の Oracle7 のデータベース用に設計されたアプリケーションが、移行された Oracle8i のデータベースでも正しく動作することを保証する責任があります。アプリケーション開発者は、移行された Oracle8i のデータベースに対するアプリケーションのテストを頻繁に行い、使用する Oracle8i の新機能を決定します。

Oracle7 の本番データベースを移行する前に、DBA またはアプリケーション開発者は、Oracle8i のテスト・データベースをインストールする必要があります。それから、アプリケーションが元の機能（または拡張された Oracle8i の機能）で動作するまで、アプリケーション開発者は必要に応じてそのアプリケーションをテストおよび変更できます。

次の参照先には、特定のアプリケーションに影響するような、移行された Oracle8i のデータベースでの相違点を判断するための情報があります。アプリケーション開発者は、これらの相違点を利用して、既存のアプリケーションへの変更を行うことができます。

- 第 8 章「互換性および相互運用性」では、Oracle のリリースの違いによって発生する可能性のある互換性および相互運用性問題について説明します。
- 第 9 章「アプリケーションのアップグレード」では、(Oracle7 のデータベースにアクセスする) 既存のアプリケーションが Oracle8i のデータベースにアクセスできるようにするために必要な変更点について説明します。また、Oracle8i の機能を利用するために Oracle7 のアプリケーションをアップグレードする方法についても説明しています。
- 『Oracle8i の理解』では、Oracle8i で使用可能な多くの新機能について説明しています。
- 付録 B「初期化パラメータの変更」では、バージョン 8 の新規の初期化パラメータ、変更された初期化パラメータおよび廃止された初期化パラメータをリストしています。
- 付録 C「静的データ・ディクショナリ・ビューの変更」では、バージョン 8 の新規の静的データ・ディクショナリ・ビュー、変更された静的データ・ディクショナリ・ビューおよび廃止された静的データ・ディクショナリ・ビューをリストしています。
- 付録 D「動的パフォーマンス・ビューの変更」では、バージョン 8 の新規の動的パフォーマンス・ビュー、変更された動的パフォーマンス・ビューおよび廃止された動的パフォーマンス・ビュー (VS ビュー) をリストしています。
- 付録 E「新しい内部データ型と SQL 関数」では、バージョン 8 で追加された新しいデータ型および SQL 機能をリストしています。
- 『Oracle8i Parallel Server 概要および管理』および『Oracle8i SQL リファレンス』では、Oracle8i での変更点と新機能について説明しています。
- バージョン 8.0 以前のデータベース・システムにアドバンスト・レプリケーションをインストール済みの場合は、『Oracle8i レプリケーション・ガイド』の付録 B「移行および互換性」を参照してください。
- アプリケーションの計画およびインプリメントの情報については、『Oracle8i アプリケーション開発者ガイド 基礎編』、『Oracle8i アプリケーション開発者ガイド ラージ・オブジェクト』および『Oracle8i アプリケーション開発者ガイド アドバンスト・キューイング』を参照してください。

Oracle8i では、既存のアプリケーションを Oracle8i にアップグレードすることを支援するいくつかの機能があります。たとえば、次のような機能です。

- Net8 および SQL*Net バージョン 2 は、Oracle の各バージョン間の通信をサポートしています。
- プログラミング・インタフェースは、Oracle の各バージョン間でも未変更のままです。
- Oracle の下位互換性では、異なるバージョンとリリース間で非互換な部分はわずかしかなかったりありません。

2

移行の準備

この章では、本番データベースを移行する前に完了しなければならない作業について説明します。また、この章では、移行プロセス（第 1 章ですでに概説）のステップ 1 からステップ 3 までを詳しく説明します。

この章で説明するトピックは、次のとおりです。

- [移行の準備](#)
- [移行処理のテスト](#)
- [移行したテスト・データベースのテスト](#)

この章の情報は一般的なものであり、Oracle7 およびバージョン 6 の本番データベース全般に適用されます。

参照： アドバンスド・レプリケーションをインストール済みの以前のデータベース・システムを移行する場合は、『Oracle8i レプリケーション・ガイド』の付録 B「移行および互換性」を参照してください。

移行の準備

移行のための準備のために、必ず次の作業を行ってください。

- [新しいデータベースの機能の理解](#)
- [移行方法の選択](#)
- [システム要件と使用可能なリソースの見積り](#)
- [新しいリリースに対する Oracle ホーム・ディレクトリの選択](#)
- [共通移行問題の回避](#)
- [バックアップ計画の準備](#)
- [テスト計画の作成](#)

新しいデータベースの機能の理解

移行処理を計画する前に、Oracle8i のデータベースの新機能を理解する必要があります。あるコンポーネントの新機能の情報を得るには、Oracle ドキュメント・ライブラリにある各機能固有のドキュメントを参照してください。たとえば、Oracle Parallel Server における変更は、『Oracle8i Parallel Server 概要および管理』を参照してください。

移行方法の選択

データベースを Oracle8i へ移行するには、次の 3 つの方法のいずれかを選択します。

- 移行ユーティリティを使用して、Oracle7 データベースを Oracle8i へ移行。使用しているプラットフォーム・システムで移行ユーティリティによって移行できる最初のリリースについては、オペレーティング・システム固有の Oracle ドキュメントを参照してください。たとえば一部のオペレーティング・システムでは、移行ユーティリティによって移行できるのはリリース 7.1.4 以降のデータベースのみです。

移行ユーティリティは、Oracle7 から Oracle8i への完全なデータベースの移行のためのコマンド行ユーティリティです。データ・ファイル・ヘッダーは変更しますが、実際のデータは未変更のままです。データはコピーしません。

- Oracle Data Migration Assistant を使用して、Oracle7 データベースを Oracle8i へ移行。使用しているプラットフォーム・システムで Oracle Data Migration Assistant によって移行できる最初のリリースについては、オペレーティング・システム固有の Oracle ドキュメントを参照してください。通常、Oracle Data Migration Assistant がサポートする最初のリリースは、移行ユーティリティがサポートする最初のリリースと同じです。

Oracle Data Migration Assistant には、完全なデータベースの移行のためのグラフィカル・ユーザー・インタフェース (GUI) があります。データ・ファイル・ヘッダーは変更しますが、実際のデータは未変更のままです。データはコピーしません。

- バージョン 7 (またはバージョン 6) のソース・データベースの全体または部分的エクスポートの実施。その後で、Oracle8i のターゲット・データベースへ全体または部分的なインポートを行います。

エクスポート / インポートによって、データベースの各部を移行できます。データ・ファイル・ヘッダーと実際のデータは変更せず、データの新しいコピーを作成します。

- COPY コマンドまたは CREATE TABLE コマンドの AS 句を使用した、ソース・データベースから Oracle8i のデータベースへのデータのコピー。

データをコピーすると、データベースの各部を移行できます。データをコピーすると、データ・ファイル・ヘッダーと実際のデータは変更せず、データの新しいコピーを作成します。

表 2-1 に、これらの方法のそれぞれの長所をまとめます。表 2-2 に、これらの方法のそれぞれの短所をまとめます。

表 2-1 各移行方法の長所

移行ユーティリティ	Oracle Data Migration Assistant	エクスポート/インポート	データのコピー
<p>自動的に実行され、DBA の介入が最小限で済む。</p> <p>変更されるオブジェクトはデータ・ディクショナリ・オブジェクトのみであるため、データベースのサイズに関係なく比較的高速である。</p> <p>本来、移行できるデータベースのサイズには制限がない。</p> <p>通常、他の移行方法と比べて、通常は比較的少量のディスク領域しか必要としない。</p> <p>Oracle Data Migration Assistant よりも、移行処理の制御を強化できる。</p>	<p>使用しやすい GUI のガイドに従って移行処理を行うことができる。</p> <p>自動的に実行され、DBA の介入が最小限で済む。</p> <p>変更されるオブジェクトはデータ・ディクショナリ・オブジェクトのみであるため、データベースのサイズに関係なく比較的高速である。</p> <p>本来、移行できるデータベースのサイズには制限がない。</p> <p>通常、他の移行方法と比べて、比較的少量のディスク領域しか必要としない。</p> <p>リリース間でのアップグレード操作、たとえば 8.0.5 から 8.1.5 へのアップグレードを実行できる。</p>	<p>バージョン 6 および Oracle7 のデータベースを Oracle8i へ移行できる。</p> <p>データベースの特定の部分を移行できる。</p> <p>Oracle のバージョン間のダウングレード、たとえば Oracle8i から Oracle7 へのダウングレードを実行できる。</p> <p>リリース間でのアップグレード操作またはダウングレード操作、たとえば 8.0.5 から 8.1.5 へのアップグレードを実行できる。</p> <p>データ・ファイルの断片化を解消し、データを圧縮して移行できるので、パフォーマンスを向上できる。</p> <p>変更した表領域または新しい表領域を使用するか、表をパーティション化することでデータベースを再構築できる。</p> <p>異なるオペレーティング・システムおよびハードウェア・プラットフォームへ移行できる。</p>	<p>データ・ファイルの断片化を解消し、データを圧縮して移行できるので、パフォーマンスを向上できる。</p> <p>変更した表領域または新しい表領域を使用してデータベースを再構築できる。</p> <p>バージョン 6 または Oracle7 のデータベースを Oracle8i へ移行できる。</p> <p>データベースの特定の部分を移行できる。</p> <p>リリース間でのアップグレード操作またはダウングレード操作、たとえばリリース 8.0.5 からリリース 8.1.5 へのアップグレードを実行できる。</p> <p>異なるオペレーティング・システムおよびハードウェア・プラットフォームへ移行できる。</p>

表 2-2 各移行方法の短所

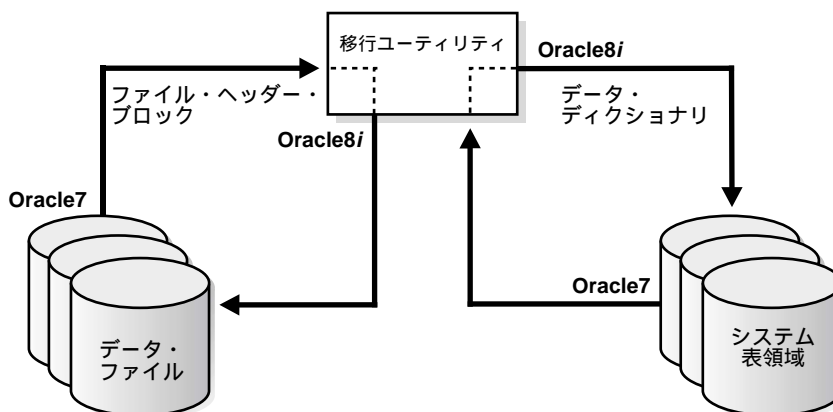
移行ユーティリティ	Oracle Data Migration Assistant	エクスポート/インポート	データのコピー
<p>Oracle7 から Oracle8i への移行しか実行できず、Oracle7 へのダウングレードはできない。</p> <p>リリース 7.0 データベースまたは特定の 7.1 リリース以前のデータベース上での直接移行は実行できない。特定の 7.1 リリースは、特定のオペレーティング・システムでのみ使用できる。</p> <p>リリース間でのアップグレード操作、たとえば 8.0.5 から 8.1.5 へのアップグレードを実行できない。</p> <p>データベースの選択した部分のみを移行することはできない。データベース全体を移行する。</p> <p>異なるオペレーティング・システムまたはハードウェア・プラットフォームへ移行できない。</p>	<p>移行処理が高度に自動化されているため、その他の方法より汎用性がない。GUI が最も重要な移行選択肢のみをカバーしている。</p> <p>ほかの方法より移行処理の制御を強化できない。</p> <p>Oracle7 へのダウングレードができない。</p> <p>リリース 7.0 データベースや特定の 7.1 リリース以前のデータベース上での直接移行は実行できない。特定の 7.1 リリースは、オペレーティング・システムでのみ使用できる。</p> <p>データベースの選択した部分のみを移行することはできない。データベース全体を移行する。</p> <p>異なるオペレーティング・システムまたはハードウェア・プラットフォームへ移行できない。</p> <p>Oracle Parallel Server をインストールしたシステムを移行できない。</p>	<p>小規模なデータベースを除いて非常に遅い。データ型 LONG の使用とデータの量に応じて所要時間が伸びる。数ギガバイトもある非常に大規模なデータベースでは数時間かかることがあり、テラバイト規模のデータベースでは数日かかることもある。</p> <p>エクスポート・ファイルへデータをコピーするために大量のディスク領域が必要である。</p>	<p>小規模なデータベースを除いて非常に遅い。データ型 LONG の使用とデータの量に応じて所要時間が伸びる。数ギガバイトもある非常に大規模なデータベースでは数時間かかることがあり、テラバイト規模のデータベースでは数日かかることもある。</p> <p>コピー操作中にソース・データベースとターゲット・データベースの両方が同時に使用可能である必要がある。</p>

次の項ではそれぞれの移行方法について詳しく説明し、それぞれに必要な時間と領域、およびどのような状況にどの方法が適しているかを説明します。

移行ユーティリティ

移行ユーティリティは、データ・ファイルのファイル・ヘッダーのみを変更し、また必要に応じてファイル内のデータの定義を変更して、Oracle7 のソース・データベースのファイルと構造を Oracle8i の形式に変換するコマンド行ユーティリティです。移行ユーティリティがデータ・ファイル内のデータ部分を変更することではなく、その形式や内容を変更することはありません。

図 2-1 移行ユーティリティ



移行ユーティリティの第一の利点は、そのスピードと使いやすさです。移行ユーティリティは、エクスポート / インポートに比べてかなり短時間で実行することができます。また、このユーティリティを使用する移行処理は、標準化された一連の簡単なステップしか必要としません。さらに、移行ユーティリティを使用してデータベースを移行するのにかかる時間は、データベースのサイズよりはデータ・ディクショナリ内のオブジェクト数によって決まります。

移行ユーティリティは、ソース・データベース全体を迅速に移行するときに特に便利です。エクスポート / インポートとは異なり、移行ユーティリティでは特定のデータ・ファイルのみを選択して移行することはできません。ただし、大量のデータおよび大きなデータ型、その他いくつか Oracle7 の機能を使用するデータベースの場合は、エクスポート / インポート機能は実用的ではなく、唯一の実用的なオプションは移行ユーティリティまたは Oracle Data Migration Assistant のみです。

移行ユーティリティが必要とするのは、Oracle7 (ソース) と Oracle8i (ターゲット) のデータ・ディクショナリを両方とも同時に保持できる、SYSTEM 表領域内の十分な一時領域のみです。

移行ユーティリティは、データベース・ファイル、ロールバック・セグメントおよび制御ファイルを含むデータベース全体を変換します。実際に Oracle7 のデータベースを移行する前ならばいつでも、Oracle7 のインスタンスでデータをオープンし、アクセスできます。ただし、移行ユーティリティによって Oracle7 のソース・データベースを Oracle8i に移行した後で Oracle7 に戻る方法は、Oracle7 のソース・データベースの全体バックアップを復元する以外にありません。

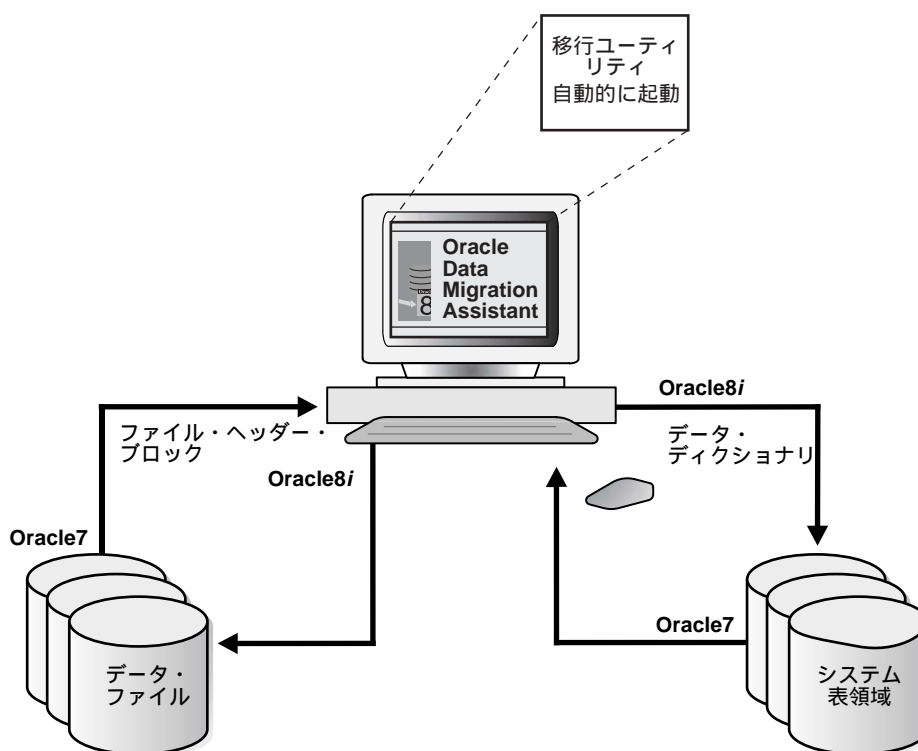
移行ユーティリティは、リリース 7.0 データベースまたは特定の 7.1 リリース以前のデータベース上での直接移行を実行できません。特定の 7.1 リリースの要件は、オペレーティング・システムでのみ使用できます。たとえば一部のオペレーティング・システムでは、移行ユーティリティによって Oracle8i へ移行できるのはリリース 7.1.4 以降のデータベースのみです。使用しているリリースが、使用しているオペレーティング・システムで移行ユーティリティがサポートするリリースよりも低い場合は、移行ユーティリティを使用して Oracle8i へ移行する前に、まずデータベースがサポートされる Oracle7 リリースまでご使用のデータベースをアップグレードまたは移行する必要があります。使用しているプラットフォーム・システムで移行ユーティリティによってサポートされている最初のリリースについては、オペレーティング・システム固有の Oracle マニュアルを参照してください。

参照： 移行ユーティリティの使用方法の詳細は、[第 3 章「移行ユーティリティによる移行」](#)を参照してください。

Oracle Data Migration Assistant

Oracle Data Migration Assistant には、移行処理をガイドするユーザー・フレンドリなグラフィカル・ユーザー・インタフェース（GUI）があります。Oracle Data Migration Assistant は移行ユーティリティをコールし、それをバックグラウンドで実行します。これによって、移行ユーティリティを手動でコマンド行から実行する必要がありません。

図 2-2 Oracle Data Migration Assistant



Oracle Data Migration Assistant の主な長所は、使用しやすいことです。Oracle Data Migration Assistant は移行ユーティリティをコールするため、移行ユーティリティのほとんどの長所と短所が Oracle Data Migration Assistant にも当てはまります。Oracle Data Migration Assistant と移行ユーティリティとの違いについては、2-9 ページの「[Oracle Data Migration Assistant と移行ユーティリティの選択](#)」を参照してください。

参照： Oracle Data Migration Assistant の使用方法の詳細は、[第 4 章](#)
「[Oracle Data Migration Assistant による移行](#)」を参照してください。

Oracle Data Migration Assistant と移行ユーティリティの選択

Oracle Data Migration Assistant または移行ユーティリティを選択するには、次の違いを考慮してください。

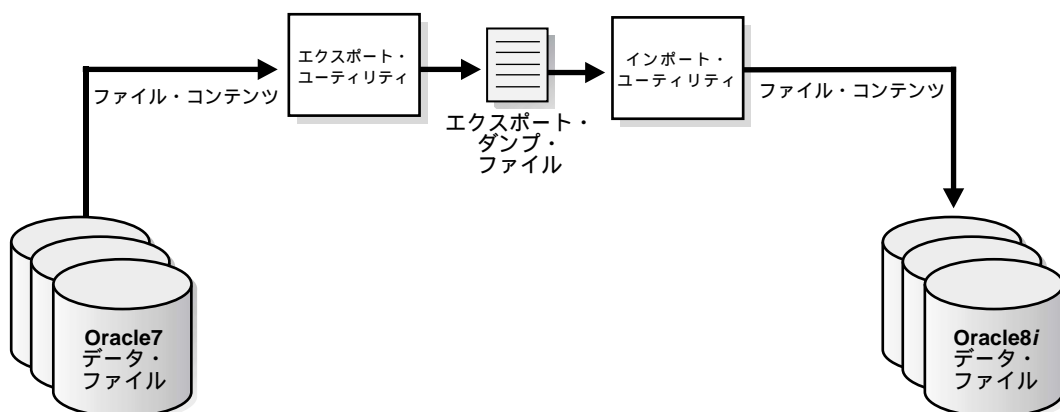
- Oracle Data Migration Assistant には、移行処理をガイドするユーザー・フレンドリな GUI があるが、移行ユーティリティはコマンド行ユーティリティである。Oracle Data Migration Assistant には、また広範なオンライン・ヘルプもあります。したがって、Oracle Data Migration Assistant は移行ユーティリティより使用しやすいと言えます。
- Oracle Data Migration Assistant は移行ユーティリティに比べて汎用性がない。複雑な処理を避けるため、Oracle Data Migration Assistant 移行処理のステップの多くを自動化しています。逆に、移行ユーティリティでは、各ステップを手動で実行する必要がありますが、移行処理中に必要な調整ができます。移行ユーティリティでは、各ステップを手動で実行する必要がありますが、移行処理中に必要な調整ができます。
- Oracle Data Migration Assistant は実行されるたびに、移行処理の全ステップを実行する。したがって、移行処理中に何らかの理由で Oracle Data Migration Assistant を終了しなければならない場合、Oracle7 データベースのバックアップを復元し、始めからやり直す必要があります。予期しないエラーを受け取った場合などは、Oracle Data Migration Assistant を終了させる必要があります。逆に、移行ユーティリティでは、処理中に明確に定義されたいくつかの主なステップがあります。特定のステップを中止させる場合でも、始めからやり直さずに前のステップの終わりから開始できます。
- Oracle Data Migration Assistant は廃止された初期化パラメータを `initSID.ora` ファイルから自動的に削除する。Oracle Data Migration Assistant によって削除される廃止された初期化パラメータのリストは、B-6 ページの「[バージョン 8 で廃止された初期化パラメータ](#)」を参照してください。逆に、移行ユーティリティでは、`initSID.ora` ファイルを変更しないため、移行ユーティリティを使用する場合は、廃止された初期化パラメータを手動で削除する必要があります。
- Oracle Data Migration Assistant は Oracle Parallel Server をインストールしたシステムの移行をサポートしない。ただし、移行ユーティリティでは、Oracle Parallel Server をインストールしたシステムを移行できます。

一般的に、コマンド行インタフェースよりグラフィカル・ユーザー・インタフェース (GUI) を希望し、ほとんど設定がない非常に自動化された処理を望む場合、Oracle Data Migration Assistant を使用します。一方、GUI よりコマンド行・インタフェースを希望し、より強力に移行処理を制御したい場合、移行ユーティリティを使用します。

エクスポート / インポート

移行ユーティリティとは異なり、エクスポート / インポート処理はソース・データベース内のデータを新しいデータベースに物理的にコピーします。ソース・データベースのエクスポート・ユーティリティは、ソース・データベースの特定部分をエクスポート・ファイルにコピーします。次に、Oracle8i のインポート・ユーティリティが、このエクスポートされたデータを新しい Oracle8i のデータベースにロードします。ただし、エクスポート・ファイルを移行する前に、新しい Oracle8i のターゲット・データベースを準備しておく必要があります。

図 2-3 エクスポート / インポート



次のセクションでは、データベースの移行にエクスポート / インポートを使用するかどうかを決める際に役に立つ、エクスポート / インポートのいくつかの局面について解説します。

参照： エクスポート / インポートによる移行方法の詳細は、[第 5 章「エクスポート / インポートによる移行」](#) および『Oracle8i ユーティリティ・ガイド』を参照してください。

エクスポート/インポートによって移行されたデータベースへの影響 エクスポート/インポートによる移行方法では、ソース・データベースを変更しません。したがって、ソース・データベースは、移行処理を通じて使用可能な状態にあります。ただし、一貫性のあるデータベースのスナップショットが必要な場合（データの整合性保持またはその他の目的のため）ソース・データベースは、制限モードで実行するか、またはエクスポート実行時に変更禁止にする必要があります。ソース・データベースを使用可能な状態にできるので、たとえば、既存の Oracle7 の本番データベースを実行しながら、エクスポート/インポートによる Oracle8i の新規データベースの作成を同時に行うことができます。この移行処理中にデータベースの完全な一貫性を維持するには、Oracle7 のデータベースのデータを変更する場合に、Oracle8i のデータベースのデータにも同じ変更を加える必要があります。

エクスポート/インポートは、データベースのアップグレードやダウングレードにも使用できます。たとえば、Oracle8i のデータベースから Oracle7 データベースへの逆方向の変換を、エクスポート/インポートによる移行方法によって実現できます。

最も重要なことは、エクスポート/インポート操作の結果としてまったく新しいデータベースが作成されることです。ソース・データベースには最終的に指定したデータのコピーが含まれますが、移行したデータベースは、元のソース・データベースとはまったく異なった方法で運用される場合があります。DBA によるデータベースの再構成またはデータの断片化解消、および新しい Oracle8i ソフトウェアの結果として、次の分野の変更が期待できます。

- パフォーマンス
- データ成長パターン
- 共有リソースの使用
- データ・ディクショナリ・サイズ
- オブジェクトの編成

エクスポート/インポートによってデータベースにもたらされる効果を最大限に利用するには、綿密な計画、熟練者による実装、および厳密なテストが必要です。これを行わなかった場合には、データベースに変更を加えたときに問題が発生することがあります。移行によってデータベースが再構成された後で、移行したデータベースが異なった動作をする場合は、その違いの原因を判断するのが困難になることがあります。

エクスポート/インポートの利点 エクスポート / インポートによるデータの移行には、次のような利点があります。

- データの断片化を解消できる。インポートすることによってデータが圧縮されるため、パフォーマンスを向上できる。
- データベースを再構成できる。つまり、新しい表領域を作成したり、既存の表や表領域、インポートによってデータがロードされる先のパーティションを変更したりできる。
- 特定のデータベース・オブジェクトまたはユーザーを移行できる。つまり、オブジェクト、ユーザー、およびその他希望する項目のみをインポートできる。
- バックアップ・アーカイブとしての役割を果たす。つまり、全データベース・エクスポートをソース・データベースのアーカイブとして使用できる。

エクスポート/インポートの制約 エクスポート / インポートによるデータの移行には、次のようないくつかの制約があります。

- エクスポート / インポートによるデータベースの移行では、熟練したデータベース管理者が必要になる。最終的に移行を試みる前に、必要とされる計画と複雑な実行作業には、通常、さまざまな段階と多くの熟練が必要になる。
- 大規模データベースの場合、データベース全体のエクスポート / インポートを行うと、エクスポート・ダンプ・ファイル用に大量の一時記憶領域が必要になる場合がある。
- オペレーティング・システムがデータベースと同等のファイル・サイズをサポートしていない場合は、エクスポートを複数のジョブに分割する必要があることがある。
- エクスポート / インポートでは、まったく新しいデータベースが作成される。ソース・データベースの状態を保持し、そのエクスポート・ダンプ・ファイル / データを Oracle8i のターゲット・データベースにインポートするには、インポートを実行する前にシステムにターゲット・データ・ファイルを作成する必要がある。
- Oracle8i への移行と断片化解消 / データベースの再構築を同時に行うなど、データベースに対して同時に複数の変更を行うと、トラブルシューティングを妨げることがある。
- 移行作業中にソース・データベースおよびターゲット・データベース内のデータの同期を保つには、ソース内のデータ変更を禁止するか、ターゲット・データベースへ移行されたデータへも同じ変更を加えてミラー化する。

エクスポート/インポートの時間要件 エクスポート/インポートを使用する全データベースの移行方法は、移行ユーティリティまたは Oracle Data Migration Assistant を使用した場合と比較して、時間が長くなる場合があります。したがって、ピーク時を外して移行するようにスケジュールしたり、移行中にソース・データベースに対して行われた変更を新しいターゲット・データベースに反映させる手配をしたりする必要があります。

エクスポート/インポートによる移行方法に必要な時間とシステム・リソース（特にディスク領域）は、DBA の技術、データベースのサイズ、移行するデータの型、特に再構築しなければならない索引の数、サイズおよび種類によって異なります。

たとえば、6 ギガバイトの比較的単純な Oracle7 のデータベースを Oracle8i に移行するのに、移行ユーティリティでは約 1 時間かかります。その同じ Oracle7 のデータベースをエクスポートする際には、2 ギガバイトの単一のエクスポート・ダンプ・ファイルが作成されます。その 1 つのエクスポート・ダンプ・ファイルをインポートするのに、20 時間かかります。5-3 ページの「[エクスポート/インポートを使用したリリース 8.0 以前のソース・データベースの移行](#)」に示されている各ステップを使用して移行を完了すると、2 日かかります。

エクスポート/インポートではデータベースの移行に時間がかかるので、次の各要素について考慮する必要があります。

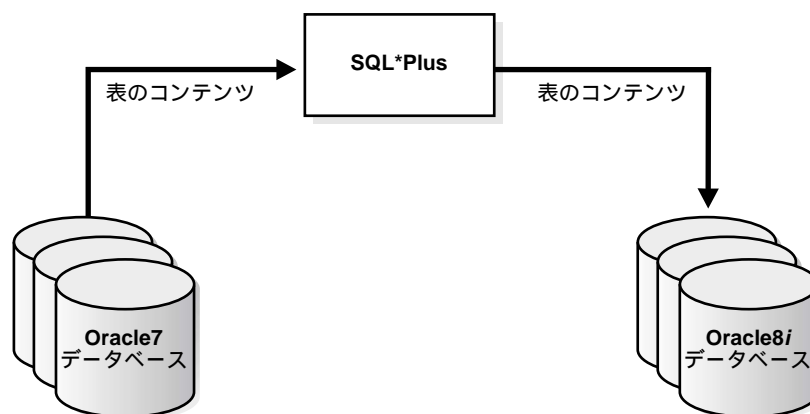
- 移行に必要な時間は、毎日のスケジュールにおいて利用できる非ピーク時または実動時間外の時間よりはるかに長い。移行処理の期間中に、業務にデータベースを使用できないようにすることは非現実的である。
- ソース・データベースを読取り専用にしたり、エクスポート実行後にソース・データベースへの変更を禁止した場合、インポートおよび移行ステップが完了するまでアプリケーションは利用できない。
- 規模の大きなデータベースについては、時間を削減し、移行の処理を最適化するために、パラレル・エクスポート・ストリームの運用を検討する。

Oracle8i インポートによるデータ定義変換 以前のバージョンからデータをインポートする場合、Oracle8i のインポート・ユーティリティでは、以前のバージョンのエクスポート・ダンプ・ファイルからデータを読み込む際にデータ定義に適切な変更を加えます。つまり、Oracle バージョン 6、バージョン 7 およびバージョン 8 のエクスポート・ユーティリティによって作成されたダンプ・ファイルが処理されます。エクスポートするソース・データベースがバージョン 6 よりも前の場合、エクスポートを実行する前に、ソース・データベースを最低でもバージョン 6 までアップグレードする必要があります。

データのコピー

データベース・リンクを使用して、ある Oracle データベースから別の Oracle データベースにデータをコピーできます。たとえば、SQL*Plus の COPY 文を使用してソース・データベースの表からターゲット・データベースの表にデータをコピーしたり、INSERT INTO コマンド、CREATE TABLE ...AS 文を使用してターゲット・データベースの中に新しい表を作成し、その表をソース・データベースのデータで埋めたりすることができます。

図 2-4 データのコピー



データのコピーを使用する移行方法には、エクスポート / インポートと同様の利点があります。どちらの移行方法を利用しても、新しい表領域を作成するか既存の表または表領域を変更することによって、データ・ファイルの断片化を解消したりデータベースを再構築したりできます。また、指定したデータベース・オブジェクトまたはユーザーのみを移行することもできます。

ただし、エクスポート / インポートとは異なり、データのコピーでは、表の特定の行を選択してターゲット・データベースに挿入できます。このため、データベース表の一部分のみを移行する場合にはデータのコピーを使用します。逆に、エクスポート / インポートを使用して Oracle7 から Oracle8i にデータを移行すると、表全体の移行のみが可能です。

たとえば、既存の表 (EMP@V7DB) のデータの一部分 (部門 10 および 20 の従業員のみ) を含む新しい表 (NEW_EMP) を作成するには、次のような SQL 文を使用します。

```

CREATE TABLE new_emp AS
  SELECT empno, ename, job, mgr, hiredate, sal, comm, deptno
  FROM emp@v7db WHERE deptno IN (10, 20);
  
```

データのコピーによる移行方法は、エクスポート / インポートの場合よりも少ないディスク領域とメモリー・バッファ領域で済みます。データのコピーでは、ソース・データベースとターゲット・データベースの両方がオンラインであればよいからです。テンポラリ・ファイルまたはエクスポート・ダンプ・ファイル用に、大量の余分な領域を割り当てる必要はありません。

大きいクラスタ表を扱っているときは、SQL*Plus COPY コマンドが役立ちます。さらに、SQL*Plus の COPY コマンドを使用すると、Net8（または SQL*Net）を使用してクラスタのさまざまな部分を並列に移動できます。データベース間でデータをコピーする方法の詳細は、『Oracle8i SQL リファレンス』の「CREATE TABLE コマンド」および『SQL*Plus ユーザーズ・ガイドおよびリファレンス』の「COPY コマンド」を参照してください。

システム要件と使用可能なリソースの見積り

移行を成功させるために必要となるシステム・リソースを見積もります。移行方法が異なると、リソース要件も異なることがあります。そのため、使用方法がよくわからない場合は、既存のデータベースを Oracle8i に移行するための各方法ごとに見積もりを行います。

見積もりに際して、次の各要素について考慮する必要があります。

- オペレーティング・システムとハードウェアの両方の構成要件
- 既存の本番データベースのサイズ
- Oracle8i をインプリメントするときに関連するデータベースに対して可能なサイズ調整

Oracle8i のバイナリ・ファイルは、Oracle7 の場合の 3 倍のディスク領域を必要とすることがあります。必要な領域サイズが 3 倍に増えるということは、大規模なバッチ・システム（数十個または数百個単位の実行可能ファイルを生成する）においては特に注意が必要です。実行可能ファイルに必要な領域は、Oracle8i の環境に組み込むオプションによっても異なります。たとえば、次のようなオプションがあります。

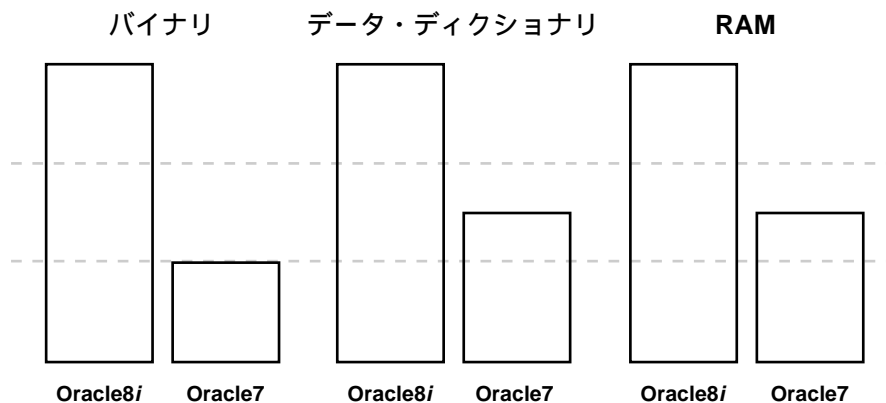
- Oracle Parallel Server（『Oracle8i Parallel Server 概要および管理』を参照）
- Net8 または SQL*Net の使用（『Oracle8i Net8 管理者ガイド』を参照）

また、Oracle8i データ・ディクショナリは、SYSTEM 表領域内に Oracle7 の場合の 2 倍のデータ・ディクショナリ領域を必要とすることがあります。移行ユーティリティを使用する場合、SYSTEM 表領域の要件は移行ユーティリティを CHECK_ONLY モードで実行して見積もることができます。

また、Oracle8i は Oracle7 の 2 倍の RAM を必要とすることがあります。必要な RAM の大きさは、Oracle8i の環境に組み込むオプションによっても異なります。

図 2-5 では、Oracle7 と Oracle8i 間でのシステム要件の違いを説明します。

図 2-5 移行に関するシステム要件



移行方法を選択して要件を見積もった後、移行を成功させるのに必要なリソースを確保します。

参照： システム要件の詳細は、ご使用のオペレーティング・システム固有の Oracle ドキュメントを参照してください。

同時アクセスのメモリー要件の見積り

Oracle8i のシステムのメモリー・サイズも、同時アクセスとその実現方法によって決まります。Oracle8i では、次の接続オプションをサポートしています。

- オプション 1:** 専用サーバー・アーキテクチャでローカル接続を使用する（「2 タスク共有」ともいう）。このオプションを、Oracle7 と同じように設定してください。
- オプション 2:** SQL*Net によるリモート接続を使用する。このオプションを、Oracle7 と同じように設定してください。
- オプション 3:** ローカル接続とリモート接続の両方にマルチスレッド共有サーバーを使用する。移行の後、『Oracle8i 管理者ガイド』に示されているとおりオプションの初期化パラメータを設定してください。
- オプション 4:** トランザクション・プロセッサ (TP)・モニターを使用する。

オプション 1 は、オプション 2 またはオプション 3 よりも多くのメモリーを必要とします。オプション 1 では、クライアント・アプリケーションとその Oracle Server（またはシャドウ）プロセスの両方が同じコンピュータ上に常駐する場合、その両方のためにメモリーが必要です。たとえば、100 個のクライアント・アプリケーション・プロセスを Oracle8i に接続した場合、100 個の Oracle Server プロセスがシステムに追加される結果となり、プロセスは合計 200 個になります。

オプション 2 の場合、Oracle プロセスのみがシステムに常駐し、クライアント・プロセスはリモートで接続されます。したがって、考慮する必要があるのは、Oracle Server プロセスのサイズおよび使用可能な共有メモリーのサイズのみです。

オプション 3 は、マルチスレッド・サーバー機能を使用し、専用の Oracle シャドウ・プロセスのかわりに、ローカル・クライアントまたはリモート・クライアントの複数のプロセスを単一のディスパッチャ・プロセスに接続できます。マルチスレッド・サーバー構成はパフォーマンス向上を意図したものではありませんが、Oracle8i のサーバー上で多数の同時接続を実現するようになるので、スループットが向上します。複数のクライアントが単一のディスパッチャに接続できるため、同時ユーザー接続のためのメモリー使用量は減少します。Oracle8i のマルチスレッド・サーバー機能の詳細は、『Oracle8i 概要』および『Oracle8i 管理者ガイド』を参照してください。

オプション 4 は、TP モニターを使用し、多数の（数百人を超える）ユーザー全員が OLQP/OLTP タイプのトランザクションを実行する必要があるシステムのための選択肢です。通常、そのようなトランザクションは持続時間が短く、ユーザーはデータベースに直接接続する必要はありません。すべてのトランザクションは、TP（トランザクション・プロセス）モニター・サービスを経由したメッセージを使用して実行されます。TP レイヤーは名前付きサービスを提供し、Oracle を含めた多様な DBMS システムとサービス要求を調整します。

注意： TP モニターを使用するための要件は非常に多様であり、このマニュアルでは説明しません。システム要件については、該当の TP モニターのベンダーに問い合わせてください。

単一システムの場合には、次の要因を考慮すればシステム・メモリー要件を見積もることができます。

- オープン・カーソル、および特定の Oracle アプリケーション・セッションでソートを発生させることのあるカーソルの平均数
- オープン・カーソルおよびソート領域を含む Oracle シャドウ・プロセスの平均サイズ
- システムの同時ユーザーのピーク数
- Oracle フロントエンド・アプリケーションの平均メモリー・サイズ

新しいリリースに対する Oracle ホーム・ディレクトリの選択

Oracle7 Oracle ホーム・ディレクトリから独立した Oracle8i リリースに対する Oracle ホーム・ディレクトリを選択する必要があります。Oracle8i ソフトウェアは、Oracle7 で使用した Oracle ホーム・ディレクトリにはインストールできません。

独立したインストレーション・ディレクトリを使用すると、ご使用の Oracle7 ソフトウェアをインストールしたままで、Oracle8i ソフトウェアを使用できます。この方法によって、本番環境全体を交換する前に、Oracle7 テスト・データベース上で移行処理をテストできます。

共通移行問題の回避

データベースを移行する前に共通移行問題を排除することで、時間を節約できます。共通問題の領域には、次の各項目があります。

- 列またはアプリケーション・コードに格納された ROWID を使用する場合は、[第 11 章「物理 ROWID の移行について」](#)を参照してください。バージョン 8 では ROWID の形式が異なるので旧 ROWID は無効であり、変換する必要があります。
- すべての Oracle 製品のバージョン、オペレーティング・システムのバージョン、およびサード・パーティー製ソフトウェアのバージョンが、使用しているハードウェア・プラットフォームでの Oracle8i に対して保証されていることを確認してください。詳細は、オペレーティング・システム固有の Oracle マニュアルを参照してください。

バックアップ計画の準備

移行が最終的に成功するかどうかは、適切なバックアップ計画の立案と実行次第で決まります。バックアップ計画を展開するには、次のような問題について考慮する必要があります。

- 業務上許容できない状況になるまでのどの程度の期間なら本番データベースが作動不能状態でもかまわないか。
- 可用性要件を満たすには、どのバックアップ計画を使用すべきか。
- サイトから離れた安全な場所にバックアップをアーカイブする必要があるか。
- どのくらいの時間でバックアップを復元できるか（オフサイト記憶域でのバックアップを含む）。
- 回復手順は正常にテストされているか。

バックアップ計画は、これらの問題のすべてに答え、データベースを正常にバックアップおよび回復するための手順を備えている必要があります。

参照： Oracle7 データベースの詳細は、『Oracle7 Server 管理者ガイド』を参照してください。Oracle8i データベースの詳細は、『Oracle8i バックアップおよびリカバリ・ガイド』を参照してください。

テスト計画の作成

移行処理のすべての段階を検証するために、一連のテストを慎重に設計してください。緻密なテストを実行し、テストが正常に完了すれば、本番データベースの移行処理を十分に理解し、予測できるようになり、移行処理の成功が確実になります。本番データベースを移行する前に、できる限り多くのテストを実行してください。テスト計画は、非常に重要です。

注意： 移行前に緻密なテストを実行しないことは危険であり、予測できない結果を招くことがあります。

テスト計画の中で、次のようなテストを実行する必要があります。

移行テスト

移行テストでは、ソース・データベースから移行されたデータベースへの移行パスを計画およびテストする必要があります。移行ユーティリティ、Oracle Data Migration Assistant またはエクスポート / インポートを使用するか、データのコピー方法を使用して本番データベースのデータをターゲット・データベースへ移行します。これらの方法については、[第4章「Oracle Data Migration Assistant による移行」](#)および[第5章「エクスポート / インポートによる移行」](#)を参照してください。

どの移行方法を選択する場合でも、移行計画を確立し、テストして検証することが必要です。

最小テスト

最小テストでは、ソース・データベースのアプリケーションの全部または一部をターゲット・データベースに移動し、ターゲット・データベースの新機能を使用可能にしないでアプリケーションを実行します。最小テストは非常に限定的なテストであり、「現実の」本番環境で現れるような潜在的問題は明らかにされません。ただし、アプリケーションの起動または呼出しに関して問題があれば、ただちに明らかになります。

機能テスト

機能テストとは、システムの新機能と既存機能を移行後にテストする一連のテストです。機能テストには、RDBMS システムのすべてのコンポーネントおよびネットワーキング、アプリケーション・コンポーネントのテストが含まれます。機能テストの目的は、システムの各コンポーネントが移行前と同様に機能し、新機能が正しく動作していることを検証することにあります。

統合テスト

統合テストでは、システムの各コンポーネントの相互作用を試験します。統合テストを計画するときは、次の各要素について考慮する必要があります。

- ターゲット・データベースのインスタンスで稼働する Pro*C/C++ アプリケーションは、新しいソフトウェアで問題がないことを確認するためにテストする必要がある。
- GUI インタフェースをその他のコンポーネントでテストする必要がある。
- Oracle8i のインスタンスにアプリケーションが直接接続されるかどうかに関係なく、データ型やデータ・ディクショナリのデータの変更（データ・ディクショナリへの行の追加、オブジェクト型の変更など）といったターゲット・データベースのわずかな変更でも、フロントエンド・アプリケーションにまで影響することがある。
- 2 つのコンポーネント間の接続に Net8 または SQL*Net がかわるときは、ストレス・テストとともにその接続のテストも実行しなければならない。

パフォーマンス・テスト

ターゲット・データベースのパフォーマンス・テストでは、ターゲット・データベースでのさまざまな SQL 文のパフォーマンスを、ソース・データベースでの SQL 文のパフォーマンスとそれぞれ比較します。移行する前に、ソース・データベースでのアプリケーションのパフォーマンス・プロファイルを理解する必要があります。特に、アプリケーションがデータベース・カーネルに対して実行するコールを理解する必要があります。

たとえば、Oracle Parallel Server を使用して、新しいリリースへ移行したときにキャッシュ・フュージョンによってもたらされるパフォーマンスの向上を測定する場合、移行の前にご使用のシステムの統計を必ず記録してください。キャッシュ・フュージョンのために、V\$SYSSTAT、V\$LOCK_ACTIVITY および V\$LOCK_CLASS_PING ビューからの統計を記録します。それによって、キャッシュ・フュージョン・パフォーマンス統計の前後を比較できます。

最高の結果を得るために、SQL スクリプト `utlbstat.sql` および `utlestat.sql` を実行して、特定の期間の V\$SYSSTAT 統計を収集します。一貫性のあるトランザクション・アクティビティ・レベルのピーク本番ロードを、最も一貫して反映している収集タイムフレームを使用します。V\$LOCK_ACTIVITY および V\$LOCK_CLASS_PING からデータを取得するには、統計収集期間の最初と最後で SELECT* 文を使用します。キャッシュ・フュージョンを新しいリリースで実行した後にこの処理を繰り返し、『Oracle8i Parallel Server 概要および管理』の説明に従って、システムのパフォーマンスを見積もります。

参照： チューニングの詳細は、『Oracle8i チューニング』を参照してください。ソース・データベースでのアプリケーションのパフォーマンス・プロファイルを完全に理解するには、TKPROF で SQL_TRACE およびプロファイルを使用可能にしてください。

ボリューム / ロード・ストレス・テスト

ボリューム / ロード・ストレス・テストでは、移行したデータベース全体を、大きいボリュームとロードでテストします。ボリュームとは、操作されるデータの量を表します。ロードとは、システム上の同時要求のレベルを表します。ボリューム / ロード・テストの目的は、さまざまなボリュームとロードでの本番システムの動作をエミュレートすることです。

ボリューム / ロード・ストレス・テストは非常に重要ですが、一般に見過ごされがちです。ユーザーはどのようなボリューム / ロード・ストレス・テストも実行しないことが多いようです。そのかわりに、ビジネス・アプリケーションを特に考慮しているわけではないベンチマークが広く利用されています。アプリケーションのベンチマークは、機能、パフォーマンスおよび統合に関する問題点を解明するために実行してください。ボリューム / ロード・ストレス・テストのかわりになるものではありません。

ソース・データベースを移行した後、すべてのデータにアクセスできてアプリケーションが正しく機能することを確認するために、データをテストする必要があります。データベースの調整が必要かどうか判断する必要もあります。可能な場合は、これらのテスト手続きを自動化してください。

テスト計画は、サイトで実行される作業を反映する必要があります。ソースの本番データベースのすべてのアプリケーションの機能やパフォーマンスをテストする必要があります。通常の使用状況とピーク時の使用状況の両方についてパフォーマンスの統計数値を収集してください。

特定の移行前後のテスト

テスト計画には、次のようなテスト項目を組み込みます。

- タイミングのテスト
- データ・ディクショナリ成長の監視
- ロールバック、一時セグメントの使用など、データベース・リソース使用率の監視

この情報を収集することによって、ソース・データベースと移行したターゲット・データベースを比較しやすくなります。

Oracle で各 SQL 文を実行するための実行計画を決定するには、ソース・データベースとターゲット・データベースの両方で EXPLAIN PLAN を使用します。この情報を複数の表に保存するには、INTO 句を使用します。

移行後に、移行されたデータベースの実行計画とソース・データベースの実行計画を比較できます。相違点がある場合は、移行されたデータベース上でコマンドを実行し、そのパフォーマンスを、ソース・データベース上で実行したコマンドのパフォーマンスと比較します。

移行処理のテスト

現行の本番データベースに干渉しないテスト環境を作成します。テスト環境は、選択した移行方法によって異なります。

- 移行ユーティリティまたは Oracle Data Migration Assistant を使用する場合は、ソース・データベースのテスト・バージョン（通常はサブセット）を作成してテスト移行を行う。
- エクスポート / インポートを使用する場合は、実際の本番ソース・データベースからほんの一部分をテストとしてエクスポート / インポートする。

テスト環境を使用してデータベースの移行を行います。最良の移行テストとは、ダウンサイズしたコピーまたはテスト・データに対してではなく、移行するデータベースの正確なコピーに対して実行することです。

注意： このデータベースのテスト・サブセットを正常に移行して、次のステップで説明するアプリケーションでテストするまでは、実際の本番データベースを移行しないでください。

Oracle8i のデータベースで使用する予定の OCI とプリコンパイラ・アプリケーションを、必ずアップグレードしてください。そうすることによって、本番データベースを移行する前にそれらのアプリケーションをサンプルの Oracle データベースでテストできます。詳細は、9-2 ページの「[OCI アプリケーションおよびプリコンパイラ・アプリケーションのアップグレード](#)」を参照してください。

移行したテスト・データベースのテスト

ソース・データベースおよび Oracle8i へ移行したテスト・データベースに対して、計画したテストを実行します。結果を比較し、相違点を記録します。必要に応じて移行を繰り返します。

移行された Oracle8i のデータベースで既存のアプリケーションが正常に動作するか確認するために、この新たに移行された Oracle8i のテスト・データベースをテストします。また、利用可能な Oracle8i の機能を使用するための機能を追加して、機能強化についてもテストします。ただし、アプリケーションがソース・データベースでの動作状況と同様に動作するかを、最初に確認してください。

参照： Oracle8i におけるアプリケーションの使用方法の詳細は、[第 9 章「アプリケーションのアップグレード」](#)を参照してください。

移行ユーティリティによる移行

この章では、移行ユーティリティを使用して Oracle7 のデータベースを Oracle8i へ移行する処理について説明します。この章で説明するトピックは、次のとおりです。

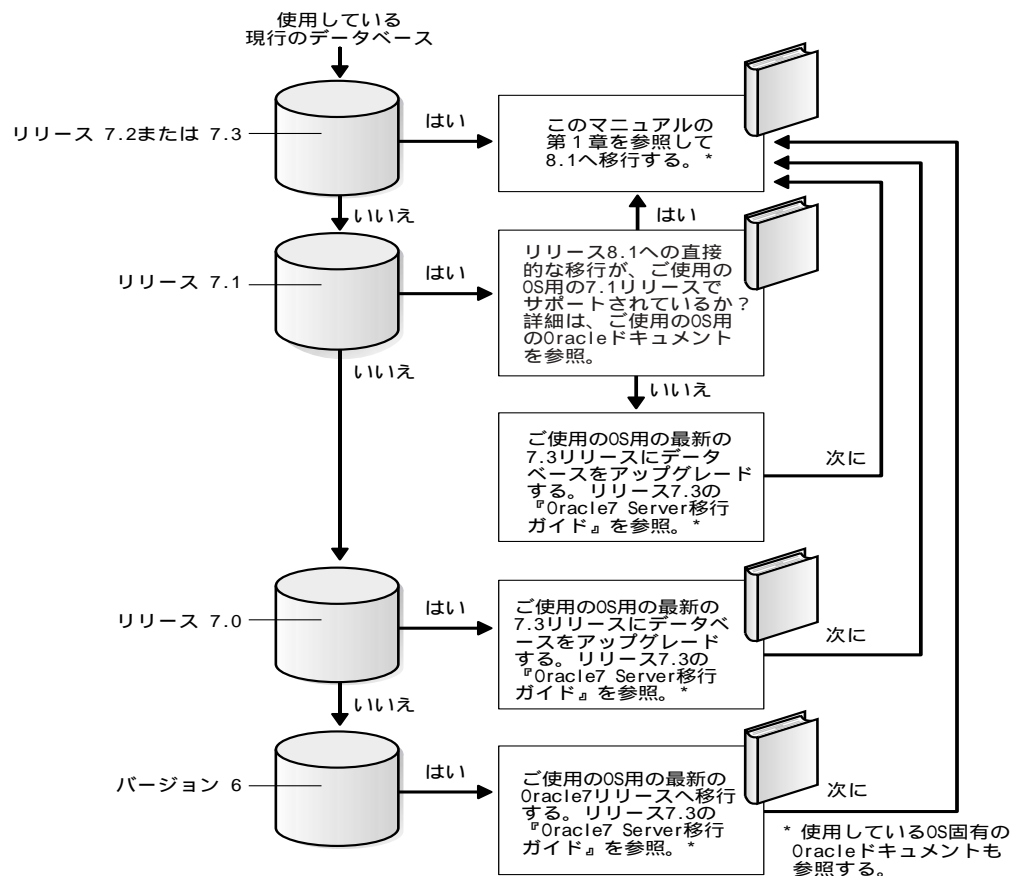
- [移行ユーティリティを使用するためのドキュメント・ロードマップ](#)
- [移行ユーティリティを使用した移行の概要](#)
- [システムの考慮事項と要件](#)
- [移行のための Oracle7 ソース・データベースの準備](#)
- [リリース 8.1 Oracle ソフトウェアのインストール](#)
- [移行ユーティリティ・コマンド行オプションの検討](#)
- [Oracle7 のソース・データベースの移行](#)
- [移行時のエラーのトラブルシューティング](#)
- [移行の中止](#)

参照： 移行処理の一部は、オペレーティング・システム固有のものです。移行の詳細は、ご使用のオペレーティング・システム固有の Oracle ドキュメントを参照してください。

移行ユーティリティを使用するためのドキュメント・ロードマップ

図 3-1 は、ご使用の現行リリースの Oracle に基づいて、データベースをリリース 8.1 へ移行するために使用するドキュメントを示すロードマップです。

図 3-1 移行ユーティリティを使用するためのドキュメント・ロードマップ



移行ユーティリティを使用した移行の概要

移行すると、Oracle7 データベースのデータ・ディクショナリまたは構造が Oracle8i 形式に変換されます。データベースを移行するには、まず Oracle8i ソフトウェアをインストールし、migpprep スクリプトを実行して Oracle7 環境へ移行ファイルをコピーします。次に、Oracle7 データベースで移行ユーティリティを実行します。そして、新しい Oracle8i のデータベースに対して一連の ALTER DATABASE コマンドを実行し、u0703040.sql 変換スクリプトを実行します。

これらの処理の結果として、次のような Oracle7 の構造体が、Oracle8i で使用できる構造体に変換されます。

- データ・ファイル（ファイル・ヘッダーのみ）
- データ・ディクショナリ
- 制御ファイル
- ロールバック・セグメント

移行処理の概要

以下のセクションでは、移行処理の概要について説明します。

Oracle7 環境での移行処理

- Oracle8i の移行ユーティリティを実行します。この移行ユーティリティは、Oracle7 のデータベースのデータ・ディクショナリに基づいて新しいデータ・ディクショナリを作成およびデータ移入し、さらに Oracle7 のデータベースの制御ファイルに基づいて 1 つのバイナリ・ファイルを作成します。このバイナリ・ファイルを、変換ファイルと呼びます。

注意： Oracle8i でデータベースをオープンしなくても、Oracle8i の移行ユーティリティを複数回実行できます。それでも、Oracle7 のデータベースへ戻れなくなることはありません。ただし、移行ユーティリティを実行すると Oracle7 データベースのカatalog・ビューが自動的に削除されます（3-30 ページの「[移行の中止](#)」を参照）。

Oracle8i 環境での移行処理

- ALTER DATABASE CONVERT コマンドを実行します。このコマンドは、移行ユーティリティが生成した変換ファイルに基づいて新しい制御ファイルを作成し、すべてのオンライン・データ・ファイル・ヘッダーを Oracle8i 形式に変換し、Oracle8i のデータベースをマウントします。

オフライン・データ・ファイルおよび読取り専用表領域のファイル・ヘッダーが、移行処理中に更新されることはありません。オフライン・データ・ファイルのファイル・ヘッダーはファイルがオンラインになったときに変換され、読取り専用表領域のファイル・ヘッダーは、移行後に読取り / 書込みになったときに変換されます。ただし、必ず読取り / 書込みになるとは限りません。

- ALTER DATABASE OPEN RESETLOGS 文を実行します。この文は、新しいディクショナリ内で定義されているすべてのオブジェクトとユーザーを Oracle8i の仕様に自動的に変換し、すべてのロールバック・セグメントを Oracle8i 形式に変換します。

Oracle8i のデータベースをオープンするときに、オフライン状態の表領域にソース・データベースのロールバック・セグメントがある場合、そのロールバック・セグメントは Oracle8i のデータベース形式にすぐには変換されません。そのかわり、表領域が Oracle8i で最初にオンラインになるときに変換されます。

- データベースの変換スクリプト実行します。主な変換スクリプトは u0703040.sql スクリプトです。このスクリプトは、特定のシステム表を作成および変更し、MIGRATE ユーザーを削除します。また、システム・カタログ・ビューおよび PL/SQL の使用に必要なすべてのパッケージを作成する、catalog.sql および catproc.sql を実行します。

その他の変換スクリプトは、特定のコンポーネントを現在のリリースへ変換するために必要な処理を実行します。たとえば、catrep.sql スクリプトは Advance Replication 用の変換スクリプトの 1 つです。

移行ユーティリティの使用

このセクションでは、移行ユーティリティを使用する際の重要な考慮事項について説明します。

移行ユーティリティがサポートする Oracle7 データベースからの開始

バージョン 6 データベースは、Oracle8i への移行の前に、少なくとも Oracle7 へ移行される必要があります。また、移行ユーティリティはいくつかの Oracle7 リリースを移行できません。使用しているプラットフォーム・システムで移行ユーティリティによってサポートされている最初のリリースについては、オペレーティング・システム固有の Oracle ドキュメントを参照してください。

たとえば、移行ユーティリティでは、オペレーティング・システムによってはリリース 7.1.4 以降のデータベースのみ移行でき、リリース 7.1.4 以前のリリース（バージョン 7.0、リリース 7.1.3 など）を移行できません。データベースのリリース番号が、使用しているオペレーティング・システムで移行ユーティリティがサポートするリリースよりも低い場合は、データベースを必要なリリースまでアップグレードまたは移行してください。

参照： データベースを必要なリリースまで移行またはアップグレードする方法については、『Oracle7 Server 移行ガイド リリース 7.3』を参照してください。その後で、この『Oracle8i 移行ガイド』を使用して Oracle8i へ移行します。

ダウングレード

ダウングレードは、既存の Oracle データベースを以前のバージョンまたはリリースに変換する処理です。移行ユーティリティでは、Oracle8i のデータベースを Oracle7 へ戻せません。状況によっては、エクスポート / インポート、バックアップからの復元およびその他の機能を使用してダウングレードすることができます。

参照： ダウングレードの情報は、[第 12 章](#)および[第 13 章](#)を参照してください。

システムの考慮事項と要件

次の項では、移行ユーティリティを使用する際のシステムの考慮事項と要件について説明します。

領域要件

Oracle8i のバイナリ・ファイルは、Oracle7 の場合の 3 倍のディスク領域を必要とすることがあります。これによって、移行処理中にディスク領域がなくなることがあります。ただし、移行ユーティリティでは、一時領域は比較的少量で済みます。十分な領域を必要とするのは、既存の Oracle7 のデータ・ディクショナリと同時に新しい Oracle8i のデータ・ディクショナリを保持するための SYSTEM 表領域のみです。

Oracle データ・ディクショナリを保持するのに必要なディスク領域は、データベース内のオブジェクトの数によって異なります。通常、新しい Oracle8i のデータ・ディクショナリは、Oracle7 の 2 倍のソース・データ・ディクショナリが必要です。必要ならば、SYSTEM 表領域にディスク領域を追加してください。

また、変換スクリプト (u0703040.sql スクリプトなど) を実行して移行を完了するには、SYSTEM 表領域またはロールバック・セグメントにより多くの領域が必要になることがあります。領域が不足していると、変換スクリプトの実行時に「unable to extend」という警告が表示されます。変換スクリプトの実行に必要な領域の実際の量は、データベース内のオブジェクトの数によって異なります。変換スクリプトの実行時に「unable to extend」という警告が表示された場合、SYSTEM 表領域またはロールバック・セグメントを拡張した後、スクリプトを再実行します。

ブロック・サイズの考慮事項

Oracle7 のデータベースおよび移行した Oracle8i のデータベース内の DB_BLOCK_SIZE (init.ora ファイル内のパラメータ) の値は同じである必要があります。Oracle8i での、最小ブロック・サイズは 2048 バイト (2KB) です。この値より大きい場合は、オペレーティング・システムの物理ブロック・サイズの整数倍にします。ただし、2KB の倍数、特に 2 乗 (つまり、2KB、4KB、8KB、16KB など) が最も堅牢な操作を実現します。

Oracle8i のブロック・サイズ設定が次の基準を満たすようにしてください。

- Oracle7 の設定と一致している。
- 最低でも 2048 バイト (2KB) である。Oracle7 のブロック・サイズが 2KB より小さい場合は、Oracle8i の移行ユーティリティによってエラー・メッセージが表示されます。
- オペレーティング・システムの物理ブロック・サイズの整数倍である。できれば 2KB の倍数にします。

レプリケーション環境のための考慮事項

Oracle7 のレプリケーション環境は、Oracle8i へ移行できます。Oracle7 のサイトは、レプリケーション環境内でバージョン 8 のサイトと共存し、正常に実行できます。ただし、各システムにインプリメントされているさまざまなレプリケーション機能に適応するように、特別の注意を払ってください。データベースの異なるバージョンの共存のサポートは、Oracle Parallel Server ごとにオペレーティング・システムで固有です。

参照： レプリケーション機能を使用したシステムの移行に関する詳細は、『Oracle8i レプリケーション・ガイド』の付録 B「移行および互換性」を参照してください。

Oracle Parallel Server をインストールしたシステムの移行

Oracle Parallel Server をインストールしたシステムを移行する場合は、この章で説明されている処理のほとんどをシステムの 1 つのノードのみに実行する必要があります。したがって、特定のステップに特に断りのない限り、この章に説明されている処理を 1 つのノードのみで実行してください。

異なるオペレーティング・システムへの移行

Oracle8i の移行ユーティリティでは、オペレーティング・システムが異なるコンピュータ・システムにデータベースを移行することはできません。たとえば、Solaris 上の Oracle7 のデータベースを Windows NT 上の Oracle8i に変換できません。ただし、エクスポート/インポートを使用すれば、異なるオペレーティング・システムへのデータベースの移行が可能です。

注意： リリース 8.1 以降では、移行処理の際、ワード・サイズの変更はサポートされています。ワード・サイズの変更には、同じオペレーティング・システム内で 32 ビットから 64 ビット・アーキテクチャへの切替えが伴います。詳細は、1-4 ページの「[ワード・サイズの変更](#)」を参照してください。

キャラクタ・セットに関する考慮事項

移行ユーティリティを使用して移行している間は、キャラクタ・セットを変更できません。つまり、Oracle7 ソース・データベースと移行済み Oracle8i データベースは同じキャラクタ・セットを持っている必要があります。Oracle8i のデータベースのすべての文字データは、データベースを作成した CREATE DATABASE コマンドで指定したキャラクタ・セットになっています。

ただし、エクスポート / インポートを完全に実行すれば、キャラクタ・セットを変更できます。また、ALTER DATABASE [NATIONAL] CHARACTER SET 文を使用してキャラクタ・セットを変更できますが、これは、新規のキャラクタ・セットが既存のキャラクタ・セットの真のスーパーセットの場合のみです。

参照： 各国語サポート (NLS) の詳細、キャラクタ・セットの指定方法および ALTER DATABASE [NATIONAL] CHARACTER SET 文で使用できるキャラクタ・セットの全リストは、『Oracle8i NLS ガイド』を参照してください。

移行のための Oracle7 ソース・データベースの準備

Oracle7 のデータベースを Oracle8i へ移行する前に、次のステップを実行してください。

1. データベースのリリース番号が、使用しているオペレーティング・システムで移行ユーティリティがサポートするリリースよりも低い場合は、データベースをサポートされるリリースまでアップグレードまたは移行します。

参照： 詳細は、3-5 ページの「[移行ユーティリティがサポートする Oracle7 データベースからの開始](#)」を参照してください。

2. Oracle7 データベースの完全なバックアップを行います。

参照： Oracle7 データベースのバックアップの詳細は、『Oracle7 Server 管理者ガイド』を参照してください。

3. プロシージャ・オプションがインストールされていなければ、Oracle7 のインストール・メディアを使用してインストールします。詳細は、ご使用のオペレーティング・システム固有の Oracle ドキュメントを参照してください。

プロシージャ・オプションがインストールされているかどうかを判断できない場合は、Server Manager を起動して確認できます。次に、Server Manager を起動したときに表示されるメッセージを示します。

Oracle Server Manager Release 2.3.3.0.0 - Production

Copyright (c) Oracle Corporation 1994, 1995. All rights reserved.

Oracle7 Server Release 7.3.4.0.0 - Production

With the distributed, replication, parallel query, Parallel Server
and Spatial Data options

PL/SQL Release 2.3.4.0.0 - Production

表示されるメッセージは、インストールしたオプションおよびリリース番号によって少し異なる場合があります。前述の列の最終行にあるように、メッセージ中に「PL/SQL」が表示される場合は、プロシージャ・オプションがインストールされています。「PL/SQL」が表示されない場合は、インストールされていません。

4. すべてのデータ・ファイルおよび表領域が正常にオンラインまたはオフラインであることを確認します。

データ・ファイルに回復が必要かどうかを判断するには、次の SQL 文を発行します。

```
SELECT * FROM v$recover_file;
```

「0 rows selected」というメッセージが表示されます。これは、すべてのデータ・ファイルが正常にオンラインまたはオフラインであることを示します。1 つでもデータ・ファイルがリストされている場合は、データベースを移行する前にそのデータ・ファイルを回復する必要があります。VSDATAFILE 動的パフォーマンス・ビューを使用して、データ・ファイル番号に基づくデータ・ファイル名を見つけることができます。メディア回復を必要とするデータ・ファイルが存在する場合、Oracle8i の移行ユーティリティは、先へ進まずにエラーを表示します。

確実にオフラインにできなかった表領域は、移行前に削除するかオンラインにする必要があります。これは、そうしないと移行後に Oracle8i で表領域を使用できないからです。通常、ALTER TABLESPACE OFFLINE IMMEDIATE または ALTER TABLESPACE OFFLINE TEMPORARY コマンドを使用してオフラインにする表領域には、メディア回復が必要です。

移行後、Oracle8i のデータベースをオープンするときにオフラインになっている表領域は、Oracle7 のデータベース・ファイル形式のまま残ります。オフラインの表領域は、移行後ならいつでもオンラインにできますし、ファイル・ヘッダーはそのときに Oracle8i 形式に変換されます。さらに、障害が発生した場合に大規模な復元を避けたいければ、SYSTEM と ROLLBACK を除くすべての表領域をオフラインにし、別の移行処理を実行する必要がある場合はその後で SYSTEM と ROLLBACK のデータ・ファイルのみを復元することができます。

5. OUTLN という名前のユーザーまたはロールがないことを確認します。これは、Oracle8i をインストールするときに、このスキーマが自動的に作成されるためです。OUTLN という名前のユーザーまたはロールがある場合、そのユーザーまたはロールを削除し、別の名前で再作成する必要があります。

OUTLN という名前のユーザーをチェックするには、次の SQL 文を発行します。

```
SELECT username FROM dba_users WHERE username = 'OUTLN';
```

OUTLN という名前のユーザーがない場合、どの行も選択されません。

OUTLN という名前のロールをチェックするには、次の SQL 文を発行します。

```
SELECT role FROM dba_roles WHERE role = 'OUTLN';
```

OUTLN という名前のロールがない場合、どの行も選択されません。

6. MIGRATE という名前のユーザーまたはロールがないことを確認します。これは、Oracle8i の移行ユーティリティがこのスキーマを作成し、既存のユーザーまたはロールをこの名前で置き換え、最終的にはそれをシステムから削除するからです。

MIGRATE という名前のユーザーをチェックするには、次の SQL 文を発行します。

```
SELECT username FROM dba_users WHERE username = 'MIGRATE';
```

MIGRATE という名前のユーザーがない場合、どの行も選択されません。

MIGRATE という名前のロールをチェックするには、次の SQL 文を発行します。

```
SELECT role FROM dba_roles WHERE role = 'MIGRATE';
```

MIGRATE という名前のロールがない場合、どの行も選択されません。

7. SYSTEM ロールバック・セグメントに OPTIMAL 設定がないようにします。OPTIMAL 設定により、移行中にエラーが発生することがあります。

SYSTEM ロールバック・セグメントの OPTIMAL 設定をチェックするには、次の SQL 文を発行します。

```
SELECT a.usn, a.name, b.optsize
FROM v$rollname a, v$rollstat b
WHERE a.usn = b.usn AND name = 'SYSTEM';
```

次のように表示されます。

USN	NAME	OPTSIZE
0	SYSTEM	

1 row selected.

OPTSIZE 列に値がある場合は、NULL に設定するために次の SQL 文を発行します。

```
ALTER ROLLBACK SEGMENT SYSTEM STORAGE (OPTIMAL NULL);
```

移行が完了したときに、OPTIMAL をリセットできます。

参照： A-4 ページの「[SYSTEM ロールバック・セグメントの OPTIMAL 設定](#)」にあるトラブルシューティング情報を参照してください。

8. ALTER ROLLBACK SEGMENT 文の STORAGE 句にある MAXEXTENTS パラメーターを変更して、SYSTEM ロールバック・セグメントの最大エクステント数を増やします (オプション)。

次に、ALTER ROLLBACK SEGMENT 文の例を示します。

```
ALTER ROLLBACK SEGMENT system
STORAGE (MAXEXTENTS 121);
```

正常に移行するには、SYSTEM ロールバック・セグメントにより多くの領域が必要な場合があります。SYSTEM ロールバック・セグメントに十分な領域がない場合は、Oracle7 環境で移行ユーティリティ実行中にエラーが発生する場合があります。

参照： Windows NT で Oracle Parallel Server を移行している場合の Operating System Dependent (OSD) レイヤーの詳細は、『Oracle Parallel Server for Windows NT スタート・ガイド』を参照してください。

リリース 8.1 Oracle ソフトウェアのインストール

リリース 8.1 ソフトウェアをインストールするには、次のステップを実行します。

1. ご使用のオペレーティング・システム固有の Oracle ドキュメントの指示に従って、Oracle Universal Installer のインストールおよび起動する準備を行います。

Oracle Parallel Server をインストールしたシステムを移行する場合は、インストールの詳細について、『Oracle8i Parallel Server セットアップおよび構成ガイド』を参照してください。

2. Oracle Universal Installer の「Welcome」画面で、「Next」をクリックします。

ある画面でヘルプが必要な場合や Oracle Universal Installer のドキュメントについて知りたい場合は、「Help」ボタンをクリックしてオンライン・ヘルプをオープンします。

3. 「File Locations」画面で、「Destination」がリリース 8.1 Oracle ホームへの絶対パスであることを確認します。その後、「Next」をクリックします。

注意： 新しい Oracle8.1 リリースを Oracle7 リリースとは別に、Oracle ホームにインストールする必要があります。

4. 「Available Products」画面で、Oracle8i Enterprise Edition または Oracle8i から、インストールする Oracle8i のエディションを選択します。その後、「Next」をクリックします。

注意： Oracle Parallel Server オプションは Oracle8i Enterprise Edition でだけ使用できます。

5. 「Installation Types」画面で、「Custom」または「Minimal」を選択します。Oracle ソフトウェアと一緒に開始データベースをインストールする場合を除いては、「Typical」を選択しないでください。「Custom」または「Minimal」を選択すると、開始データベースのインストールを回避できます。通常、既存のデータベースを移行する場合は、開始データベースをインストールしないでください。

注意： Oracle Parallel Server に対して、Minimal はサポートされていません。

選択後、「Next」をクリックします。

「Custom」を選択した場合、「Upgrading or Migrating an Existing Database」画面が表示されるまで、カスタム・インストール設定の指定ができる画面に従ってください。

特定のオプションの使用を中止しない場合、Oracle7 データベースにインストールしていたオプションすべてをインストールすることを確認します。たとえば、Oracle7 にアドバンスド・レプリケーションをインストールしていた場合、それを Oracle8i にインストールする必要があります。

6. Oracle Parallel Server をインストールする場合、「Cluster Node Selection screen」画面で、ソフトウェアをインストールするノードを選択します。その後、「Next」をクリックします。
7. 「Upgrading or Migrating an Existing Database」画面で、「Upgrade or Migrate an Existing Database」チェックボックスをチェックしないでください。その後、「Next」をクリックします。

「Upgrade or Migrate an Existing Database」チェック・ボックスをチェックすると、インストール後、Oracle Data Migration Assistant が自動的に起動します。移行ユーティリティを使用したデータベースの移行の指示に従っているので、Oracle Data Migration Assistant を起動してはいけません。

注意： Oracle Data Migration Assistant は、Oracle Parallel Server の移行をサポートしません。

8. 「Create Database」画面で、「No」ラジオ・ボタンを選択します。それによって、既存のデータベースを移行中であるため、データベースを作成しないことを示します。その後、「Next」をクリックします。
9. 「Summary」画面で、インストールの設定および選択が正しいかどうかを確認します。その後、「Install」をクリックします。これによって、Oracle Universal Installer がインストールを実行します。ある程度時間がかかる場合があります。

インストールが正常に完了したら、「Exit」ボタンをクリックして「Universal Installer」をクローズします。

参照： Windows NT で Oracle Parallel Server を移行する場合、Operating System Dependent (OSD) レイヤーの詳細は、『Oracle Parallel Server Getting Started for Windows NT』を参照してください。

移行ユーティリティ・コマンド行オプションの検討

移行処理の次の作業は、Oracle8iの移行ユーティリティを実行することです。作業を始める前に、次の移行ユーティリティのコマンド行オプションを検討してください。移行処理の中でその一部を使用する場合があるからです。また、移行ユーティリティのコマンド行オプションに関する詳細が、オペレーティング・システムに固有の Oracle ドキュメントに記載されていることがあります。

CHECK_ONLY	TRUE にすると、移行ユーティリティは、移行を実行せずに領域使用率を計算します。FALSE にすると、移行ユーティリティは領域使用率の計算と移行の両方を実行します。このコマンド行オプションは、NO_SPACE_CHECK と相互に排他的です。
DBNAME	移行するデータベース名 (<code>init^{sid}.ora</code> ファイル内の <code>DB_NAME</code>) を指定します。
MULTIPLIER	Oracle8i の <code>i_file#_block#</code> 索引の初期サイズを、Oracle7 の <code>i_file#_block#</code> 索引との比例で指定します。たとえば、 <code>MULTIPLIER=30</code> とすると、索引が作成されたときの初期サイズが 3 倍になります。 <code>MULTIPLIER</code> コマンド行オプションを指定しない場合、移行ユーティリティは <code>i_file#_block#</code> 値 15 を使用し、Oracle8i 用に作成される <code>i_file#_block#</code> 索引は、Oracle7 の <code>i_file#_block#</code> 索引よりも 1.5 倍大きくなります。
NEW_DBNAME	移行したデータベースに新しい名前を指定します。デフォルト名に「 <code>DEFAULT</code> 」は使用しないでください。より意味のある名前を選んでください。
NLS_NCHAR	Oracle8i のデータベースの <code>PROPS\$</code> に各国語サポート (NLS) <code>NCHAR</code> キャラクタ・セットを指定します (たとえば、 <code>W52DEC</code> または <code>US7ASCII</code>)。 <code>NLS_NCHAR</code> オプションを指定しなければ、移行ユーティリティは Oracle7 のデータベースのキャラクタ・セットを使用します。
NO_SPACE_CHECK	TRUE にすると、移行ユーティリティは移行前に領域使用率チェックを行いません。FALSE にすると、移行ユーティリティは移行前に領域使用率チェックを行います。このコマンド行オプションは、 <code>CHECK_ONLY</code> と相互に排他的です。

PFILE	<p>初期化パラメータ・ファイルの名前を指定します。PFILE コマンド行 オプションを指定しない場合、移行ユーティリティはデフォルトの <code>init.ora</code> ファイルを使用します。</p> <p>注意: UNIX 上では、次の例のように <code>"\"</code> 記号でエスケープされた二重引用符でパス名を囲む必要があります。</p> <pre>mig PFILE=\"/tmp/mig/pfile\"</pre>
SPOOL	<p>出力をスプールするファイルのファイル名を指定します。</p> <p>注意: UNIX 上では、次の例のように <code>"\"</code> 記号でエスケープされた二重引用符でパス名を囲む必要があります。</p> <pre>mig SPOOL=\"/tmp/mig/spool\"</pre>

Oracle7 のソース・データベースの移行

移行ユーティリティを使用して Oracle7 のソース・データベースを Oracle8i へ移行するには、以下のセクションに示すステップを実行してください。

移行に対する Oracle7 の環境の準備

`migprep` ユーティリティは、必要な移行ファイルを Oracle8i Oracle ホームから Oracle7 Oracle ホームへコピーして、移行のために Oracle7 の環境を整えます。新しいリリース Oracle8.1 Oracle ホームを示す環境変数を使用して、次の方法で `migprep` を実行します。

```
migprep new_oracle_home old_oracle_home
```

`new_oracle_home` は、新しい Oracle8i Oracle ホーム・ディレクトリの絶対パスです。`old_oracle_home` は、古い Oracle7 Oracle ホーム・ディレクトリの絶対パスです。

たとえば、UNIX で、新しい Oracle8i Oracle ホームが `/oracle/product/8.1` で、古い Oracle7 Oracle ホームが `/oracle/product/7.3` である場合、次のように入力して、`migprep` を実行します。

```
migprep /oracle/product/8.1 /oracle/product/7.3
```

Windows NT で、新しい Oracle8i Oracle ホームが E ドライブの `¥oracle¥product¥8.1` で、古い Oracle7 Oracle ホームが D ドライブの `¥oracle¥product¥7.3` である場合、次のように入力して、`migprep` を実行します。

```
migprep e:¥oracle¥product¥8.1 d:¥oracle¥product¥7.3
```

Oracle7 の環境での移行ステップ

Oracle7 の環境では、次の移行ステップを実行してください。

1. 次の環境変数が、Oracle7 のディレクトリを指定することを確認します。

- ORACLE_HOME
- PATH
- LD_LIBRARY_PATH
- ORA_NLS
- ORACLE_BASE
- ORACLE_PATH (設定されている場合)

注意： これらの環境変数の中には、ご使用のオペレーティング・システムへ適用できないものもあります。

注意： Oracle Parallel Server に対して、すべてのノードでこのステップを実行します。

2. オペレーティング・システム固有のドキュメントを参照し、作成する Oracle8i の新しいデータベース・オブジェクトの配置を制御するシステム設定を確認します。

次の例は、オペレーティング・システム固有の変数の設定を示します。

- UNIX システムでは、TWO_TASK 環境変数が設定解除されている必要があります。
- Windows NT システムでは、TWO_TASK が設定状態のままである必要があります。
- VMS では、ORA_DFLT_HOSTSTR 論理変数が設定解除されている必要があります。

注意： Oracle Parallel Server に対して、すべてのノードでこのステップを実行します。

3. ORA_NLS33 環境変数をご使用の Oracle7 環境の次のディレクトリに設定します。

`$ORACLE_HOME/migrate/nls/admin/data`

4. NLS_LANG 環境変数が、ご使用のデータベースで使用するキャラクタ・セットに設定されていることを確認します。

ご使用のキャラクタ・セットを確認するために、次の SQL 文を発行します。

```
SELECT * FROM v$nls_parameters
WHERE parameter = 'NLS_LANGUAGE'
OR parameter = 'NLS_TERRITORY'
OR parameter = 'NLS_CHARACTERSET';
```

この問合せが戻す 3 つの値すべてを使用して、NLS_LANG を設定します。たとえば、ご使用の NLS_LANGUAGE が AMERICAN、NLS_TERRITORY が AMERICA および NLS_CHARACTERSET が WE8ISO8859P1 の場合、NLS_LANG を次のように設定します。

```
AMERICAN_AMERICA.WE8ISO8859P1
```

参照： NLS_LANG の詳細は、『Oracle8i NLS ガイド』を参照してください。

5. Oracle8i の移行ユーティリティを実行するのに必要な DBA 権限を持っていることを確認します。

DBA 権限を持っていることを確認するには、DBA_ROLE_PRIVS 静的データ・ディクショナリ・ビューを問い合わせます。たとえば、ユーザー SYSTEM として接続している場合、次の SQL 文を入力します。

```
SELECT * FROM dba_role_privs WHERE grantee = 'SYSTEM';
```

DBA がユーザーの GRANTED_ROLE 列にリストされている場合、DBA 権限を持っています。DBA 権限がない場合、DBA 権限を持つユーザーとして接続します。

6. RESTRICTED SESSION 権限を持つ他の DBA が、移行ユーティリティの実行中にデータベースへ接続しないようにします。また、通常のユーザーは移行処理中にデータベースへ接続することはできません。
7. SHUTDOWN NORMAL または SHUTDOWN IMMEDIATE コマンドを使用して、Oracle7 のデータベースを完全に停止します。SHUTDOWN ABORT は使用しないでください。Oracle7 のソース・データベースは、完全に停止する必要があります。これによって、REDO 情報やコミットされていないトランザクションは残りません。

```
SVRMGR> SHUTDOWN IMMEDIATE
```

Oracle Parallel Server を使用している場合、インスタンスを完全に停止します。

注意： 移行を開始する前に Oracle7 のデータベースが停止していない場合は、移行ユーティリティが停止してエラー・メッセージが表示されません。

8. SYSTEM 表領域に十分な領域があることを確認します (オプション)。

一般的な移行問題は、移行中に SYSTEM 表領域の領域を使用し切ってしまうことです。十分なディスク領域が SYSTEM 表領域に割り当てられていないと、移行ユーティリティは移行処理を完了しません。システム・プロンプトで次のように入力して、移行を成功させるために必要なディスク領域を調べるために、CHECK_ONLY コマンド行オプションを TRUE に設定して Oracle8i の移行ユーティリティを実行します。

```
mig CHECK_ONLY=TRUE
```

CHECK_ONLY コマンド行オプションを指定すると、移行ユーティリティは、移行に必要なディスク領域を算定し、利用可能な領域の大きさをチェックし、ディスク領域の要件に関する情報メッセージを発行します。CHECK_ONLY コマンド行オプションを TRUE に設定すると、移行ユーティリティは、Oracle8i のデータ・ディクショナリを構築したりその他の移行処理を実行したりしません。

CHECK_ONLY コマンド行オプションが SYSTEM 表領域に領域を追加する必要があることを示している場合、次のようなコマンドを発行して、新しいデータ・ファイルのかわりとなる適切なディレクトリ・パス名を追加する必要のある領域の大きさを指定します。

```
ALTER TABLESPACE system
  ADD DATAFILE '/home/user1/mountpoint/oradata/db1/system02.dbf'
  SIZE 50M;
```

SYSTEM 表領域に領域を追加する場合、終了時にデータベースを停止することを忘れないでください。

9. システム・プロンプトで次の移行ユーティリティ・コマンドを入力して、Oracle8i 移行ユーティリティを実行します。

```
mig
```

オペレーティング・システム固有の Oracle ドキュメントに明示的に表記されていない限り、コマンドは mig です。mig のみ入力して移行ユーティリティをデフォルト設定のオプションで実行するか、mig を入力した後に 1 つまたは複数の選択オプションを続けます。

参照： コマンド行オプションの詳細は、3-14 ページの「移行ユーティリティ・コマンド行オプションの検討」を参照してください。

10. 移行ユーティリティの実行後に結果をチェックします。移行ユーティリティは情報メッセージを生成し、`migrate.bsq` スクリプトの実行時にその進行状況をエコーします。移行ユーティリティが ORA エラーで終了する場合、エラーおよび問題解決のために実行する処置の詳細は、付録 A「移行問題のトラブルシューティング」を参照してください。

移行ユーティリティは、Oracle7 の制御ファイルの情報を含む変換ファイルを作成します。後に移行処理の中で、この変換ファイルは Oracle8i の新しい制御ファイルを作成するときに ALTER DATABASE CONVERT によって使用されます。

変換ファイルの名前と位置は、オペレーティング・システムによって異なります。たとえば、UNIX オペレーティング・システムでは、Oracle7 の環境の場合のデフォルト位置は `$ORACLE_HOME/dbs` であり、このディレクトリ内のデフォルト・ファイル名は `convsid.dbf` です。この *sid* は、バージョン 7 のインスタンス ID です。Windows NT では、デフォルト位置は Oracle7 環境の `$ORACLE_HOME\database` であり、このディレクトリ内のデフォルト・ファイル名は `convert.ora` です。

注意： Oracle8i の移行ユーティリティによって停止された Oracle7 のデータベースはオープンしないでください。データ・ファイルのバージョンを確実に統合するには、データベースを Oracle8i へ変換するときに、ディクショナリ内の SCN、変換ファイルおよびファイル・ヘッダーのすべてに一貫性がなければなりません。移行ユーティリティの実行後に Oracle7 のデータベースをオープンすると、SCN のチェックが、データベースを Oracle8i へ変換するときに失敗し、「Oracle7 データ・ファイルは Oracle8 に移行していません。」という ORA-01211 エラーが表示されます。したがって、Oracle7 のデータベースをオープンする場合は、移行ユーティリティを再実行し、ステップ 7 から開始する必要があります。

Oracle7 のソース・データベースの保存

移行ユーティリティが正常に終了したら、Oracle7 のデータベースのコールド・バックアップを取ってください。このバックアップは、次の目的で使用します。

- Oracle8i で ALTER DATABASE CONVERT コマンドを実行後に Oracle7 のデータベースに戻りたくなった場合は、バックアップを復元して Oracle7 のデータベースを開始し、3-30 ページの「**移行の中止**」の手順に従うことができます。
- Oracle8i を回復するための最初の Oracle8i バックアップとして使用できます。
- Oracle8i のデータベースの変換時（ALTER DATABASE CONVERT または ALTER DATABASE OPEN RESETLOGS）にエラーが発生した場合は、このバックアップを復元し、問題を修正して変換処理を継続することができます。ただし、移行ユーティリティの実行前に取ったバックアップを復元する場合は、移行ユーティリティを再実行する必要があります。

参照： Oracle7 データベースでのバックアップの実行および回復操作の詳細は、『Oracle7 Server 管理者ガイド』を参照してください。

さらに、Oracle7 のホーム・ディレクトリを含む、Oracle7 で配布されるソフトウェア全体のバックアップを作成します。そのバックアップには、次のファイル群が必ず含まれるようにしてください。

- すべてのサブディレクトリ
- 制御ファイル
- データ・ファイルとオンライン REDO ログ・ファイル（Oracle7 のデータベースのデータ・ファイルがなくなるか読み取れない場合に備えて）。ただし、これらのファイルには未処理の REDO 情報は含まれていないはずです。
- パラメータ・ファイル
- 変換ファイル
- Oracle7 のデータベースにオブジェクトを作成するスクリプト
- 必要に応じて元のデータベースを復元できるスクリプト

Oracle8i の環境での移行ステップ

Oracle8i の環境では、次の移行ステップを実行してください。

参照： Windows NT で Oracle Parallel Server を移行している場合、Operating System Dependent (OSD) レイヤーの詳細は、『Oracle Parallel Server Getting Started for Windows NT』を参照してください。

1. 次の環境変数が、Oracle8i の実行可能ファイルを指定することを確認します。

- ORACLE_HOME
- PATH
- LD_LIBRARY_PATH
- ORA_NLS
- ORACLE_BASE
- ORACLE_PATH (設定されている場合)

ORACLE_HOME が Oracle7 の実行可能ファイルを指している場合は、移行処理の後に ALTER DATABASE CONVERT コマンドを実行すると、ORA-223 エラー「変換ファイルが無効またはバージョンが正しくありません。」が表示されます。

注意： これらの環境変数の中には、ご使用のオペレーティング・システムへ適用できないものもあります。

注意： Oracle Parallel Server に対して、すべてのノードでこのステップを実行します。

参照： ご使用のオペレーティング・システムでのその他の重要な環境変数の設定は、オペレーティング・システム固有の Oracle8i インストレーション・ドキュメントを参照してください。

2. データベースの制御ファイルを、削除または改名します。または、CONTROL_FILES 初期化パラメータを使用して制御ファイルの新しい名前を指定します。CONTROL_FILES 初期化パラメータは通常 `inittsid.ora` ファイルで設定されますが、Oracle Parallel Server を使用している場合、かわりに `initdb_name.ora` ファイルで設定される場合があります。

ALTER DATABASE CONVERT コマンドによって、新しい制御ファイルが自動的に作成されます。CONTROL_FILES パラメータを使用しない場合は、このコマンドが移行前のデータベース (CONVERT ファイルから派生した) の制御ファイルの名前を使用し、その制御ファイルがすでに存在していたらエラーを戻します。したがって、この場合は、制御ファイルを削除するか改名する必要があります。

ただし、CONTROL_FILES パラメータを使用して新しい制御ファイルを指定する場合は、指定した名前を持つ新しい制御ファイルが ALTER DATABASE CONVERT コマンドによって作成され、古い制御ファイルを削除する必要はありません。

制御ファイルは、Oracle7 よりも Oracle8i の方がかなり大きくなっています。たとえば、Oracle7 で数百キロバイトの制御ファイルは Oracle8i では数十メガバイトに拡大される場合があります。Oracle8i でサイズが大きくなるのは、制御ファイル内にバックアップや表領域レコードなどの情報がさらに記憶されるからです。制御ファイルが RAW デバイス上にある場合、または使用可能なディスク領域が制限されている場合は、このサイズの増大が重要になります。

注意： CONTROL_FILES は、カンマで区切られた 1 つまたは複数の制御ファイルの名前を指定します。オラクル社では、異なる複数のデバイスで複数のファイルを使用するか、オペレーティング・システム・レベルでファイルをミラー化することをお勧めします。詳細は、『Oracle8i 管理者ガイド』を参照してください。

注意： Oracle Parallel Server に対して、すべてのノードでこのステップを実行します。

3. 移行に重要なファイルを Oracle7 Oracle ホーム以外の場所にコピーします。
 - a. 変換ファイルを Oracle7 Oracle ホーム・ディレクトリから Oracle8i Oracle ホーム・ディレクトリへ移動またはコピーします。ほとんどの UNIX オペレーティング・システムでは、変換ファイル `convsid.dbf` (`sid` は Oracle8i データベース名) は、Oracle7 および Oracle8i 環境ともに `$ORACLE_HOME/dbs` の中になります。Windows NT では、変換ファイル `convert.ora` は、Oracle7 および Oracle8i 環境ともに `$ORACLE_HOME¥database` の中になります。
 - b. Oracle7 Oracle ホーム内に存在するパスワード・ファイルがある場合、そのパスワード・ファイルを Oracle7 Oracle ホーム・ディレクトリ以外の場所へ移動またはコピーします。

パスワード・ファイルの名前および位置は、オペレーティング・システムによって異なります。たとえば、UNIX オペレーティング・システムでは、デフォルトのパスワード・ファイルは `$ORACLE_HOME/dbs/orapwsid` ですが、Windows NT では、デフォルトのパスワード・ファイルは `$ORACLE_HOME¥database¥pwdsid.ora` です。両方の場合とも、`sid` はご使用の Oracle インスタンス ID です。
 - c. 使用する `initSID.ora` ファイルが Oracle7 Oracle ホーム内に存在する場合、Oracle ホーム以外の場所へ移動またはコピーします。以前のリリースでは、`initSID.ora` ファイルのデフォルト位置は、UNIX の場合は `$ORACLE_HOME/dbs`、Windows NT の場合は `$ORACLE_HOME¥database` でした。リリース 8.1 では、デフォルト位置は `$ORACLE_BASE/admin/sid/pfile` で、`sid` はご使用の Oracle インスタンス ID です。`initSID.ora` ファイルはどこにでも置けます。ただし、Oracle8i へ移行後は、Oracle7 Oracle ホーム中に置くことはできません。
 - d. `initSID.ora` ファイルが IFILE (インクルード・ファイル) エントリを含み、Oracle7 Oracle ホーム内に存在する場合、IFILE エントリで指定されているファイルを Oracle ホーム以外の場所へ移動またはコピーします。
 - e. Oracle Parallel Server を使用し、ご使用の `initdb_name.ora` ファイルが Oracle7 Oracle ホーム内に存在する場合、`initdb_name.ora` ファイルを Oracle ホーム以外の場所へ移動またはコピーします。

注意： Oracle Parallel Server に対して、すべてのノードでこのステップを実行します。

4. Oracle8i 環境内の `initSID.ora` ファイルを調整して、Oracle8i で使用できるようにします。具体的には、次のステップを実行します。
 - a. ご使用の `initSID.ora` ファイル内の COMPATIBLE 初期化パラメータを 8.0.5 または 8.1.0 などの有効なバージョン 8 設定へ設定します。COMPATIBLE パラメータが Oracle7 のリリースに設定されていないことを確認してください。設定されていると、Oracle8i データベースを開始できず、移行が失敗します。詳細は、8-6 ページの「[COMPATIBLE パラメータの設定](#)」を参照してください。
 - b. 廃止されたパラメータを削除して、変更されたパラメータを調整します。Oracle7 の特定の初期化パラメータは、バージョン 8 では廃止されています。廃止されたパラメータは、Oracle8i のインスタンスを開始する `initSID.ora` ファイルからすべて削除してください。廃止されたパラメータを Oracle8i のデータベースで使用する、エラーになることがあります。また、バージョン 8 で構文が変更されているパラメータを変更することも必要です。新規のパラメータおよび改名されたパラメータ、廃止されたパラメータのリストは、[付録 B「初期化パラメータの変更](#)」を参照してください。また、Oracle Parallel Server を使用している場合には、廃止されている Oracle Parallel Server 初期化パラメータの詳細は、『Oracle8i Parallel Server 概要および管理』を参照してください。
 - c. JOB_QUEUE_PROCESSES 初期化パラメータを使用してスナップショットを自動的に更新する場合は、`initSID.ora` ファイルにあるこのパラメータをコメント行にしてください。データベースを移行した後、このコメントを削除して、このパラメータを通常に使用できます。
 - d. 必要に応じて、LARGE_POOL_SIZE 設定を調節します。Oracle のリリース 8.1 では、LARGE_POOL_SIZE 設定は Oracle によって自動的に計算されます。自動設定が大きすぎる場合、移行に要する時間が長くなることがあります。詳細は、B-9 ページの「[大規模プールから割り当てられたパラレル実行](#)」を参照してください。
 - e. Oracle Parallel Server を使用する場合、PARALLEL_SERVER 初期化パラメータを FALSE に設定します。移行が完了した後に、それを TRUE に戻します。
 - f. UNIX オペレーティング・システムで分散ロック・マネージャ (DLM) を使用する場合、LM_PROCS 初期化パラメータを、Oracle7 で使用される DLM のロック、リソースおよびプロセス・パラメータに等しく設定してくることを確認します。
 - g. DB_DOMAIN 初期化パラメータが正しく設定されていることを確認します。

参照： この初期化パラメータの詳細は、B-9 ページの「[DB_DOMAIN パラメータ](#)」を参照してください。

- h. `initssid.ora` ファイルを新しい場所で、古い Oracle7 Oracle ホームの完全なパスになるようにパス名のクエスチョン・マーク (?) を変更して、編集します。
Oracle ホームの完全パス名が記述されている場合、`initssid.ora` ファイルを編集する必要はありません。`initssid.ora` ファイルは相対的パス名ではなく、完全なパス名のみを持っています。
- i. `initssid.ora` ファイルに `IFILE` エントリがある場合、`initssid.ora` ファイルの `IFILE` エントリを変更して、ステップ 3 のコピー先の新しい場所を指定するようにします。次に、`IFILE` エントリに指定されているファイルをサブステップ a から h までで `initssid.ora` ファイルを編集した方法と同じように編集します。
- j. Oracle Parallel Server を使用している場合、`initdb_name.ora` ファイルをステップ a から h までで `initssid.ora` ファイルを変更した方法と同じように変更します。

これらの調整後、変更したファイルのすべてを保存してください。

注意： Oracle Parallel Server に対して、すべてのノードでこのステップを実行します。

- 5. Oracle8i `DB_NAME` が Oracle7 `DB_NAME` と異なる場合は、次のステップを実行してください。同じ場合は、ステップ 6 へ進みます。
 - a. UNIX オペレーティング・システムで、`convssid.dbf` ファイルを改名して、Oracle8i `DB_NAME` と一致させます。たとえば、Oracle7 `DB_NAME` が `DBMS7` で、Oracle8i `DB_NAME` が `DBMS8` である場合は、変換ファイルを `convDBMS7.dbs` から `convDBMS8.dbs` へ改名します。Windows NT では、この処理は不要です。
 - b. `initssid.ora` ファイル内の `DB_NAME` パラメータを Oracle8i データベース名に設定します。
- 6. すべてのオンライン・データ・ファイルがアクセス可能で、正しいディレクトリに入っていることを確認します。RAW ディスクを使用する場合は、ログ・ファイルもアクセス可能でなければなりません。
- 7. `$ORACLE_HOME/rdbms/admin` ディレクトリへ変更します。現在の位置は Oracle8i Oracle ホームである必要があります。
- 8. Server Manager を起動します。
- 9. データベース・インスタンスへ接続します。

```
SVRMGR> CONNECT INTERNAL
```

10. Oracle8i の新しいデータベースをマウントせずに、Oracle8i のデータベースのインスタンスを開始します。

```
SVRMGR> STARTUP NOMOUNT
```

注意： データベースのインスタンスを他のモードで開始すると、そのデータベースを破損する可能性があります。

PFILE オプションを使用して、使用する `inittssid.ora` ファイルの場所を指定することができます。

廃止されたパラメータをリストするエラー・メッセージが表示されることがあります。その場合には、データベースを停止し、ご使用の `inittssid.ora` ファイルを編集して、リストされるパラメータを削除し、その後 STARTUP NOMOUNT を再度実行します。

11. Oracle8i のデータベースの新しい制御ファイルを作成し、次のコマンドを発行して、すべてのオンライン表領域のファイル・ヘッダーを Oracle8i 形式に変換します。

```
SVRMGR> ALTER DATABASE CONVERT;
```

このコマンドの実行が成功すると、このデータベースについては Oracle7 に戻れなくなります。ただし、必要に応じてバックアップから Oracle7 のデータベースを復元することはできます。

このステップでエラーが発生した場合は、エラーの原因となっている条件を訂正し、移行ユーティリティを再実行してください。3-16 ページのステップ 1 から再度開始します。そうでない場合は、移行ユーティリティを実行した後に実行したバックアップを復元します。

参照： このステップで発生するエラーと問題解決のために実行する処置の詳細は、A-16 ページの「[ALTER DATABASE CONVERT コマンドにおける問題](#)」を参照してください。

12. 次のコマンドで Oracle8i データベースをオープンします。

```
SVRMGR> ALTER DATABASE OPEN RESETLOGS;
```

Oracle8i のデータベースがオープンされているときは、オンライン状態のロールバック・セグメントもすべて新しい Oracle8i 形式に変換されます。

このコマンドを発行するときエラーが発生する場合は、始めから移行処理をやり直し、移行ユーティリティ実行後にデータベースが Oracle7 環境でオープンしていないことを確認してください。この章の最初から開始してください。ただし、[第 2 章](#)で説明されている移行前のステップを完了していることを確認してください。

13. 成功したかどうかを後で確認するために、結果をログ・ファイルにスプールするようにシステムを設定します。


```
SVRMGR> SPOOL catoutm.log
```

実行するスクリプトの出力を画面で確認する場合は、SET ECHO ON 文の発行も確認できます。

```
SVRMGR> SET ECHO ON
```

14. Oracle8i のデータベースの変換スクリプト u0703040.sql を実行します。

```
SVRMGR> @u0703040.sql
```

u0703040.sql スクリプトは、使用するオペレーティング・システムの移行ユーティリティでサポートする 7.1、7.2 および 7.3 リリースすべてのデータベースの変換スクリプトです。u0703040.sql スクリプトは、特定のシステム表を作成および変更し、MIGRATE ユーザーを削除します。また、システム・カタログ・ビューおよび PL/SQL の使用に必要なすべてのパッケージを作成する、catalog.sql と catproc.sql を実行します。

このスクリプトか、残りのステップのスクリプトのいずれかを実行中に何らかの問題が発生した場合は、問題を解決してスクリプトを再実行してください。必要に応じて、この章に説明されているスクリプトのどれかを実行できます。

参照： スクリプトの実行中のエラーを判断する詳細は、1-3 ページの「[スクリプトの実行](#)」を参照してください。

注意： u0703040.sql スクリプトの実行が極端に長い場合、LARGE_POOL_SIZE の設定がそのインスレーションには大きすぎるため、発生していることがあります。VSPARAMETER ビューを使用して LARGE_POOL_SIZE の設定を確認します。大きすぎるようであれば、initSID.ora ファイルの値を小さくします。詳細は、B-9 ページの「[大規模プールから割り当てられたパラレル実行](#)」を参照してください。

15. Oracle システムにアドバンスド・レプリケーションをインストールしている場合は、次のステップを実行します。

- a. `catrep.sql` を実行します。

```
SVRMGR> @catrep.sql
```

- b. `r0703040.sql` を実行します。

```
SVRMGR> @r0703040.sql
```

この `r0703040.sql` スクリプトは、`catrep.sql` 後のアドバンスド・レプリケーションに関連するアップグレードを実行します。

16. Oracle システムに Oracle Parallel Server をインストールしている場合は、新しいリリースに付属する次のカタログ・スクリプトを実行します。

```
SVRMGR> @catparr.sql
```

17. `utlrlp.sql` を実行します (オプション)。

```
SVRMGR> @utlrlp.sql
```

`utlrlp.sql` スクリプトは、以前 `INVALID` 文中にあったすべての既存の PL/SQL モジュールを再コンパイルします。それらには、パッケージ、プロシージャ、型などがあります。これらの処理はオプションです。ただし、これらの処理によって、再コンパイルの費用を将来ではなくインストレーション中に確認できます。

オラクル社は、このオプションのステップを実行することをお勧めします。

18. スクリプト結果のログ・ファイルへのスプーリングをオフにします。

```
SVRMGR> SPOOL OFF
```

次にスプール・ファイルをチェックして、すべてのパッケージとプロシージャのコンパイルが成功したかどうかを検証してください。ステップ 13 でスプール・ファイルに名前を付けましたが、その名前は `catoutm.log` でした。

領域が不足していることを警告するエラーやスクリプトが実行しないエラーは、調査する必要があります。これらのタイプのエラーが発生する場合、移行は成功していない場合があります。ただし、存在しないオブジェクトの変更や削除の障害についてのエラーは一般に無視できます。

`SET ECHO ON` を指定した場合は、ここで `SET ECHO OFF` を行うことができます。

```
SVRMGR> SET ECHO OFF
```

19. Oracle8i のデータベース上で、SHUTDOWN を実行します。

```
SVRMGR> SHUTDOWN IMMEDIATE
```

注意： SHUTDOWN NORMAL または SHUTDOWN IMMEDIATE を使用してください。SHUTDOWN ABORT は使用しないでください。

この正しい停止を実行すると、すべてのキャッシュをフラッシュし、バッファを消去し、その他の内部的な不整合の解消を行います。これらの手段は、新規に移行した Oracle8i のデータベースの整合性と一貫性を保証するための重要な最終ステップです。

COMPATIBLE 初期化パラメータは、ご使用のデータベースの互換レベルを制御します。移行したデータベース用の互換レベルに基づいて、COMPATIBLE 初期化パラメータをご使用の `initSID.ora` ファイルに設定します。

参照： 詳細は、8-6 ページの「[COMPATIBLE パラメータの設定](#)」を参照してください。

20. 第 6 章「データベースの移行後」に記載の手順を実行します。

注意： 古い Oracle7 ソフトウェアをそのまま残したい場合は、決して古い Oracle7 ソフトウェアを使用して移行したデータベースを開始しないでください。新しい Oracle8i インストレーションの実行可能ファイル以外で、データベースを開始しないでください。

移行時のエラーのトラブルシューティング

エラーが発生する原因としては、次のような操作が行われたか、または省略されたことが考えられます。

- 誤った順序で移行ステップを実行する
- 移行の前提条件を満たしていない
- 変換の例外を検出する

参照： 移行中に発生する可能性のあるエラーとそのエラーの対処措置については、[付録 A「移行問題のトラブルシューティング」](#) および『Oracle8i エラー・メッセージ』を参照してください。

移行の中止

移行ユーティリティを実行する前に使用する Oracle7 データベースのバックアップを行った場合、移行を中止する最も簡単な方法は、バックアップを復元することです。ただし、バックアップを取っていない場合、または移行ユーティリティ実行後にバックアップを行った場合には、移行を中止するためにこの項に説明されている手順を実行する必要があります。

Oracle8i の移行ユーティリティは複数回実行できますが、それでも Oracle7 のデータベースへ戻れなくなることはありません。ただし、移行ユーティリティを実行すると、Oracle7 のデータベースのカatalog・ビューが自動的に削除されます。したがって、移行ユーティリティの実行後に Oracle7 のデータベースに戻るには、Oracle7 の catalog.sql スクリプトを実行して、Oracle7 のデータベースのカatalog・ビューを復元する必要があります。

注意： すでに ALTER DATABASE CONVERT コマンドを実行している場合は、移行を中止するために次の手順は使用できません。このコマンドを実行し Oracle7 へ戻りたい場合は、[第 13 章「Oracle7 へのダウングレード」](#)の手順を実行します。

移行を中止するには、一般的に Oracle7 環境中の次のステップを実行して Oracle7 のデータベースを復元する必要があります。

1. Server Manager を使用して Oracle7 のデータベースを起動します。

2. ユーザー MIGRATE を削除します。

```
SVRMGR> DROP USER MIGRATE CASCADE;
```

3. catalog.sql および catproc.sql を再実行します。

```
SVRMGR> @catalog.sql
```

```
SVRMGR> @catproc.sql
```

4. Server Manager がインストールされている場合は、catsvrmg.sql を実行します。

```
SVRMGR> @catsvrmg.sql
```

5. Parallel Server がインストールされている場合は、catparr.sql を実行します。

```
SVRMGR> @catparr.sql
```

6. アドバンスド・レプリケーションがインストールされている場合は、catrep.sql を実行します。

```
SVRMGR> @catrep.sql
```

注意： Oracle8i の移行ユーティリティは、リリース 7.1 およびリリース 7.2 のデータベースをリリース 7.3 にアップグレードします。元の Oracle7 の本番データベースがリリース 7.1 または 7.2 であり、移行が実行されても Oracle8i への変換前に中止された場合、Oracle7 のデータベースはディクショナリがリリース 7.3 のままになります。ただし、そのような場合、リリース 7.3 からリリース 7.1 または 7.2 へダウングレードする必要はありません。7.1 または 7.2 ソフトウェアは、それ以上の操作を必要としないデータ・ディクショナリで動作します。

Oracle Data Migration Assistant による移行

この章では、Oracle Data Migration Assistant を使用して Oracle7 データベースを Oracle8i へ移行する処理について説明します。この章で説明するトピックは、次のとおりです。

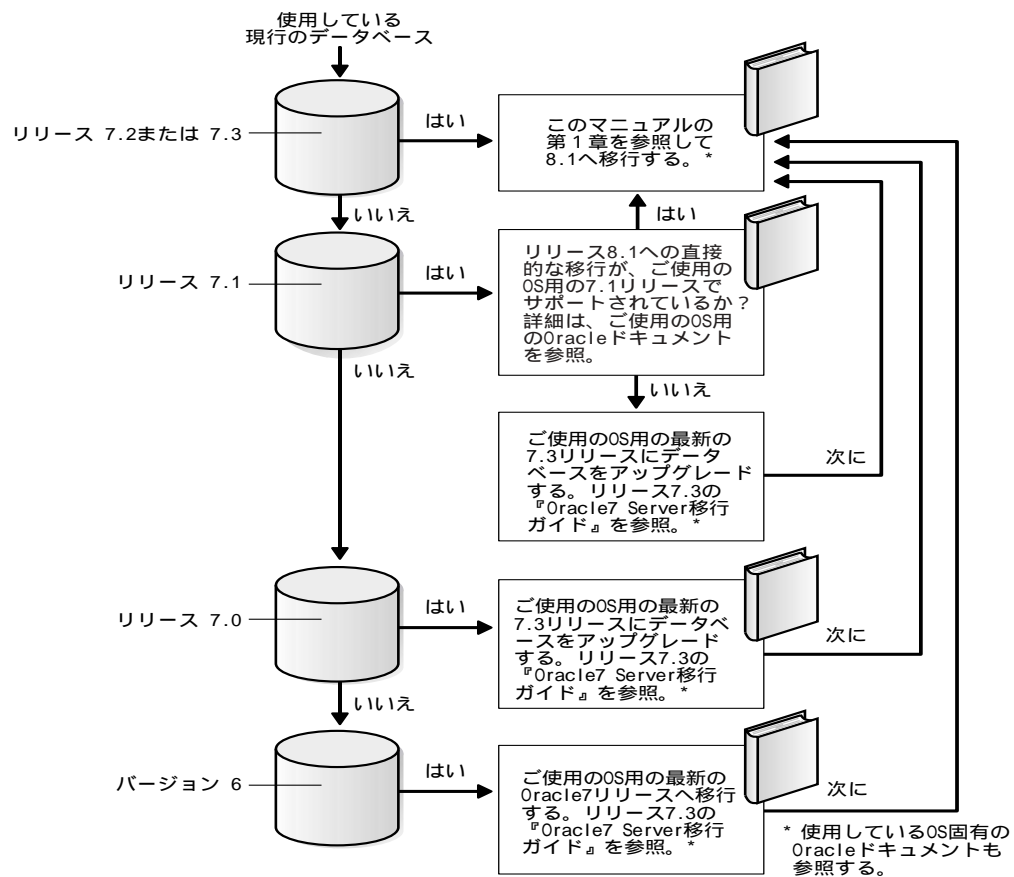
- [Oracle Data Migration Assistant を使用するためのドキュメント・ロードマップ](#)
- [Oracle Data Migration Assistant を使用した移行の概要](#)
- [システムの考慮事項と要件](#)
- [移行のための Oracle7 ソース・データベースの準備](#)
- [Oracle ソフトウェア リリース 8.1 のインストールおよびそのデータベースの移行](#)
- [移行時のトラブルシューティング](#)
- [移行の中止](#)

参照： 移行処理の一部には、オペレーティング・システム固有の部分が
あります。移行の追加情報については、ご使用のオペレーティング・シス
テム固有の Oracle ドキュメントを参照してください。

Oracle Data Migration Assistant を使用するためのドキュメント・ロードマップ

図 4-1 に、ご使用の現行リリースの Oracle に基づいて、データベースをリリース 8.1 に移行するために使用するドキュメントを指定するロードマップを示します。

図 4-1 Oracle Data Migration Assistant を使用するためのドキュメント・ロードマップ



Oracle Data Migration Assistant を使用した移行の概要

この項では、Oracle Data Migration Assistant を使用する際の重要な考慮事項について説明します。

注意： Oracle Data Migration Assistant では、インストールされている Oracle Parallel Server を使用したシステムの移行はサポートされていません。Oracle Parallel Server がすでにインストールされている場合、別の移行方法（移行ユーティリティまたはインポート / エクスポート）を使用してデータベースを移行してください。

Oracle Data Migration Assistant がサポートする Oracle7 データベースからの開始

バージョン 6 のデータベースは、Oracle8i に移行する前に、Oracle7 以降のバージョンに移行する必要があります。また、Oracle7 のリリースには、Oracle Data Migration Assistant では移行できないものがあります。ご使用のオペレーティング・システムにある Oracle Data Migration Assistant がサポートする最初のリリースについての情報は、オペレーティング・システム固有の Oracle ドキュメントを参照してください。

たとえば、オペレーティング・システムによっては、Oracle Data Migration Assistant はリリース 7.1.4 以降のデータベースのみ移行できますが、リリース 7.1.4 以前のリリース（リリース 7.0 またはリリース 7.1.3 など）は移行できません。データベースのリリース番号が、使用しているオペレーティング・システムにある Oracle Data Migration Assistant がサポートするリリースよりも低い場合は、データベースを必要なリリースまでアップグレードまたは移行してください。

参照： データベースを必要なリリースに移行またはアップグレードについては、『Oracle7 Server 移行ガイド リリース 7.3』を参照してください。その後で、この『Oracle8i 移行ガイド』マニュアルを参照して Oracle8i へ移行します。

ダウングレード

ダウングレードとは、既存の Oracle データベースを以前のバージョンまたはリリースに変換する処理です。Oracle Data Migration Assistant では、Oracle8i データベースを Oracle7 へ戻せません。状況によっては、エクスポート / インポート、バックアップからの復元およびその他のファンクションなどの機能を使用してダウングレードできます。

参照： ダウングレードの詳細は、[第 12 章](#)および[第 13 章](#)を参照してください。

システムの考慮事項と要件

次の項では、Oracle Data Migration Assistant を使用する際のシステムの考慮事項と要件について説明します。

領域要件

Oracle8i のバイナリ・ファイルは、Oracle7 の場合の 3 倍のディスク領域を必要とすることがあります。これによって、移行処理中にディスク領域がなくなることがあります。ただし、Oracle Data Migration Assistant では、一時領域は比較的少量で済みます。十分な領域を必要とするのは、既存の Oracle7 のデータ・ディクショナリと同時に新しい Oracle8i のデータ・ディクショナリを保持するための SYSTEM 表領域のみです。

Oracle データ・ディクショナリを保持するのに必要なディスク領域は、データベース内のオブジェクトの数によって異なります。通常、新しい Oracle8i のデータ・ディクショナリは、Oracle7 のソース・データ・ディクショナリの 2 倍の領域が必要です。必要ならば、SYSTEM 表領域にディスク領域を追加してください。十分な領域が SYSTEM 表領域に割り当てられていないと、Oracle Data Migration Assistant は移行処理を完了しません。

SYSTEM 表領域に領域を追加する必要がある場合は、次のようなコマンドを発行して新しいデータ・ファイルのかわりとなる適切なディレクトリ・パス名を追加する必要のある領域の大きさを指定します。

```
ALTER TABLESPACE system
  ADD DATAFILE '/home/user1/mountpoint/oradata/db1/system02.dbf'
  SIZE 50M;
```

制御ファイルは、Oracle7 よりも Oracle8i の方がかなり大きくなっています。たとえば、Oracle7 の数百キロバイトの制御ファイルは、Oracle8i では数十メガバイトのサイズに拡大される場合があります。Oracle8i においてサイズが大きくなるのは、バックアップや表領域レコードなどの情報が制御ファイル内にさらに記憶されるからです。制御ファイルが RAW デバイス上にある場合、または使用可能なディスク領域が制限されている場合は、このサイズの増大が重要になります。

ブロック・サイズの考慮事項

Oracle7 データベースと移行した Oracle8i データベース内の DB_BLOCK_SIZE (INITsid.ora ファイル内のパラメータ) の値は同じでなければなりません。Oracle8i の最小ブロック・サイズは 2048 バイト (2KB) です。この値より大きい場合は、オペレーティング・システムの物理ブロック・サイズの整数倍にします。ただし、2KB の倍数、特に 2 乗 (つまり、2KB、4KB、8KB、16KB など) が最も堅牢な操作を実現します。

Oracle8i のブロック・サイズ設定が次の基準を満たすようにしてください。

- Oracle7 の設定と一致している。
- 最低でも 2048 バイト (2KB) である。Oracle7 のブロック・サイズが 2KB より小さい場合は、Oracle Data Migration Assistant によってエラー・メッセージが表示されます。
- オペレーティング・システムの物理ブロック・サイズの整数倍である。できれば 2KB の倍数にします。

レプリケーション環境のための考慮事項

Oracle7 のレプリケーション環境は、Oracle8i へ移行できます。Oracle7 のサイトは、レプリケーション環境内でバージョン 8 のサイトと共存し、正常に実行できます。ただし、各システムにインプリメントされているさまざまなレプリケーション機能に適応するように、特別の注意を払ってください。

参照： レプリケーション機能を使用したシステムの移行に関する詳細は、『Oracle8i レプリケーション・ガイド』の付録 B「移行および互換性」を参照してください。

異なるオペレーティング・システムへの移行

Oracle Data Migration Assistant では、オペレーティング・システムが異なるコンピュータ・システムにデータベースを移行することはできません。たとえば、Solaris 上の Oracle7 を Windows NT 上の Oracle8i に変換できません。ただし、エクスポート / インポートを使用すると、異なるオペレーティング・システムのデータベースの移行が可能です。

注意： リリース 8.1 以降では、移行処理の際、ワード・サイズの変更はサポートされています。ワード・サイズの変更には、同じオペレーティング・システム内で 32 ビットから 64 ビット・アーキテクチャへの切替えが伴います。詳細は、1-4 ページの「[ワード・サイズの変更](#)」を参照してください。

キャラクタ・セットに関する考慮事項

Oracle Data Migration Assistant を使用して移行している間は、キャラクタ・セットを変更できません。つまり、Oracle7 ソース・データベースおよび移行済み Oracle8i データベースのキャラクタ・セットは同じでなければなりません。Oracle8i のデータベースのすべての文字データは、データベースを作成した CREATE DATABASE コマンドで指定したキャラクタ・セットになっています。

ただし、エクスポート / インポートを完全に実行すれば、キャラクタ・セットを変更できます。また、ALTER DATABASE [NATIONAL] CHARACTER SET 文を使用してキャラクタ・セットを変更することができますが、これは、新規のキャラクタ・セットが既存のキャラクタセットの真のスーパーセットの場合のみです。

参照： 各国語サポート (NLS) の詳細、キャラクタ・セットの指定方法および ALTER DATABASE [NATIONAL] CHARACTER SET 文で使用できるキャラクタ・セットの全リストは、『Oracle8i NLS ガイド』を参照してください。

移行のための Oracle7 ソース・データベースの準備

Oracle7 データベースを Oracle8i へ移行する前に、次のステップを実行してください。

1. データベースのリリース番号が、使用しているオペレーティング・システムで Oracle Data Migration Assistant がサポートするリリースよりも低い場合は、データベースをサポートされているリリースまでアップグレードまたは移行します。

参照： 詳細は、4-3 ページの「[Oracle Data Migration Assistant がサポートする Oracle7 データベースからの開始](#)」を参照してください。

2. Oracle7 データベースの完全なバックアップを行います。

参照： Oracle7 データベースのバックアップの詳細は『Oracle7 Server 管理者ガイド』を参照してください。

3. プロシージャ・オプションがインストールされていなければ、Oracle7 のインストレーション・メディアを使用してインストールします。詳細は、ご使用のオペレーティング・システム固有の Oracle ドキュメントを参照してください。

プロシージャ・オプションがインストールされているかどうかを判断できない場合は、Server Manager を起動して確認できます。次に、Server Manager を起動した時に表示されるメッセージの例を示します。

Oracle Server Manager Release 2.3.3.0.0 - Production

Copyright (c) Oracle Corporation 1994, 1995. All rights reserved.

Oracle7 Server Release 7.3.4.0.0 - Production

With the distributed, replication, and Spatial Data options

PL/SQL Release 2.3.4.0.0 - Production

表示されるメッセージは、インストールしたオプションおよびリリース番号によって少し異なる場合があります。前述の列の最終行にあるように、メッセージ中に「PL/SQL」が表示される場合は、プロシージャ・オプションがインストールされています。「PL/SQL」が表示されない場合は、インストールされていません。

4. すべてのデータ・ファイルおよび表領域が正常にオンラインまたはオフラインであることを確認します。

データ・ファイルに回復が必要かどうかを判断するために、次の SQL 文を発行します。

```
SELECT * FROM v$recover_file;
```

「0 rows selected」というメッセージが表示されます。これは、すべてのデータ・ファイルがオンラインまたはオフラインのいずれかであることを示しています。データ・ファイルが1つでも表示される場合は、データベースを移行する前にデータ・ファイルを復元する必要があります。V\$DATAFILE 動的パフォーマンス・ビューを使用して、データ・ファイル番号に応じたデータ・ファイル名を検索できます。メディア回復を必要とするデータ・ファイルが存在する場合は、先へ進まずに、移行中にエラーが表示されず。

確実にオフラインにできなかった表領域は、移行前に削除するかオンラインにする必要があります。これは、そうしないと移行後に Oracle8i で表領域を使用できないからです。通常、ALTER TABLESPACE OFFLINE IMMEDIATE または ALTER TABLESPACE OFFLINE TEMPORARY コマンドを使用してオフラインにする表領域には、メディア回復が必要です。

移行後、Oracle8i のデータベースをオープンするときにオフラインになっている表領域は、Oracle7 のデータベース・ファイル形式のまま残ります。オフラインの表領域は、移行後ならいつでもオンラインにできますし、ファイル・ヘッダーはそのときに Oracle8i 形式に変換されます。さらに、障害が発生した場合に大規模な復元を避けたいければ、SYSTEM と ROLLBACK を除くすべての表領域をオフラインにし、別の移行処理を実行する必要がある場合はその後で SYSTEM と ROLLBACK のデータ・ファイルのみを復元することができます。

5. OUTLN という名前のユーザーまたはロールがないことを確認します。これは、Oracle8i をインストールするときにこのスキーマを自動的に作成するからです。OUTLN という名前のユーザーまたはロールがある場合は、そのユーザーまたはロールを削除し、別の名前で再作成する必要があります。

OUTLN という名前のユーザーをチェックするには、次の SQL 文を発行します。

```
SELECT USERNAME FROM dba_users WHERE USERNAME = 'OUTLN';
```

OUTLN という名前のユーザーがない場合は、どの行も選択されません。

OUTLN という名前のロールをチェックするには、次の SQL 文を発行します。

```
SELECT ROLE FROM dba_roles WHERE ROLE = 'OUTLN';
```

OUTLN という名前のロールがない場合は、どの行も選択されません。

6. MIGRATE という名前のユーザーまたはロールがないようにします。これは、Oracle Data Migration Assistant がこのスキーマを作成し、既存のユーザーまたはロールをこの名前で置き換え、最終的にはそれをシステムから削除するからです。

MIGRATE という名前のユーザーをチェックするには、次の SQL 文を発行します。

```
SELECT USERNAME FROM dba_users WHERE USERNAME = 'MIGRATE';
```

MIGRATE という名前のユーザーがない場合は、どの行も選択されません。

MIGRATE という名前のロールをチェックするには、次の SQL 文を発行します。

```
SELECT ROLE FROM dba_roles WHERE ROLE = 'MIGRATE';
```

MIGRATE という名前のロールがない場合は、どの行も選択されません。

7. SYSTEM ロールバック・セグメントに OPTIMAL 設定がないようにします。移行中、OPTIMAL 設定によってエラーになることがあります。

SYSTEM ロールバック・セグメントの OPTIMAL 設定をチェックするには、次の SQL 文を発行します。

```
SELECT a.usn, a.name, b.optsize
FROM v$rollname a, v$rollstat b
WHERE a.usn = b.usn AND name='SYSTEM';
```

次のように表示されます。

USN	NAME	OPTSIZE
0	SYSTEM	

1 row selected.

OPTSIZE 列に値がある場合は、NULL に設定するために次の SQL 文を発行します。

```
ALTER ROLLBACK SEGMENT SYSTEM STORAGE (OPTIMAL NULL);
```

移行が完了したときに OPTIMAL をリセットできます。

参照： A-4 ページの「[SYSTEM ロールバック・セグメントの OPTIMAL 設定](#)」にあるトラブルシューティング情報を参照してください。

8. ALTER ROLLBACK SEGMENT 文の STORAGE 句の MAXEXTENTS パラメータを変え
ることによって、SYSTEM ロールバック・セグメントのエクステントの最大数を増加
して下さい (オプション)。

次に、ALTER ROLLBACK SEGMENT 文の例を示します。

```
ALTER ROLLBACK SEGMENT system
STORAGE (MAXEXTENTS 121);
```

正常に移行を完了するには、SYSTEM ロールバック・セグメントより多くの領域が必要
な場合があります。SYSTEM ロールバック・セグメントに十分な領域がない場合は、
Oracle Data Migration Assistant 実行中にエラーが発生する場合があります。

9. SYSTEM 表領域に十分な空き領域を確保して、Oracle8i のデータ・ディクショナリおよび既存の Oracle7 のデータ・ディクショナリを同時に保持します。

SYSTEM 表領域の空き領域を確認するには、次の SQL 文を発行します。

```
SELECT sum(bytes) FROM dba_free_space  
WHERE tablespace_name='SYSTEM';
```

この文はシステム表領域の空きバイト数を表示します。

参照： 詳細は、4-4 ページの「[領域要件](#)」および 2-15 ページの「[システム要件と使用可能なリソースの見積り](#)」を参照してください。

10. すべてのオンライン・データ・ファイルがアクセス可能で、正しいディレクトリに入っていることを確認します。RAW ディスクを使用する場合は、ログ・ファイルもアクセス可能でなければなりません。
11. `initSID.ora` ファイルを調整して Oracle8i で使用できるようにします。
`initSID.ora` ファイルが Oracle7 環境内に存在している場合があります。Oracle Data Migration Assistant は、実行時にそのファイルを Oracle8i 環境にコピーします。

具体的には、次のステップを実行します。

- a. JOB_QUEUE_PROCESSES 初期化パラメータを使用してスナップショットを自動的に更新する場合は、`initSID.ora` ファイルにあるこのパラメータをコメント行にしてください。データベースを移行した後、このコメントを削除して、このパラメータを通常に使用できます。
- b. 必要に応じて、LARGE_POOL_SIZE 設定を調整します。リリース 8.1 では、LARGE_POOL_SIZE 設定は Oracle によって自動的に計算されます。自動設定が大きすぎる場合、移行にかかる時間が長くなることがあります。詳細は、B-9 ページの「[大規模プールから割り当てられたパラレル実行](#)」を参照してください。
- c. DB_DOMAIN 初期化パラメータが正しく設定されていることを確認します。

参照： この初期化パラメータ設定の詳細は、B-9 ページの「[DB_DOMAIN パラメータ](#)」を参照してください。

- d. `initSID.ora` ファイルに IFILE (ファイルを含む) エントリがある場合、IFILE エントリで指定されているファイルを編集します。編集方法はサブステップの a から d で `initSID.ora` ファイルを編集した場合と同様です。

これらの調整後、`initSID.ora` ファイルおよび IFILE エントリで指定されたファイルが存在する場合は、これらのファイルを保存してください。

Oracle ソフトウェア リリース 8.1 のインストールおよびそのデータベースの移行

リリース 8.1 ソフトウェアをインストールおよびそのデータベースを移行するには、次のステップを実行します。

1. ご使用のオペレーティング・システム固有の Oracle ドキュメントの指示に従って、Oracle Universal Installer のインストールおよび起動する準備を行います。
2. Oracle Universal Installer の「Welcome」画面で、「Next」をクリックします。
ある画面でヘルプが必要な場合や Oracle Universal Installer のドキュメントについて知りたい場合は、「Help」ボタンをクリックしてオンライン・ヘルプをオープンします。
3. 「File Locations」画面で、「Destination」がリリース 8.1 Oracle ホームへの絶対パスであることを確認します。その後、「Next」をクリックします。
4. 「Available Products」画面で、Oracle8i Enterprise Edition または Oracle8i から、インストールする Oracle8i のエディションを選択します。その後、「Next」をクリックします。
5. 「Installation Types」画面では、インストレーション・タイプを選択します。その後、「Next」をクリックします。

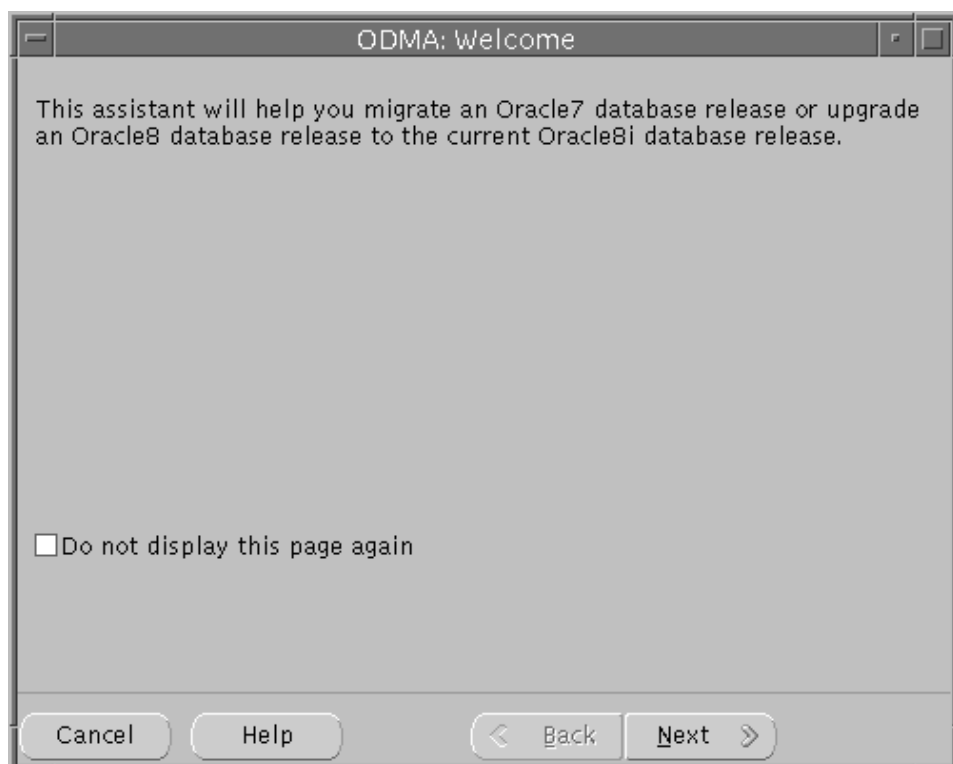
「Custom」を選択した場合、「Upgrading or Migrating an Existing Database」画面が表示されるまで、カスタム・インストレーション設定の指定ができる画面に従ってください。

特定のオプションの使用を中止しない場合、Oracle7 データベースにインストールしていたオプションすべてをインストールすることを確認します。たとえば、Oracle7 にアドバンスド・レプリケーションをインストールしていた場合、それを Oracle8i にインストールする必要があります。

6. 「Upgrading or Migrating an Existing Database」画面では、次のステップを実行してください。
 - a. 「Upgrade or Migrate an Existing Database」チェック・ボックスを選択します。
 - b. 移行する Oracle7 データベースを選択します。
 - c. 「Next」をクリックします。
7. 「Summary」画面で、インストレーションの設定および選択が正しいかどうかを確認します。その後、「Install」をクリックします。これによって、Oracle Universal Installer がインストレーションを実行します。ある程度時間がかかる場合があります。
インストールが終了したら、1 つ以上のアシスタントが起動する場合があります。Oracle Data Migration Assistant が起動すると、移行を続行できます。
8. Oracle Data Migration Assistant の「Welcome」画面で、「Next」をクリックします。実際の画面は、オペレーティング・システムによって、[図 4-2](#) に示すものとは異なる場合があります。

注意： Oracle Data Migration Assistant は、SHUTDOWN IMMEDIATE 文を使用してデータベースをシャットダウンします。したがって、Oracle Data Migration Assistant が起動するときはどのユーザーもログインすることは避けてください。ログイン済みのユーザーは切断されます。

図 4-2 Oracle Data Migration Assistant の「Welcome」画面



ある画面でヘルプが必要な場合や Oracle Data Migration Assistant のドキュメントについて知りたい場合は、「Help」ボタンをクリックしてオンライン・ヘルプをオープンします。

9. 「Before You Migrate or Upgrade」画面で、移行している Oracle7 データベースが指定した条件を満たすことを確認します。その後、「Next」をクリックします。
10. 「Select a Database Instance」画面で、移行している Oracle7 データベースのデータベース・インスタンスを選択します。その後、「Next」をクリックします。
11. 「Database Password and INIT.ORA File」画面で、次のステップを実行します。
 - a. 指定された新規の Oracle ホームが正しいことを確認します。
 - b. 指定された `initSID.ora` ファイルの場所が、移行している Oracle7 データベースの `initSID.ora` ファイルへの絶対パスであることを確認します。
 - c. 指定された古い Oracle ホームが、移行している Oracle7 データベースの Oracle ホームへの絶対パスであることを確認します。
 - d. 「Next」をクリックします。
12. 「Choose Migration Type」画面では、移行タイプを選択します。その後、「Next」をクリックします。

「Custom」を選択した場合、「Confirm Backup」画面が表示されるまで、カスタム・インストール設定の指定ができる画面に従ってください。
13. 「Confirm Backup」画面で、ご使用の Oracle7 データベースのバックアップを取っている場合、「Next」をクリックします。そうでない場合は、バックアップを取り、移行を繰り返します。
14. 「Start the Migration or Upgrade」画面では、仕様のすべてが正しいことを確認します。不適切な仕様がある場合は、それを訂正できるまで「Back」をクリックします。すべてが正しい場合は、「Next」をクリックします。

Oracle Data Migration Assistant が移行を実行します。処理過程はステータス・バーに表示されます。
15. 「Listener.ora Migration Confirmation」画面では、`listener.ora` ファイルを自動的に変更するのにアシスタントが必要な場合は「Yes」ボタンをクリックします。`listener.ora` ファイルを変更するのにアシスタントが必要でない場合は「No」ボタンをクリックします。

データベースが Oracle Enterprise Manager で正常に動作するためには、`listener.ora` ファイルにある特定の変更が必要となります。Oracle Enterprise Manager を使用する場合、`listener.ora` ファイルを自動的に変更するには「Yes」ボタンをクリックします。ただし、Oracle Enterprise Manager を使用しない場合は、「No」ボタンをクリックします。

「Yes」ボタンをクリックした場合、Oracle Data Migration Assistant は listener.ora ファイルを次のように変更します。

- a. アシスタントは古いリスナーおよび Oracle8i リスナーをシャットダウンします。
- b. アシスタントは Oracle8i listener.ora にある SID_DESC エントリを変更します。変更方法は次のうちのいずれかです。

簡単な場合: 古い listener.ora に次の SID_DESC エントリがあるとします。

```
...
  (SID_DESC =
    (SID_NAME = ORCL)
  )
...
```

データベース名が SAL、ドメイン名が COM、Oracle ホームが /oracle/product/8.1 の場合、アシスタントは次のエントリを追加します。

```
...
  (SID_DESC =
    (GLOBAL_DBNAME = sal.com)
    (ORACLE_HOME = /oracle/product/8.1)
    (SID_NAME = SAL)
  )
...
```

少し複雑な場合: 古い listener.ora に次の SID_DESC エントリがあるとします。

```
...
  (SID_DESC =
    (GLOBAL_DBNAME = an_entry)
    (SID_NAME = ORCL)
  )
...
```

an_entry が移行したデータベースの GLOBAL_DBNAME と一致しない場合、データベース名が SAL、ドメイン名が COM および Oracle ホームが /oracle/product/8.1 の場合は、アシスタントは次のエントリを追加します。

```
...
(SID_DESC =
  (GLOBAL_DBNAME = sal.com)
  (ORACLE_HOME = /oracle/product/8.1)
  (SID_NAME = SAL)
)
...
```

このエントリは簡単な場合のエントリと同じですが、アシスタントはエントリ *an_entry* を SERVICE_NAMES パラメータに追加します。したがって、アシスタントは SERVICE_NAMES パラメータを次のように変更します。

```
SERVICE_NAMES = sal.com, an_entry
```

- c. Windows NT では、アシスタントは移行されたデータベースを古い listener.ora ファイルから削除します。UNIX では、アシスタントはこの操作を行いません。
- d. アシスタントは Oracle8i リスナーを起動します。

Oracle Data Migration Assistant 単独での実行

既存のデータベースの移行を指定しないで Oracle8i をインストールした場合、Oracle8i のインストール終了後、Oracle Data Migration Assistant を単独で実行できます。

Oracle Data Migration Assistant を単独で実行させるには、次のステップを実行します。

1. Oracle Data Migration Assistant を起動します。

Unix の場合、システムプロンプトに次のコマンドを入力します。

```
odma
```

Windows NT の場合、次のように選択します。

スタート > プログラム > Oracle - *ORACLE_HOME_NAME* > Migration Utilities > Oracle Data Migration Assistant

Oracle Data Migration Assistant を起動すると、「Welcome」画面が表示されます（4-12 ページの図 4-2 を参照）。

2. Oracle Data Migration Assistant 上の各ウィンドウの質問に答えていき、次のウィンドウへ移動する準備ができたなら、「Next」をクリックします。

参照： 詳細は、4-11 ページの「Oracle ソフトウェア リリース 8.1 のインストールおよびそのデータベースの移行」のステップ 8 ~ 15 を参照してください。

移行の終了

Oracle Data Migration Assistant が正常に実行できたら、次のステップを実行します。

1. Server Manager を起動します。
2. データベース・インスタンスに接続します。

```
SVRMGR> CONNECT INTERNAL;
```

3. STARTUP RESTRICT を実行します。

```
SVRMGR> STARTUP RESTRICT
```

`init.ora` ファイルの位置を指定する `PFFILE` オプションを使用する場合があります。

4. 成功したかどうかを後で確認するために、結果をログ・ファイルにスプールするようにシステムを設定します。

```
SVRMGR> SPOOL catoutma.log
```

実行したスクリプトの出力を画面上で確認する場合は、`SET ECHO ON` 文の発行でも確認できます。

```
SVRMGR> SET ECHO ON
```

5. Oracle システムがアドバンスト・レプリケーションをインストールした場合は、次のステップを実行します。

- a. `catrep.sql` を実行します。

```
SVRMGR> @catrep.sql
```

- b. `r0703040.sql` を実行します。

```
SVRMGR> @r0703040.sql
```

この `r0703040.sql` スクリプトは `catrep.sql` 後のアドバンスト・レプリケーションに関連するアップグレードを実行します。

これらのスクリプト、または残りのステップ中のスクリプトを実行中に問題が発生した場合は、問題の原因を修正してスクリプトを再実行します。この章で説明するどのスクリプトも、必要に応じて何度でも再実行できます。

6. utlrlp.sql を実行します (オプション)。

```
SVRMGR> @utlrlp.sql
```

utlrlp.sql スクリプトは、以前に INVALID 状態にあった既存の PL/SQL モジュール (パッケージ、プロシージャ、型など) をすべて再コンパイルします。この処理はオプションです。ただし、この処理を行うと、再コンパイルは、インストール後ではなく、インストール中に発生します。

オラクル社はこのオプションのステップを実行することをお勧めします。

7. スクリプト結果のログ・ファイルへのスプーリングをオフにします。

```
SVRMGR> SPOOL OFF
```

次にスプール・ファイルをチェックして、パッケージとプロシージャのコンパイルが成功したかどうかを検証します。ステップ 4 でスプール・ファイルに名前を付けましたが、その名前は catoutma.log でした。

領域が不足していることを警告するエラーやスクリプトが実行しないエラーは、調査する必要があります。これらのタイプのエラーが発生する場合、移行は成功していない場合があります。ただし、存在しないオブジェクトの変更や削除の障害についてのエラーは、一般に無視できます。

SET ECHO ON を指定した場合は、ここで SET ECHO OFF を行うことができます。

```
SVRMGR> SET ECHO OFF
```

8. Oracle8i のデータベース上で、SHUTDOWN を実行します。

```
SVRMGR> SHUTDOWN IMMEDIATE
```

注意: SHUTDOWN NORMAL または SHUTDOWN IMMEDIATE を使用してください。SHUTDOWN ABORT は使用しないでください。

この正しい停止を実行すると、すべてのキャッシュをフラッシュし、バッファを消去し、その他の内部的な不整合の解消を行います。これらの手段は、新規に移行した Oracle8i のデータベースの整合性と一貫性を保証するための重要な最終ステップです。

9. Oracle8i 用に、`init.ora` ファイルを調整します。

バージョン 8 で構文が変更されているパラメータを変更することも必要です。新規のパラメータ、改名されたパラメータ、廃止されたパラメータのリストは、[付録 B「初期化パラメータの変更」](#)を参照してください。

また、[付録 B「初期化パラメータの変更」](#)に示す新規パラメータを理解し、移行済みデータベースでパラメータを決定してください。

また、Oracle Data Migration Assistant は COMPATIBLE 初期化パラメータを 8.0.5. に設定します。COMPATIBLE 初期化パラメータのリセットの詳細は、[第 8 章「互換性および相互運用性」](#)を参照してください。

参照： 初期化パラメータの詳細は『Oracle8i リファレンス・マニュアル』を参照してください。

10. [第 6 章「データベースの移行後」](#)に記載の手順を実行します。

注意： 古い Oracle7 ソフトウェアを保持している場合、Oracle7 ソフトウェアで移行済みデータベースを絶対に起動させないでください。新規の Oracle8i インストールディレクトリにある実行ファイルでのみ、データベースを起動してください。

移行時のトラブルシューティング

エラーが発生する原因としては、次のような操作が行われたか、あるいは省略されたことが考えられます。

- 誤った順序で移行ステップを実行する
- 移行の前提条件を満たしていない
- 変換の例外を検出する

参照： 移行中に発生する可能性のあるエラーとそのエラーの対処措置については、[付録 A「移行問題のトラブルシューティング」](#)および『Oracle8i エラー・メッセージ』を参照してください。

移行の中止

Oracle Data Migration Assistant による移行を行う場合は、必ず Oracle7 データベースのバックアップを取ってください。移行を中止する場合は、バックアップから復元することになります。

5

エクスポート / インポートによる移行

この章では、エクスポート / インポートを使用してデータベースを Oracle7 から Oracle8i へ移行する方法について説明します。

この章で説明するトピックは次のとおりです。

- [エクスポート / インポートの基本](#)
- [エクスポート / インポートを使用したリリース 8.0 以前のソース・データベースの移行](#)

参照： インポートおよびエクスポート・ユーティリティの使用方法は、『Oracle8i ユーティリティ・ガイド』を参照してください。

エクスポート / インポートの基本

エクスポート / インポートを使用してデータベースを移行するには、次の 3 つの基本ステップを行います。

1. 移行元のデータベース（ソース・データベース）からデータをエクスポートします。このエクスポートでは、データを物理的にエクスポート・ダンプ・ファイルへコピーします。
2. エクスポートしたデータをインポートする Oracle8i のデータベース（ターゲット・データベース）を作成します。
3. エクスポートしたデータを新しい Oracle8i のターゲット・データベースにインポートします。

参照： エクスポート / インポートによる移行方法の評価については、2-3 ページの「[移行方法の選択](#)」および『Oracle8i ユーティリティ・ガイド』を参照してください。

エクスポート要件

データベースを移行またはアップグレードするには、ソース・データベースのリリースに付属のエクスポート・ユーティリティを使用します。エクスポートの後、インポート・ユーティリティを使用して、エクスポート・ダンプ・ファイルのデータをターゲット・データベースにコピーします。エクスポートしたデータをインポート・ユーティリティでターゲット・データベースに移行する前に、ターゲット・データベースが作成済みですでに操作可能な状態にある必要があります。

たとえば、リリース 7.3 からリリース 8.1 へ移行する場合、エクスポート・ユーティリティ リリース 7.3 を使用してください。

注意： ソースとなる Oracle データベースがバージョン 6 以前である場合は、エクスポートを行う前に、ソース・データベースを最低でもバージョン 6 まで移行してください。

インポート要件

データベースを移行、アップグレードまたはダウングレードするには、ターゲット・データベースのリリースに付属のインポート・ユーティリティを使用します。たとえば、リリース 7.3 からリリース 8.1 へ移行する場合、リリース 8.1 のエクスポート・ユーティリティを使用してください。

その他のオプション

その他のオプションをインストールしている場合は、次の関連情報を参照してください。

- Trusted Oracle データベースから機能をエクスポートする場合や、Trusted Oracle データベースへ機能をインポートする場合、Trusted Oracle データベースの機能の移行の詳細は、Trusted Oracle のドキュメントを参照してください。
- アドバンスド・レプリケーションがインストール済みのデータベース・システムを移行する場合は、『Oracle8i レプリケーション・ガイド』の付録 B「移行および互換性」を参照してください。

エクスポート / インポートを使用したリリース 8.0 以前のソース・データベースの移行

エクスポート / インポート・ユーティリティを使用してバージョン 6 または Oracle7 データベースを移行するには、次のステップを実行します。

1. ソース・データベースに付属のエクスポート・ユーティリティを使用してソース・データベースをエクスポートします。ソース・データベースでのエクスポート・ユーティリティの使用方法については、ソース・データベースのサーバー・ユーティリティのドキュメントを参照してください。Oracle7 のデータベースおよびバージョン 6 のデータベースのエクスポートは両方とも、Oracle8i にインポートできます。

一貫性のあるエクスポート結果を得るには、エクスポートの実行中と実行後はソース・データベースが更新不可でなければなりません。エクスポート実行後にユーザーがソース・データベースを更新できるようにする場合は、インポート完了後にソース・データベースに対する変更を Oracle8i のターゲット・データベースにコピーする手順を実行し、その後でソース・データベースを使用可能にします。

2. Oracle8i ソフトウェアをインストールします。インストールは、オペレーティング・システム固有の作業です。Oracle8i のインストール手順は、ご使用のオペレーティング・システム固有の Oracle8i ドキュメントを参照してください。
3. 新しい Oracle8i のデータベースに既存のソース・データベースと同じ名前を付ける場合は、新しい Oracle8i のデータベースを作成する前に既存のデータベースを停止する必要があります。
4. Oracle8i ターゲット・データベースを作成します。

参照： Oracle8i データベース作成の詳細は、『Oracle8i 管理者ガイド』を参照してください。

5. Server Manager を Oracle8i 環境で起動します。
6. データベース・インスタンスに接続します。

```
SVRMGR> CONNECT INTERNAL
```

7. STARTUP を使用して、Oracle8i データベース・インスタンスを開始します。
8. 記憶領域パラメータを変更してディスク領域の使用状況を改善するなどの目的のために、ターゲット・データベースに表領域およびユーザー、表を事前に作成しておきます。SQL*Plus や Server Manager を使用して表を事前に作成する場合は、元のデータベースとの互換モードで実行するか、またはインポート中に発生する特定のデータ定義の変換について特に配慮する必要があります。

注意： 新しい Oracle8i のデータベースをソース・データベースと同じコンピュータ上に作成し、ソース・データベースのデータ・ファイルを上書きしない場合は、事前に表領域を作成し、インポート時に IGNORE=Y および DESTROY=N を指定する必要があります。

9. Oracle8i のインポート・ユーティリティを使用して、ソース・データベースからエクスポートしたオブジェクトをインポートします。インポート・セッションからファイルに情報メッセージやエラー・メッセージを保存するには、LOG パラメータを組み込んでください。

参照： インポート・ユーティリティの詳細は、『Oracle8i ユーティリティ・ガイド』を参照してください。

10. 移行作業が完了したら、どのオブジェクトのどのインポートが正常に終了してどれが失敗したか（失敗したものがある場合）を、インポート・ログ・ファイルを見てチェックします。

参照： エラー処理情報については、『Oracle8i ユーティリティ・ガイド』および Oracle8i Server の README.doc ファイルを参照してください。

11. その他のインポート方法（『Oracle8i ユーティリティ・ガイド』を参照）を使用するか、またはソース・オブジェクトを作成する SQL スクリプトを使用して、インポートに失敗したオブジェクトを作成し直します（または完全に新規のインポートを開始します）。
12. エクスポートの実行後にソース・データベースに変更を加えた場合は、そのデータベースを使用可能にする前に、それらの変更が Oracle8i のデータベースに反映されていることを確認します。詳細は、5-3 ページのステップ 1 を参照してください。
13. [第 6 章「データベースの移行後」](#)に記載の手順を実行します。

6

データベースの移行後

この章では、移行ユーティリティ、Oracle Data Migration Assistant、エクスポート / インポートまたはデータ・コピーを使用した移行を完了した後で実行する手順について説明します。また、1-9 ページの「[ステップ 6: 新しい本番データベースのチューニングおよび調整](#)」について詳しく説明します。

この章で説明する移行後のステップは、次のとおりです。

- [移行したデータベースのバックアップ](#)
- [無効な日付制約のチェック](#)
- [使用不可のビットマップ索引の再構築](#)
- [パラレル実行を使用する問題の回避](#)
- [パーティション・ビューからパーティション表への移行](#)
- [OUTLN ユーザーのパスワードの変更](#)
- [Net8 の新しいリリースへの移行またはアップグレード（オプション）](#)
- [ご使用の listener.ora ファイルの変更](#)
- [データベースのテストと結果の比較](#)
- [移行されたデータベースのチューニング](#)
- [新機能の適宜追加](#)
- [必要な新管理手順の作成](#)

移行したデータベースのバックアップ

新たに移行した本番データベースの完全なバックアップを、必ず作成してください。このバックアップは、データ・ファイル、制御ファイル、オンライン REDO ログ・ファイル、パラメータ・ファイル、新しいデータベース内のオブジェクトを作成するための SQL スクリプトなどのすべてを含む完全なバックアップでなければなりません。ホット・バックアップは完全な回復可能性を提供できないので、完全なバックアップを作成するには全データベース・エクスポートまたはコールド・バックアップが必要です。後続のステップによって、新しく移行されたデータベースに悪い影響が出るような場合は、必要に応じてここで作業を中止することもできます。

参照： データベースのバックアップの詳細は、『Oracle8i バックアップおよびリカバリ・ガイド』を参照してください。

注意： 移行ユーティリティを使用すると、ソース・データベースが変換されます。したがって、移行後は 3-20 ページの「[Oracle7 のソース・データベースの保存](#)」で作成したバックアップ以外は、ソース・データベースがなくなります。このバックアップは、新規に移行したデータベースを回復するための最初の Oracle8i のバックアップとしても使用できます。

無効な日付制約のチェック

無効な日付制約は、無効日付操作を必要とします。無効日付操作は、日付の中の世紀を暗黙的に想定するものであり、西暦 2000 年に問題が発生します。utlconst.sql スクリプトは、データベース内のすべてのチェック制約で実行され、無効日付操作を組み込む場合は制約を無効に設定します。このスクリプトは、最後にすべての無効な制約を選択します。

utlconst.sql スクリプトを実行するには、次のステップを実行します。

1. システム・プロンプトで、\$ORACLE_HOME/rdbms/admin ディレクトリへ移動します。
2. Server Manager を起動します。
3. データベース・インスタンスへ接続します。

```
SVRMGR> CONNECT INTERNAL
```


4. 次のように入力します。

```
SVRMGR> SPOOL utlresult.log
SVRMGR> @utlconst.sql
SVRMGR> SPOOL OFF
```

スクリプトを実行した後、utlresult.log ログ・ファイルには、無効な日付制約を持つすべての制約が入っています。

注意： utlconst.sql スクリプトは無効な制約を訂正せずに、使用禁止にします。必要な変更を加えた後に、無効な制約を削除するか、再作成する必要があります。

使用不可のビットマップ索引の再構築

移行中に、一部のビットマップ索引が使用不可になることがあります。これらの索引を検索するには、次の SQL 文を発行します。

```
SELECT index_name, index_type, table_owner, status
FROM dba_indexes
WHERE index_type = 'BITMAP'
AND status = 'UNUSABLE';
```

リストされた使用不可のビットマップ索引を再構築します。

参照： ビットマップ索引の使用の詳細は、『Oracle8i チューニング』および『Oracle8i 概要』を参照してください。

パラレル実行を使用する問題の回避

リリース 8.1 移行では、パラレル実行メッセージは大規模プールから割り当てられます。以前のリリースでは、この割当ては共有プールから行われました。この変更の結果生じる問題を避けるために、initSID.ora ファイルの次のパラメータを調整する必要があります。

- SHARED_POOL_SIZE
- LARGE_POOL_SIZE

参照： これらのパラメータの調整の詳細は、B-9 ページの「[大規模プールから割り当てられたパラレル実行](#)」を参照してください。

パーティション・ビューからパーティション表への移行

Oracle8i の新しいアプリケーションで、パーティション・ビューの使用はお薦めしません。また、既存のパーティション・ビューはパーティション表へ変換する必要があります。ALTER TABLE コマンドの EXCHANGE PARTITION オプションを使用して、Oracle7 のデータベース用に作成されたパーティション・ビューをパーティション表へ変換できます。

参照： パーティション・ビューのパーティション表への変換方法の詳細は、『Oracle8i 管理者ガイド』を参照してください。パーティション・ビューおよびパーティション表のバックグラウンド情報は、『Oracle8i 概要』を参照してください。

注意： パーティション・ビューは Oracle の将来のリリースではサポートされません。

OUTLN ユーザーのパスワードの変更

OUTLN ユーザーは Oracle8i のインストール中に自動的に作成されます。このユーザーは DBA 権限を持っています。ALTER USER コマンドを使用して、このユーザーのパスワードを変更します。Oracle8i は OUTLN ユーザー・スキーマを追加して、プラン・スタビリティをサポートします。OUTLN ユーザーはストアド・アウトラインに関連付けられるメタデータを中枢的に管理する場所として機能します。

Net8 の新しいリリースへの移行またはアップグレード（オプション）

Net8 の新しいリリースへの移行またはアップグレードは必須ではありません。ただし、Net8 は SQL*Net V2 に比べて非常に便利で、簡素化された設定および拡張された機能が含まれます。Net8 の新しいリリースも、次の点で Net8 および SQL*Net の以前のリリースよりも優れています。

- **サービス命名**によって、クライアントは特定のデータベースのインスタンスではなく、サービス名を使用して、全体としてサービスにアクセスできます。サービス命名はサービス名を論理的に特定なインスタンス名と区別し、SID パラメータを交換して、1 つのインスタンスが複数のサービスで利用できるようにします。個々のインスタンスも複数のサービスの一部となり得ます。
- **インスタンス登録**は自動で行われます。インスタンスは開始されるときそれ自体をリスナーに登録します。以前のリリースは、インスタンスの情報は手動で listener.ora ファイルに設定されていました。

- **フェイルオーバー**は自動で行われます。インスタンスが停止した場合は、クライアント接続要求は自動的に別のリスナーへ送信されます。
- **ロード・バランシング**は接続を利用可能なリスナーへ分散します。

参照： Net8 の利点および Net8 の新しいリリースへの移行またはアップグレード方法の詳細は、『Oracle8i Net8 管理者ガイド』を参照してください。

ご使用の listener.ora ファイルの変更

Oracle Enterprise Manager を使用してデータベース・オブジェクトを管理する場合、データベースの情報に関して listener.ora ファイルを設定する必要があります。

```
sid_list_listener_name =
  (sid_list =
    (sid_desc =
      (global_dbname = global_database_name)
      (oracle_home = oracle_home)
      (sid_name = sid)
    )
  )
```

GLOBAL_DBNAME はグローバル・データベース名を示します。その名前はデータベース名とデータベースのドメイン名から構成されてます。ORACLE_HOME は Oracle ソフトウェアがインストールされるディレクトリを示します。SID_NAME はデータベース・インスタンスの Oracle システム識別子を示します。

たとえば、次の情報は LISTENER というリスナー名を持つデータベースに関するものです。SAL.COM のグローバル・データベース名は SAL の SID を持つインスタンスによって参照されます。そのとき、/oracle/product/8.1 ディレクトリにインストールされたソフトウェアが使用されます。

```
sid_list_listener =
  (sid_list =
    (sid_desc =
      (global_dbname = sal.com)
      (oracle_home = d:/oracle/product/8.1)
      (sid_name = sal)
    )
  )
```

Oracle Enterprise Manager を使用していない場合は、データベースの情報は設定する必要がありません。それは、Oracle8i のデータベース・インスタンスが自動的にそれ自体をリスナーに登録するためです。

参照： ファイルと SID_LIST パラメータの詳細は、『Oracle8i Net8 管理者ガイド』を参照してください。

データベースのテストと結果の比較

2-19 ページの「[テスト計画の作成](#)」で作成したテスト計画を使用して、Oracle8i のデータベースをテストします。テスト結果を元のデータベースで得られたテスト結果と比較し、同じ結果が得られているかまたは改善しているかを確認します。

一般的に、移行した Oracle8i のデータベースでのパフォーマンスは、ソース・データベースでのパフォーマンス以上になります。Oracle8i でデータベースのパフォーマンス低下が見られる場合は、初期化パラメータが正しく設定されていることを確認してください。初期化パラメータが不正に設定されていると、パフォーマンスに害を与えることがあります。

移行したテスト・データベースのテストに加えて、元のソース・データベースと新たに移行した Oracle8i のデータベースをしばらくの間同時に実行し、新しいデータベースが正しく設定されていて適切に稼動することを確認することもできます。両方のデータベースを同時に実行するには、Oracle の構成を変更する必要があります。もちろん、両方のデータベースに同期した更新が必要です。理想的には、自動化プロセスの一部として更新を行います。

移行されたデータベースのチューニング

移行したデータベースのパフォーマンスを改善したい場合は、データベースを調整してください。Oracle7 のデータベースおよび関連アプリケーションのチューニングに使用した処理のほとんどは、Oracle8i のデータベースにも同様に有効な場合と、不適切な場合があります。したがって、ソース・データベースおよびアプリケーションをチューニングするために行った処置が原因で、移行した Oracle8i のデータベースのパフォーマンスが低下することのないようにしてください。

参照： チューニングの詳細は、『Oracle8i チューニング』を参照してください。

新機能の適宜追加

Oracle8i では、使用可能な多くの新機能が追加されましたが、これらのどの機能が移行されたデータベースの役に立つかを判断して、これらの機能を使用する計画を立ててください。

移行した Oracle8i のデータベースを使用するためにすぐに変更する必要はありません。データベースおよびそれに対応するアプリケーションに、これらの拡張機能を徐々に取り入れてもかまいません。

第 9 章「[アプリケーションのアップグレード](#)」では、Oracle8i の新機能を利用するためにアプリケーションを拡張する方法について説明します。ただし、Oracle8i 新機能をインプリメントする前にアプリケーションをテストし、移行データベース上でそれらを正常に実行する必要があります。

必要な新管理手順の作成

Oracle8i の機能に慣れてきたら、データベース管理用のスクリプトや手順を見直し、変更が必要かどうかを判断します。

それぞれのアプリケーションに必要な変更を、データベースにも行う必要があります。たとえば、データベースで整合性制約を使用可能にした場合、アプリケーションでのこのデータ整合機能のいくつかを取り除けます。

Oracle8i の新しいリリースへのアップグレード

この章では、Oracle の現行のリリースから Oracle8i の新しいリリースへのアップグレードについて説明します。この章で説明する内容は、Oracle リリース 8.0 以降のインストール環境のみに適用されます。現行のリリースが Oracle7 やバージョン 6 といったリリース 8.0 より前のリリースで、そこから Oracle8i へ移行する場合は、このマニュアルの最初に記載されている指示に従い、第 1 章「概要」から始めてください。

この章で説明するトピックは、次のとおりです。

- [アップグレード・パス](#)
- [Oracle8i の新しいリリースへのアップグレード](#)
- [特定のコンポーネントのアップグレード](#)
- [新リリースへのアップグレードの後](#)
- [現在のリリースのワード・サイズの変更](#)

参照： アップグレード処理の一部は、オペレーティング・システム固有のものです。ご使用のオペレーティング・システムでのアップグレードに関するその他の指示は、オペレーティング・システム固有の Oracle ドキュメントを参照してください。

Oracle Parallel Server のアップグレード： Oracle Parallel Server をインストールしたシステムをアップグレードする場合は、この章で説明されている処理のほとんどはシステムのノードの 1 つのみで実行されます。したがって、特定のステップに特に断りのない限り、この章に説明されている処理を 1 つのノードのみで実行してください。

アップグレード・パス

新しいリリースへデータベースをアップグレードするために必要なパスは、ご使用のリリースに依存します。表 7-1 には、Oracle の古いリリースごとに必要なアップグレード・パスが含まれています。現在ご使用のリリースに固有のアップグレード・パスおよびマニュアルを使用します。

表 7-1 アップグレード・パス

古いリリース	アップグレード・パス
8.0.1	ダイレクト・アップグレードはサポートされていません。新しいリリースへアップグレードするには、次のステップを実行します。 <ol style="list-style-type: none">リリース 8.0.2 の README.doc ファイルの指示を使用して、リリース 8.0.2 へアップグレードします。リリース 8.0.5 の README.doc ファイルの指示を使用して、リリース 8.0.5 へアップグレードします。7-3 ページの「Oracle8i の新しいリリースへのアップグレード」の指示に従って、新しいリリースへアップグレードします。
8.0.2 8.0.4S	ダイレクト・アップグレードはサポートされていません。新しいリリースへアップグレードするには、次のステップを実行します。 <ol style="list-style-type: none">リリース 8.0.5 の README.doc ファイルの指示を使用して、リリース 8.0.5 へアップグレードします。7-3 ページの「Oracle8i の新しいリリースへのアップグレード」の指示を使用して、新しいリリースへアップグレードします。
8.0.3 8.0.4 8.0.5 8.1.3 8.1.4	ダイレクト・アップグレードはサポートされています。7-3 ページの「 Oracle8i の新しいリリースへのアップグレード 」の指示に従って、新しいリリースへアップグレードします。
8.1.1 8.1.2	新しいリリースへのアップグレードは、サポートされていません。

Oracle8i の新しいリリースへのアップグレード

この項では、データベースを Oracle8i へアップグレードする処理について説明します。

注意： Oracle8i Enterprise Edition から Oracle8i (以前は Workgroup Server) へアップグレードする場合は、アップグレードの実行前に、Oracle8i Enterprise Edition の拡張機能を使用するアプリケーションを変更し、それらの拡張機能を使用しないようにしなければなりません。

アップグレードの準備

アップグレード処理を開始するには、次のステップを実行します。

1. リリース 8.1.4 からアップグレードし、保存する Java ソース、クラスまたはリリースオブジェクトがある場合は、アップグレード前にこれらの Java オブジェクトをエクスポートし、アップグレード後にそれらをインポートします。

注意： Java オブジェクトをリリース 8.1.5 以前のデータベースと現在のリリース間でエクスポートおよびインポートすることは、保証されていません。

参照： Java ソース、クラスおよびリソース・オブジェクトのエクスポートおよびインポートの詳細は、『Oracle8i ユーティリティ・ガイド』を参照してください。

2. リリース 8.1.3 または 8.1.4 からアップグレードしており、リリース 8.1.3 またはリリース 8.1.4 を使用してリリース 8.1 互換キュー表を作成した場合は、データベースをアップグレードする前に 8.1 互換キュー表を削除します。リリース 8.1.3 または 8.1.4 以外のリリースからアップグレードする場合は、ステップ 4 へ進みます。

既存のリリース 8.1 互換キュー表がデータベースに存在するかどうかを判断するには、次の SQL 文を発行します。

```
SELECT owner, queue_table FROM dba_queue_tables where  
compatible like '8.1%';
```

たとえば、前述のキューの発行の結果、次のテーブルが表示されることがあります。

```
OWNER                                QUEUE_TABLE
-----
AQUSER1                             RAW_MSG_TABLE
1 row selected.
```

次の例に示されているように、DBMS_AQADM.DROP_QUEUE_TABLE を使用してリストしたキュー表を削除できます。

```
EXECUTE dbms_aqadm.drop_queue_table(queue_table=>'AQUSER1.RAW_MSG_TABLE',
                                     force => TRUE);
```

3. DB_DOMAIN 初期化パラメータが正しく設定されていることを確認します。

参照： この初期化パラメータの詳細は、B-9 ページの「[DB_DOMAIN パラメータ](#)」を参照してください。

4. Server Manager を起動します。
5. データベース・インスタンスへ接続します。

```
SVRMGR> CONNECT INTERNAL
```

6. 8.0 リリースからアップグレードする場合は、OUTLN という名前のユーザーまたはロールがないことを確認します。これは、Oracle8i のインストールの際、このスキーマが自動的に作成されるからです。OUTLN という名前のユーザーまたはロールがある場合、そのユーザーまたはロールを削除し、別の名前で作成する必要があります。

注意： 8.1 リリースからアップグレードする場合は、このチェックを実行する必要はありません。それは、以前の 8.1 リリースをインストールするときに OUTLN ユーザーが作成されているためです。したがって、リリース 8.1 からアップグレードする場合は、ステップ 7 へ進みます。以前の 8.1 リリースからアップグレードする場合は、OUTLN ユーザーを削除しないでください。

OUTLN という名前のユーザーをチェックするには、次の SQL 文を入力します。

```
SELECT username FROM dba_users WHERE username = 'OUTLN';
```

OUTLN という名前のユーザーがない場合、どの行も選択されません。

OUTLN という名前のロールをチェックするには、次の SQL コマンドを入力します。

```
SELECT role FROM dba_roles WHERE role = 'OUTLN';
```

OUTLN という名前のロールがない場合、どの行も選択されません。

7. SYSTEM 表領域、また必要に応じて、ロールバック・セグメントを保存している表領域に空き領域を追加します。

新しいリリースへアップグレードするには、SYSTEM 表領域およびロールバック・セグメントを保存している表領域により大きな空き領域が必要です。システムに十分な空き領域がない場合は、これらの表領域へより大きな領域を追加することを考慮してください。アップグレード中に領域から実行する場合は、アップグレードを再実行する必要があります。

次の SQL 文は、より多くの領域を表領域へ追加する方法を説明しています。

```
ALTER TABLESPACE system
  ADD DATAFILE '/home/user1/mountpoint/oradata/db1/system02.dbf'
  SIZE 50M;
```

8. データベース上で、SHUTDOWN IMMEDIATE を実行します。

```
SVRMGR> SHUTDOWN IMMEDIATE
```

Oracle Parallel Server を使用している場合、インスタンスを完全に停止します。

9. データベースの全オフライン・バックアップを実行します。

参照： 詳細は、『Oracle8i バックアップおよびリカバリ・ガイド』を参照してください。

データベースの移行

アップグレード方法を選択し、その指示に従ってアップグレードします。

アップグレード方法の選択

ご使用のデータベースをリリース 8.1 へアップグレードするには、2 つの方法があります。Oracle Data Migration Assistant を使用してアップグレードを完了するか、手動でアップグレードを実行できます。

Oracle Data Migration Assistant によって、データベースの完全に自動化されたアップグレードができます。処理の各ステップをガイドするグラフィカル・ユーザー・インターフェース (GUI) を使用します。また、Oracle Data Migration Assistant には、広範なオンライン・ヘルプがあります。Oracle Data Migration Assistant は現在のリリースに適切なスクリプトを実行し、廃止された初期化パラメータを `inittssid.ora` ファイルから削除し、任意で `listener.ora` ファイルを設定します。廃止された初期化パラメータのリストは、[付録 B 「初期化パラメータの変更」](#) を参照してください。

一方、Oracle Data Migration Assistant を使用すると、ある程度汎用性や制御が失われます。特に初期化パラメータの設定などに関して、アップグレード処理を完全に制御したい場合は、アップグレードを手動で行います。

注意： Oracle Data Migration Assistant は Oracle Parallel Server がインストールされているシステムをアップグレードできません。Oracle Parallel Server がインストールされている場合は、データベースを手動でアップグレードする必要があります。

どの方法を使用してデータベースをアップグレードするかを決定して、それからその方法に合った次の項の 1 つのステップを実行します。

- [Oracle Data Migration Assistant を使用したデータベースのアップグレード](#) (7-7 ページ)
- [データベースの手動でのアップグレード](#) (7-13 ページ)

Oracle Data Migration Assistant を使用したデータベースのアップグレード

Oracle Data Migration Assistant を使用してデータベースをアップグレードするには、次のステップを実行します。

1. ご使用のオペレーティング・システム固有の Oracle ドキュメントを参照して、Oracle Universal Installer をインストールおよび起動する準備を行います。
2. Oracle Universal Installer の「Welcome」画面で、「Next」をクリックします。
ある画面でヘルプが必要な場合または Oracle Universal Installer のマニュアルについて知りたい場合は、「Help」ボタンをクリックしてオンライン・ヘルプをオープンします。
3. 「File Locations」画面で、「Destination」がリリース 8.1 Oracle ホーム用に使用するディレクトリへの絶対パスであることを確認します。その後、「Next」をクリックします。

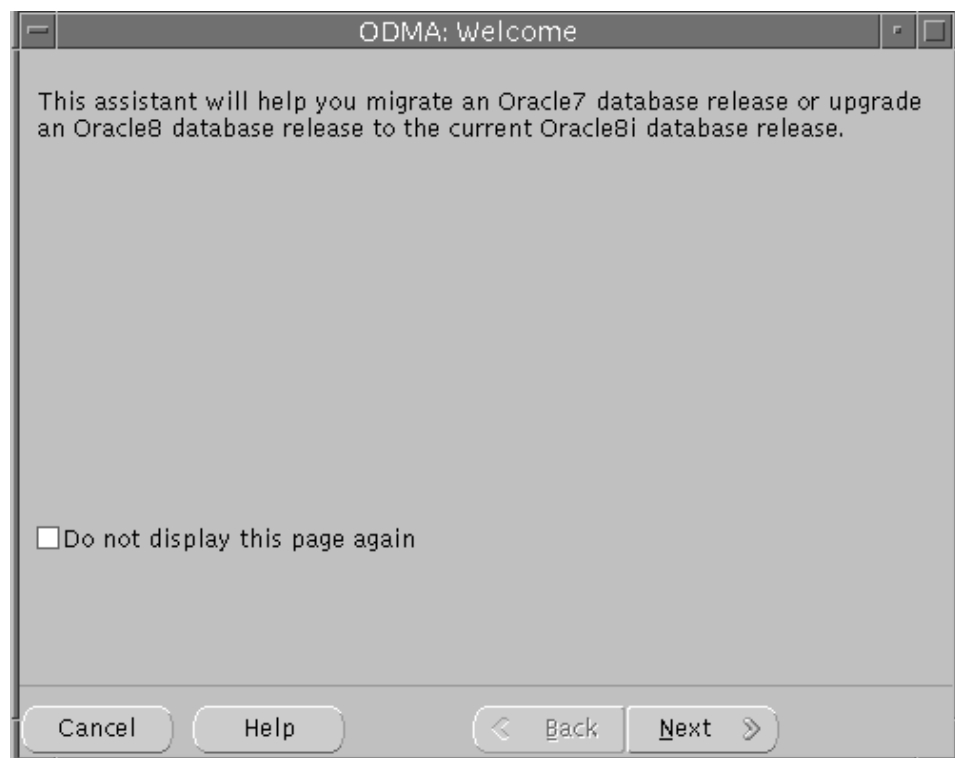
注意： 8.0 リリースからのアップグレードの場合、新しい Oracle8.1 リリースを 8.0 リリースとは別に、Oracle ホームにインストールする必要があります。ただし、8.1 リリースからのアップグレードの場合、この制限は適用されません。新しいリリースを以前のリリースのように同じ Oracle ホームにインストールできます。

4. 「Available Products」画面で、Oracle8i Enterprise Edition または Oracle8i から、インストールする Oracle8i のエディションを選択します。その後、「Next」をクリックします。
5. 「Installation Types」画面で、インストールのタイプを選択します。その後、「Next」をクリックします。
「Custom」を選択した場合、「Upgrading or Migrating an Existing Database」画が表示されるまで、カスタム・インストール設定の指定ができる画面に従ってください。
特定のオプションの使用を中止しない場合は、Oracle8 データベースにインストールしていたオプションすべてをインストールすることを確認します。たとえば、Oracle8 にアドバンスド・レプリケーションをインストールしていた場合、それを Oracle8i にインストールしなければなりません。
6. 「Upgrading or Migrating an Existing Database」画面で、次のステップを実行します。
 - a. 「Upgrade or Migrate an Existing Database」チェック・ボックスを選択します。
 - b. アップグレードするデータベースを選択します。
 - c. 「Next」をクリックします。
7. 「Summary」画面で、インストールの設定および選択が正しいかどうかを確認します。その後、「Install」をクリックします。Oracle Universal Installer はインストールを実行します。この作業は、ある程度時間がかかる場合があります。

インストレーションが終了したら、1 つ以上のアシスタントが起動する場合があります。Oracle Data Migration Assistant が起動すると、アップグレードを続行できます。

8. Oracle Data Migration Assistant の「Welcome」画面で、「Next」をクリックします。実際の画面は、オペレーティング・システムによって、[図 7-1](#) に示されるものとは異なる場合があります。

図 7-1 Oracle Data Migration Assistant の「Welcome」画面



ある画面でヘルプが必要な場合や Oracle Data Migration Assistant のマニュアルについて知りたい場合は、「Help」ボタンをクリックしてオンライン・ヘルプをオープンします。

9. 「Before You Migrate or Upgrade」画面で、アップグレードするデータベースが指定した条件を満たすことを確認します。その後、「Next」をクリックします。
10. 「Select a Database Instance」画面で、アップグレード中のデータベースのデータベース・インスタンスを選択します。その後、「Next」をクリックします。

注意： 選択するデータベースはリリース 8.0 以上でなければなりません。データベースが Oracle7 以前のデータベースの場合、アップグレードではなく、データベースの移行を実行しなければなりません。データベースが Oracle7 の場合は、Oracle Data Migration Assistant を終了して、[第 1 章](#)を参照して移行処理を開始します。

11. 「Database Password and INIT.ORA File」画面で、次のステップを実行します。
 - a. 指定された Oracle ホームが正しいことを確認します。
 - b. 指定された `initSID.ora` ファイルの場所がアップグレード中のデータベースの `initSID.ora` ファイルへの絶対パスであることを確認します。
 - c. 指定される古い Oracle ホームが、アップグレード中のデータベースの Oracle ホームへの絶対パスであることを確認します。
 - d. 「Next」をクリックします。
12. 「Choose Migration or Upgrade Type」画面で、移行のタイプを選択します。その後、「Next」をクリックします。
 「Custom」を選択した場合、「Confirm Backup」画面が表示されるまで、カスタム移行設定の指定ができる画面に従います。
13. データベースをバックアップした場合、「Confirm Backup」画面で、「Next」をクリックします。そうでない場合はバックアップを取り、アップグレードを繰り返します。
14. 「Start the Migration or Upgrade」画面で、すべての仕様が正しいことを確認します。不適切な設定がある場合、その設定が訂正されるまで「Back」をクリックします。すべて正しい場合は、「Next」をクリックします。

Oracle Data Migration Assistant がアップグレードを実行します。COMPATIBLE 初期化パラメータがアップグレード前に設定されていなかった場合は、アシスタントによって COMPATIBLE は 8.0.5 へ設定されます。ただし、COMPATIBLE がアップグレードより前に設定された場合は、元の設定が移行の後に設定されます。COMPATIBLE 初期化パラメータの再設定は、[第 8 章「互換性および相互運用性」](#)を参照してください。

アップグレード中に次のような一連のメッセージが表示される場合があります。

```
ORA-00604: 再帰 SQL レベル : 1 でエラーが発生しました。  
ORA-00001: 一意制約 : (SYSTEM.AQ$_QUEUES_CHECK) に反しています。  
ORA-06512: 2023 行 : "SYS.DBMS_AQADM"  
ORA-06512: 2 行
```

これらのメッセージは無視できます。

注意: 古い Oracle ソフトウェアをそのまま残したい場合は、決して古いソフトウェアを使用してアップグレードしたデータベースを開始しないでください。新しいリリース 8.1 インストール・ディレクトリの実行可能ファイル以外で、データベースを開始しないでください。

15. アシスタンスによって listener.ora ファイルを自動的に修正する場合、「Listener.ora Migration Confirmation」画面で、「Yes」ボタンをクリックします。また、アシスタンスによって listener.ora ファイルを修正しない場合、「No」をクリックします。

データベースが Oracle Enterprise Manager で正常に動作するためには、listener.ora ファイルにある特定の変更が必要となります。Oracle Enterprise Manager を使用する場合、listener.ora ファイルを自動的に変更するには「Yes」ボタンをクリックします。ただし、Oracle Enterprise Manager を使用しない場合は、「No」ボタンをクリックします。

「Yes」ボタンをクリックした場合、次の方法で、Oracle Data Migration Assistant は listener.ora ファイルを変更します。

- a. アシスタントは、古いリスナーおよび Oracle8i リスナーをシャットダウンします。
- b. アシスタンスは、次の方法のいずれかで、Oracle8i listener.ora の移行されたデータベースに対して、SID_DESC エントリを変更します。

簡単な場合: 古い listener.ora に次の SID_DESC エントリがあるとします。

```
...  
    (SID_DESC =  
      (SID_NAME = ORCL)  
    )  
...
```


データベース名が SAL、ドメイン名が COM および Oracle ホームが /oracle/product/8.1 の場合、アシスタントは次のエントリを追加します。

```
...
(SID_DESC =
  (GLOBAL_DBNAME = sal.com)
  (ORACLE_HOME = /oracle/product/8.1)
  (SID_NAME = SAL)
)
...
```

少し複雑な場合: 古い listener.ora に次の SID_DESC エントリがあるとします。

```
...
(SID_DESC =
  (GLOBAL_DBNAME = an_entry)
  (SID_NAME = ORCL)
)
...
```

an_entry が移行したデータベースの GLOBAL_DBNAME と一致しない場合、データベース名が SAL、ドメイン名が COM および Oracle ホームが /oracle/product/8.1 の場合、アシスタントは次のエントリを追加します。

```
...
(SID_DESC =
  (GLOBAL_DBNAME = sal.com)
  (ORACLE_HOME = /oracle/product/8.1)
  (SID_NAME = SAL)
)
...
```

このエントリは簡単な場合のエントリと同じですが、アシスタントはエントリ *an_entry* の SERVICE_NAMES パラメータへの追加も行います。したがって、アシスタントは次のように SERVICE_NAMES パラメータを変更します。

```
SERVICE_NAMES = sal.com, an_entry
```

- c. Windows NT では、アシスタントは移行されたデータベースのエントリを古い listener.ora ファイルから削除します。UNIX オペレーティング・システムでは、アシスタントはこの処理を行いません。
- d. アシスタントは Oracle8i リスナーを起動します。

Oracle Data Migration Assistant の単独での実行 既存のデータベースを移行するまたはアップグレードするかを指定しないで、Oracle8i をインストールした場合は、Oracle8i のインストール終了後に Oracle Data Migration Assistant を単独で実行できます。

Oracle Data Migration Assistant を独立して実行するために、次のステップを実行します。

1. Oracle Data Migration Assistant の起動

UNIX の場合、システム・プロンプトに次のコマンドを入力します。

```
odma
```

Windows NT の場合、次のように選択します。

スタート > プログラム > Oracle - *ORACLE_HOME_NAME* > Migration Utilities > Oracle Data Migration Assistant

Oracle Data Migration Assistant を起動すると、その「Welcome」画面が表示されます（7-8 ページの図 7-1 を参照）。

2. 各「Oracle Data Migration Assistant」のそれぞれのウィンドウの質問に答え、次のウィンドウへ移動する準備ができたら「Next」をクリックします。

参照： 詳細は、7-7 ページの「[Oracle Data Migration Assistant を使用したデータベースのアップグレード](#)」のステップ 8 ~ 15 を参照してください。

データベースの手動でのアップグレード

データベースを手動でアップグレードするには、次のステップを実行します。

参照： Windows NT で Oracle Parallel Server を移行している場合は、Operating System Dependent (OSD) レイヤーの詳細は、『Parallel Server for Windows NT スタート・ガイド』を参照してください。

1. Oracle Universal Installer を使用して Oracle8i をインストールします。
 - a. Oracle Universal Installer のインストールと開始は、オペレーティング・システム固有の Oracle ドキュメントを参照してください。

Oracle Parallel Server をインストールしたシステムを移行する場合、インストールの詳細は、『Oracle8i Parallel Server セットアップおよび構成ガイド』を参照してください。
 - b. Oracle Universal Installer の「Welcome」画面で、「Next」をクリックします。

ある画面でヘルプが必要な場合または Oracle Universal Installer のドキュメントについて知りたい場合は、「Help」ボタンをクリックしてオンライン・ヘルプをオープンします。
 - c. 「File Locations」画面で、「Destination」がリリース 8.1 Oracle ホーム用に使用するディレクトリへの絶対パスであることを確認します。その後、「Next」をクリックします。

注意： 8.0 リリースからのアップグレードの場合、新しい Oracle8.1 リリースを 8.0 リリースとは別に、Oracle ホームにインストールしなければなりません。ただし、8.1 リリースからのアップグレードの場合、この制限は適用されません。新しいリリースを以前のリリースのように同じ Oracle ホームにインストールできます。

- d. 「Available Products」画面で、Oracle8i Enterprise Edition または Oracle8i から、インストールする Oracle8i のエディションを選択します。その後、「Next」をクリックします。

注意： Oracle Parallel Server オプションは Oracle8i Enterprise Edition でだけ使用できます。

- e. 「Installation Types」画面で、「Custom」または「Minimal」を選択します。Oracle ソフトウェアと一緒に開始データベースをインストールする場合を除いては、「Typical」を選択しないでください。「Custom」または「Minimal」を選択すると、開始データベースのインストールを回避できます。通常、既存のデータベースを移行する場合は、開始データベースをインストールしないでください。

注意： Oracle Parallel Server インストレーションに対して、Minimal はサポートされていません。

選択後、「Next」をクリックします。

「Custom」を選択した場合、「Upgrading or Migrating an Existing Database」画面が表示されるまで、カスタム・インストレーション設定の指定ができる画面に従ってください。

特定のオプションの使用を中止しない場合は、Oracle8 データベースにインストールしていたオプションすべてをインストールすることを確認します。たとえば、Oracle8 にアドバンスド・レプリケーションをインストールしていた場合、それを Oracle8i にインストールしなければなりません。

- f. Oracle Parallel Server をインストールする場合、「Cluster Node Selection screen」画面で、ソフトウェアをインストールするノードを選択します。その後、「Next」をクリックします。
- g. 「Upgrading or Migrating an Existing Database」画面で、「Upgrade or Migrate an Existing Database」チェック・ボックスをチェックしないでください。その後、「Next」をクリックします。

「Upgrade or Migrate an Existing Database」チェック・ボックスをチェックすると、インストレーション後、Oracle Data Migration Assistant が自動的に起動します。手動によるデータベースのアップグレードの指示に従っているので、Oracle Data Migration Assistant を起動してはいけません。

注意： Oracle Data Migration Assistant は、Oracle Parallel Server の移行をサポートしません。

- h. 「Summary」画面で、インストールの設定および選択が正しいかどうかを確認します。その後、「Install」をクリックします。これによって、Oracle Universal Installer がインストールを実行します。ある程度時間がかかる場合があります。

インストールが正常に完了したら、「Exit」ボタンをクリックして「Universal Installer」をクローズします。

2. 使用する `initSID.ora` ファイルが古い環境の Oracle ホーム内に存在する場合、Oracle ホーム以外の場所へ移動またはコピーします。以前のリリースでは、`initSID.ora` ファイルのデフォルト位置は、UNIX 場合は `$ORACLE_HOME/dbs`、Windows NT の場合は `$ORACLE_HOME\database` でした。リリース 8.1 では、デフォルト位置は `$ORACLE_BASE/admin/SID/pfile` で、`SID` はご使用の Oracle インスタンス ID です。`initSID.ora` ファイルはどこにでも置けます。ただし、Oracle8i へ移行後は、Oracle7 Oracle ホーム中に置くことはできません。

`initSID.ora` ファイルが `IFILE` (インクルード・ファイル) エントリを持ち、`IFILE` 内に指定されたファイルが古い環境の Oracle ホーム内に存在する場合、`IFILE` エントリで指定されているファイルを古い環境の Oracle ホーム以外の場所へコピーします。`IFILE` エントリ内に指定されたファイルは、追加の初期化パラメータを持っています。このファイルをコピーした後で、次のステップに記述されているように、`initSID.ora` ファイルを編集したのと同じ方法で編集します。

注意： Oracle Parallel Server に対して、すべてのノードでこのステップを実行します。また、ご使用の `initDB_name.ora` ファイルが古い環境の Oracle ホーム内に存在する場合、`initDB_name.ora` ファイルを古い環境 Oracle ホーム以外の場所へ移動またはコピーします。

3. `initSID.ora` ファイルを調整して、新しいリリースで使用できるようにします。

特定の初期化パラメータは、新しい 8.1 リリースでは廃止されています。廃止されたパラメータは、新しいリリース 8.1 を開始する初期化パラメータ・ファイルからすべて削除してください。廃止されたパラメータはエラーを発生することがあります。また、新しい 8.1 リリースで構文が変更されているパラメータを変更することも必要です。新規のパラメータ、改名されたパラメータおよび廃止されたパラメータのリストは、[付録 B 「初期化パラメータの変更」](#) を参照してください。また、Oracle Parallel Server を使用する場合、廃止されている Oracle Parallel Server 初期化パラメータの詳細は、『Oracle8i Parallel Server 概要および管理』を参照してください。

JOB_QUEUE_PROCESSES 初期化パラメータを使用してスナップショットを自動的に更新する場合は、`initSID.ora` ファイルにあるこのパラメータをコメント行にしてください。また、アドバンスド・キューイングを使用し、伝播スケジュールがある場合は、JOB_QUEUE_PROCESSES および AQ_TM_PROCESSES 初期化パラメータの両方をコメント行にしてください。データベースのアップグレード後、このコメントを削除してこれらのパラメータを正常に使用できます。

リリース 8.1 では、LARGE_POOL_SIZE 設定は Oracle によって自動的に計算されます。自動設定が大きすぎる場合、アップグレードに要する時間が長くなることがあります。詳細は、B-9 ページの「[大規模プールから割り当てられたパラレル実行](#)」を参照してください。

`initSID.ora` ファイルに IFILE エントリがある場合、`initSID.ora` ファイルの IFILE エントリを変更して、ステップ 2 のコピー先の新しい場所を指定するようにします。

Oracle Parallel Server を使用する場合、`initDB_name.ora` ファイルを `initSID.ora` ファイルを変更したのと同じ方法で変更します。

これらの調整後、変更したファイルのすべてを保存することを確認します。

注意： Oracle Parallel Server に対して、すべてのノードでこのステップを実行します。また、PARALLEL_SERVER 初期化パラメータを FALSE に設定します。アップグレード処理が終了した後に、それを TRUE に戻せます。

4. 次の環境変数が、新しいリリース 8.1 ディレクトリを指定することを確認します。

- ORACLE_HOME
- PATH
- ORA_NLS
- ORACLE_BASE
- LD_LIBRARY_PATH (UNIX の場合)
- ORACLE_PATH (設定されている場合)

注意： これらの環境変数の中には、ご使用のオペレーティング・システムへ適用できないものもあります。

注意： Oracle Parallel Server に対して、すべてのノードでこのステップを実行します。

参照： ご使用のオペレーティング・システムでのその他の重要な環境変数の設定は、オペレーティング・システム固有の Oracle8i インストール・ドキュメントを参照してください。

5. システム・プロンプトで、`$ORACLE_HOME/rdbms/admin` ディレクトリへ移動します。
6. Server Manager を起動します。
7. データベース・インスタンスへ接続します。

```
SVRMGR> CONNECT INTERNAL
```

8. STARTUP RESTRICT を実行します。

```
SVRMGR> STARTUP RESTRICT
```

PFILE オプションを使用して、使用する `initSID.ora` ファイルの場所を指定することができます。

廃止されたパラメータをリストするエラー・メッセージが表示されることがあります。その場合には、`initSID.ora` ファイルを編集して、リストされたパラメータを削除し、その後 STARTUP RESTRICT を再度実行します。

9. 成功したかどうかを後で確認するために、結果をログ・ファイルにスプールするようにシステムを設定します。

```
SVRMGR> SPOOL catoutu.log
```

実行するスクリプトの画面出力を参照する場合は、SET ECHO ON 文も発行できます。

```
SVRMGR> SET ECHO ON
```

10. `uold_release.sql` を実行します。ここで、`old_release` とはアップグレード前にインストールしたリリースを指します。適正スクリプトの選択は、表 7-2 を参照してください。各スクリプトで、"Old Release" 列で指定されたリリースからのダイレクト・リリースが行えます。"Old Release" はアップグレードする元のリリースです。

スクリプトを実行するには、次のように入力します。

```
SVRMGR> @uold_release.sql
```

表 7-2 アップグレード・スクリプト

古いリリース	実行スクリプト
8.0.1	ダイレクト・アップグレードはサポートされない。詳細は、7-2 ページの「アップグレード・パス」を参照。
8.0.2	ダイレクト・アップグレードはサポートされない。詳細は、7-2 ページの「アップグレード・パス」を参照。
8.0.3	u0800030.sql
8.0.4	u0800040.sql
8.0.4S	ダイレクト・アップグレードはサポートされない。詳細は、7-2 ページの「アップグレード・パス」を参照。
8.0.5	u0800050.sql
8.1.1	新しいリリースへのアップグレードは、サポートされない。
8.1.2	新しいリリースへのアップグレードは、サポートされない。
8.1.3	u0801030.sql
8.1.4	u0801040.sql

注意： アップグレードの前にインストールした古いリリースが表 7-2 にリストされていない場合は、新しいインストールの readme ファイルを参照して、適切なアップグレード・スクリプトを実行してください。

スクリプトを実行する場合、次のガイドラインに確実に従います。

- 新しいリリース 8.1 インストールで提供されるスクリプトのバージョンを使用する必要がある。
- 新しいリリースの 8.1 環境でスクリプトを実行する必要がある。
- アップグレードがいくつかのリリースにわたる場合でも、1 つのスクリプトのみを実行する必要がある。たとえば、古いリリースが 8.0.4 であった場合、`u0800040.sql` のみを実行します。

実行するスクリプトは、一定のディクショナリ表を作成および変更します。また、アップグレード先のリリースに付属するシステム・カタログ・ビューおよび PL/SQL の使用に必要なすべてのパッケージを作成する、`catalog.sql` および `catproc.sql` を実行します。

このスクリプトまたは残りのステップのスクリプトのいずれかを実行中に何らかの問題が発生した場合は、問題を解決してスクリプトを再実行してください。必要に応じて、この章に説明されているスクリプトのいずれかを実行できます。

参照： スクリプトの実行中のエラーを判断する詳細は、1-3 ページの「[スクリプトの実行](#)」を参照してください。

アップグレード中に次のような一連のメッセージが表示される場合があります。

```
ORA-00604: 再帰 SQL レベル : 1 でエラーが発生しました。  
ORA-00001: 一意制約 : (SYSTEM.AQS_QUEUES_CHECK) に反しています。  
ORA-06512: 2023 行 : "SYS.DBMS_AQADM"  
ORA-06512: 2 行
```

これらのメッセージは無視できます。

注意： アップグレード・スクリプトの実行が極端に長時間の場合、`LARGE_POOL_SIZE` の設定がそのインストールには大きすぎるために発生していることがあります。VSPARAMETER ビューを使用して `LARGE_POOL_SIZE` の設定を確認します。大きすぎるようであれば、`init sid.ora` ファイルの値を小さくします。詳細は、B-9 ページの「[大規模プールから割り当てられたパラレル実行](#)」を参照してください。

11. スクリプト結果のログ・ファイルへのスプーリングをオフにします。

```
SVRMGR> SPOOL OFF
```

次にスプール・ファイルをチェックして、すべてのパッケージとプロシージャのコンパイルが成功したかどうかを検証してください。ステップ 9 でスプール・ファイルに名前を付けましたが、その名前は `catoutm.log` でした。このファイルに問題を見つけた場合、それを訂正してください。

SET ECHO ON を指定した場合は、ここで SET ECHO OFF を行うことができます。

```
SVRMGR> SET ECHO OFF
```

12. ALTER SYSTEM DISABLE RESTRICTED SESSION を実行します。

```
SVRMGR> ALTER SYSTEM DISABLE RESTRICTED SESSION;
```

13. Oracle8i のデータベース上で、SHUTDOWN を実行します。

```
SVRMGR> SHUTDOWN IMMEDIATE
```

注意： SHUTDOWN NORMAL または SHUTDOWN IMMEDIATE を使用してください。SHUTDOWN ABORT は使用しないでください。

この正しい停止を実行すると、すべてのキャッシュをフラッシュし、バッファを消去し、その他の内部的な不整合の解消を行います。これらの手段は、新規にアップグレードした Oracle8i のデータベースの整合性と一貫性を保証するための重要な最終ステップです。

ご使用のデータベースが新しい 8.1 リリースへアップグレードされました。

注意： 古い Oracle ソフトウェアをそのまま残したい場合は、絶対に古い Oracle7 ソフトウェアを使用してアップグレードしたデータベースを開始しないでください。新しいリリース 8.1 インストレーション・ディレクトリの実行可能ファイル以外で、データベースを開始しないでください。

特定のコンポーネントのアップグレード

Oracle データベースのコンポーネントのいくつかは、一般のアップグレード処理とは異なるアップグレード処理が必要です。この項では、特定のコンポーネントのアップグレード情報を説明します。7-3 ページの「[Oracle8i の新しいリリースへのアップグレード](#)」の指示に従ってデータベースをアップグレードした後に、これらの項に説明されている処理を実行します。

次のアップグレード手順には、エクスポート / インポートが関係するものがあります。エクスポート / インポートについては、『Oracle8i ユーティリティ・ガイド』を参照してください。

アドバンスド・レプリケーションのアップグレード

Oracle システムにアドバンスド・レプリケーションをインストールしている場合は、次のステップを実行します。

1. データベースを実行している場合は、停止します。

```
SVRMGR> SHUTDOWN IMMEDIATE
```

注意： Oracle Parallel Server に対して、PARALLEL_SERVER 初期化パラメータを FALSE に設定します。アップグレード処理が終了した後に、それを TRUE に戻せます。

2. システム・プロンプトで、\$ORACLE_HOME/rdbms/admin ディレクトリへ移動します。
3. Server Manager を起動します。
4. データベース・インスタンスへ接続します。

```
SVRMGR> CONNECT INTERNAL
```

5. STARTUP RESTRICT を実行します。

```
SVRMGR> STARTUP RESTRICT
```

PFILE オプションを使用して、使用する `initsid.ora` ファイルの場所を指定することができます。

- 成功したかどうかを後で確認するために、結果をログ・ファイルにスプールするようにシステムを設定します。

```
SVRMGR> SPOOL catoutrep.log
```

実行するスクリプトの画面出力を参照する場合は、SET ECHO ON 文も発行できます。

```
SVRMGR> SET ECHO ON
```

- catrep.sql を実行します。

```
SVRMGR> @catrep.sql
```

- rold_release.sql を実行します。ここで、old_release とはアップグレード前にインストールしたリリースを指します。適正スクリプトの選択は、表 7-3 を参照してください。各スクリプトで、"Old Release" 列で指定されたリリースからのアドバンスト・レプリケーション用のダイレクト・リリースが行えます。"Old Release" はアップグレードする元のリリースです。

スクリプトを実行するには、次のように入力します。

```
SVRMGR> @rold_release.sql
```

表 7-3 アドバンスト・レプリケーションのアップグレード・スクリプト

古いリリース	実行スクリプト
8.0.1	ダイレクト・アップグレードはサポートされない。詳細は、7-2 ページの「アップグレード・パス」を参照。
8.0.2	ダイレクト・アップグレードはサポートされない。詳細は、7-2 ページの「アップグレード・パス」を参照。
8.0.3	r0800030.sql
8.0.4	r0800040.sql
8.0.4S	ダイレクト・アップグレードはサポートされない。詳細は、7-2 ページの「アップグレード・パス」を参照。
8.0.5	r0800050.sql
8.1.1	新しいリリースへのアップグレードは、サポートされない。
8.1.2	新しいリリースへのアップグレードは、サポートされない。
8.1.3	アップグレード・スクリプトは不要。アドバンスト・レプリケーションは新しいリリース 8.1 データベースで動作する。
8.1.4	アップグレード・スクリプトは不要。アドバンスト・レプリケーションは新しいリリース 8.1 データベースで動作する。

注意： アップグレードの前にインストールした古いリリースが表 7-3 にリストされていない場合は、新しいインストールの readme ファイルを参照して、アドバンスト・レプリケーションに適切なアップグレード・スクリプトを実行します。

スクリプトを実行する場合、次のガイドラインに確実に従います。

- 新しいリリース 8.1 インストールで提供されるスクリプトのバージョンを使用する必要がある。
- 新しいリリースの 8.1 環境でスクリプトを実行する必要がある。
- アップグレードがいくつかのリリースにわたる場合でも、1 つのスクリプトのみを実行する必要がある。たとえば、古いリリースが 8.0.4 であった場合、`r0800040.sql` のみを実行します。

9. スクリプト結果のログ・ファイルへのスプーリングをオフにします。

```
SVRMGR> SPOOL OFF
```

次にスプール・ファイルをチェックして、すべてのパッケージとプロシージャのコンパイルが成功したかどうかを検証してください。ステップ 6 でスプール・ファイルに名前を付けましたが、その名前は `catoutrep.log` でした。このファイルに問題を見つけたら、それを訂正してください。

SET ECHO ON を指定した場合は、ここで SET ECHO OFF を行うことができます。

```
SVRMGR> SET ECHO OFF
```

10. ALTER SYSTEM DISABLE RESTRICTED SESSION を実行します。

```
SVRMGR> ALTER SYSTEM DISABLE RESTRICTED SESSION
```

アドバンスト・レプリケーションは新しいリリースへアップグレードされました。

Oracle Parallel Server のアップグレード

ご使用の Oracle システムに Oracle Parallel Server をインストールしている場合は、次のステップを実行します。

1. SHUTDOWN IMMEDIATE を使用して、すべてのインスタンスを停止します。

```
SVRMGR> SHUTDOWN IMMEDIATE
```

注意： Oracle Parallel Server に対して、PARALLEL_SERVER 初期化パラメータを FALSE に設定します。アップグレード処理が完了した後に、それを TRUE に戻せます。

2. システム・プロンプトで、\$ORACLE_HOME/rdbms/admin ディレクトリへ移動します。
3. Server Manager を起動します。
4. データベース・インスタンスへ接続します。

```
SVRMGR> CONNECT INTERNAL;
```

5. STARTUP RESTRICT を実行します。

```
SVRMGR> STARTUP RESTRICT
```

PFILE オプションを使用して、使用する `initsid.ora` ファイルの場所を指定することができます。

6. 成功したかどうかを後で確認するために、結果をログ・ファイルにスプールするようにシステムを設定します。

```
SVRMGR> SPOOL catoutpar.log
```

実行するスクリプトの画面出力を参照する場合は、SET ECHO ON 文も発行できます。

```
SVRMGR> SET ECHO ON
```

7. catparr.sql を実行します。

```
SVRMGR> @catparr.sql
```

8. スクリプト結果のログ・ファイルへのスプーリングをオフにします。

```
SVRMGR> SPOOL OFF
```

次にスプール・ファイルをチェックして、すべてのパッケージとプロシージャのコンパイルが成功したかどうかを検証してください。ステップ 6 でスプール・ファイルに名前を付けましたが、その名前は `catoutpar.log` でした。このファイルに問題を見つけたら、それを訂正してください。

SET ECHO ON を指定した場合は、ここで SET ECHO OFF を行うことができます。

```
SVRMGR> SET ECHO OFF
```

9. ALTER SYSTEM DISABLE RESTRICTED SESSION を実行します。

```
SVRMGR> ALTER SYSTEM DISABLE RESTRICTED SESSION
```

Oracle Parallel Server は新しいリリースへアップグレードされました。

スナップショットのアップグレード

リリース 8.0 からアップグレードされた、またはリリース 8.0 データベースからインポートされたスナップショットは、リリース 8.1 で利用できる新しいサマリー管理機能を使用できません。これらの新しい機能を使用する場合は、アップグレードされたスナップショットおよびリリース 8.0 からインポートされたスナップショットのそれぞれに次のステップを実行します。

1. 問合せ再書き込み権限をスナップショットの所有者に付与します。問合せ再書き込みに対しては、ローカルのスナップショットのみ利用できます。
2. スナップショットに ALTER SNAPSHOT ENABLE QUERY REWRITE コマンドを発行します。

たとえば、SSORDERS という名前のスナップショットには、次のコマンドを発行します。

```
ALTER SNAPSHOT ssorders ENABLE QUERY REWRITE;
```

注意：「スナップショット」という言葉は、「マテリアライズド・ビュー」と同義語です。

アドバンスト・キューイング・オプションのアップグレード

この項では、アドバンスト・キューイング (AQ) オプションをアップグレードするために必要な処理を説明します。

拡張アドレス・フィールドの使用

注意： この項は Oracle のリリース 8.0.3 からアップグレードしたシステムにのみ適用されます。Oracle のリリース 8.0.3 をご使用のシステムで一度も使用したことがない場合は、この項の手順を実行するありません。7-29 ページの「[キュー表のアップグレード](#)」へ移動してください。

リリース 8.0.4 では、AQ\$_AGENT データ型の拡張アドレス・フィールドが導入されました。アドレス・フィールドは 1024 バイトに拡張されています。リリース 8.0.3 をインストールした場合、この項の手順を実行して、拡張アドレス・フィールドを使用します。

また、リリース 8.0.3 をインストールしているが、リリース 8.0.4 などの以前のリリースへアップグレードするときにアドレス・フィールドを拡張するため、この項で説明されているステップを実行した場合、次のステップを実行する必要はありません。または、AQ を使用しない場合は、これらのステップを実行する必要はありません。

ただし、リリース 8.0.3 をインストールして、以前のリリースでこれらのステップを実行せず、拡張アドレス・フィールドを使用したい場合は、ここで、この手順を実行しなければなりません。オラクル社では、拡張アドレス・フィールドを使用することをお勧めします。

拡張アドレス・フィールドを使用するときは、次のステップを実行してください。

1. データベースを停止します。

```
SVRMGR> SHUTDOWN IMMEDIATE
```

注意： Oracle Parallel Server に対して、PARALLEL_SERVER 初期化パラメータを FALSE に設定します。アップグレード処理が完了した後に、それを TRUE に戻せます。

2. `initsid.ora` フィールドで、`JOB_QUEUE_PROCESSES` および `AQ_TM_PROCESSES` 初期化パラメータをコメント行にしてください。この手順が完了した後に、これらのパラメータのコメントを削除できます。
3. システム・プロンプトで、`$ORACLE_HOME/rdbms/admin` ディレクトリへ移動します。

4. Server Manager を起動します。
5. データベース・インスタンスへ接続します。

```
SVRMGR> CONNECT INTERNAL;
```

6. STARTUP RESTRICT を実行します。

```
SVRMGR> STARTUP RESTRICT
```

PFILE オプションを使用して、使用する `init \textit{sid} .ora` ファイルの場所を指定することができます。

7. 成功したかどうかを後で確認するために、結果をログ・ファイルにスプールするようにシステムを設定します。

```
SVRMGR> SPOOL catoutaq.log
```

実行するスクリプトの画面出力を参照する場合は、SET ECHO ON 文も発行できます。

```
SVRMGR> SET ECHO ON
```

8. エクスポート・ユーティリティを使用して既存のキュー表すべての内容をエクスポートします。

データベースの既存のキュー表に存在するかどうかを判断するには、次の SQL 文を発行します。

```
SELECT owner, queue_table FROM dba_queue_tables;
```

SYSTEM.DEF\$_AQCALL および SYSTEM.DEF\$_AQERROR キュー表もエクスポートし、ステップ 12 でインポートする必要があります。これらのデフォルト・キュー表はアドバンスト・レプリケーションによって使用されます。

参照： キュー表のエクスポートの必要な手順は、『Oracle8i アプリケーション開発者ガイド アドバンスト・キューイング』を参照してください。

9. DBMS_AQADM.DROP_QUEUE_TABLE 手順を使用して、キュー表のすべてを削除します。SYSTEM.DEF\$_AQCALL および SYSTEM.DEF\$_AQERROR キュー表が削除されたことを確認します。

参照： DBMS_AQADM.DROP_QUEUE_TABLE プロシージャの詳細は、『Oracle8i アプリケーション開発者ガイド アドバンスト・キューイング』を参照してください。

10. catnoqueue.sql を実行して、既存の AQ ディレクトリ表を削除します。

```
@catnoqueue.sql
```

11. catqueue.sql を実行して、新しい型およびディクショナリ表を再定義します。

```
@catqueue.sql
```

12. インポート・ユーティリティを使用してステップ 8 でエクスポートしたキュー表をインポートします。
13. SYSTEM.DEF\$_AQCALL および SYSTEM.DEF\$_AQERROR キュー表を含むステップ 8 でエクスポートしたすべてのキュー表がステップ 12 に正しくインポートされたかを次の SQL 文を発行して確認します。

```
SELECT owner, queue_table FROM dba_queue_tables;
```

次の例では、この SQL 文を発行したときに表示される出力が示されています。

OWNER	QUEUE_TABLE
SYSTEM	DEF\$_AQCALL
SYSTEM	DEF\$_AQERROR
AQUSER1	QUEUE_TABLE1
AQUSER2	QUEUE_TABLE2

14. スクリプト結果のログ・ファイルへのスプーリングをオフにします。

```
SVRMGR> SPOOL OFF
```

次にスプール・ファイルをチェックして、すべてのパッケージとプロシージャのコンパイルが成功したかどうかを検証してください。ステップ 7 でスプール・ファイルに名前を付けましたが、その名前は catoutaq.log でした。このファイルに問題を見つけたら、それを訂正してください。

SET ECHO ON を指定した場合は、ここで SET ECHO OFF を行うことができます。

```
SVRMGR> SET ECHO OFF
```

15. ALTER SYSTEM DISABLE RESTRICTED SESSION を実行します。

```
SVRMGR> ALTER SYSTEM DISABLE RESTRICTED SESSION;
```

拡張アドレス・フィールドを使用できます。

キュー表のアップグレード

既存のキュー表をアップグレードした場合のみ、次のリリース 8.1 の AQ 拡張機能が利用できます。

- 伝播されたメッセージの元のメッセージ ID 列の追加
- 送信者の ID 列の追加
- キューおよびシステム・レベル権限
- ルール・ベースの加入
- リリース 8.0 の VARRAY に格納された履歴管理情報の個別の記憶域

既存のキュー表をアップグレードするには、DBMS_AQADM.MIGRATE_QUEUE_TABLE プロシージャを実行して、COMPATIBLE オプション用の 8.1 の指定を行います。たとえば、ユーザー SCOTT が所有する TB_QUEUE というキュー表の場合は、次のコマンドを発行します。

```
EXECUTE dbms_aqadm.migrate_queue_table (  
    queue_table => 'scott.tb_queue',  
    compatible => '8.1');
```

リリース 8.1 互換である新しいキュー表を作成するには、キュー表の所有者として接続し、DBMS_AQADM.CREATE_QUEUE_TABLE プロシージャを実行し、次の例のように COMPATIBLE オプションに 8.1 を指定します。

```
EXECUTE dbms_aqadm.create_queue_table(  
    queue_table => 'scott.tkagqtpeqt',  
    queue_payload_type => 'message',  
    sort_list => 'priority,enq_time',  
    multiple_consumers => true,  
    comment => 'Creating queue with priority and enq_time sort order',  
    compatible => '8.1');
```

注意： キュー表をアップグレードし、新しいリリース 8.1 互換キュー表を作成するために、COMPATIBLE 初期化パラメータを 8.1.0 以上に設定する必要があります。

ユーザー定義データ型のアップグレード

リリース 8.0 からリリース 8.1 へデータベースをアップグレードするときに、既存のユーザー定義データ型（オブジェクト型、ネストした表、VARRAY 型など）はリリース 8.0 表現形式のままです。表現形式はリリース 8.1 では変更され、パフォーマンスを向上するために、ディスク領域使用を最適化します。

リリース 8.1 でもリリース 8.0 の形式を使用できますが、改善された表現形式を利用できません。したがって、既存のユーザー定義データ型をアップグレードすることをお勧めします（必須ではありません）。

既存のユーザー定義データ型に新しい書式を使用するには、次のステップを実行します。

1. エクスポート・ユーティリティを使用して、リリース 8.0 ユーザー定義オブジェクト型、ネストした表、VARRAY 型を含む表（キュー表を含む）のすべてをエクスポートします。
2. エクスポートした表を削除します。
3. COMPATIBLE 初期化パラメータを設定して、データベースの互換レベルが 8.1.0 以上であることを確認します。
4. インポート・ユーティリティを使用して、ステップ 1 でエクスポートした表をインポートします。

リカバリ・カタログのアップグレード

アップグレードされたデータベースのリカバリ・カタログ・スキーマが、アップグレードしたデータベースとは別のデータベースに存在します。Recovery Manager 実行可能ファイルをリリース 8.1 へアップグレードした場合は、リカバリ・カタログもリリース 8.1 へアップグレードしなければなりません。

また、単一のリカバリ・カタログによって管理される異なるリリースの複数のデータベースがある場合は、特定の Recovery Manager リリースとリカバリ・カタログ・リリース間での互換性問題を考慮する必要があります。たとえば、Recovery Manager のリリース 8.1.3 と 8.1.4 はリリース 8.1.5 以上のリカバリ・カタログへアクセスできません。したがって、この場合は、リリース 8.1.5 以上へのリカバリ・カタログによって管理されるデータベースのすべてをアップグレードしなければなりません。Recovery Manager を使用するリカバリ・カタログの詳細は、8-39 ページの「[Recovery Manager](#)」を参照してください。

リカバリ・カタログを手動でアップグレードするには、次のステップを実行します。

1. Recovery Manager へログインして、リカバリ・カタログを接続します。

たとえば、RCAT/RCAT がリカバリ・カタログ所有者のユーザー名およびパスワードで、RECDB がネットワーク・サービス名の場合、次のように入力します。

```
rman rcvcat rcat/rcat@recdb
```

Recovery Manager の 8.1 リリースを使用して古いリカバリ・カタログへ最初に接続するときは、メッセージ RMAN-06186 が表示されます。これは、リカバリ・カタログをアップグレードしなければならないことを表します。

2. UPGRADE CATALOG コマンドを使用して、リカバリ・カタログを最新のリリースへアップグレードします。Recovery Manager によって、カタログ・アップグレードの確認のためにコマンドを 2 回入力するように指示されます。アップグレード中にエラーが発生した場合は、Recovery Manager ログ中に表示されます。

次に、リリース 8.0.4 からリカバリ・カタログをアップグレードするセッションのログを示します。

```
Recovery Manager: Release 8.1.5.0.0
```

```
RMAN-06008: 回復カタログ・データベースに接続されました。
```

```
RMAN-06186: PL/SQL パッケージ rcat.DBMS_RCVCAT バージョン 08.00.04 (RCVCAT  
データベース内) は旧式です。
```

```
RMAN> upgrade catalog
```

```
RMAN-06435: 回復カタログの所有者は rcat です。
```

```
RMAN-06442: UPGRADE CATALOG コマンドを再入力して、カタログの更新を確認してください。
```

```
RMAN> upgrade catalog
```

```
RMAN-06408: 回復カタログがバージョン 08.01.05 にアップグレードされました。
```

無効な PL/SQL モジュールの再コンパイル

utlrlp.sql スクリプトは、以前 INVALID 文中にあったすべての既存の PL/SQL モジュール（パッケージ、プロシージャ、型など）を再コンパイルします。これらの処理はオプションです。ただし、これらの処理を行うと、再コンパイルはインストール後ではなく、インストール中に確認できます。

utlrlp.sql スクリプトを実行するには、次のステップを実行します。

1. システム・プロンプトで、\$ORACLE_HOME/rdbms/admin ディレクトリへ移動します。
2. Server Manager を起動します。
3. データベース・インスタンスへ接続します。

```
SVRMGR> CONNECT INTERNAL;
```

4. utlrlp.sql を実行します。

```
SVRMGR> @utlrlp.sql
```

オラクル社では、このオプションの utlrlp.sql を実行することをお勧めします。

新リリースへのアップグレードの後

次の項では、新しい 8.1 リリースへアップグレードした後に実行する必要がある処理を説明します。

新しい TO_LOB 演算子の使用

リリース 8.1 では、新しい SQL 演算子 TO_LOB によってデータを 1 つの表の LONG 列から別の表の LOB へコピーできます。

参照： 詳細は、9-7 ページの「[LONG から LOB へのコピー](#)」を参照してください。

無効な日付制約のチェック

無効な日付制約は、無効日付操作を必要とします。無効日付操作は、日付の中の世紀を暗黙的に想定するものであり、西暦 2000 年に問題が発生します。utlconst.sql スクリプトは、データベース内のすべてのチェック制約で実行され、無効日付操作を組み込む場合は制約を無効に設定します。このスクリプトは、最後にすべての無効な制約を選択します。

注意： 以前のリリースへ移行またはアップグレードした後、utlconst.sql をすでに実行済みの場合は、再度実行する必要はありません。ただし、このスクリプトを複数回実行してもシステムが破損することはないため、実行したかどうか分からない場合は、ここで実行します。

utlconst.sql スクリプトを実行するには、次のステップを実行します。

1. システム・プロンプトで、\$ORACLE_HOME/rdbms/admin ディレクトリへ移動します。
2. Server Manager を起動します。
3. データベース・インスタンスへ接続します。

```
SVRMGR> CONNECT INTERNAL;
```

4. 次のように入力します。

```
SVRMGR> SPOOL utlresult.log
SVRMGR> @utlconst.sql
SVRMGR> SPOOL OFF
```

スクリプトを実行した後、utlresult.log ファイルには、無効な日付制約を持つすべての制約が入っています。utlconst.sql スクリプトは無効な制約を訂正せずに、使用禁止にします。必要な変更を加えた後に、無効な制約を削除するか、再作成する必要があります。

パラレル実行を使用する問題の回避

リリース 8.1 移行では、パラレル実行メッセージは大規模プールから割り当てられます。以前のリリースでは、この割当ては共有プールから行われました。この変更の結果生じる問題を避けるために、init`sid`.ora ファイルの次のパラメータを調整する必要があります。

- SHARED_POOL_SIZE
- LARGE_POOL_SIZE

参照： これらのパラメータの調整の詳細は、B-9 ページの「[大規模プールから割り当てられたパラレル実行](#)」を参照してください。

新しいリリース用の INIT`sid`.ORA ファイルの調整

Oracle の新しいリリースごとに新しい初期化パラメータが導入され、いくつかのパラメータが変更され、また廃止されます。ご使用のシステムに役立つ可能性がある新しい初期化パラメータの利点を使用するために、これらの変更に対してご使用の init`sid`.ora ファイルを調整する必要があります。

参照： リリース 8.1 の新規の初期化パラメータ、変更された初期化パラメータおよび廃止された初期化パラメータのリストは[付録 B「初期化パラメータの変更」](#)を参照してください。パラメータごとの詳細は、『Oracle8i リファレンス・マニュアル』を参照してください。

COMPATIBLE 初期化パラメータは、ご使用のデータベースの互換レベルを制御します。アップグレードしたデータベース用の互換レベルに基づいて、COMPATIBLE 初期化パラメータを、ご使用の init`sid`.ora ファイルに設定します。

参照： 詳細は、8-6 ページの「[COMPATIBLE パラメータの設定](#)」を参照してください。

OUTLN ユーザーのパスワードの変更

OUTLN ユーザーは Oracle8i インストール中に自動的に作成されます。このユーザーには DBA 権限が付与されています。ALTER USER コマンドを使用して、このユーザーのパスワードを変更します。Oracle8i は、OUTLN ユーザー・スキーマを追加して、プラン・スタビリティをサポートします。OUTLN ユーザーは、ストアド・アウトラインに関連付けられたメタデータを中枢的に管理する場所として機能します。

ご使用の listener.ora ファイルの変更

Oracle Enterprise Manager を使用してデータベース・オブジェクトを管理している場合、次のように listener.ora ファイルをデータベースの情報について設定する必要があります。

```
sid_list_listener_name =
  (sid_list =
    (sid_desc =
      (global_dbname = global_database_name)
      (oracle_home = oracle_home)
      (sid_name = sid)
    )
  )
```

GLOBAL_DBNAME はグローバル・データベース名を示します。これはデータベース名とデータベースのドメイン名から構成されています。ORACLE_HOME は Oracle ソフトウェアがインストールされたディレクトリを示します。SID_NAME はデータベース・インスタンスの Oracle システム識別子を示します。

たとえば、次の情報は LISTENER というリスナー名を持つデータベースです。SAL.COM のグローバル・データベース名は SAL の SID のあるインスタンスによって参照されます。そのとき、/oracle/product/8.1 ディレクトリにインストールされたソフトウェアが使用されます。

```
sid_list_listener =
  (sid_list =
    (sid_desc =
      (global_dbname = sal.com)
      (oracle_home = /oracle/product/8.1)
      (sid_name = sal)
    )
  )
```

Oracle Enterprise Manager を使用していない場合、Oracle8i データベース・インスタンスがリスナーで自動的にそれ自体を登録するため、データベースの情報を設定する必要はありません。

参照： listener.ora ファイルおよび SID_LIST パラメータの設定の詳細は、『Oracle8i Net8 管理者ガイド』を参照してください。

Java オブジェクトの削除

リリース 8.1.4 からアップグレードして、アップグレード前に Java オブジェクトを作成した場合は、これらのオブジェクトを削除しなければなりません。Java 形式はリリース 8.1.5 では変更されたため、リリース 8.1.4 の Java オブジェクトおよびリリース 8.1.5 以降の Java オブジェクトとは互換性がありません。8.0 リリースからアップグレードした場合は、この手順を実行する必要はありません。

リリース 8.1.4 で作成した Java オブジェクトを削除するには、次のステップを実行します。

1. システム・プロンプトで、\$ORACLE_HOME/rdbms/admin ディレクトリへ移動します。
2. SQL*Plus を開始して、SYS 権限を持つユーザーとして接続します。
3. 次のように入力します。

```
SQL> SPOOL utljavaout.log
SQL> @utljavarm.sql
SQL> SPOOL OFF
```

スプール・ファイルをチェックして、文が正常に実行されたことを確認します。

4. システム・プロンプトで、\$ORACLE_HOME/javavm/install ディレクトリへ移動します。
5. SQL*Plus を開始して、SYS 権限を持つユーザーとして接続します。
6. 次のように入力します。

```
SQL> SPOOL initjvmout.log
SQL> @initjvm.sql
SQL> SPOOL OF
```

スプール・ファイルをチェックして、文が正常に実行されたことを確認します。

アップグレード前に Java ソース、クラスおよびリソース・オブジェクトをエクスポートする場合、前述のステップを完了した後にインポート・ユーティリティを使用してインポートできます。

注意： Java オブジェクトを以前のリリースの 8.1.5 データベースと現在のリリース間でエクスポートおよびインポートすることは、保証されていません。

参照： Java ソース、クラスおよびリソース・オブジェクトのエクスポートおよびインポートの詳細は、『Oracle8i ユーティリティ・ガイド』を参照してください。

現在のリリースのワード・サイズの変更

この項では、現在のリリースのワード・サイズを変更する手順を説明します（32 ビット・ソフトウェアから 64 ビット・ソフトウェア、または 64 ビット・ソフトウェアから 32 ビット・ソフトウェア）。

参照： ワード・サイズ変更の詳細は、1-4 ページの「[ワード・サイズの変更](#)」を参照してください。

現在のリリースのワード・サイズを変更するには、次のステップを実行します。

1. Server Manager を起動します。
2. データベース・インスタンスへ接続します。

```
SVRMGR> CONNECT INTERNAL;
```

3. データベース上で、SHUTDOWN IMMEDIATE を実行します。

```
SVRMGR> SHUTDOWN IMMEDIATE
```

注意： Oracle Parallel Server に対して、すべてのインスタンスにこのコマンドを発行します。また、PARALLEL_SERVER 初期化パラメータを FALSE に設定します。アップグレード処理が完了した後に、それを TRUE に戻せます。

4. データベースの全オフライン・バックアップを実行します。

参照： 詳細は、『Oracle8i バックアップおよびリカバリ・ガイド』を参照してください。

5. 現在 32 ビット・インストレーションが存在する場合は、同じリリースの 64 ビット・バージョンをインストールします。または、現在 64 ビット・インストレーションが存在する場合は、同じリリースの 32 ビット・バージョンをインストールします。

注意： インストレーションは、オペレーティング・システム固有の作業です。インストレーションの指示については、Oracle8i のプラットフォーム固有のインストレーション・ドキュメントおよびご使用のプラットフォームでの Oracle8i の README ファイルを参照してください。

6. システム・プロンプトで、`$ORACLE_HOME/rdbms/admin` ディレクトリへ移動します。
7. Server Manager を起動します。
8. データベース・インスタンスへ接続します。

```
SVRMGR> CONNECT INTERNAL;
```

9. STARTUP RESTRICT を実行します。

```
SVRMGR> STARTUP RESTRICT
```

PFILE オプションを使用して、使用する `initstid.ora` ファイルの場所を指定することができます。

10. 成功したかどうかを後で確認するために、結果をログ・ファイルにスプールするようにシステムを設定します。

```
SVRMGR> SPOOL catoutw.log
```

実行するスクリプトを参照する場合は、SET ECHO ON 文も発行できます。

```
SVRMGR> SET ECHO ON
```

11. utlirp.sql を実行します。

```
SVRMGR> @utlirp.sql
```

utlirp.sql スクリプトは新しいデータベースが必要な形式で既存の PL/SQL モジュールを再コンパイルします。まず、このスクリプトは、一定のディクショナリ表を変更します。次に、パッケージ STANDARD と DBMS_STANDARD を再ロードします。これは、PL/SQL を使用するために必要です。最後に、パッケージ、プロシージャ、型などのすべての PL/SQL モジュールの再コンパイルをトリガーします。

12. スクリプト結果のログ・ファイルへのスプーリングをオフにします。

```
SVRMGR> SPOOL OFF
```

次にスプール・ファイルをチェックして、すべてのパッケージとプロシージャのコンパイルが成功したかどうかを検証してください。ステップ 10 でスプール・ファイルに名前を付けましたが、その名前は catoutw.log でした。このファイルに問題を見つけたら、それを訂正してください。

SET ECHO ON を指定した場合は、ここで SET ECHO OFF を行うことができます。

```
SVRMGR> SET ECHO OFF
```

13. ALTER SYSTEM DISABLE RESTRICTED SESSION を実行します。

```
SVRMGR> ALTER SYSTEM DISABLE RESTRICTED SESSION
```

ご使用のデータベースのワード・サイズが変更されました。データベースをオープンして通常に使用できます。

8

互換性および相互運用性

この章では、異なる Oracle のリリース間で発生する可能性のある互換性の問題、および相互運用性の問題について説明します。これらの違いは、一般的なデータベース管理および既存のアプリケーションに影響する可能性があります。

この章では、次のトピックについて説明します。

- [互換性の概要](#)
- [8.1.0 以降の互換性レベルを必要とする機能](#)
- [相互運用性の概要](#)
- [互換性および相互運用性の問題](#)

互換性の概要

Oracle の新しいリリースにアップグレードする場合、特定の新機能によって、前回のリリースとの互換性がなくなることがあります。次については、アップグレードした Oracle データベースは前回のリリースとの互換性がなくなります。

- ご使用の前回のリリースでは処理できない、すべてのデータをディスクに格納する新機能（データ・ディクショナリ変更を含む）
- 以前の環境と比較すると、新しい環境では異なった動作をする既存の機能。この種の非互換性は言語的非互換性として分類されます。

COMPATIBLE パラメータ

Oracle では、COMPATIBLE 初期化パラメータを使用して、データベースの互換性を制御できます。COMPATIBLE パラメータが `initSID.ora` ファイルに設定されない場合、最も低いリリース値（すべての 8.0 および 8.1 リリースでは 8.0.0）をデフォルトで設定します。COMPATIBLE パラメータをより高い値にリセットしない限り、ご使用のデータベースがリリース 8.0 との互換性に問題を起こすような新機能は使用できません。

このデフォルトの動作には、次のような利点があります。

- デフォルトによって前回のリリースとの互換性が維持されるため、ダウングレードしやすい。
- 1 つ以上のデータベースを使用する環境で操作する場合、アップグレードされたデータベースは、まだアップグレードされてないデータベースと互換性がある。

ただし、デフォルト設定では、COMPATIBLE パラメータを設定しない場合、新しいリリースの機能の多くが使用できません。

参照： 新しいリリースのこれらの機能については、8-9 ページの「[8.1.0 以降の互換性レベルを必要とする機能](#)」を参照してください。

Oracle のリリース 8.1 のインストール中にインストールする製品によっては、Oracle Universal Installer は COMPATIBLE パラメータをより高い値（8.1.0 など）に設定することがあります。現行の COMPATIBLE パラメータの設定に確信がない場合、`initSID.ora` ファイルを確認してください。

図 8-1 に、Oracle のリリース 8.0 およびリリース 8.1 のデフォルト設定およびその他の可能な設定を示します。

図 8-1 COMPATIBLE パラメータ

<p>デフォルトは8.0.0</p>  <p>リリース 8.0</p> <p>8.0.xにのみ 設定可能</p>	<p>デフォルトは8.0.0</p>  <p>リリース 8.1</p> <p>8.1.yまたは8.0.xに 設定可能</p>
<p>最小設定値: 8.0.0</p>	<p>最小設定値: 8.0.0</p>
<p>最大設定値: ご使用の現行リリース</p>	<p>最大設定値: ご使用の現行リリース</p>
<p>設定不可能な値</p> <ul style="list-style-type: none"> ・ Oracle7以前のリリース ・ 現行リリース以降のリリース (8.1.0以上を含む) 	<p>設定不可能な値</p> <ul style="list-style-type: none"> ・ Oracle7以前のリリース ・ 現行リリース以降のリリース (8.2.0以上を含む)

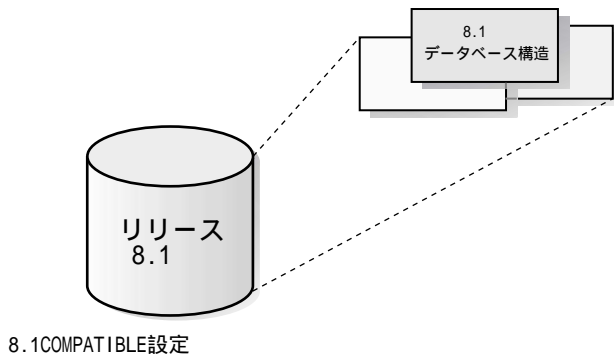
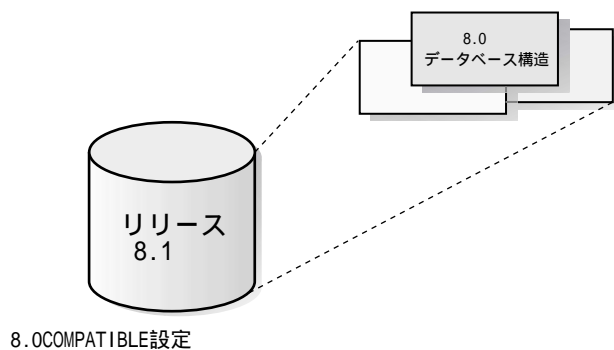
COMPATIBLE パラメータの動作

COMPATIBLE パラメータは、次のように動作します。

- Oracle データベースの動作を制御する。たとえば、COMPATIBLE パラメータを 8.0.0 に設定したリリース 8.1 データベースを実行する場合、リリース 8.1 データベースはリリース 8.0 と互換性のあるデータベース構造をディスクに生成します。つまり、COMPATIBLE の設定によって、機能を使用可能または不可能にできます。データベースと COMPATIBLE パラメータ設定との間の互換性をなくすような新機能を使用する場合、エラーが戻ります。ただし、互換性をなくすような変更をディスクに行わない新機能はすべて使用できます。

- データベースが、データベース自体の設定と互換性があることを確認する。データベースがその設定と互換性がない場合、データベースは起動せず、エラーで終了します。この場合、`initsid.ora` ファイルにある COMPATIBLE パラメータを、データベースに対する適切な値に設定する必要があります。

図 8-2 COMPATIBLE パラメータ設定によるデータベース構造



参照： データベース構造の詳細は、『Oracle8i 概要』を参照してください。

ダウングレードおよび互換性

新しいリリースへ1度アップグレードまたは移行すると、COMPATIBLE パラメータを新しいリリースと一致するように設定できます。これによって、新しいリリースの機能をすべて使用できますが、前回のリリースにダウングレードすることがより難しくなることがあります。ダウングレードする場合、ダウングレード先のリリースとの非互換性をすべて削除する必要があります。その処理には時間と手間がかかります。

参照： ダウングレードの詳細は、[第 12 章「古いバージョン 8 リリースへのダウングレード」](#)を参照してください。

ご使用の互換性のレベル

ご使用のデータベースの互換性レベルは、COMPATIBLE パラメータの設定に従います。たとえば、COMPATIBLE パラメータを 8.1.5 に設定すると、データベースは 8.1.5 の互換性レベルで実行します。

現行の COMPATIBLE パラメータ設定の確認

現行の COMPATIBLE パラメータの設定をチェックするには、次のコマンドを発行します。

```
SELECT name, value, description FROM v$parameter  
WHERE name='compatible';
```

COMPATIBLE パラメータを設定する場合

COMPATIBLE パラメータは、移行、アップグレードまたはダウングレードの各処理において特定の時点で設定する必要があります。適切な章の記載の手順に従い、設定を指示された場合のみ、COMPATIBLE パラメータを設定するようにしてください。

注意： 移行、アップグレードまたはダウングレード処理が完了すれば、必要に応じて COMPATIBLE パラメータを設定できます。

COMPATIBLE パラメータの設定

COMPATIBLE パラメータを設定するとき、次のステップを実行します。

1. COMPATIBLE パラメータ（オプション）を設定する前に、データベースのバックアップを行います。

COMPATIBLE パラメータの設定によって、データベースと Oracle の前回のリリースとの間に互換性を持たなくなることがあります。バックアップを取っていれば、必要に応じて以前のリリースに戻すことができます。

参照： バックアップの詳細は、『Oracle8i バックアップおよびリカバリ・ガイド』を参照してください。

2. COMPATIBLE パラメータを低い値に設定し直す場合、データベースがその設定と互換性があることを確認します。また、COMPATIBLE パラメータを高い値に変更する場合、ステップ 4 へ進みます。

COMPATIBLE パラメータを低くし、8.0.x に設定する場合、12-2 ページの「[非互換性の削除](#)」を参照して、リリース 8.0 との非互換性の削除に関するサブセクションにある指示に従ってください。

また、データベースを 8.1.0 以上の互換性レベルで作成した場合、Oracle は、リリース 8.0 と互換性のない特定のシステム定義型を作成します。これらの非互換性を削除するには、リリース 8.1 に付属する utldst.sql スクリプトを実行します。

```
@utldst.sql
```

3. COMPATIBLE パラメータを低い値に変更する場合、ALTER DATABASE RESET COMPATIBILITY を実行します。

```
ALTER DATABASE RESET COMPATIBILITY;
```

COMPATIBLE パラメータを高い値に変更する場合、ステップ 4 へ進みます。

参照： 詳細は、8-7 ページの「[ALTER DATABASE RESET COMPATIBILITY について](#)」を参照してください。

4. データベースを停止します。

```
SHUTDOWN IMMEDIATE
```

5. `initSID.ora` のファイルを編集して、COMPATIBLE 設定を入力または変更します。

たとえば、COMPATIBLE パラメータを 8.1.0 に設定するには、`initSID.ora` のファイルに次のように入力します。

```
COMPATIBLE=8.1.0
```

注意： 8.0 以降のリリースへ移行した場合、COMPATIBLE パラメータに Oracle7 のいかなるリリースの値も設定できません。また、現行のリリース以降の値にすることもできません。

6. STARTUP を使用してデータベースを起動します。

ALTER DATABASE RESET COMPATIBILITY について

互換性レベルを以前のリリースへ変更するように Oracle に指示するには、ALTER DATABASE RESET COMPATIBILITY 文を使用します。たとえば、一時表のような一部の Oracle の機能は 8.1.0 以上の互換性レベルを必要とします。COMPATIBLE パラメータを 8.1.0 以上へ設定して一時表を作成する場合、その表はデータベース中で、8.1.0 と互換性を持つオブジェクトになります。

ALTER DATABASE RESET COMPATIBILITY は、ありうる最低の互換性レベル（8.0.0）とは互換性のないオブジェクトを作成した各機能をチェックします。特定の機能に対して互換性のないオブジェクトが存在しないことが確認されると、その機能の互換性レベルは 0.0.0 に設定されます。これは、その機能が使用されていないことを意味します。ただし、特定の機能によって作成された互換性のないオブジェクトが確認されると、その機能に対する互換性レベルは、必要とされる互換性レベルに設定されます。

たとえば、1 つ以上の一時表領域が存在する場合、8.1.0 がその一時表領域の機能に必要とされる互換性レベルであるとき、互換性レベルは 8.1.0 に設定されます。ただし、ALTER DATABASE RESET COMPATIBILITY では、データベースの互換性レベルを上げることができないことに注意してください。8.1.0 の互換性レベルを必要とするデータベース・オブジェクトを作成する前に、まず COMPATIBLE パラメータをより高い値（8.1.0 など）に設定する必要があります。

データベースをクローズし、COMPATIBLE パラメータを低い値に変更してからデータベースをオープンすると、Oracle は各機能の互換性レベルをチェックします。init`sid`.ora のファイル内で、COMPATIBLE パラメータに指定した互換性レベルより上のレベルの機能がある場合、データベースはオープンできず、互換性のない機能であることを示すエラー・メッセージを表示します。

ご使用のデータベースの非互換性をすべて削除しても、データベースを停止する前に ALTER DATABASE RESET COMPATIBLE 文を実行できず、非互換性がなくてもデータベースはオープンしません。これは、データベースがデータベース内のオブジェクトに対する各機能の互換性レベルをチェックするように指示されなかったためです。これらの機能に対する互換性レベルをリセットしなかったため、Oracle は互換性を持たないオブジェクトが過去に作成されたことのみ記憶しています。ALTER DATABASE RESET COMPATIBLE 文を実行すると、互換性を持たないオブジェクトを明確にチェックするよう Oracle は指示され、互換性を持たないオブジェクトがない場合、互換性レベルをリセットします。

8.1.0 以降の互換性レベルを必要とする機能

次の表に示す機能を使用するには、COMPATIBLE パラメータを 8.1.0 以上に設定する必要があります。

リストに示す機能は、リリース 8.1 で紹介される新機能のすべてではありません。リストに示す機能は、8.1.0 以上の互換性レベルを必要とするリリース 8.1 の新機能のみです。つまり、一部の新機能は 8.1.0 以上の互換性レベルを必要としません。

参照： 次に示す新機能に関するエントリについては、『Oracle8i Generic Documentation Master Index』を参照してください。

アプリケーション

表 8-1 アプリケーション：8.1.0 以降の互換性レベルを必要とする機能

機能領域	8.1.0 以上の互換性レベルを必要とする機能
Java	Enterprise JavaBeans ストアド・プロシージャの Java コード SQLJ トランスレータ
Oracle コール・インタフェース (OCI)	クライアント通知のサポート

表領域

表 8-2 表領域：8.1.0 以上の互換性レベルを必要とする機能

機能領域	8.1.0 以上の互換性レベルを必要とする機能
表領域	ローカルで管理される表領域 オンライン読取り専用表領域 トランスポート可能な表領域

スキーマ・オブジェクト

表 8-3 スキーマ・オブジェクト : 8.1.0 以上の互換性レベルを必要とする機能

機能領域	8.1.0 以上の互換性レベルを必要とする機能
表	列削除のサポート 一時表
索引構成表	索引構成表のキー要約サポート 索引構成表の二次索引 索引構成表の LOB 索引構成表のパーティション化
索引	ビットマップ索引の保護 (ビットマップ索引が ALTER TABLE コマンドによって無効にならない) 拡張可能な索引作成 機能に基づいた索引 索引セグメントの結合 索引のキー要約サポート オンライン索引 (再) 構築
ハッシュ・クラスタ	単一表ハッシュ・クラスタ

パーティション化

表 8-4 パーティション化 : 8.1.0 以上の互換性レベルを必要とする機能

機能領域	8.1.0 以上の互換性レベルを必要とする機能
表のパーティション化	索引構成表のパーティション化 LOB がある表のパーティション化 オブジェクト表のパーティション化 ユーザー定義型がある表のパーティション化。オブジェクト、REF、VARRAY およびネストした表がある列を含みます。 コンポジット・メソッドおよびハッシュ・メソッドを使用する表のパーティ ション化
ハッシュ・クラスタのパーティ ション化	ハッシュ・パーティション化のサポート

組込みデータ型

表 8-5 組込みデータ型 : 8.1.0 以上の互換性レベルを必要とする機能

機能領域	8.1.0 以上の互換性レベルを必要とする機能
LOB (ラージ・オブジェクト)	LOB がある表のパーティション化 一時 LOB CLOB および NCLOB の可変幅キャラクタ・セット
ROWID	UROWID (ユニバーサル ROWID) 注意 : 新しい UROWID データ型を (表のように) 使用するには、8.1.0 の互 換性レベルが必要です。データベース・オブジェクトの一部として必要と します。UROWID 変数は、互換性レベルが 8.0 のリリース 8.1 データベ ースについては PL/SQL コードで使用できます。

8.1.0 以降の互換性レベルを必要とする機能

ユーザー定義データ型

表 8-6 ユーザー定義データ型：8.1.0 以上の互換性レベルを必要とする機能

機能領域	8.1.0 以上の互換性レベルを必要とする機能
ネストした表	コレクション・ロケータ 索引構成表内のネストした表のデータ ネストした表のビュー列にあるトリガー ネストした表のユーザー指定記憶句
オブジェクト識別子	ユーザー定義オブジェクト識別子
REF	REF 列の参照整合性制約
VARRAY	VARRAY を LOB として格納する記憶領域パラメータ 索引構成表の VARRAY データ

Oracle Parallel Server

表 8-7 Oracle Parallel Server: 8.1.0 以上の互換性レベルを必要とする機能

機能領域	8.1.0 以上の互換性レベルを必要とする機能
Oracle Parallel Server	ジョブのインスタンス親和性

データの保護

表 8-8 データの保護 : 8.1.0 以上の互換性レベルを必要とする機能

機能領域	8.1.0 以上の互換性レベルを必要とする機能
セキュリティ	アプリケーション・コンテキスト ファイン・グレイン・アクセス・コントロール N-Tier の認証と認可
データベースのバックアップ および回復	Fast Start On Demand ロールバック Oracle メディア管理 API のプロキシ・コピー・サポート

分散データベース

表 8-9 分散データベース : 8.1.0 以上の互換性レベルを必要とする機能

機能領域	8.1.0 以上の互換性レベルを必要とする機能
アドバンスド・レプリケーション	更新可能スナップショットのための列レベル・スナップショット・サブセッティング
異機種間サービス	エージェント自動登録

データ・ウェアハウジング

表 8-10 データ・ウェアハウス : 8.1.0 以上の互換性レベルを必要とする機能

機能領域	8.1.0 以上の互換性レベルを必要とする機能
マテリアライズド・ビューを使用したサマリー管理	ディメンション・スキーマ・オブジェクト 問合せリライト 問合せリライトの再書き込み権限

データ・アクセス

表 8-11 データ・アクセス : 8.1.0 以上の互換性レベルを必要とする機能

機能領域	8.1.0 以上の互換性レベルを必要とする機能
SQL および PL/SQL	自立型トランザクション 一括バインド C コール仕様 CALL 文 システム固有の動的 SQL NOCOPY パラメータ引渡しモード utlchain1.sql および utlexcpt1.sql スクリプト
アドバンスト・キューイング	伝播されたメッセージのためのオリジナル・メッセージ ID 列の追加 送信者 ID 列の追加 データベース・イベント・パブリッシュ キュー表のインスタンス親和性 非永続キュー キュー・レベル権限およびシステム・レベル権限 ルール・ベースの引用 履歴管理情報の個別記憶領域 注意 : 伝播には 8.0.4 以上の互換性レベルが必要。
パッケージ	DBMS_REPAIR パッケージ
制約	DISABLE VALIDATE 制約状態
トリガー	トリガーにおけるオブジェクト・アウトバインド ネストした表のビュー列のトリガー DATABASE および SCHEMA のトリガー トリガー本体として、プロシージャへの CALL を使用したトリガー
オブティマイザ	拡張可能なオブティマイザ オブティマイザ計画の安定性
データベース・リソース	データベース・リソース・マネージャ

Spatial 情報および Visual 情報

表 8-12 Spatial 情報および Visual 情報 : 8.1.0 以上の互換性レベルを必要とする機能

機能領域	8.1.0 以上の互換性レベルを必要とする機能
Spatial	Spatial 索引タイプ Spatial 演算子
Visual Information Retrieval (VIR)	VIR 索引タイプ VIR 演算子

注意： Spatial および VIR には、インストールする前に COMPATIBLE パラメータを 8.1.0 以上に設定する必要があります。

相互運用性の概要

相互運用性とは、異なるバージョンおよびリリース間で、分散された環境において相互に通信しともに動作する Oracle の性能を示します。Oracle の分散データベース・システムは、異なるバージョンまたはリリースの Oracle データベースを持つことができます。サポートされた Oracle のリリースはすべて、分散データベース・システムで実行できます。ただし、分散データベースとともに動作するアプリケーションは、システムの各ノードで使用できる機能を認識する必要があります。

たとえば、分散データベースのアプリケーションは、Oracle7 データベースが、リリース 8.0 以降でのみ利用可能なオブジェクト SQL 拡張子を認識することを予測できません。

注意： 相互運用性の定義は、この『Oracle8i 移行ガイド』に特有のものです。このドキュメントでは、Oracle の異なったバージョン間やリリース間で実行される移行、アップグレードおよびダウングレードについて説明しています。ただし、その他の Oracle ドキュメントは、相互運用性という用語をより幅広い定義で使用する場合があります。たとえば、相互運用性は、異なるハードウェア・プラットフォームとオペレーティング・システム間の通信を示すこともあります。

互換性および相互運用性の問題

次の項では、互換性問題および相互運用性問題を示し、その問題によって生じる問題を防ぐ処置について説明します。これらの項で説明する問題は、Oracle のリリースの違いによって発生します。

アプリケーション

リリース 8.1 の新機能を使用しない既存の (Oracle7 およびリリース 8.0) アプリケーションを変更する必要はありません。既存のアプリケーションは、同程度、または拡張された機能をリリース 8.1 に対して実行できます。リリース 8.0 ヘダウングレードしても、リリース 8.1 のデータベースに対して動作しているアプリケーションが継続して動作できるようにするには、COMPATIBLE パラメータを 8.0.5 以下に設定します。

ただし、COMPATIBLE パラメータは、ディスクのフォーマットを変更するリリース 8.1 の機能の使用を制限するのみで、その他のリリース 8.1 の機能の使用は制限しません。したがって、データベースがリリース 8.0 ヘダウングレードされる場合、8.0.5 以下に設定しても、リリース 8.1 で開発されたアプリケーションが正しく実行するという保証はありません。

参照: 第 9 章「アプリケーションのアップグレード」を参照してください。

アプリケーションの一般的な互換性問題および相互運用性問題

この項では、アプリケーションの一般的な互換性問題および相互運用性問題について説明します。

アプリケーションにアクセスされる索引構成表 アプリケーションにアクセスされる表が、通常の表から索引構成表に変更した場合、アプリケーションも変更を必要とすることがあります。この変更が必要であるかどうかは、アプリケーションが物理 ROWID またはユニバーサル ROWID (UROWID) のどちらを使用するかによって決定されます。

アプリケーションが変更を必要とするかどうかは、ROWID 値をバインドまたは定義するためにアプリケーションが使用しているホスト変数の種類によって異なります。

- アプリケーションがリリース 8.0 以降の OCI ROWID 記述子 (Pro*C の OCIROWID および Pro*COBOL の SQL-ROWID) を使用する場合、アプリケーションは変更しなくても正しく機能する。
- アプリケーションがホスト変数について DESCRIBE を常に行う場合、アプリケーションは変更しなくても正しく機能する。アプリケーションは SQLT_RDD という新しいデータ型に対応できることを確認してください。

- アプリケーションがホスト変数 SQLT_RID を使用する場合、ホスト変数 VARCHAR または ROWID 記述子を使用するようにアプリケーションを書き直す必要がある。書き直す場合、ROWID 記述子のほうが適切です。
- アプリケーションがホスト変数 CHARACTER を使用する場合、動作はそのホスト変数のサイズによる。そのサイズによって主キーを保持でき、変数が可変長文字列である場合、アプリケーションは変更せずに正しく機能します。ただし、アプリケーションが、固定サイズで 18 文字列を使用する場合、より長い可変文字列または OCI 記述子を使用するようにアプリケーションを変更する必要があります。

UROWID を使用するアプリケーションでは、ホスト変数 VARCHAR が ROWID を保持できるほど大きくない場合があります。その場合、変数の最大サイズを増加するか、OCI ROWID 記述子を使用するようにアプリケーションを変更します。OCI ROWID 記述子は不透明であり、自動的にサイズを変更するので、OCI ROWID 記述子を使用するほうが適切です。

リリース 7.1 XA コールの非互換性 Oracle8i データベースではリリース 7.1XA コールはサポートしていませんが、リリース 7.2 および 7.3 コールはサポートしています。したがって、リリース 7.1 データベースを Oracle8i へ移行した後に、対応付けられた Tuxedo アプリケーション（および XA コールを使用するその他の対応アプリケーション）を Oracle8i の XA ライブラリに再リンクします。リリース 7.2 またはリリース 7.3 のデータベースを移行する場合、再リンクする必要はありません。

ANALYZE TABLE VALIDATE STRUCTURE コマンドの動作の変更 リリース 8.1 から、ANALYZE TABLE VALIDATE STRUCTURE コマンドは最初のエラーで実行を停止しません。この変更を考慮して、この動作に基づくアプリケーションを変更します。

OCI アプリケーション

この項では、OCI アプリケーションに関する互換性問題および相互運用性問題について説明します。

参照： 詳細は『Oracle コール・インタフェース・プログラマーズ・ガイド』も参照してください。

共有構造と相互運用性 共有構造は、リリース 8.1 のライブラリとリンクした Oracle7 クライアントではサポートされていません。共有構造の利点を利用するには、アプリケーションをリリース 8.1 以降の OCI で作成する必要があります。また、リリース 8.1 以降の Oracle データベース・サーバと通信する必要があります。

リリース 8.0 の Oracle データベース・サーバにアクセスしているリリース 8.1 の OCI クライアントのみが、共有構造の利点の一部を利用できます。クライアントおよび Oracle データベース・サーバの両方がリリース 8.0 以前の場合、共有構造はサポートされません。

スレッド・セーフティ ORLON コールおよび OLON コールは、バージョン 8 ではサポートされていません。ただし、単一スレッド・アプリケーションの場合でも、OLOG を使用してください。

注意： マルチスレッド・アプリケーションの場合は OLOG コールが必要になります。

OCI アプリケーションのリンク・ライン OCI アプリケーションの場合、Oracle8i のリンク・ラインは Oracle7 のリンク・ラインとは異なります。Oracle8i の OCI アプリケーションがコンパイルされる場合の Oracle8i のリンク・ラインの使用例については、\$ORACLE_HOME/rdbms/demo/demo_rdbms.mk ファイルを参照してください。

Oracle7 クライアント Oracle7 クライアントは、Oracle7 と Oracle8i のコールを組み合わせ、Oracle8i OCI の使用を選択できます。どちらのコールを使用するかによって機能性の度合いが異なります。暗号化 API およびパスワード・リセット・コールも同様に、単独ですることもできます。次の機能を使用可能にするには、処理された文のすべてのフェーズに Oracle8i の OCI を使用します。

- フェイルオーバー
- プリフェッチ
- ビギンバック方式のコミットまたは取消し
- クライアント側の変換

Oracle7 コードと Oracle8i のコードを組み合わせる場合、Oracle7 クライアントは Oracle8i コールを使用してログインする必要があります。

プリコンパイラ・アプリケーション

この項では、プリコンパイラ・アプリケーションに関する互換性問題および相互運用性問題について説明します。

参照： 詳細は、『Pro*C/C++ プリコンパイラ・プログラマーズ・ガイド』および『Pro*COBOL プリコンパイラ・プログラマーズ・ガイド』を参照してください。

Pro*C/C++ での SYSDBA 権限による接続 Pro*C/C++ で CONNECT 文を発行する場合、SYSDBA 権限はデフォルトに使用されなくなりました。リリース 8.0 では、次の CONNECT 文を使用して Pro*C/C++ の SYSDBA 権限と接続しました。

```
EXEC SQL CONNECT :sys IDENTIFIED BY :sys_passwd;
```


リリース 8.1 では、次の CONNECT 文を発行して Pro*C/C++ で SYSDBA の権限と接続します。

```
EXEC SQL CONNECT :sys IDENTIFIED BY :sys_passwd IN SYSDBA MODE;
```

Pro*COBOL での SYSDBA 権限による接続 Pro*COBOL で CONNECT 文を発行する場合、SYSDBA 権限が、デフォルトに使用されなくなりました。リリース 8.0 では、次の CONNECT 文を使用して SYSDBA 権限と接続しました。

```
EXEC SQL
    CONNECT :sys IDENTIFIED BY :SYS-PASSWD
END-EXEC.
```

リリース 8.1 では、次の CONNECT 文を発行して SYSDBA 権限と接続します。

```
EXEC SQL
    CONNECT :sys IDENTIFIED BY :SYS-PASSWD IN SYSDBA MODE
END-EXEC.
```

バージョン 8 における Ada のサポート リリース 7.3 では、Oracle は Pro*ADA 製品を公的にはサポートしていませんでした。Pro*ADA を Ada 用の SQL*Module の最新リリースにアップグレードできます。ただし、Ada8.1 用の SQL*Module は、オブジェクトをサポートしていません。

PL/SQL の下位互換性およびプリコンパイラ PLSQL_V2_COMPATIBILITY 下位互換性 (8-20 ページの「[PL/SQL アプリケーション](#)」を参照) は、次のように DBMS プリコンパイラ・コマンド行オプションを設定することによってプリコンパイラ環境で使用できます。

```
... DBMS=Oracle7
```

PL/SQL アプリケーション

この項では、PL/SQL のアプリケーションの互換性問題および相互運用性問題について説明します。

参照： 詳細は、『PL/SQL ユーザーズ・ガイドおよびリファレンス』を参照してください。

PL/SQL V2 互換モード PL/SQL V2 互換モードは、リリース 8.0 以降の PL/SQL で使用できます。このモードは、PLSQL_V2_COMPATIBILITY 初期化パラメータによって使用可能になります。

PL/SQL V2 互換モードを設定するには、次のような 3 つの方法があります。

- `initSID.ora` に次の行を追加します。

```
PLSQL_V2_COMPATIBILITY = TRUE
```

- 次の SQL 文を発行します。

```
ALTER SYSTEM SET PLSQL_V2_COMPATIBILITY = TRUE;
```

- 次の SQL 文を発行します。

```
ALTER SESSION SET PLSQL_V2_COMPATIBILITY = TRUE;
```

PLSQL_V2_COMPATIBILITY 初期化パラメータによって、次のような状況でリリース 8.0 以降の PL/SQL と PL/SQL V2 の間に互換性が得られます。

- PL/SQL V2 コンパイラを使用すると、ソース内での定義を行う前にソース内でレコード型や索引表の型を参照できます。リリース 8.0 以降の PL/SQL では、ソース内で型を参照する前に必ず型の定義を行っておく必要があります。ただし、PL/SQL V2 の互換モードを使用可能にすると、リリース 8.0 以降の PL/SQL は PL/SQL V2 と同様に動作します。
- PL/SQL V2 コンパイラは、次のような不正な構文を許可してしまいます。

```
return variable-expression
```

この構文は誤っており、次の形式に変更する必要があります。

```
return variable-type
```

こうした不正な構文が見つかったと、リリース 8.0 以降の PL/SQL のコンパイラはエラーを表示します。ただし、PL/SQL V2 の互換モードを使用可能にすると、リリース 8.0 以降の PL/SQL は PL/SQL V2 と同様に動作し、エラーは表示しません。

- PL/SQL V2 では、IN パラメータとして渡された索引表の要素を変更または削除できません。次にその例を示します。

```
function foo (x IN table_t) is
begin
x.delete(2);
end;
```

この IN パラメータの使用方法は誤っています。リリース 8.0 以降の PL/SQL では、IN パラメータの読取り専用方法を正しく施行し、IN パラメータとして渡された索引表を索引表方式から変更することはありません。ただし、PL/SQL V2 の互換モードを使用可能にすると、リリース 8.0 の PL/SQL は PL/SQL V2 と同様に動作し、そのパラメータを認めます。

- PL/SQL V2 では、レコードである IN パラメータのフィールドを (OUT パラメータとして) 引き渡すことができますが、リリース 8.0 以降の PL/SQL では、そのような引渡しは許可しません。ただし、PL/SQL V2 の互換モードを使用可能にすると、リリース 8.0 以降の PL/SQL は PL/SQL V2 と同様に動作し、この種の引渡しを許可します。
- PL/SQL V2 コンパイラでは、レコード変数である OUT パラメータのフィールドが式のコンテキスト中で使用されることを許可します (たとえば、割当て文の右側にあるドット修飾名など)。

このような OUT パラメータの使用は許可すべきではありません。リリース 8.0 以降の PL/SQL では、OUT パラメータを式のコンテキスト中で使用することは許可していません。ただし、PL/SQL V2 の互換モードを使用可能にすると、リリース 8.0 以降の PL/SQL は PL/SQL V2 と同様に動作します。

- PL/SQL V2 では、SELECT リストの FROM 句の中で OUT パラメータを使用できます。リリース 8.0 以降の PL/SQL では、OUT パラメータのそのような使用方法是許可されません。ただし、PL/SQL V2 の互換モードを使用可能にすると、リリース 8.0 以降の PL/SQL は PL/SQL V2 と同様に動作します。

キーワード動作の違い：バージョン 7 とバージョン 8 バージョン 7 とバージョン 8 の両方に含まれるさまざまなキーワードや型は、SELECT リスト内のファンクション名として使用されたとき、バージョン 7 とバージョン 8 では次のようにわずかに異なるエラー・メッセージ識別子を出力します。

表 8-13 キーワード動作の違い

キーワード	バージョン 8 の動作	バージョン 7 の動作
CHARACTER、COMMIT、DEC、FALSE、INT、NUMERIC、REAL、SAVEPOINT、TRUE	エラーの生成 (ORA-06550 および PLS-00222)	エラーの生成 (ORA-06552 および PLS-222)

起動およびシャットダウン

この項では、Oracle データベースの起動、接続およびシャットダウンに関する互換性問題および相互運用性について説明します。

シャットダウンの変更

リリース 8.0.4 以降では、SHUTDOWN IMMEDIATE となっている間、使用されていないサーバーのプロセスは停止します。したがって、ユーザーに戻されるエラーは、過去のリリースのエラーと異なります。

初期化パラメータ

新規のパラメータ、変更されたパラメータおよび廃止されたパラメータのリストについては、[付録 B「初期化パラメータの変更」](#)を参照してください。また、特定の初期化パラメータに関する互換性問題の詳細は、B-8 ページの「[初期化パラメータの互換性の問題](#)」を参照してください。

表領域とデータ・ファイル

この項では、表領域およびデータ・ファイルに関する互換性問題および相互運用性問題について説明します。

トランスポート可能な表領域

2つのデータベース間で表領域をトランスポートする場合、互換性問題が発生します。

参照： 互換性の問題については、『Oracle8i 管理者ガイド』を参照してください。

テンプファイル

リリース 8.1 では、テンプファイルを導入します。テンプファイルの詳細は、データ・ファイルの詳細とは異なった、静的データ・ディクショナリ・ビューおよび動的パフォーマンス・ビューにあります。テンプファイルの詳細を表示するには、DBA_TEMP_FILES 静的データ・ディクショナリ・ビューおよび次に示す動的パフォーマンス・ビューを参照します。

- V\$TEMPFILE
- V\$TEMP_EXTENT_MAP
- V\$TEMP_EXTENT_POOL
- V\$TEMP_SPACE_HEADER
- V\$TEMPSTAT
- V\$TEMP_PING

Oracle は、データ・ファイルとテンプファイルの両方に自動的に番号を割り当てます。2つのデータ・ファイル間では同じ番号を共有できません。同様に、テンプファイル間でも同じ番号は共有できません。ただし、テンプファイルとデータ・ファイルの間では同じ番号を共有できます。

読取り専用表領域へのアクティブ・トランザクション制限

リリース 8.1 以前のリリースでは、表領域を読取り専用にするまで、データベースではいかなるアクティブ・トランザクションも実行できませんでした。リリース 8.1 では、COMPATIBLE パラメータを 8.1 以上に設定すると、この制限がなくなります。8.1.0 以上の互換性レベルでデータベースを使用すると、ALTER TABLESPACE...READ ONLY コマンドはアクティブ・トランザクションが完了するまで待ってから、表領域を読取り専用に変更します。ただし、COMPATIBLE パラメータを 8.1.0 以下に設定する場合、制限が適応されません。

データ・ディクショナリ

この項では、データ・ディクショナリの変更によって生じる互換性問題および相互運用性問題について説明します。

参照： 廃止されたビューのリストおよび詳細は、付録 C「静的データ・ディクショナリ・ビューの変更」および付録 D「動的パフォーマンス・ビューの変更」も参照してください。

データ・ディクショナリの保護

新しい Oracle8i データ・ディクショナリの保護のメカニズムによって、SYS スキーマにユーザー表を作成し、'ANY' 権限を使用してその表にアクセスするアプリケーションに問題が生じる場合があります。たとえば、ユーザーは、AUD\$ 表内の監査レコードを削除する DELETE 文を使用するために DELETE CATALOG ROLE を所有する必要があります。

SYS スキーマでユーザー表を作成し、それらにアクセスすることは安全性の面で保証できません。したがって、アプリケーション側ではオブジェクトを別のスキーマに移動することが要求されます。一時的な互換性を確保するために O7_DICTIONARY_ACCESSIBILITY という初期化パラメータを使用しますが、このスイッチは暫定的に使用するだけです。

アプリケーションから SYSDBA オプションなしでユーザー SYS に接続することは避けるべきです。ユーザー SYS に接続してパスワードを共有するのではなく、一般ユーザーに DBA 権限を付与し、ユーザーが SYSDBA としてデータベース接続して SYS スキーマに接続します。

廃止されたデータ・ディクショナリ・ビュー

catalog5.sql ファイルおよび catalog6.sql ファイルで作成され、Oracle バージョン 5 およびバージョン 6 に対する下位互換性のためにバージョン 7 で保持されていた特定のデータ・ディクショナリ・ビューは、バージョン 8 では廃止されています。これらのデータ・ディクショナリ・ビューに対する参照元は、データベース・ツールとアプリケーションから削除します。

スキーマ・オブジェクト

この項では、スキーマ・オブジェクトに関する互換性問題および相互運用性問題について説明します。

ビットマップ索引の保護

リリース 8.1 以前のリリースでは、特定の SQL 文を発行することによって、ビットマップ索引を意図せずに無効にできました。ビットマップ索引が無効になる最も一般的な原因は、次に示す種類の文によるものです。

- あらかじめ主キーを持たない既存の表に主キーを設定する ALTER TABLE 文
- あらかじめ NOT NULL 制約を持たない列に NOT NULL 制約を設定する ALTER TABLE 文

COMPATIBLE パラメータが 8.1.0 以上に設定される場合、リリース 8.1 ではこれらの意図しない無効化が起こらないようにします。ただし、COMPATIBLE パラメータが 8.1.0 以下に設定される場合、ビットマップ索引は保護されず、以前に説明した特定の SQL 文は、ビットマップ索引を無効にする可能性があります。

データ型

この項では、データ型に関する互換性問題および相互運用性問題について説明します。

ラージ・オブジェクト (LOB)

この項では、LOB に関する互換性問題および相互運用性問題について説明します。

CLOB および NCLOB の可変幅キャラクタ・セット リリース 8.0 では、データベース・キャラクタ・セットが可変幅の場合、SYSTEM 以外のユーザーは CLOB または NCLOB のデータ型がある表を作成できませんでした。リリース 8.1 では、可変幅キャラクタ・セットを持つ表で CLOB および NCLOB データ型をサポートします。データは UCS2 (2 バイトの固定値ユニコード) として格納されます。

COMPATIBLE パラメータを 8.1.0 以下に設定する場合、この機能は次のように制限されます。

- データベース・キャラクタ・セットが可変幅の場合、SYSTEM ユーザーでないユーザーは CLOB および NCLOB データ型を含む表を作成できません。
- SYSTEM ユーザーは、CLOB および NCLOB データ型を含む表を作成できても、CLOB または NCLOB 列にデータを格納できません。

COMPATIBLE パラメータを 8.1.0 以上に設定すると、これらの制限は適用されません。

LOB 索引句 LOB 索引句は、Oracle の今後のリリースではサポートされない予定です。LOB 格納用の表領域から離れた表領域で、LOB 索引データを格納するために LOB 索引句を使用した場合、索引データは、LOB と同じ表領域に属するように再配置されます。これには、リリース 8.1 で次の操作のいずれかを実行する必要があります。

- LOB についてエクスポート / インポートを実行する
- LOB をパーティション表へ移動する

Oracle7 から Oracle8i へ移行するために、エクスポート / インポートを使用した場合、索引データは移行の間に自動的に再配置されます。ただし、移行ユーティリティを使用した場合、索引データは再配置されません。

また、リリース 8.1 で新しい表を作成し、非パーティション表の LOB 索引のために表領域を指定する場合、表領域の仕様は無視され、LOB 索引は LOB と同じ表領域に配置されます。

LOB 索引の格納を確認するには、SYSDBA として接続し、次の SQL 文を発行します。

```
SELECT index_name, index_type, tablespace_name
FROM dba_indexes
WHERE index_type = 'LOB';
```

Oracle ROWID

この項では、ROWID に関する互換性問題および相互運用性問題について説明します。

新規の物理 ROWID データ型書式 バージョン 8 では、物理 ROWID の書式が変更されました。列またはアプリケーション・コードに格納された物理的 ROWID を使用する場合、以前の物理 ROWID は無効であるため、変換する必要があります。

参照： 新規の物理 ROWID 書式の詳細は、[第 11 章「物理 ROWID の移行について」](#)を参照してください。

UROWID データ型 UROWID (ユニバーサル ROWID) データ型はリリース 8.1 で導入された新機能です。リリース 8.1 以前のクライアントは、キャラクタ・ホスト変数のみを使用して UROWID データ型の列にアクセスできます。つまり、他の変数の型はサポートされません。

NCHAR および NLS の使用

バージョン 8 では、特定の列および属性、PL/SQL 変数、パラメータ、戻り値に各国文字 (NCHAR) を使用することを宣言できます。そのような明示的な宣言がない限り、NCHAR および NLS の使用はほとんどが透過的に処理され、バージョン 8 の他の機能には影響しません。例外としては、PROPSS または VALUES ディクショナリ・ビューにある SELECT 文で、CHARACTER_SET_NAME 列または NLS_NCHAR_CHARACTERSET 行を戻す場合があります。

NCHAR および NLS に伴う移行問題 PROPSS ディクショナリ表には、CREATE DATABASE 文で指定されたキャラクタ・セットを記述する 2 つの行があります。NAME='NLS_CHARACTERSET' を含む行では、VALUES 列にデータベース・キャラクタ・セットの名前があります。NAME='NLS_CHARACTERSET' を含む行では、VALUES 列に各国文字キャラクタ・セットの名前があります。

リリース 7.3 と比較すると、さまざまなビューに、次の値を持つ新規列 CHARACTER_SET_NAME が含まれます。

```
DECODE(x$.CHARSETFORM,
      1, 'CHAR_CS',
      2, 'NCHAR_CS',
```

ここで、x\$ は実表の 1 つを表します。ビューの DATA_TYPE または COLTYPE 列値は、キャラクタ・セットの選択項目を表示するようには変更されません。

NCHAR と NLS の互換性および相互運用性 リリース 7.1 以降のクライアントは、データを各国文字キャラクタ・セットで保持するバージョン 8 のサーバーとやりとりできます。出力された各国文字データがバインド・ハンドルまたは定義ハンドルを通る場合、クライアントのデータベース・キャラクタ・セットへの変換は OCI によって透過的に処理されます。

データが入力バインド値として必要な場合、および各国文字キャラクタ・セット文字列のみが許可されている場所でデータを使用する場合、その値を使用する SQL または PL/SQL コードでバインド変数を使用している箇所を囲みます。バインド変数を各国文字キャラクタ・セットに変換する CSCONVERT() へのコールによって、このバインド変数はデータベース・キャラクタ・セットを保持しているとみなされます。この場合、クライアントは、クライアントのデータベース・キャラクタ・セットでしかデータを渡せません。クライアントのデータベース・キャラクタ・セットには、各国文字セットの全種類が揃っていない可能性があります。

バージョン 6 や Oracle7 のクライアントは、ANY_CS パラメータを使用して、サブプログラムに対するリモート・プロシージャ・コール (RPC) を行えます。ANY_CS パラメータは、その実引数にサーバーのデータベース・キャラクタ・セットが使用されているものと解釈します。バージョン 6 や Oracle7 のクライアントは、各国文字キャラクタ・セットの使用を宣言しているパラメータを持つサブプログラムに対しては RPC を行えません。また、戻り値に各国文字キャラクタ・セットが使用されている関数に対しても RPC を行えません。このような禁止事項の違反は、コンパイル時ではなく実行時に検出されます。

PL/SQL に埋め込まれた SQL コードを使用しているバージョン 6 または Oracle7 のクライアントは、SQL 文で CSCONVERT() 関数または 2 番目の引数を持つ CHR() を直接使用することができません。これは、バージョン 6 や Oracle7 のクライアントの PL/SQL パッケージである STANDARD に、これらの関数が含まれていないためです。この非互換性を訂正するために、サーバー上に次のような 2 つのストアド・プロシージャを作成することができます。

```
CREATE OR REPLACE FUNCTION to_nchar_cs(t VARCHAR2) RETURN NCHAR VARYING
IS BEGIN
    RETURN csconvert(t, nchar_cs);
END;
/
```

```
CREATE OR REPLACE FUNCTION to_char_cs(t nchar varying) RETURN VARCHAR2
IS BEGIN
    RETURN csconvert(t, char_cs);
END;
/
```

NCHAR/NLS 環境変数および互換性 次に示すように、使用環境に NLS_LANG を設定する必要があります。

- リリース 7.3.x の環境に ORA_NLS32 環境変数を設定する
- バージョン 8 の環境に ORA_NLS33 環境変数を設定する

クライアントに正しい NLS キャラクタ・セットの環境変数があることを確認します。リリース 7.3 の NLS コードがバージョン 8 のキャラクタ・セットをロードしようとすると、エラーが発生します。

ユーザー定義データ型

リリース 8.1 では、ユーザー定義データ型に新しい書式を導入しています。新しい書式によって、リリース 8.0 で使用されていた書式でよりも、パフォーマンスが著しく向上します。新しいユーザー定義データ型書式を使用するには、COMPATIBLE パラメータを 8.1.0 以上に設定する必要があります。

リリース 8.0 のユーザー定義データ型をリリース 8.1 のデータベースで使用しても互換性の問題はありません。ただし、データベースは、新しい書式によるパフォーマンスの向上という利点は得られません。

リリース 8.0 ユーザー定義データ型へアクセスするリリース 8.1 クライアント

ユーザー定義データ型書式は、クライアントおよびサーバー間の互換性のやりとりの一部として交渉されます。リリース 8.0 データベース・サーバー、または COMPATIBLE パラメータを 8.0.x に設定したリリース 8.1 データベース・サーバーをご使用の場合、リリース 8.1 クライアントはデータベースにアクセスできますが、そのクライアントはリリース 8.0 として設定されます。

リリース 8.1 ユーザー定義データ型へアクセスするリリース 8.0 クライアント

リリース 8.0 クライアントがリリース 8.1 ユーザー定義データ型のあるサーバーにアクセスする場合は、データベースがユーザー定義データ型をリリース 8.0 書式に変換します。したがって、リリース 8.0 のクライアントはデータにアクセスできますが、パフォーマンスは低下することがあります。

ネストした表

リリース 8.0 のクライアントは、リリース 8.1 のネストした表の次の機能はサポートしません。

- コレクション・ロケータ
- 索引構成表にあるネストした表のデータ記憶域を含む、コレクション列のユーザー指定記憶域

したがって、リリース 8.0 のクライアントがリリース 8.1 のサーバーにアクセスしようとし、ネストした表がロケータとして戻されるように指定されてる場合、またはネストした表の記憶域がユーザ指定の場合、非互換エラーとなりアクセスできません。

LOB として格納される VARRAY

リリース 8.0 のクライアントは、VARRAY を LOB として格納する記憶領域パラメータの仕様をサポートしません。したがって、リリース 8.0 のクライアントが、VARRAY を LOB として格納する格納パラメータ仕様のあるリリース 8.1 のサーバーにアクセスしようとすると、非互換エラーとなりアクセスできません。

SQL と PL/SQL

この項では、SQL および PL/SQL に関する、互換性問題および相互運用性問題について説明します。

参照： SQL および PL/SQL の詳細は、『Oracle8i SQL リファレンス』および『PL/SQL ユーザーズ・ガイドおよびリファレンス』を参照してください。

関数の非サポート GREATEST_LB、LEAST_UB および TO_LABEL

リリース 8.1.5 以降の Oracle8i では、Trusted Oracle ラベルで動作する内蔵 PL/SQL 関数 (GREATEST_LB、LEAST_UB および TO_LABEL) はサポートされません。

PL/SQL のネイティブ動的 SQL

次の項では、PL/SQL のネイティブ動的 SQL に関する相互運用性の問題を説明します。

サーバー側の PL/SQL リリース 8.1 以降の互換性レベルを持つ Oracle データベース・サーバーでは、リモート・サーバーにあるオブジェクトへの参照を含むネイティブ動的 SQL 文をどの互換性レベルでも実行できます。

たとえば、次のプロシージャはネイティブ動的 SQL 文を持ち、リモートの Oracle データベース・サーバーにリンクします。

```
PROCEDURE dyn1 is
BEGIN
    EXECUTE IMMEDIATE 'insert into tab@remote_link
        values ('a', 10)';
END;
```

前述の例の *remote_link* は、リリース 7.3、8.0 または 8.1 などのすべての Oracle バージョンへのリンクとして使用できます。

ネイティブ動的 SQL および RPC コール RPC コールを作成するクライアントのリリースにかかわらず、RPC コールのターゲットである PL/SQL プログラムはネイティブ動的 SQL を使用できます。たとえば、リリース 7.3 または 8.0 のクライアントは、リリース 8.1.0 以上の互換性レベルで Oracle データベース・サーバに RPC コールを発行することができます。

SQL スクリプト utlchain.sql および utlchain1.sql

リリース 8.1 のインストールでは、移行された行および連鎖された行を格納する表を、utlchain.sql および utlchain1.sql の 2 つのスクリプトを使用して作成します。utlchain1.sql スクリプトは、通常の表および索引構成表の両方で実行できますが、utlchain.sql スクリプトは、通常の表でのみ実行でき、索引構成表では実行できません。

正しい実行スクリプトは、ご使用のデータベースの互換性レベルによって異なります。

- 互換性レベルが 8.1.0 未満の場合、utlchain.sql を実行
- 互換性レベルが 8.1.0 以降の場合、utlchain1.sql を実行

SQL スクリプト utlexcpt.sql および utlexcpt1.sql

リリース 8.1 のインストールでは、使用可能にする制約の例外を格納する表を utlexcpt.sql および utlexcpt1.sql の 2 つのスクリプトを使用して作成します。utlexcpt1.sql スクリプトは、通常の表および索引構成表の両方で実行できますが、utlexcpt.sql スクリプトは、通常の表でのみ実行でき、索引構成表では実行できません。

正しい実行スクリプトは、ご使用のデータベースの互換性レベルによって異なります。

- 互換性レベルが 8.1 未満の場合、utlexcpt.sql を実行
- 互換性レベルが 8.1 以降の場合、utlexcpt1.sql を実行

AS 副問合せを持つパラレル CREATE TABLE 文の動作変更

8.0 以降のリリースで、AS 副問合せを持つ CREATE TABLE 文に PARALLEL 句を使用すると、INITIAL 記憶パラメータは無視され、NEXT 記憶パラメータが使用されます。Oracle7 では、INITIAL 記憶パラメータは無視されません。

次に、SQL 文の例を示します。

```
CREATE TABLE tb_2 STORAGE (INITIAL 1M NEXT 500K)
PARALLEL (DEGREE 2)
AS SELECT * FROM tb_1;
```

8.0 以降のリリースでは、INITIAL の値は 500K となりますが、Oracle7 では 1M となります。

アドバンスト・キューイング (AQ)

この項では、AQ の互換性問題および相互運用性問題について説明します。

参照： AQ の詳細は、『Oracle8i アプリケーション開発者ガイド アドバンスト・キューイング』を参照してください。次の項では、新しい AQ 機能の互換性および相互運用性のみ説明します。使用方法の詳細は、『Oracle8i アプリケーション開発者ガイド アドバンスト・キューイング』を参照してください。

キュー・レベル権限およびシステム・レベル権限

キュー・レベル権限およびシステム・レベル権限を使用するには、キュー表の互換性レベルが 8.1.0 以上である必要があります。特に、DBMS_AQADM パッケージに次のプロシージャを使用してキュー・レベル権限を付与する場合、キュー表の互換性レベルは 8.1 以上である必要があります。

- GRANT_QUEUE_PRIVILEGE
- REVOKE_QUEUE_PRIVILEGE

相互運用性および送信者の ID 列

リリース 8.1 では、送信者の ID を付加的な属性としてメッセージ・プロパティにマップします。リリース 8.0 およびリリース 8.1 データベース間で通信する場合は、新しい属性は無視されます。

OCI のアプリケーション用に、メッセージ・プロパティの記述子の新しい属性として送信者の ID 属性を利用できます。リリース 8.1 の OCI クライアントは、メッセージ・プロパティをサーバーへ送信またはサーバーから受信するために新しい RPC コードを使用します。

ルール・ベースの引用

DBMS_AQADM.MIGRATE_QUEUE_TABLE プロシージャを使用してリリース 8.0 からリリース 8.1 へキュー表を移行する場合は、既存のサブスライバは NULL ルールを持つサブスライバに自動的にアップグレードされます。

メッセージ・ストリーミング

メッセージ・ストリーミングは、ソース・データベースおよび接続先データベースがともにリリース 8.1 以降である場合にのみサポートされます。メッセージ・ストリーミングを使用するために、COMPATIBLE パラメータを 8.1.0 に設定する必要はありません。リリース 8.1 データベースで COMPATIBLE パラメータが 8.0.5 以下に設定されていても、メッセージ・ストリーミングはサポートされます。

プロシージャおよびパッケージ

この項では、プロシージャおよびパッケージに関する互換性問題および相互運用性問題について説明します。

DBMS_LOB パッケージおよび NOCOPY

COMPATIBLE パラメータが 8.1.0 以上に設定されている場合、DBMS_LOB パッケージは LOB パラメータ用の新しい NOCOPY 構文を使用します。また、DBMS_LOB パッケージに渡された LOB ロケータは、新しい NOCOPY の意味に従います。

COMPATIBLE パラメータが 8.1 以下に設定されている場合は、NOCOPY 構文はサポートされません。したがって、互換性レベルが 8.0 である場合は、次の操作は行わないでください。

- DBMS_LOB パッケージでの新しい NOCOPY 構文の使用
- 新しい dbmslob.sql スクリプトの実行

DBMS_REPAIR パッケージ

DBMS_REPAIR パッケージを使用するには、COMPATIBLE パラメータを 8.1.0 以上に設定する必要があります。互換性レベルが 8.1.0 未満の場合は、DBMS_REPAIR パッケージを実行できません。

SET_SESSION_LONGOPS プロシージャの構文変更

バージョン 8 では、DBMS_APPLICATION_INFO.SET_SESSION_LONGOPS プロシージャを変更しています。新しい構文の詳細は、dbmsapin.sql ファイルを参照してください。ご使用のアプリケーションがこのプロシージャを使用している場合は、それに応じてアプリケーションを変更します。

Oracle オプティマイザ

COMPATIBLE パラメータを 8.1.0 以上に設定すると、オプティマイザの実行計画の選択が向上することがあります。互換性レベルが 8.1.0 の場合、オプティマイザは HIST_HEAD\$ データ・ディクショナリ表の新しい列および AVGCLN を使用して実行計画を選択できます。

Oracle Parallel Server

1 つの Oracle Parallel Server 環境で複数の Oracle リリースを使用する場合、リリースのサポートはオペレーティング・システムによって異なります。ご使用のオペレーティング・システムが、単一の Oracle Parallel Server 環境で異なるリリースの共存をサポートするかどうかについては、ご使用のオペレーティング・システム固有の Oracle ドキュメントを参照してください。

8.0 以降のリリースでは、データベースに属し、ハードウェア・クラスタで実行される Parallel Server モードでリンクされた Oracle インスタンスは、すべて GMX 実行可能ファイルのワード・サイズに合致している必要があります。したがってインスタンスはすべて、32 ビットまたは 64 ビットの実行可能ファイルを実行できなければなりません。

またリリース 8.0 では、ワード・サイズの異なる Oracle Parallel Server 実行可能ファイルを複数のデータベースで併用することはできませんが、Oracle Parallel Server モードでリンクされていない Oracle 実行可能ファイルでは可能です。リリース 8.1 では、GMS プロセスが排除されており、したがってこの制限も緩和されています。複数のデータベースに属する Oracle インスタンスは、リリース 8.1 では異なるワード・サイズで実行可能な場合があります。

INSTANCES キーワードの PARALLEL 句からの削除

今後の Oracle リリースでは、PARALLEL 句の INSTANCES キーワードを廃止する予定です。リリース 8.1 で INSTANCES キーワードを使用することはできますが、旧リリースとは異なって解釈されます。Oracle7 およびリリース 8.0 では、次のようなコマンドの PARALLEL 句で INSTANCES キーワードを使用することができました。

- ALTER CLUSTER
- ALTER DATABASE ...RECOVER
- ALTER INDEX ... REBUILD
- ALTER TABLE
- CREATE CLUSTER
- CREATE INDEX
- CREATE TABLE

また、ヒントでも使用することができました。INSTANCES キーワードは、パラレル操作で使用される Oracle Parallel Server の数を指定するのに使用されました。

またリリース 8.1 以降では、PARALLEL 句に並列度を指定する構文が変更されています。次の例のように、PARALLEL の後ろに数値を置くことで簡単に並列度を指定できます。

```
ALTER TABLE emp PARALLEL 5;
```

ただし、使用できるように選択した場合、キーワードである DEGREE は有効なままです。前述の構文は次の文と等価です。

```
ALTER TABLE emp PARALLEL (DEGREE 5 INSTANCES 1);
```

構文にかかわらず、パラレル操作で使用される問合せスレッドの数と同じ数値を指定します。いずれの構文も、問合せを実行するために必要なインスタンスの数に影響することはありません。いくつのインスタンスを使用するかは、使用可能なインスタンスの数および各インスタンスの負荷に基づいてシステムが決定します。したがって、どの構文でも結果は同じとなります。

リリース 8.1 での INSTANCES キーワードの使用 リリース 8.1 では、旧構文を引き続き使用して INSTANCES および DEGREE を指定することができますが、並列度を指定する単一のキーワードとして解釈されます。したがって、リリース 8.1 では廃止されたコマンド構文は引き続き受け入れられますが、これまでのリリースとは解釈が異なる場合があります。表 8-14 に、廃止された構文を引き続き使用した場合に、可能な INSTANCES および DEGREE の設定を Oracle が解釈する方法を示します。表 8-14 の各列は、次を示します。

- **並列度**は、PARALLEL 句の DEGREE キーワードの設定を示します。
- **インスタンス**は、PARALLEL 句の INSTANCES キーワードの設定を示します。
- **8.1 並列度**は、DEGREE および INSTANCES に基づいてリリース 8.1 での Oracle の並列度の解釈を示します。

表 8-14 リリース 8.1 での INSTANCES キーワードの変換

並列度	インスタンス	8.1 並列度
解除または 1	解除または 1	1
x	DEFAULT	x
x	解除または 1	x
解除または 1	DEFAULT	DEFAULT
DEFAULT	y	y
解除または 1	y	y
DEFAULT	解除または 1	DEFAULT
x	y	x × y

表中の x および y は整数値を表す変数です。

解除されたキーワードには、1 をその値として使用します。

これらの解釈のために、リリース 8.1 における Oracle の異なった動作を次に示します。

- DEGREE に x、および INSTANCES に 1 を設定しても、パラレル操作で使用するインスタンスが 1 つのみになるとは保証されません。
- DEGREE に 1、および INSTANCES に y を設定しても、並列操作で 1 つのインスタンスについて使用する問合せスレッドが 1 つのみになるとは保証されません。
- DEGREE に x、および INSTANCES に y を設定しても、いずれの設定も保証されません。かわりに、Oracle は x に y を掛けた値を並列度に設定することを試みます。

オラクル社は、予期しない動作を避けるため、INSTANCES キーワードを使用しないようにすることをお勧めします。PARALLEL_INSTANCE_GROUP 初期化パラメータを使用することをお勧めします。

参照： PARALLEL 句および PARALLEL_INSTANCE_GROUP 初期化パラメータの詳細は、『Oracle8i SQL リファレンス』を参照してください。

データベース・セキュリティ

この項では、データベース・セキュリティに関する互換性問題および相互運用性問題について説明します。

パスワード管理

バージョン 7 以前のアプリケーションをバージョン 8 のパスワード管理に対応させるには、アプリケーションで次の変更を行う必要があります。

- バージョン 8 の OCI コールである *OCISessionBegin()* を使用してサーバーに接続します。サーバーが SUCCESS_WITH_INFO を戻す場合、パスワードが有効でまだ猶予期間にあるかどうかを確認します。パスワードの期限が切れていても、まだ猶予期間内であれば、ユーザーに警告を出し、*OCIChangePassword()* コールを使用してユーザーのパスワードを変更します（パスワード変更コールはオプションですが使用することをお勧めします）。
- パスワードの期限が切れていてエラーが戻された場合は、バージョン 8 の OCI コールである *OCIChangePassword()* を使用してユーザーのパスワードを変更する必要があります。*OCIChangePassword()* を使用してパスワードを変更しない場合は、パスワード検証ルーチンは、新しいパスワードが古いパスワードと等しいかどうかを確認しません。同じ場合は変更を拒否しません。

Oracle7 のアプリケーションで前述の変更を行わなかった場合、ユーザー・アカウントの期限が切れた後にパスワードを変更するには、SQL*Plus などの Oracle Tools が必要になります。

このバージョン 8 のパスワード管理機能は、デフォルトでオフになっています。バージョン 8 のサーバー・システムでパスワード期限切れ機能をインプリメントしていない場合は、Oracle7 のクライアント側でパスワード管理について変更を行う必要はありません。DEFAULT プロファイルでは、すべてのパラメータが UNLIMITED に設定されており、パスワード複雑度チェック・ルーチンは NULL に設定されています。

パスワード検証ルーチンは、そのプロファイル定義とともにエクスポートまたはインポートされます。ユーザーの履歴表もバージョン 8 でインポートおよびエクスポートできます。

リリース 8.0 以降のサーバーのある Oracle7 以前のクライアント Oracle7 クライアントは、Oracle7 OCI コールを使用してサーバーに接続します。このため、リリース 8.0 以降のパスワードの有効期限は検出できません。ただし、リリース 8.0 以降のその他のパスワード管理機能は、Oracle7 のクライアントでも動作します。パスワード満了の処理も含めたリリース 8.0 以降のパスワード管理の全機能は、Oracle7 のログイン・コールをリリース 8.0 のログイン・コールに置き換えるだけで、Oracle7 のクライアントでも動作します。

Oracle7 以前のサーバーのあるリリース 8.0 以降のクライアント リリース 8.0 以降のクライアントをコード化して、Oracle7 以前のサーバーで機能させることができます。クライアント用のコーディング例を、以下に示します。

```
OCISessionBegin(...) /* call release 8.0 and higher logon OCI call */
if (SUCCESS_WITH_INFO) then
{ /* Check for password expiration and take appropriate action*/
...
OCIChangePassword(...);
...
}
```

エンタープライズ・ユーザー管理

この項では、エンタープライズ・ユーザー管理に関する互換性問題および相互運用性問題について説明します。

注意： リリース 8.0 で使用可能なグローバル・ユーザー機能が変更され、現在はベータ版のカスタムのみに使用可能です。この機能は、今後のリリースでは Oracle8i の一部になります。

リリース 8.0 との相互運用性 リリース 8.0 サーバーは、リリース 8.1 サーバーでグローバル・ユーザーおよびロールを共有できません。また、リリース 8.0 および 8.1 の現行ユーザー・データベース・リンクはサポートされません。

Oracle7 およびバージョン 6 リリースとの相互運用性 バージョン 7 またはバージョン 6 のサーバーではグローバル・ユーザーを作成または認可できないため、これらのサーバーはグローバル・ユーザーまたはロールをバージョン 8 と共有できません。また、バージョン 8 からバージョン 6 またはバージョン 7 への現行のユーザー・データベース・リンクは、サポートされません。

データベース・バックアップおよび回復

この項では、データベースのバックアップおよび回復に関する互換性および相互運用性の問題について説明します。

Recovery Manager

Recovery Manager の実行可能ファイルは、Recovery Manager が動作するデータベースと同じリリースである必要があります。したがって、リリース 8.0 の Recovery Manager の実行可能ファイルはリリース 8.1 データベース上では使用することができません。また、リリース 8.1 の Recovery Manager の実行可能ファイルはリリース 8.0 のデータベース上では使用できません。

リカバリ・カタログ 一般に、Recovery Manager はリカバリ・カタログおよびターゲット・データベースのリリース以上のリリースと互換性があります。ただし、その一般規則にはいくつかの例外があります（特にベータ・リリースに関する場合）。次の表では、リカバリ・カタログおよびターゲット・データベースと Recovery Manager の互換性を示します。

表 8-15 Recovery Manager の互換性

Recovery Manager のリリース	互換性のあるリカバリ・カタログ・リリース	互換性のあるターゲット・データベース・リリース
8.1.5	8.1.5 以上	8.1.5 以上
8.1.4 ベータ	8.1.4 ベータのみ	8.1.4 ベータのみ
8.1.3 ベータ	ベータ・リリース 8.1.3 または 8.1.4	ベータ・リリース 8.1.3 または 8.1.4
8.0.5	8.0.5 以上 ベータ・リリース 8.1.3 および 8.1.4 は除く	8.0.5 以上 ベータ・リリース 8.1.3 および 8.1.4 は除く
8.0.4	8.0.4 以上 ベータ・リリース 8.1.3 および 8.1.4 は除く	8.0.4 以上 ベータ・リリース 8.1.3 および 8.1.4 は除く
8.0.3	8.0.3 のみ	8.0.3 のみ

参照： リカバリ・カタログのアップグレードについては 7-30 ページの「[リカバリ・カタログのアップグレード](#)」を参照してください。

Recovery Manager コマンド Recovery Manager のリリース 8.1 では Recovery Manager コマンドがいくつか変更されています。ただし、以前のリリースで使用されていたコマンドはすべて Recovery Manager のリリース 8.1 で同様に動作します。

たとえば、clone コマンドは duplicate コマンドに変更されていますが、clone コマンドは適切に実行できます。また、allocate および connect コマンドの clone オプションは、現在 auxiliary オプションとなっていますが、clone オプションは同様に実行できます。同様に、copy コマンドおよび set コマンドの clonename キーワードは、現在 auxname となっていますが、clonename キーワードも実行できます。

バックアップ管理 : EBU および Recovery Manager

EBU および Recovery Manager は、Oracle データベースのバックアップを管理するためのクライアント側のユーティリティです。ただし、バージョン 8 のデータベースのバックアップを管理するには、Recovery Manager を使用する必要があります。EBU は、バージョン 8 では使用できません。

EBU および Recovery Manager は両方とも、Legato または EMC などのサード・パーティの記憶サブシステムと通信するためにメディア管理言語 (MML) を使用します。EBU と Oracle7 用のテープ・サブシステム管理モジュールに対して行った投資は、Recovery Manager とバージョン 8 でも再利用できます。ただし、バックアップ・ボリューム形式は再利用できません。Recovery Manager は異なる形式を作成し、Oracle7 のバックアップはバージョン 8 での復元には一般的に役に立たないので、バージョン 8 の下で記憶サブシステムに新しいバックアップを書き込む必要があります。

注意： Recovery Manager のスクリプト記述言語は、EBU のスクリプト記述言語とはまったく異なります。

データ・ファイル・バックアップ

Oracle7 で使用されたデータ・ファイル・バックアップはバージョン 8 のリリースにおいては、次の例外を除いて、復元することができません。移行ユーティリティの実行後に取られた Oracle7 データベースのバックアップは、バージョン 8 のどのリリースでも、復元または回復できます。EBU を使用して Oracle7 データベースをバックアップし、そのデータベースをバージョン 8 で回復させるために、後で復元しなければならない場合、バージョン 8 で回復する前に EBU を使用して、データ・ファイルを復元する必要があります。Oracle7 データベースが、オペレーティング・システム・コマンドでディスク・ファイルにバックアップされる場合、そのディスク・ファイルは、**catalog datafilecopy** コマンドを使用して、Recovery Manager に登録できます。

ファイルをバックアップするリリースとファイルを回復するリリース間のダイレクト・アップグレード・パスが、7-2 ページの表 7-1 の「アップグレード・パス」でサポートされている場合、バージョン 8 で取られたデータ・ファイルのバックアップは、バージョン 8 以降のどのリリースでも復元および回復できます。データ・ファイルの内容が以前のリリースと互換性がある場合、バージョン 8 のバックアップをバージョン 8 以前のリリースに復元および回復することもできます。

スタンバイ・データベース

スタンバイ・データベースは Oracle のリリース 7.3 以上のみで動作します。次の互換性に関する制限はスタンバイ・データベースに適用されます。

- プライマリ・データベースおよびスタンバイ・データベースは同じオペレーティング・システムで実行します。また、プライマリ・データベースとスタンバイ・データベースはオペレーティング・システムの同じリリースで実行することをお勧めします。
- プライマリ・データベースおよびスタンバイ・データベースは Oracle の同じメンテナンス・リリースを実行する必要があります。たとえば、プライマリ・データベースがリリース 8.0.5 を実行する場合、スタンバイ・データベースは、8.0.3、8.0.4 または 8.0.5 のような本番リリース 8.0 を実行できます。ただし、この場合、スタンバイ・データベースは Oracle7 またはリリース 8.1 を実行することはできません。

参照： スタンバイ・データベースに関するオペレーティング・システムの要件の詳細は、オペレーティング・システム固有の Oracle ドキュメントを参照してください。

スタンバイ・データベースの Oracle8i への移行 スタンバイ・データベースを Oracle7 から Oracle8i へ移行するには、次のステップを実行します。

1. Oracle7 で作成した REDO ログをすべて適用します。
2. プライマリ・データベースが Oracle8i の下で正常にオープンすることを確認します。
3. スタンバイ・データベースを Oracle8i へ移行します。
4. プライマリ・データベースの制御ファイルの変更ファイルを作成する ALTER DATABASE CREATE STANDBY CONTROLFILE AS *file_name* コマンドを発行して、スタンバイ・データベースの制御ファイルを作成します。
5. ステップ 4 で作成した制御ファイルをスタンバイ・データベース・サイトにコピーします。

6. 初期データ・ファイルをプライマリ・データベースからスタンバイ・データベース・サイトにコピーします。初期データ・ファイルが分からない場合、次の SQL 文を使用して参照します。

```
SELECT file_name, file_id FROM dba_data_files WHERE file_id = 1;
```

参照： スタンバイ・データベースの詳細は『Oracle8i バックアップおよびリカバリ・ガイド』を参照してください。

Fast Start On Demand ロールバックおよび Fast Start パラレル・ロールバック

セッションまたはインスタンスが正常に終了しない場合、回復プロセスとして、Oracle はコミットされていないトランザクションをロールバックします。Oracle8i には、ロールバック・パフォーマンスを改善する、Fast Start On Demand ロールバックおよび Fast Start パラレル・ロールバックの 2 つの新機能が備わっています。

使用不能のトランザクションが、他のトランザクションが必要とする行に行ロックをかけている場合、Fast Start On Demand ロールバックは、自動的にその新規トランザクションが必要とするデータ・ブロックを回復します。どの新規トランザクションの進行もブロックしない他のデータ・ブロックおよびトランザクションは、バックグラウンドでロールバックされます。Fast Start On Demand ロールバックは、COMPATIBLE 初期化パラメータを 8.1.0 以上に設定した場合にのみ使用可能です。

Fast Start パラレル・ロールバックは、複数のサーバー・プロセスを使用して使用不能の各トランザクションを回復し、バックグラウンド・ロールバック・パフォーマンスを改善します。COMPATIBLE 初期化パラメータが 8.0 または 8.1 リリースに設定されている場合、Fast Start パラレル・ロールバックを使用することができます。Fast Start パラレル・ロールバックは、次の条件が満たされた場合にのみ、複数のサーバー・プロセスを使用して使用不能のトランザクションを回復します。

- 使用不能の各トランザクションに 1 つ以上のプロセスを割り当てるサーバー・プロセスが十分ある。
- COMPATIBLE が 8.0 リリースに設定された場合、トランザクションは複数のサーバー・プロセスで作成される。トランザクションが 1 つのサーバー・プロセスのみで作成される場合、1 つのサーバー・プロセスのみがロールバック操作で使用されます。この制限は COMPATIBLE が 8.1.0 以上に設定された場合には適用されません。

参照： Fast Start On Demand ロールバックの詳細は、『Oracle8i 概要』を参照してください。

REDO ログのアーカイブ

リリース 8.1 では、オンライン REDO ログ・ファイルを、ローカル・ディスクベースのファイルまたは特定のスタンバイ・データベースなど、複数のアーカイブ先にアーカイブすることができます。この新機能によって、この項で説明する互換性および相互運用性の問題が起こる可能性があります。

以前アーカイブされたオンライン REDO ログの再アーカイブ リリース 8.1 以前は、正常かつ完全にアーカイブされたオンライン REDO ログ・ファイルを再アーカイブすることが可能でした。さらに、REDO ログ・ファイルを、以前正常にアーカイブしたアーカイブ先に再アーカイブすることも可能でした。

リリース 8.1 から、次の制限が適用されます。

- 正常にアーカイブされたオンライン・REDO ログは再アーカイブされない。
- 正常にアーカイブしたアーカイブ先には再アーカイブされない。

アーカイブ操作エラー検出動作 リリース 8.1 以前は、エラーが検出された場合、アーカイブ操作がただちに停止し、そのエラーをアラート・ログに報告して、ユーザーにエラーの警告をしていました。

リリース 8.1 以降、アーカイブ操作は、アーカイブ先のすべてが処理されない場合にのみ、プロセスを停止します。1 つ以上のアーカイブ先でのエラーのみではアーカイブ操作は停止しません。アーカイブ操作は、すべてのアーカイブ先が処理できない場合に停止します。

LogMiner

LogMiner はリリース 8.1 以上のインスタンスで実行し、次の基準を満たすすべてのデータベースの REDO ログ・ファイルを分析します。

- 分析 Oracle インスタンスと同様の DBCS (データベース・キャラクタ・セット) を持つ。
- 分析 Oracle インスタンスと同様のハードウェア・プラットフォームで実行する。
- リリース 8.0 以上のデータベース。

LogMiner は、REDO ログ・ファイルを分析するために、マウントされたデータベースを必要としません。ただし、REDO ログ・ファイルの内容を完全に翻訳するには、LogMiner ディクショナリ (カタログ) へアクセスする必要があります。LogMiner はそのディクショナリを使用して、内部オブジェクト識別子およびデータ型をオブジェクト名および外部データ形式に翻訳します。PL/SQL パッケージ DBMS_LOGMNR_D を使用して、データベース・ディクショナリを外部ファイルに抽出します。そのファイルは、後で REDO ログ・ファイルの分析に使用できます。ディクショナリがなければ、LogMiner は内部オブジェクト識別子を戻し、データを 16 進で表示します。

他のデータベースのアーカイブされた REDO ログ・ファイルの分析 異なるデータベースの REDO ログ・ファイルを分析すると同時に、データベースのインスタンスに LogMiner を実行できます。他のデータベースのアーカイブされた REDO ログ・ファイルを分析する場合、LogMiner は次の条件を備えている必要があります。

- REDO ログ・ファイルと同じデータベースおよび同じデータベースのキャラクタ・セットで作成されるディクショナリ・ファイルへアクセスする。
- 同じシステム上である必要はないが、ログ・ファイルを生成する同様のハードウェア・プラットフォームで実行する。
- Oracle リリース 8.0 以上から回復に適用される REDO ログ・ファイルを使用する。

Oracle メディア管理 API およびプロキシ・コピー

Oracle メディア管理 API バージョン 2 以降、プロキシ・コピーの機能がサポートされます。Recovery Manager プロキシ・バックアップを試みる場合、Oracle はプロキシ・コピー機能をサポートしない Oracle メディア管理 API リリース 1.1 またはバージョン 2 とリンクされます。その後 Recovery Manager はエラーを戻し、バックアップは失敗します。

分散データベース

この項では、分散データベースに関連する互換性および相互運用性問題について説明します。

基本レプリケーション

リリース 8.1 以前は、Oracle スナップショットは、常にスナップショット実表および実表のビューから構成されていました。たとえば、スナップショット SNAP_EMP を作成することによって、ビュー SNAP_EMP および実表（通常 SNAPS_SNAP_EMP と呼ばれる）が作成されました。リリース 8.1 では、ほとんどのスナップショットは、そのスナップショットと同じ名前の実表のみを持ちます。ビューは作成されません。

ビューは次の場合にスナップショットに追加されます。

- リリース 8.0 などのリリース 8.1 以前のデータベースからスナップショットがインポートされた。
- COMPATIBLE パラメーターが 8.1.0 以下に設定されている。
- スナップショットが非表示列を必要とする（つまり、副問合せを含む ROWID スナップショットおよび高速リフレッシュ可能なスナップショット）。

アドバンスト・レプリケーション

レプリケート環境には、次の互換性に関する制限が適用されます。

- Oracle の異なるリリースがあるレプリケート環境の場合、古いリリースと互換性がないデータはレプリケートできません。たとえば、リリース 8.1.0 の互換性レベルのデータベースとリリース 8.0.0 の互換性レベルのデータベースがあるレプリケート環境では、リリース 8.0 と互換性のないデータは、これらのデータベース間でレプリケートできません。
- リリース 8.1 以前のリリースの環境では外部に存在したアドバンスト・レプリケーション・パッケージの多くはリリース 8.1 では内部に存在します。これは、パフォーマンスを向上し、データの整合性を保つためです。このような内部パッケージは『Oracle8i レプリケーション・ガイド』に一覧で示されています。

N-way マスター・サイトのうち 1 つ以上がリリース 8.1 以前のものであれば、次のプロシージャで GENERATE_80_COMPATIBLE フラグを解除または TRUE に設定します。

- GENERATE_REPLICATION_SUPPORT
- CREATE_SNAPSHOT_REOBJECT
- GENERATE_SNAPSHOT_SUPPORT

ただし、アドバンスト・レプリケーション環境にリリース 8.1 以上のリリースで動作するマスターしかない場合は、GENERATE_80_COMPATIBLE を FALSE に設定します。これによって、レプリケート生成 RPC 用のリリース 8.1 プロトコルを使用して、top flavor でインスタンス化されたサイトからデータをレプリケートできます。

異機種間サービス・エージェント

この項では、異機種間サービス・エージェントに関する互換性問題および相互運用性問題について説明します。

異なるリリースのサーバー間の相互運用性 リリース 8.0.3 以降のリリースのサーバーは、リリース 8.0.3 以降のリリースの異機種間サービス・エージェントに接続および使用できます。異なるリリースのサーバー間で接続する場合、リリースの低いほうの機能に制限されます。

マルチスレッド・サービス・エージェント リリース 8.1 から、マルチスレッド異機種間サービス・エージェントがサポートされています。既存のエージェントが存在し、マルチスレッド機能を利用する場合は、エージェント初期化ファイルを作成し、エージェント制御ユーティリティを使用してそのエージェントを起動します。

参照： 異機種間サービスの概要、エージェント初期化ファイルの作成およびエージェント制御ユーティリティを使用したエージェントの起動については『Oracle8i 分散システム』を参照してください。

SQL*Net または Net8

バージョン 7 およびバージョン 8 のリリースでは SQL*Net V2 または Net8 を使用できます。SQL*Net バージョン 1 は、さまざまなネットワーク・アドレス指定スキームを使用していたが、バージョン 8 では使用できません。したがって、アップグレードしたアプリケーションには、次の要件が適用されます。

- クライアントおよびサーバーの両方で、SQL*Net V2 または Net8 を使用していなければならない。
- マルチスレッド・サーバーには、SQL*Net V2 または Net8 がサーバー上に必要。したがって、マルチスレッド・サーバーを使用してクライアントと接続するには、クライアント上の SQL*Net V2 または Net8 を使用する必要があります。

SQL*Net V1 の SQL*Net V2 または Net8 へのアップグレード

SQL*Net V1 を SQL*Net V2 または Net8 へアップグレードするには、次の変更を行います。

- SQL*Net V2 または Net8 をインストールします。
- 各接続文字列を次のバージョンの接続記述子として再作成します。SQL*Net V2 は『SQL*Net 管理者ガイド』で概説している構文を使用し、Net8 は『Oracle8i Net8 管理者ガイド』で概説している構文を使用します。
- SQL*Plus および SQL*Forms などを含む、SQL*Net V2 または Net8 で使用するプリコンパイラ・プログラムおよび Oracle 実行可能ファイルを再リンクします。

参照： V1 から V2 への SQL*Net のアップグレードの詳細は、『SQL*Net Version 2.0 Administrator's Guide』および『Oracle8i SQL*Net V2 Migration Guide』を参照してください。SQL*Net V1 の Net8 へのアップグレードの詳細は、『Oracle8i Net8 管理者ガイド』を参照してください。

サービス命名および接続ロード・バランシング

リリース 8.1 およびそれ以降のリリースは、1 つ以上のデータベース・インスタンスを含むサービスのサービス命名および接続ロード・バランシングをサポートします。各サービスは複数インスタンスを含むことができ、かつ各インスタンスは複数ハンドルを含むことができます。このサポートによって、ある特定のデータベース・インスタンスよりむしろサービスにアクセスできるようになります。また、ある特定のインスタンス名からサービス名を論理的に分けることができます。

複数インスタンスを含むサービスをサポートするために、次のような接続記述子の新規パラメータを使用します。

- SERVICE_NAME
- INSTANCE_NAME

新規パラメータは、次のようなプロセスで発生する要求を処理することで接続ロード・バランシングを使用可能にします。

1. クライアント・プログラムが接続先のサービスの名前を指定します。
2. TNS リスナーがサービス内で最も負荷の少ないインスタンスを検索します。
3. TNS リスナーがサービス内で最も負荷の少ないハンドラを検索します。
4. 必要であれば、TNS リスナーがクライアントを最適のハンドラにリダイレクトするか、クライアント接続をハンドラに渡します。

接続ロード・バランシングを使用するために、次の処理を実行してください。

- 接続拡張子の SID パラメータの使用を中止します。
- `initSID.ora` ファイルの SERVICE_NAMES および INSTANCE_NAME パラメータを使用します。
- SERVICE_NAME パラメータを `tnsnames.ora` ファイルで使用します。

注意： TNS リスナーが同じインスタンス名を持つ 2 つ以上のインスタンスを処理できるように設定する前に、どのクライアント・プログラムも SID パラメータに基づく接続を使用していないことを確認してください。

参照： 接続ロード・バランシングおよび SERVICE_NAME パラメータの使用の詳細は、『Oracle8i Net8 管理者ガイド』を参照してください。

エクスポート/インポート

バージョン 8.0 のエクスポート・ユーティリティでは、バージョン 8.0 以前のインポート・ユーティリティと下位互換性のないダンプ・ファイルが作成されます。エクスポートしたデータは、リリース 8.0 以前のインポート・ユーティリティではインポートできません。

バージョン 5 から、エクスポート・ダンプ・ファイルは今後の Oracle のすべての主要リリース、パッチ・リリースおよびメンテナンス・リリースにインポート可能になります。[表 8-16](#) に、このサポートの詳細を示します。

表 8-16 エクスポート・ダンプ・ファイルの上位互換性

ダンプ・ファイル	今後のリリースにインポート可能
バージョン 5 およびバージョン 6 注意: バージョン 5 では、リリース 5.1.22 以降のリリースのエクスポート・ダンプ・ファイルがサポートされています。	Oracle7、リリース 8.0、リリース 8.1 および今後のリリース
Oracle7、リリース 8.0 およびリリース 8.1	リリース 8.0、リリース 8.1 および今後のリリース

エクスポート・ダンプ・ファイルは、前リリースのエクスポート/インポート・ユーティリティを使用して、現在のリリースにインポートできます。[表 8-17](#) に、このサポートの詳細を示します。

表 8-17 エクスポート・ダンプ・ファイルの下位互換性

ダンプ・ファイルのリリース	以前のリリースにインポート可能	使用するエクスポート/インポート・ユーティリティ
リリース 8.0	リリース 7.3	リリース 7.3
リリース 8.1	リリース 8.0	リリース 8.0

リリース 8.0 以降のリリースのデータを Oracle7 (以前) データベースにエクスポートするためには、catexp7.sql スクリプトをリリース 8.0 以降のリリースのデータベースで実行した後、Oracle7 エクスポート・ユーティリティを使用する必要があります。このスクリプトは \$ORACLE_HOME/rdbms/admin ディレクトリにあります。

ダイレクト・パス・エクスポートは、従来型パス・エクスポートとは異なるコード化形式を使用します。したがって、生成されるエクスポート・ダンプ・ファイルも、ダイレクト・パス・エクスポートと従来型パス・エクスポートでは異なります。

ダイレクト・パス・エクスポートまたは従来型パス・エクスポートで生成されたダンプ・ファイルは、両方ともリリース 8.0 またはそれ以降のリリースのインポート・ユーティリティで使用できます。リリース 8.0 以前のダンプ・ファイルは、リリース 8.0 以降のインポート・ユーティリティと上位互換性を持ちます。ダイレクト・パス・エクスポートの現行または以前のバージョンおよびリリース、または従来型パス・エクスポートによって生成されるダンプ・ファイル・データは、Oracle Server の将来のバージョンおよびリリースとの上位互換性を持ちます。

Oracle7 と互換性がないデータへのエクスポート / インポートの使用方法

Oracle7 エクスポート・ユーティリティを使用して、リリース 8.0 以降のリリースのデータベースからデータをエクスポートする場合、Oracle7 と互換性のないデータはエクスポートされません。たとえば、パーティション表は Oracle7 エクスポート・ユーティリティではエクスポートされません。リリース 8.0 以降のリリースのパーティション表を Oracle7 データベースへ移動する必要がある場合、まずその表を非パーティション表に再編成します。同様に、リリース 8.0 以降のリリースから Oracle7 に互換性のない別の型のデータを移動する必要がある場合、まずそのデータが Oracle7 で互換性を持つように再編成します。

その他の互換性および相互運用性の問題

この項では、Oracle インストレーションのその他の互換性問題および相互運用性問題について説明します。

複数バージョン管理

同一のコンピュータ・システム上で Oracle の異なるバージョンを同時に実行できます。ただし、それぞれのバージョンはそのバージョンに対応するデータベースにのみアクセスできます。たとえば、バージョン 7 およびバージョン 8 を同一のコンピュータ・システムにインストールしている場合は、バージョン 7 のサーバーはバージョン 7 のデータベースにはアクセスできますが、バージョン 8 のデータベースにはアクセスできません。また、バージョン 8 のサーバーはバージョン 8 のデータベースにはアクセスできますが、バージョン 7 のデータベースにはアクセスできません。

参照： オペレーティング・システムでの複数バージョン管理の詳細は、オペレーティング・システム固有の Oracle ドキュメントを参照してください。オペレーティング・システムによっては制限がある場合があります。

2GB ファイル・サイズ依存関係

Oracle リリース 8.0.4 以上のリリースは、2GB 以上のファイルにアクセスできます。ただし、このアクセスは次のようなオペレーティング・システム依存関係に影響されます。

- **ファイル・モード：** そのファイルは、ファイル・システム・ファイルまたは RAW デバイス・ファイルですか？多くの UNIX プラットフォームは、2GB を超えるファイルサイズを RAW デバイスでのみサポートします。
- **非同期 IO：** オペレーティング・システムはファイルの非同期 IO を RAW およびファイル・システム・ファイルの両方についてサポートしていますか？非同期 IO は 2GB を超えるファイルをサポートしていますか？
- **オペレーティング・システムのリビジョン：** オペレーティング・システムのリリース番号は 2GB を超えるファイルサイズをサポートしていますか？たとえば、Solaris2.5.1 では、RAW デバイスについてのみ 2GB を超えるファイル・サイズをサポートしています。ただし、Solaris2.6 では、2GB を超える RAW およびファイル・システム・ファイルをサポートします。
- **オペレーティング・システムの IO サブシステムの問題：** 2GB を超えるファイル・サイズをサポートするために、オペレーティング・システムがファームウェアのアップグレードを必要としていますか？2GB を超えるファイル・サイズをサポートするようになったのは最近のことで、多くのディスク・アレイまたは IO サブシステムは、2GB を超えるファイル・サイズをサポートするために、ファームウェアのアップグレードを必要としています。オペレーティング・システムの発売元に、大きいサイズのファイルをサポートするためにどのファームウェア・パッチが必要かを確認してください。

2GB を超えるサイズのファイルを使用する前に、これらのオペレーティング・システム依存関係を確認してください。

9

アプリケーションのアップグレード

この章では、リリース 8.1 の新機能を利用するためにアプリケーションをアップグレードする方法について説明します。リリース 8.1 の新機能を使用しない既存の Oracle7 およびリリース 8.0 のアプリケーションを変更する必要はありません。既存のアプリケーションは、リリース 8.1 のデータベース上でも同様の機能、または拡張された機能を実行できます。

この章で説明するトピックは、次のとおりです。

- [Oracle8i へのアプリケーションのアップグレードの概要](#)
- [OCI アプリケーションおよびプリコンパイラ・アプリケーションのアップグレード](#)
- [SQL*Plus スクリプトのアップグレード](#)
- [Oracle7 の Forms または Developer/2000 アプリケーションのアップグレード](#)
- [LONG から LOB へのコピー](#)

Oracle8i へのアプリケーションのアップグレードの概要

次に示すバージョン 8 の機能は、アプリケーションのリリース 8.1 へのアップグレード・プロセスで役に立ちます。

- SQL*Net バージョン 2 および Net8 は、両方とも Oracle のさまざまなバージョンおよびリリースで動作します。その結果、バージョン 7 のデータベースとバージョン 8 のデータベースは、SQL*Net バージョン 2 と Net8 を使用して通信できます。

注意： SQL*Net バージョン 1 は、異なるネットワーク・アドレス指定スキームを使用していたため、バージョン 8 では使用できません。

- アプリケーションの作成と管理は、プログラミング・インタフェース（たとえば、Oracle プリコンパイラおよび Oracle コール・インタフェース）が異なるバージョン間で変更されていません。たとえば、SQL*Net または Net8 を使用できます。または、以前のリリースのデータベース用に設計されたアプリケーションを再リンクして、バージョン 8 のリリースまたは Oracle7 のリリースでそのアプリケーションを実行できます。
- Oracle の異なるリリース間での多少の非互換性に対応するため、下位互換性が組み込まれています。
- 移行後、多くの新機能および拡張機能を自動的に利用できます。こういった新機能の中には、パフォーマンスを向上させるものもあります。

OCI アプリケーションおよびプリコンパイラ・アプリケーションのアップグレード

Oracle データベースを Oracle7 から Oracle8i へ移行する前に、Oracle8i のデータベースで使用する予定の OCI とプリコンパイラ・アプリケーションをアップグレードしてください。そうすることによって、本番データベースを移行する前にそれらのアプリケーションを Oracle8i のサンプル・データベースでテストできます。

これらのアプリケーションをアップグレードするのに必要な作業量は、プログラム・インタフェースと Oracle8i を利用する程度によって異なります。簡単なものから困難なものの順に、次の選択が可能です。

- 既存のパフォーマンスと機能を維持する
- アプリケーションのパフォーマンスを向上させる
- Oracle8i のデータベースの新機能を利用する

次の項では、OCI アプリケーションおよびプリコンパイラ・アプリケーションについて説明します。

OCI アプリケーションのアップグレード

アプリケーション開発者が既存の OCI アプリケーションにリリース 8.1 の機能を取り込む場合、次のオプションを使用します。

- アプリケーションが Oracle7 OCI コールを使用する場合、新規のリリース 8.1 OCI コールのみを使用するためアプリケーションを完全に再書込みします。
- アプリケーションが Oracle7 OCI コールを使用する場合、Oracle7 コールをいくつかの操作に使用している間に、リリース 8.1 OCI を既存のアプリケーションに取り込みます。
- アプリケーションがバージョン 8 コールのみを使用する場合、新規のリリース 8.1 OCI コールを既存のアプリケーションに取り込みます。

参照： OCI アプリケーションの書込みおよびアップグレードの詳細は、『Oracle コール・インタフェース・プログラマーズ・ガイド』を参照してください。

文の実行に対するバッチ・エラー・モードの使用

リリース 8.1 から、OCI アプリケーションは、*OCIStmtExecute* を使用して DML 配列を実行するときにバッチ・エラー・モードを使用できます。バッチ・エラー・モードを使用するには、OCI ライブラリおよびサーバー・ライブラリがリリース 8.1 以上でなければなりません。

OCI_BATCH_ERRORS にモード・パラメータを設定し、この機能に必要な新規コードを追加することによって、既存のアプリケーションを変更してバッチ・エラー・モードを使用できます。その後、リリース 8.1 クライアント・ライブラリに、変更したアプリケーションを再コンパイルおよび再リンクします。

クライアントへの通知サポート

リリース 8.1 から、パブリッシュ / 加入インタフェースを使用する OCI アプリケーションでクライアント通知がサポートされます。クライアント通知によって、アプリケーションがデータベース・イベント・パブリッシュおよびアドバンスト・キューイング機能を利用できます。クライアント通知機能を使用するには、クライアント・アプリケーションがリリース 8.1 以上のクライアント・ライブラリとリンクしている必要があります。

アドバンスト・キューイング・オプションでの LISTEN コール・サポート

リリース 8.1 から、OCI アプリケーションで LISTEN コールがサポートされます。LISTEN コールはアドバンスト・キューイング・オプションで使用でき、一連のキューに対してメッセージがないか監視するために使用されます。LISTEN コールを使用するには、クライアント・アプリケーションがリリース 8.1 以上のクライアント・ライブラリとリンクしている必要があります。

プリコンパイラ・アプリケーションのアップグレード

既存のプリコンパイラ・アプリケーションをリリース 8 のデータベースで使用するには、次のステップを実行してください。

1. 既存アプリケーションのリリース 8.1 の新機能を利用する場合、既存アプリケーションが Oracle7 であるか Oracle8 であるかによって次の処理を実行します。
 - Oracle7 のアプリケーションの場合、Oracle7 と Oracle8i の違いを反映するように、Oracle7 アプリケーションの再コーディングや、新規アプリケーションの作成を行います。
 - Oracle8 のアプリケーションで、リリース 8.1 の新機能を使用するとき、リリース 8.1 の新機能性コードをバージョン 8 の既存アプリケーションに取り込みます。既存アプリケーションでリリース 8.1 の機能を利用しない場合は、この手順を省略してステップ 2 へ進んでください。
2. プリコンパイラに付属するリリース 8.1 のランタイム・ライブラリ (SQLLIB) に、アプリケーションを再リンクします。リリース 8.1 でアプリケーションを使用する場合は、少なくともこのステップを実行する必要があります。

注意： Oracle8i のデータベースに対して、既存の Oracle7 プリコンパイラ・アプリケーションを実行する場合、アプリケーションの再コンパイルや再コンパイルは必要ありません。

既存 Oracle7 プリコンパイラ・アプリケーションの簡易アップグレード

Pro*C 2.2 などの、Oracle7 のプリコンパイラで動作するように作成されたアプリケーションは、Pro*C 8.1 などの Oracle8i のプリコンパイラへ非常に簡単にアップグレードできます。Oracle7 のプリコンパイラ・クライアントは Oracle8i のサーバーとともに運用でき、Oracle8i のプリコンパイラ・クライアントは Oracle7 のサーバーとともに運用できます。

次に、プリコンパイラとサーバーの互換性について概要を示します。例として Pro*C を使用しますが、Pro*COBOL にも適用します。

- Oracle7 の Pro*C 2.2 を使用するアプリケーションは、Oracle8i のサーバーに対しても同様に動作します（SQLLIB 関数のコールはすべて Oracle8i のサーバーに対して動作する）。
- Pro*C 8.1 を使用するアプリケーションは、プリコンパイラまたはサーバーのオブジェクト機能などのバージョン 8 の新機能を使用しない場合に限り、Oracle7 のサーバーに対して動作します。
- Oracle7 および Oracle8i の SQLLIB 関数のコールは、同一アプリケーションおよび同一トランザクション内に混在できます。

既存の Oracle7 のアプリケーションで使用可能なアップグレード・パスには、次の 3 つの選択肢があります。再コンパイルや再プリコンパイルは必要ありません。例として Pro*C を使用しますが、Pro*COBOL にも適用します。

- Oracle7 のクライアント・アプリケーションに変更を加えずに保存します。アプリケーションは、新しい Oracle8i のサーバーに対して同様に動作します。
- Oracle8i のクライアントへアップグレードするが、アプリケーションは変更しません。Oracle7 のクライアントを Oracle8i のクライアントにアップグレードするには、新しいバージョンの SQLLIB を再リンクするのみです。アプリケーションを再コンパイルする必要はありません。再リンクされた Oracle7 のアプリケーションは、Oracle8i のサーバーに対して同様に動作します。
- Oracle8i へアップグレードしてアプリケーションに変更を加えることによって、Oracle8i の新しいプリコンパイラ機能を利用してパフォーマンスや拡張性を向上したり、Oracle8i のサーバーのオブジェクト機能を使用したりします。既存のアプリケーションを Oracle8i のサーバー上で実行するには、既存の Oracle7 アプリケーションを Oracle8i の新しい SQLLIB コールを使用して再コーディングし、新しい SQLLIB と再リンクする必要があります。Oracle8i のサーバーのオブジェクト機能を使用するには、どのアプリケーションであってもこのアップグレード・パスが必要となります。さらに、アプリケーション開発者が Oracle8i のサーバーのオブジェクト機能を使用する場合、Pro*C8.0 以上を使用するにはクライアントをアップグレードする必要があります。

参照： プリコンパイラ・アプリケーションの詳細は、『Pro*C/C++ プリコンパイラ・プログラマーズ・ガイド』および『Pro*COBOL プリコンパイラ・プログラマーズ・ガイド』を参照してください。

SQL*Plus スクリプトのアップグレード

SQL*Plus リリース 8.0 以上、リリース 8.0 以上のデータベースおよび PL/SQL リリース 8.0 以上の機能を使用するときは、次のステップを実行してください。

- SQL*Plus 3.x のスクリプトに次のような変更を加えて、SQL*Plus 8.0 以上のスクリプトに変換します。
 - スクリプトに SET COMPATIBILITY VERSION 7 という行がある場合は、それを SET COMPATIBILITY VERSION 8 に変更します。または、この行を削除してデフォルト設定 (VERSION 8) を使用します。
 - login.sql ファイルをチェックし、SET COMPATIBILITY VERSION 7 という行があれば SET COMPATIBILITY VERSION 8 に変更します。
 - SQL*Plus の新機能の詳細は、『SQL*Plus ユーザーズ・ガイドおよびリファレンス』を参照する。
- Oracle8i の新機能を使用するには、既存の SQL スクリプトを変更して Oracle8i の構文に適合させます。Oracle8i の新機能を使用しない場合は、既存の SQL スクリプトに変更を加えなくても Oracle8i で実行できます。

参照： SQL スクリプトのアップグレードの詳細は、『Oracle8i SQL リファレンス』を参照してください。

注意： PL/SQL プロシージャについては、変更する必要はありません。

Oracle7 の Forms または Developer/2000 アプリケーションのアップグレード

Forms アプリケーションは、Oracle7 でも Oracle8i でも同じように動作します。Oracle8i の新機能が Forms アプリケーションとどのように相互作用するかについては、『Oracle Forms リファレンス・マニュアル Vol 1』および『Oracle Forms リファレンス・マニュアル Vol 2』、『Oracle Forms 開発者ガイド』および『Oracle Forms 上級テクニック』を参照してください。

LONG から LOB へのコピー

LOB データ型 (BFILE、BLOB、CLOB および NCLOB) は LONG データ型よりも多くの長所があります。LOB データ型と LONG データ型の違いの詳細は、『Oracle8i 概要』を参照してください。

リリース 8.1 では、SQL の新演算子、TO_LOB は表の LONG 列から LOB 列へデータをコピーします。コピーを成功させるために LONG データ型および LOB データ型は同じでなければなりません。たとえば、LONG RAW データは BLOB データに、LONG データは CLOB データにコピーされる必要があります。

次のプロシージャの例では、LONG_TAB 表の LONG_COL という名の LONG 列が、LOB_TAB 表の LOB_COL という名の LOB 列にコピーされます。この表は表の各行の識別番号を持つ ID 列を含みます。

LONG 列から LOB 列にデータをコピーするには次のステップを実行してください。

1. LONG 列を含む表と同じ定義を持つ新規の表を作成します。ただし、LONG データ型のかわりに LOB データ型を使用します。

たとえば、次の定義の表があるとします。

```
CREATE TABLE long_tab (  
    id NUMBER,  
    long_col LONG);
```

次の SQL 文を使用して新規の表を作成します。

```
CREATE TABLE lob_tab (  
    id NUMBER,  
    clob_col CLOB);
```

注意： 新規の表を作成するとき、整合性制約、トリガー、付与、索引など、表のスキーマを保存するようにしてください。TO_LOB 演算子はデータをコピーするだけで、表のスキーマは保存しません。

2. TO_LOB 演算子を使用して INSERT コマンドを発行し、LONG データ型のある表から LOB データ型のある表にデータを挿入します。

たとえば、次の SQL 文を発行します。

```
INSERT INTO lob_tab
  SELECT id,
  TO_LOB(long_col)
  FROM long_tab;
```

3. コピーが成功したことを確認し、LONG 列の表を削除します。

たとえば、次の SQL コマンドを発行して、LONG_TAB 表を削除します。

```
DROP TABLE long_tab;
```

4. LONG データがある表の名前を使用して、新規の表のシノニムを作成します。そのシノニムによって、使用しているデータベースおよびアプリケーションが適切に機能していることを確認できます。

たとえば、次の SQL 文を発行します。

```
CREATE SYNONYM long_tab FOR lob_tab;
```

コピーが完了したら、LOB データを使用するために、表を使用するすべてのアプリケーションを変更する必要があります。

参照： LOB データを使用するためのアプリケーション変更の詳細は、『Oracle8i アプリケーション開発者ガイド ラージ・オブジェクト』を参照してください。

Server Manager から SQL*Plus への移行

Server Manager 行モードは、今後のリリースでは廃止されます。Server Manager 行モードを使用して SQL スクリプトを実行する場合、SQL*Plus と互換性を持たせるためにこれらのスクリプトを変更し、SQL*Plus を使用して実行します。この章では、SQL*Plus で動作する Server Manager の行モード・スクリプトの変更について説明します。

この章は次のトピックについて説明します。

- [起動の違い](#)
- [コマンド](#)
- [構文の違い](#)

参照： SQL*Plus リリース 8.1 の使用についての詳細は、『SQL*Plus ユーザーズ・ガイドおよびリファレンス』を参照してください。

注意： 簡潔にするため、Server Manager 行モードはこの章の後半で Server Manager として表記されます。

起動の違い

Server Manager と SQL*Plus の起動メソッドは異なります。SQL スクリプトを変更し、SQL*Plus を正しく起動する必要があります。次の項は起動の違いを説明し、SQL*Plus 起動のオプションを提供します。

Server Manager の起動

Server Manager を起動するには、システム・プロンプトで Server Manager のプログラム名を入力します。このプログラム名はオペレーティング・システム固有です。Server Manager 起動後、次の例のように CONNECT コマンドを使用して接続します。

```
CONNECT scott/tiger
```

SQL*Plus の起動

次のセクションでは、SQL*Plus を起動するさまざまな方法について説明します。

NOLOG オプションを使用した SQL*Plus の起動

Server Manager と同じような処理を SQL*Plus で行う場合、SQL*Plus 起動時に次のように NOLOG オプションを使用します。

```
sqlplus /nolog
```

SQL*Plus が起動し、CONNECT コマンドを使用してユーザーとして接続できます。

接続情報を使用した SQL*Plus の起動

SQL*Plus を起動する他のオプションとして、プログラム開始時に接続情報を入力する方法があります。たとえば、SQL*Plus を起動し、SCOTT/TIGER として接続するには、次のように入力します。

```
sqlplus scott/tiger
```

SQL*Plus が起動し、SCOTT というユーザーとして接続します。

オプションや接続情報なしの SQL*Plus の起動

オプションや接続情報なしで SQL*Plus を起動するには、次のように入力します。

```
sqlplus
```

ユーザー名とパスワードを入力するように求められます。有効なユーザー名とパスワードを入力すると、SQL*Plus が起動し、プロンプトで指定したユーザーとして接続します。ただし、SQL スクリプトで、ユーザー名とパスワードの入力を省略することもできます。

コマンド

Server Manager と SQL*Plus は、両方のプログラムで同じ処理を行う特定のコマンドを共有しています。ただし、Server Manager での処理と SQL*Plus での処理が異なるコマンドもあります。Server Manager から SQL*Plus への移行を成功させるために、これらの違いと類似点を理解する必要があります。次の項では、SQL*Plus によって正しく解釈されるコマンドを使用するための SQL スクリプトの変更について説明します。

新 SQL*Plus リリース 8.1 のコマンド

[表 10-1](#) に SQL*Plus で現在使用可能である Server Manager コマンドを示します。SQL*Plus リリース 8.1 で実行している SQL スクリプトで、これらのコマンドを使用できます。

注意： リリース 8.1 以前のデータベース（Oracle7 およびリリース 8.0）でこれらのコマンドを含むスクリプトを実行するには、Server Manager を使用します。SQL*Plus リリース 8.1 以前の SQL*Plus のバージョンは、これらの新しいコマンドを含むスクリプトを実行しません。

コマンド

表 10-1 新 SQL*Plus リリース 8.1 のコマンド

コマンド	説明
ARCHIVE LOG	オンライン REDO ログ・ファイルの自動アーカイブを開始または停止する、特定の REDO ログ・ファイルを手動で（明示的に）アーカイブする、またはアーカイブについての情報を表示します。
RECOVER	1 つ以上の表領域、1 つ以上のデータ・ファイルまたはデータベース全体で、メディア回復を実行します。
SET AUTORECOVERY	ON にすると、回復中に必要なアーカイブ REDO ログ・ファイルのデフォルトのファイル名を、RECOVER コマンドに自動的に適用します。AUTORECOVERY が ON に設定されている場合、必要なファイルが所定の名前で所定の位置に配置されているので対話は不要です。
SET INSTANCE	セッションに対するデフォルト・インスタンスを指定のインスタンス・パスに変更します。データベースへは接続しません。デフォルトのインスタンスは、インスタンスが指定されていない場合にコマンドに対して使用されます。
SET LOGSOURCE	回復中に、アーカイブ・ログが取り出される場所を指定します。デフォルト値は、LOG_ARCHIVE_DEST 初期化パラメータによって設定されます。パス名なしで SET LOGSOURCE コマンドを発行すると、デフォルトの位置を復元します。
SHOW AUTORECOVERY	自動回復が可能かどうかを示します。
SHOW INSTANCE	デフォルト・インスタンスに対する接続文字列を示します。SHOW INSTANCE を使用していない場合または SHOW INSTANCE コマンドの LOCAL オプションを使用している場合、SHOW INSTANCE は LOCAL 値を戻します。
SHOW LOGSOURCE	アーカイブ・ログの位置に対する現行の設定を示します。LOG_ARCHIVE_DEST 初期化パラメータによって指定されたように、デフォルト値が有効の場合、DEFAULT を表示します。
SHOW PARAMETERS	1 つ以上の初期化パラメータに対する現行の値を表示します。コマンドの後ろに文字を指定せずに SHOW PARAMETERS コマンドは、初期化パラメータをすべて表示します。
SHOW SGA	現行インスタンスのシステム・グローバル領域についての情報を表示します。
SHUTDOWN	現在実行中の Oracle インスタンスを停止し、任意でデータベースをクローズしてアンマウントします。
STARTUP	データベースのマウントおよびオープンを含むいくつかのオプションで、Oracle インスタンスを開始します。

Server Manager と SQL*Plus に共通のコマンド

表 10-2 に示すコマンドは、Server Manager と SQL*Plus の両方において使用可能であり、Oracle の以前のリリースの両方のプログラムでも使用可能です。SQL*Plus で使用するために、SQL スクリプトのこれらのコマンドを変更する必要はありません。

注意： 2 つのプログラムのこれらのコマンドに対する出力において、わずかな形式の違いがある可能性があります。

表 10-2 既存の SQL*Plus コマンドに対応する Server Manager コマンド

コマンド	説明
CONNECT	指定されたユーザー名を使用して、データベースに接続します。
DESCRIBE	関数、パッケージ、パッケージ本体、プロシージャ、表またはビューについて説明します。たとえば、表の場合、表の各列の定義を表示します。
REMARK	通常、SQL スクリプト・ファイルにコメントを入力します。
SET COMPATIBILITY	互換モードを V7、V8 または NATIVE に設定します。互換性モードの設定は、文字列、整合性制約およびロールバック・セグメントの記憶領域パラメータの仕様に影響します。NATIVE はデータベースのバージョンと一致します。
SET ECHO	コマンド・ファイルの各コマンドの実行時に、START コマンドがコマンドを一覧表示するかどうかを制御します。ON はコマンドをリストし、OFF はリストを停止します。
SET NUMWIDTH	数を表示するデフォルトの幅を設定します。
SET SERVEROUTPUT	ストアド・プロシージャの出力（つまり、DBMS_OUTPUT.PUT_LINE）または SQL*Plus の PL/SQL ブロックを表示するかどうかを制御します。OFF は DBMS_OUTPUT.PUT_LINE の出力を停止し、ON は出力を表示します。
SET TERMOUT	コマンド・ファイルから実行されるコマンドによって生成される出力の表示を制御します。OFF は、画面で出力を参照しなくてもコマンド・ファイルからの出力をスプールできるよう表示を停止します。ON は出力を表示します。
SHOW ALL	ERRORS、PARAMETERS および SGA を除き、SET コマンドで設定されたシステム変数をすべてアルファベット順で表示します。
SHOW ERRORS	プロシージャ、パッケージまたは関数の前回のコンパイルで生成されたエラーがあれば、それを示します。
SPOOL	問合せ結果をオペレーティング・システム・ファイル（任意で SQL*Plus）に格納し、そのファイルをプリンタに送ります。 注意： スプール・ファイルの拡張機能は、SQL*Plus および Server Manager で異なります。拡張機能を確認するには、SPOOL コマンドを発行するときに指定します。

Server Manager コマンドに等しい SQL*Plus コマンド

表 10-3 に Server Manager コマンドと名前は異なるが対応している SQL*Plus コマンドを示します。SQL スクリプトでこれらの Server Manager コマンドのいずれかを使用する場合、スクリプトを変更して、かわりに SQL*Plus コマンドを使用します。

表 10-3 Server Manager コマンドに等しい SQL*Plus コマンド

Server Manager コマンド	SQL*Plus コマンド	説明
SET CHARWIDTH SET DATEWIDTH SET LONGWIDTH	COLUMN FORMAT	<p>SQL*Plus で COLUMN FORMAT コマンドを使用して、文字列、日付列および数値列の列幅を設定できます。SQL スクリプトで、Server Manager の SET CHARWIDTH、SET DATEWIDTH および SET LONGWIDTH を SQL*Plus の COLUMN FORMAT コマンドに置き換えます。</p> <p>たとえば、SQL スクリプトに次のエントリがあるとしたします。</p> <pre>SET CHARWIDTH 5</pre> <p>このコマンドは、Server Manager のすべての文字列幅を 5 に設定します。</p> <p>ENAME のような特定の列を 5 文字幅で表示するには、次の SQL*Plus コマンドを入力します。</p> <pre>COLUMN ENAME FORMAT A5</pre>
SET STOPONERROR	WHenever SQLERROR Whenever OSERROR	<p>WHenever SQLERROR および Whenever OSERROR コマンドを使用して、SQL エラーまたはオペレーティング・システム・エラーが発生したときに、終了または継続するよう SQL*Plus に指示します。Server Manager の SET STOPONERROR のかわりに、SQL スクリプトでこれらのコマンドを使用します。</p> <p>WHenever SQLERROR と Whenever OSERROR のいずれの場合でも、EXIT 句が SQL*Plus に終了の指示を出し、CONTINUE 句が SQL*Plus に継続の指示を出します。他の用語と句もこれらのコマンドに使用可能です。</p>

SET TIMING コマンド機能が異なる場合

SET TIMING コマンドは、Server Manager と SQL*Plus の両方に使用可能ですが、オペレーティング・システムによっては 2 つのプログラムで異なった機能を持つ場合もあります。詳細は、使用しているオペレーティング・システム固有の Oracle ドキュメントを参照してください。使用しているオペレーティング・システムの 2 つのプログラムで、SET TIMING コマンドが異なる機能を持つ場合、このコマンドが SQL*Plus で正しく機能するように、SQL スクリプトを変更します。

SQL*Plus で使用不可な Server Manager コマンド

次の Server Manager コマンドは、SQL*Plus リリース 8.1 では使用できません。

- SET MAXDATA
- SET RETRIES

使用している SQL スクリプトからこれらのコマンドを削除してください。

構文の違い

次のセクションでは、Server Manager と SQL*Plus の構文の違いを説明します。SQL*Plus を使用してスクリプトを実行する前に、SQL スクリプトを変更し、SQL*Plus 構文の表記規則に適合させます。

コメント

SQL*Plus は、次の型のコメントを認識します。

- SQL*Plus REMARK コマンド（または REM）
- SQL コメント・デリミタ、/*...*/
- ANSI/ISO コメント、--

SQL*Plus コードでこれらの型のコメントを使用するための詳細は、『SQL*Plus ユーザーズ・ガイドおよびリファレンス』を参照してください。

Server Manager はこれらの型のコメントをサポートしますが、中には動作が異なるものもあります。Server Manager では使用可能だが、SQL*Plus では使用できない特定の型のコメントもあります。次のセクションでは、各コメント型と Server Manager および SQL*Plus の構文の違いについて説明します。

REMARK コマンド（または REM）

一般に、REMARK コマンドは Server Manager と SQL*Plus で同じ動作をします。使用している SQL スクリプトで REMARK コマンドの発生を変更する必要はありません。ただし、違いが 1 つあります。SQL*Plus は、Server Manager とは異なり、REMARK コマンドを終了するハイフンを解釈します。違いについては、10-11 ページの「[分割行として使用するハイフン](#)」を参照してください。

SQL コメント・デリミタ、/*...*/

Server Manager では、SQL コメント・デリミタをセミコロンの後に置くことができますが、SQL*Plus では、SQL コメント・デリミタをセミコロンの後に置くことはできません。この 1 つの違いを除き、SQL コメント・デリミタは、Server Manager と SQL*Plus で同様に動作します。

使用している SQL スクリプトに、セミコロンの後に置かれた SQL コメント・デリミタが含まれる場合、コメント自体を次の行に移動するか、セミコロンを削除して次の行にスラッシュを置き、SQL 文を終了します。

たとえば、使用している SQL スクリプトの 1 つに、次の Server Manager コードがあるとして

```
SELECT * FROM scott.emp
      WHERE job = 'CLERK'; /* Includes only clerks. */
```

SQL*Plus では、このコードを次のエントリのいずれかに置き換えます。

```
SELECT * FROM scott.emp
      WHERE job = 'CLERK';
/* Includes only clerks. */

SELECT * FROM scott.emp
      WHERE job = 'CLERK' /* Includes only clerks. */
/
```

ANSI/ISO コメント、--

Server Manager では、ANSI/ISO コメントをセミコロンの後に置くことができますが、SQL*Plus では、ANSI/ISO コメントをセミコロンの後に置くことはできません。この 1 つの違いを除き、ANSI/ISO コメントは、Server Manager と SQL*Plus で同様に動作します。

使用している SQL スクリプトに、セミコロンの後に置かれた ANSI/ISO コメントが含まれる場合、コメント自体を次の行に移動するか、セミコロンを削除して次の行にスラッシュを置き、SQL 文を終了します。

たとえば、使用している SQL スクリプトの 1 つに、次の Server Manager コードがあるとします。

```
SELECT * FROM scott.emp
      WHERE job = 'CLERK'; -- Includes only clerks.
```

SQL*Plus では、このコードを次のエントリのいずれかに置き換えます。

```
SELECT * FROM scott.emp
      WHERE job = 'CLERK';
-- Includes only clerks.

SELECT * FROM scott.emp
      WHERE job = 'CLERK' -- Includes only clerks.
/
```

Server Manager シャープ (#)・コメント

Server Manager は、コメント行を示すシャープ符号 (#) の使用をサポートします。使用しているスクリプトにこれらのコメントが含まれる場合、(#) を (--) に変更し、SQL*Plus を使用するスクリプトを実行します。

たとえば、使用している SQL スクリプトの 1 つに、次の Server Manager コードがあるとします。

```
# This statement returns only clerks.
SELECT * FROM scott.emp
      WHERE job = 'CLERK';
```

SQL*Plus では、このコードを次のエントリに置き換えます。

```
-- This statement returns only clerks.
SELECT * FROM scott.emp
      WHERE job = 'CLERK';
```

ブランク行

Server Manager は、SQL 文内のブランク行を認めますが、SQL*Plus では認められません。SQL*Plus は、最初のブランク行で SQL 文をバッファします。これによって、特定の SQL 文を簡単に再実行できます。

使用している SQL スクリプトのいずれかが、SQL 文内に特定のブランク行を含む場合、SQL*Plus を使用するこれらのスクリプトを実行する前に、ブランク行を削除します。たとえば、使用している SQL スクリプトの 1 つに、次の SQL 文があるとします。

```
SELECT empno, ename, sal, comm

      FROM scott.emp

      WHERE job = 'MANAGER';
```

SQL*Plus では、ブランク行を削除して、このコードを次の SQL 文に置き換えます。

```
SELECT empno, ename, sal, comm
      FROM scott.emp
      WHERE job = 'MANAGER';
```

注意： SQL 文の間にブランク行を置くことはできます。この制限事項は、SQL 文内のブランク行にのみ適用されます。

ハイフン継続文字

SQL*Plus は、長い SQL 文または SQL*Plus コマンドに対する継続文字として、ハイフンの使用をサポートします。たとえば、次のように継続文字を使用できます。

```
SELECT empno, ename, sal, comm FROM scott.emp -
WHERE job = 'MANAGER';
```

Server Manager は、継続文字としてのハイフンの使用をサポートしませんが、使用している SQL スクリプトにおいてハイフンを他の用途に使用することはできます。ハイフンを使用すると、SQL*Plus はハイフンを継続文字として解釈し、予期しない出力がされることがあります。

次のセクションは、ハイフンを別の用途で使用しているのに、SQL*Plus が SQL スクリプトのハイフンの使用を継続文字として解釈する場合について説明します。使用している SQL スクリプトがハイフンを使用しているかどうかを確認し、次の記述と同様の例を避けるために、SQL スクリプトを変更します。

分割行として使用するハイフン

使用している SQL スクリプトが、コードの分割行として REMARK コマンドの後に長いハイフン行を使用することがあります。SQL スクリプトからの次のサンプル行を参考例として示します。

```
Rem -----  
SELECT empno, ename, job  
      FROM scott.emp;
```

この文で、SQL*Plus は、SELECT 文の最初の行を前の行の継続として、つまり REMARK コメントとして解釈します。このため、FROM 行は SQL 文の最初の行として解釈され、SQL*Plus は次のエラーを戻します。

```
unknown command beginning "FROM scott..." - rest of line ignored.
```

使用している SQL スクリプトの分割行としてハイフンを使用する場合、SQL*Plus を使用するスクリプトを実行する前に、ハイフンの前の REM コマンドを削除します。

マイナス記号として使用するハイフン

ハイフンは、マイナス記号と同じキーボード文字であるため、行の最後にハイフンを付けることができます。SQL スクリプトからの次のサンプル行を参考例として示します。

```
CREATE TABLE xx (  
  a int,  
  b int,  
  c int);  
  
INSERT INTO xx VALUES (10, 20, 30);  
  
SELECT a + b -  
      c FROM xx;
```

マイナス記号は継続文字として解釈されるため、SQL*Plus は (c) を別名として解釈します。

```
SELECT a + b c FROM xx;
```

このため、SQL*Plus は次の予期しない出力を戻します。

```
      C  
-----  
      30
```

ただし、Server Manager はこのコードを次のように解釈します。

```
SELECT a + b - c FROM xx;
```

このため、Server Manager は次の所定の出力を戻します。

```
A+B-C
-----
      0
```

使用している SQL スクリプトの行の最後にマイナス記号がないことを確認してください。

アンパサンド

SQL*Plus はアンパサンド (&) を置換変数として解釈し、Server Manager はアンパサンドを標準の文字列として解釈します。アンパサンドに続くテキストに定義された値がない場合、アンパサンドがコメントで囲まれても、SQL*Plus は未定義値として解釈し、ユーザーに入力を求めます。このため、アンパサンドは SQL*Plus において予期しない出力を発生する可能性があります。

アンパサンドを標準テキストの文字列として使用する SQL スクリプトがある場合、2 つのオプションがあります。

- SET ESCAPE コマンドを使用して、各アンパサンドの前にエスケープ文字を置きます。
- SET DEFINE OFF コマンドを使用して、置換変数の認識を使用禁止にします。

注意： 他に有効な置換変数がある場合、SET DEFINE OFF コマンドを使用しないでください。使用すると、他の変数が認識されなくなります。

たとえば、次の SQL 文はユーザーに SQL*Plus での入力を求めます。

```
CREATE TABLE "Employees & Managers" (  
    Employees varchar(16),  
    Managers varchar(16));  
  
Enter value for managers:
```

SET ESCAPE コマンドの使用

ユーザー・プロンプトを避けるには、SET ESCAPE コマンドを使用してエスケープ文字を設定できます。その後、アンパサンドの前にエスケープ文字を置きます。バックスラッシュ (\) はエスケープ文字として使用されます。

前の例のプロンプトを避けるには、SET ESCAPE コマンドを使用し、次のようにエントリを変更します。

```
SET ESCAPE \

CREATE TABLE "Employees \& Managers" (
    Employees varchar(16),
    Managers varchar(16));
```

SETDEFINE OFF コマンドの使用

前の例のプロンプトを避けるには、SET DEFINE OFF コマンドを使用し、次のようにエントリを変更します。

```
SET DEFINE OFF

CREATE TABLE "Employees & Managers" (
    Employees varchar(16),
    Managers varchar(16));
```

CREATE TYPE および CREATE LIBRARY コマンド

SQL*Plus は、CREATE TYPE および CREATE LIBRARY コマンドを PL/SQL ブロックとして扱います。このため、SQL*Plus では、別の行にスラッシュ (/) を使用してこれらのコマンドを終了する必要があります。Server Manager では、セミコロンでこれらのコマンドを終了することができます。

使用している SQL スクリプトにおいて、セミコロンで CREATE TYPE または CREATE LIBRARY コマンドを終了する場合、セミコロンを削除し、次の行にスラッシュを置きます。たとえば、次の SQL 文は SQL*Plus では認識されません。

```
CREATE OR REPLACE TYPE sys.dummy AS OBJECT (data CHAR(1));
CREATE OR REPLACE LIBRARY DBMS_SPACE_ADMIN_LIB TRUSTED AS STATIC;
```

SQL*Plus で実行する前に、次のようにこれらの文を編集します。

```
CREATE OR REPLACE TYPE sys.aq$_dummy_t AS OBJECT (data CHAR(1))
/
CREATE OR REPLACE LIBRARY DBMS_SPACE_ADMIN_LIB TRUSTED AS STATIC
/
```

COMMIT コマンド

SQL*Plus では、COMMIT コマンドはセミコロン (;) またはスラッシュ (/) で終了する必要がありますが、Server Manager では、COMMIT コマンドに終了文字は必要ありません。このため、終了文字なしで SQL スクリプトの COMMIT コマンドを使用する場合、これらのスクリプトを編集して終了文字を組み込みます。

たとえば、SQL スクリプトに次の COMMIT コマンドがあるとします。

```
commit
```

次の例のいずれかのように、コマンドに終了文字を組み込みます。

```
commit;
```

```
commit  
/
```

物理 ROWID の移行について

バージョン 8 の物理 ROWID には、バージョン 8 の 2 つの新機能であるパーティション表およびグローバル索引を使用可能にする新しい内部形式および外部形式が含まれています。

参照： 詳細は『Oracle8i アプリケーション開発者ガイド 基礎編』および『Oracle8i 概要』を参照してください。

この章は、新しいバージョン 8 の物理 ROWID に関連する次の移行の問題について説明します。

- [アプリケーションおよびデータの移行](#)
- [DBMS_ROWID パッケージ](#)
- [変換プロシージャの例](#)
- [スナップショット・リフレッシュ](#)
- [バージョン 8 以前のクライアントとの互換性の問題](#)
- [ROWID 関連の移行に関する質問および回答](#)

注意： この章に出てくる ROWID という単語は、物理 ROWID のことです。この章には、UROWID（ユニバーサル ROWID）データ型についての説明はありません。UROWID データ型に関連する互換性の問題の詳細は、[第 8 章「互換性および相互運用性」](#)を参照してください。

アプリケーションおよびデータの移行

ROWID は ROWID データ型および文字型の列に格納できます。ただし、格納されていたバージョン 7 の ROWID は、バージョン 7 からバージョン 8 へのデータベースの移行後には無効になります。したがって、バージョン 7 の ROWID はバージョン 8 に変換する必要があります。

アプリケーション

ROWID のアセンブルおよび逆アセンブルを手動で行わないアプリケーションは、変更や再コンパイルの必要がありません。新しい ROWID がホスト変数についての現行の格納要件を満たしているからです。

ROWID の作成やその内容分析を試みるアプリケーションでは、バージョン 8 に付属の新規パッケージ DBMS_ROWID を使用して、新しいバージョン 8 の ROWID の形式や内容进行操作する必要があります。このパッケージには、バージョン 7 の ROWID から直接使用できる情報（ファイルおよびブロック・アドレスを含む）を抽出する関数およびデータ・オブジェクト番号が含まれます。

データ

ROWID 値（ROWID データ型または文字形式）を含む列は、値が参照する表をバージョン 8 へ移行する場合と一緒に移行する必要があります。そうしないと、格納された値を使用して行を取り出すことができなくなります。一方、バージョン 8 の表に格納された ROWID 値がバージョン 7 またはバージョン 6 の表を指定している場合、列の移行は必要ありません。

列の移行は 2 段階で行われます。定義の移行およびデータの移行です。列の定義は、バージョン 7 からバージョン 8 へのディクショナリの移行時に自動的に調整されます。ROWID ユーザー列の最大サイズは拡張ディスク ROWID のサイズにまで増やされ、ROWID 列用の COL\$ の LENGTH 列を 6 バイトから 10 バイトに変更します。

データの移行は、システムがバージョン 8 でオープンされた後でないと実行できません。ユーザーは、異なる表を別々に移行したり、複数の表を並行して一度に移行したりできます。ただし、移行は、バージョン 7 のデータベース・ファイル制限を超える前に行う必要があります。これは、あいまいなブロック・アドレスが作成される場合があるためです。

ユーザーのインストレーション環境で利用できる既存の ROWID リフレッシュ手順またはバージョン 8 の DBMS_ROWID 機能を使用して、格納済みの ROWID をバージョン 7 形式からバージョン 8 形式に移行できます。

この移行ユーティリティまたは Oracle Data Migration Assistant を使用したデータの移行は、ユーザー定義列に格納された ROWID にのみ適用されます。システムによって格納された ROWID（ローカル索引内の ROWID など）はすべて、移行ユーティリティまたは Oracle Data Migration Assistant を使用した移行の後でも有効であるため、移行のための特別な操作は必要ありません。また、移行ユーティリティおよび Oracle Data Migration Assistant を使用した移行中に、非パーティション表の索引は制限付き ROWID データ型形式を使用できるため、索引が無効になることはありません。

注意： ROWID が含まれる列をインポートすると、ROWID の妥当性を再設定するために特別な注意が必要になることを示す警告メッセージが出力されます。そのような特別な注意は、インポートするすべての ROWID について必要になります。したがって、エクスポート / インポートを使用したバージョン 8 への移行には、ユーザー定義列だけでなく ROWID が含まれるすべての列について特別な注意が必要です。

DBMS_ROWID パッケージ

バージョン 8 では、DBMS_ROWID PL/SQL パッケージを提供しています。このパッケージには、次の機能が含まれます。

- バージョン 7 形式およびバージョン 8 形式での ROWID の作成。
- バージョン 7 の ROWID およびバージョン 8 の ROWID の解釈。
- バージョン 7 の ROWID およびバージョン 8 の ROWID 間での変換。

格納済み ROWID の移行は、次の項で説明するように変換関数を使用して実行できます。

ROWID 変換型

ユーザーは、変換する ROWID の型を指定する必要があります。ROWID が ROWID データ型のユーザー列に格納されているのか CHAR データ型または VARCHAR データ型のユーザー列に格納されているのかによって、ROWID 変換関数による変換の実行方法が異なるからです。

ROWID データ型の列の場合、変換プロシージャの呼出し元は、プロシージャ・パラメータとして次の値を渡す必要があります。

```
rowid_convert_internal constant integer := 0;
```

CHAR または VARCHAR データ型の列の場合、変換プロシージャの呼出し元は、プロシージャ・パラメータとして次の値を渡す必要があります。

```
rowid_convert_external constant integer := 1;
```

ROWID 変換関数

次の関数は、ROWID の変換を行います。

- ROWID_TO_EXTENDED は、ROWID をバージョン 7 (制限) 形式からバージョン 8 (拡張) 形式に変換します。
- ROWID_TO_RESTRICTED は、ROWID をバージョン 8 (拡張) 形式からバージョン 7 (制限) 形式に変換します。
- ROWID_VERIFY は、指定の ROWID をバージョン 7 形式からバージョン 8 形式に変換できるかどうかをチェックします。

次に、ROWID_TO_EXTENDED プロシージャおよび ROWID_VERIFY プロシージャについて詳しく説明します。

ROWID_TO_EXTENDED 変換プロシージャ

ROWID_TO_EXTENDED は、次のパラメータを使用します。

- **Rowid** - 外部文字形式で変換する ROWID を指定します。
- **Schema Name** - ROWID を拡張形式に変換する行を含む表の所有者スキーマ名を指定します。
- **Table Name** - ROWID を拡張形式に変換する行を含む表の表名を指定します。
- **Conversion Type** - 変換中の ROWID の型を指定します。

参照: 詳細は、11-3 ページの「[ROWID 変換型](#)」を参照してください。

ROWID_TO_EXTENDED は外部文字形式のバージョン 8 (拡張) ROWID を戻し、そのパラメータを次の方法で解釈します。

- ターゲット表のスキーマ名および表名が指定されていない場合 (NULL) ROWID_TO_EXTENDED は、変換する ROWID によって指定されたページを取り出そうと試みます。ROWID_TO_EXTENDED は、この ROWID に格納されたファイル番号を絶対ファイル番号として扱いますが、移行前にファイルが削除され、その番号が再利用されている場合には、問題が発生する可能性があります。取り出されたページが有効な表に属している場合、ROWID はその表のデータ・オブジェクト ID を使用して拡張形式に変換されます。ただし、これは非常に非効率的な変換方法であり、ターゲット表がわからない場合など、最終手段としてのみ使用することをお勧めします。ユーザーは、変換された値を使用するときに、正しい表名を知っていなければなりません。

- スキーマ名および表名が指定されている場合（好ましいアプローチ）、ROWID_TO_EXTENDED は表に対する SELECT 権限を検証し、この表のデータ・オブジェクト番号を使用して ROWID を拡張形式に変換します。変換された ROWID がこの表内の実際の行を参照するかどうかは、ROWID の変換時にも ROWID の使用時にも保証されません。
- ROWID に NULL 値が指定されている場合、プロセスによって表の指定が無視され、NULL 値が戻されます。
- ROWID に 0 の値または、より一般的な $\langle n \rangle 0.\langle m \rangle 0.\langle p \rangle 0$ という値が指定されている場合、表名は無視され、00000000.0000.0000 という形式の制限 ROWID が戻されます。
- バージョン 8 の ROWID が指定されている場合は、ROWID 内のデータ・オブジェクトが実際のデータ・オブジェクト番号（表名の指定によって決まる）に対して検証されます。これら 2 つの数字が一致しない場合は、INVALID ROWID エラーが発生します。一致した場合は、元の ROWID が戻されます。

ROWID_VERIFY

バージョン 8 では、ROWID_VERIFY という ROWID 検証プロセスが提供されています。このプロセスは ROWID を拡張形式に変換できた場合、ROWID_TO_EXTENDED と同じパラメータを使用して 0 を戻します。そうでない場合は 1 を戻します。

ただし、ユーザーが基礎表の SELECT 権限を持っていない場合や、表が存在しない場合には、ROWID_VERIFY はセキュリティ違反エラーまたは「object not found」というエラーを戻します。ROWID_VERIFY は ROWID_TO_EXTENDED プロセスで移行をする前に、不正 ROWID を識別します。

変換プロセスの例

ROWID の変換プロセスの例は次のとおりです。

例 1

表 SCOTT.T には、ROWID データ型の列 C が入っていると仮定します。これらの ROWID はすべて単一の表、SCOTT.T1 を参照します。

列 C の値は、次の文を使用して拡張形式に変換できます。

```
UPDATE SCOTT.T SET C = DBMS_ROWID.ROWID_TO_EXTENDED(C, 'SCOTT', 'T1', 0);
```

例 2

より一般的な状況では、列 C に格納された ROWID は別のさまざまな表を参照する可能性があります。表名は同じ行内にある他の列の値を基にして見つけられます。たとえば、表 T の列 TNAME に、列 C の ROWID によって参照される表の名前が入っているとします。

この場合、列 C の値は、次の文を使用して拡張形式に変換できます。

```
UPDATE SCOTT.T SET C = DBMS_ROWID.ROWID_TO_EXTENDED(C, 'SCOTT', TNAME, 0);
```

例 3

ROWID_TO_EXTENDED 関数を CREATE...AS SELECT 文の中で使用できます。たとえば、変換によって ROWID データ型のユーザー列のサイズが増え（特定のポートによって異なりますが、通常は 6 ~ 10 バイト増えます）それによって間接行が作成されることもあるので、場合によっては ROWID_TO_EXTENDED 関数を使用します。

この場合、UPDATE よりも CREATE...AS SELECT を選択します。

```
CREATE TABLE SCOTT.TNEW (A, B, C)
  AS SELECT A, B, DBMS_ROWID.ROWID_TO_EXTENDED(C, 'SCOTT', 'T1', 0) FROM SCOTT.T;
```

例 4

列 C に格納された ROWID のターゲット表が不明な場合、変換は次の文を使用して実行できます。

```
UPDATE SCOTT.T SET C = DBMS_ROWID.ROWID_TO_EXTENDED(C, NULL, NULL, 0);
```

例 5

次の SQL 文を使用して、変換の前にいくつかの不正な ROWID を検索できます。

```
SELECT ROWID, C FROM SCOTT.T WHERE DBMS_ROWID.ROWID_VERIFY(C, NULL, NULL, 0) = 1;
```

スナップショット・リフレッシュ

バージョン 8 の ROWID データ型では、マスター・サイトおよびスナップショット・サイトの両方をバージョン 8 へアップグレードするとき、すべての ROWID スナップショットの完全リフレッシュを実行する必要があります。

参照： 詳細は『Oracle8i レプリケーション・ガイド』の付録 B「移行および互換性」を参照してください。

バージョン 8 以前のクライアントとの互換性の問題

リリース 8.0 以前のクライアントはバージョン 8 のデータベースにアクセスでき、バージョン 8 のクライアントもリリース 8.0 以前のデータベースにアクセスできます。リリース 8.0 以前のデータベースからバージョン 8.0 のデータベースへ戻される疑似列 ROWID および ROWID データ型の列のバイナリ値と文字値は、常にリリース 8.0 の前の（制限付きの）形式となります。これは、リリース 8.0 以前のシステムでは拡張形式 ROWID を認識できないためです。

バージョン 8 に付属の DBMS_ROWID パッケージは、バージョン 7 の ROWID の内容解釈およびバージョン 7 形式での ROWID の作成に使用できます。

バージョン 8 以前のクライアントがバージョン 8 のデータベースにアクセスすると、ROWID をバージョン 8 の拡張形式で受け取ります。したがって、バージョン 8 以前のクライアントは、バージョン 8 のサーバーが戻す ROWID の内容を解釈できません。

バージョン 8 のスナップショットの互換性は、リリース 7.1.4 以降に制限されます。さらに、マスター・サイトがアップグレードされた場合は、バージョン 8 のアップグレード・スクリプトによってログを無効にする必要があります。これによって、スナップショットでは、再度高速リフレッシュが可能になる前に完全リフレッシュが強制的に実行されます。

ROWID 関連の移行に関する質問および回答

問：バージョン 8 には、バージョン 7 のインポート・クライアントに対する制限はありますか？

答：バージョン 8 の表内に拡張 ROWID 値が入っている行が 1 行でもあれば、バージョン 7 のクライアントが ROWID ユーザー列のあるバージョン 8 の表をインポートすることはできません。

問：Forms3（および Forms4）では、実表の更新時に新しい ROWID データ型形式は認識されますか？

答：バージョン 8 のデータベースにアクセスしようとするフォーム・アプリケーションは、パッチ #380655 を使用して再リンクする必要があります。

問：バージョン 8 の ROWID を変更した場合、PRO* プリコンパイル済みプログラムに影響はありますか？

答：ROWID を使用しても、その形式に依存しないプログラムには影響がありません。バージョン 7 の ROWID データ型形式に依存するプログラムは、新しいパッケージ DBMS_ROWID を使用するように変更する必要があります。

問：「WHERE CURRENT of CURSOR」命令はまだ動作しますか？

答：動作します。リリース 8.0 より前のクライアントからバージョン 8 のサーバーにアクセスする場合や、バージョン 8 のクライアントからバージョン 8 より前のサーバーにアクセスする場合でも動作します。

問：現在、内部 ROWID データ型形式として動的 SQL とバインドを使用しています。malloc() を使用して領域を拡大する必要はありますか？

答：バージョン 8 の ROWID は、ホスト変数についてのバージョン 7 の記憶域要件を満たしているので、変更したり領域を追加したりする必要はありません。

問：自分の表の列を ROWID データ型に定義できますか？

答：列を ROWID データ型に定義することはできます。バージョン 7 では ROWID 列に 6 バイト必要でしたが、バージョン 8 では 10 バイト必要になります。

問：現在、バージョン 7 の ROWID データ型形式に依存しています。変換アルゴリズムは、将来、文書化されるでしょうか？

答：バージョン 8 の新しい ROWID データ型形式は、そのような用途については文書化されていません。しかし、バージョン 8 では、バージョン 8 の ROWID の内容を解釈するための DBMS_ROWID (PL/SQL) パッケージが提供されています。

問：将来、自分の索引を再作成する必要がでてくるでしょうか？

答：古い ROWID データ型形式を格納している列に対して作成された索引だけは、データの移行後に再作成する必要があります。

問：ROWID データ型を、リリース 8.0 より前の PL/SQL または RPC、FORMS で使用しています。この方法で今後も作業を続けられるでしょうか？

答：ROWID が ROWID データ型のホスト変数に戻される際の形式は同じなので、一般的に変更は必要ありません。ただし、次の特例を除きます。

dblink (異機種間 dblink) を介したバージョン 7 のサーバーからバージョン 8 のデータベースへのリモート・マップ問合せは、OCI による DTYRID 型 (CHR 変換なし) としての ROWID フェッチ時に、エラー ORA3116 で終了します。次の方法で問題を回避します。

- DTYCHR 型としてかわりに ROWID フェッチを使用すると、暗黙の変換を起動して問題を回避します。
- バージョン 7 のサーバーで SQLT_RID およびパッチ (Oracle から利用可能) を使用すると、CHR 変換を起動せずに問題を回避します。

古いバージョン 8 リリースへの ダウングレード

この章の内容は、Oracle のインストレーションのリリース 8.1 にのみ適用します。ダウングレードとは、Oracle データベースを同じバージョンの前のリリースへ変換することを意味します。たとえば、データベースをリリース 8.1 からリリース 8.0 に変換するときなどです。ダウングレードとは、Oracle データベースを前のバージョンに変換することも意味します。たとえば、データベースを Oracle8i から Oracle7 に変換するときなどです。この章では、Oracle の以前の 8.1 リリースまたは 8.0 リリースへのダウングレードについて説明します。Oracle7 にダウングレードする場合は、[第 13 章「Oracle7 へのダウングレード」](#)を参照してください。

次の項の指示順に、プロシージャを実行し、データベースをダウングレードしてください。

- [全オフライン・バックアップの実行](#)
- [非互換性の削除](#)
- [データベース互換性のリセット](#)
- [データベースのダウングレード](#)

参照： ダウングレードの一部には、オペレーティング・システム固有の部分があります。ダウングレードの詳細は、オペレーティング・システム固有の Oracle ドキュメントを参照してください。

全オフライン・バックアップの実行

ダウングレードする前に、リリース 8.1 のデータベースに全オフライン・バックアップを実行してください。

参照： 詳細は、『Oracle8i バックアップおよびリカバリ・ガイド』を参照してください。

非互換性の削除

非互換性の削除のプロセスは、以前のリリース 8.1 とリリース 8.0 のどちらにダウングレードするかによって異なります。ダウングレードするリリースの適切な項の指示に従ってください。

リリース 8.1 にダウングレードする場合の非互換性の削除

リリース 8.1.4 または 8.1.3 のいずれかにダウングレードする場合、非互換性を削除するために次の処置を実行してください。

- 単一表ハッシュ・クラスタを使用していた場合は、12-8 ページの「[全単一表ハッシュ・クラスタの削除](#)」の指示に従ってください。
- 一時表を使用していた場合は、12-6 ページの「[一時表の削除](#)」の指示に従ってください。
- Java オブジェクトを作成した場合は、12-26 ページの「[Java](#)」の指示に従ってください。リリース 8.1.4 は Java をサポートしていますが、リリース 8.1.5 では形式が変わります。したがって、リリース 8.1.5 で使用した Java オブジェクトは 8.1 リリースでは使用できません。リリース 8.1.4 にダウングレードし Java オブジェクトを保存する場合は、それらを削除する前にエクスポートし、リリース 8.1.4 にダウングレードした後インポートすることができます。

これらの処置を完了した後、次の機能を 1 つでも使用していた場合、12-42 ページの「[データベース互換性のリセット](#)」の項に進んでください。

- 単一表ハッシュ・クラスタ
- 一時表
- Java
- オンライン索引

ただし、これらの機能を使用していなかった場合、データベースの互換性を再設定する必要はありません。続けて 12-43 ページの「[データベースのダウングレード](#)」の項に進んでください。

リリース 8.0 にダウングレードする場合の非互換性の削除

使用しているデータベースの互換性水準がダウングレードするリリースよりも高い場合、ダウングレードする前に削除すべき前のリリースの非互換性が、データベース内にある可能性があります。次の SQL 文を発行して、COMPATIBLE パラメータの設定をチェックします。

```
SELECT name, value, description FROM v$parameter
WHERE name='compatible';
```

COMPATIBLE パラメータがダウングレードしようとしているリリース、またはそれ以前のリリースに設定されている場合、非互換性の削除は必要ありません。たとえば、リリース 8.0.5 にダウングレードする場合、COMPATIBLE パラメータが 8.0.5 またはそれ以下ならば、非互換性の削除は必要ありません。この場合、使用しているデータベースには、ダウングレードしようとしているリリースとの間の非互換性は存在しません。したがって、この項の残りを省略して 12-43 ページの「[データベースのダウングレード](#)」の項に進んでください。

ただし、リリース 8.0 にダウングレードする場合、COMPATIBLE パラメータがダウングレードしようとしているリリースより高い設定の場合は、非互換性がいくつか存在している可能性があります。たとえば、リリース 8.0.5 にダウングレードし、COMPATIBLE が 8.1.0 またはそれ以上の設定である場合、非互換性が存在する可能性があります。同様に、リリース 8.0.3 にダウングレードしようとするときに、COMPATIBLE が 8.0.4 またはそれ以上に設定されていると、非互換性が存在する可能性があります。

非互換性が存在する場合、次の一般的な手順でダウングレードしようとするリリースとの非互換性を削除できます。

1. システム・プロンプトで、\$ORACLE_HOME/rdbms/admin ディレクトリに変更します。
2. Server Manager を起動します。
3. データベース・インスタンスへ接続します。

```
SVRMGR> CONNECT INTERNAL
```

4. 次のステップを行って、非互換性を識別してください。
 - a. V\$COMPATIBILITY の動的パフォーマンス・ビューに問い合わせ、非互換性を識別します。

```
SVRMGR> SELECT * FROM v$compatibility WHERE release != '0.0.0.0.0';
```

RELEASE 列の値がダウングレードしようとしているリリースより高い場合は、非互換性が存在します。

注意： この問合せは、0.0.0.0.0の互換性水準の機能を表示しません。これらの機能は現在使用されておらず、その機能に対しての処置は必要ありません。

- b. utlincmpt.sql を実行します。

```
SVRMGR> SPOOL utlincmpt.out
SVRMGR> @utlincmpt.sql
SVRMGR> SPOOL OFF
```

utlincmpt.sql のスクリプトは、この章の残りに記述されている問合せをすべて実行して、非互換性を識別します。したがって、utlincmpt.sql スクリプトを実行することによって、この章のこれ以降の部分で説明するすべての SELECT 文を実行できます。

utlincmpt.sql のスクリプトを実行後、utlincmpt.out のファイルを表示し、SELECT 文が値を戻したインスタンスを探します。戻った値がリリース 8.0 との非互換性です。

5. 非互換性をすべて削除または変更し、ダウングレードしようとしているリリースと互換性のあるデータベースを作成します。

次の項では、リリース 8.0 との非互換性の削除について説明します。非互換性を削除するには、SYS ユーザーの権限を使用して処置を実行する必要があります。したがって、特に指示がない限り、SYS ユーザーでログインし、次の項に記述されている処置を SYSDBA で実行してください。

また、互換性レベルが 8.1.0 またはそれ以上でデータベースを作成した場合、リリース 8.0 と非互換のシステムが定義した型を作成します。これら非互換性を削除するには、リリース 8.1 に付属する utldst.sql のスクリプトを実行します。

```
@utldst.sql
```

注意： Oracle8i Enterprise Edition から Oracle8i (以前は Workgroup Server) へダウングレードする場合は、ダウングレードの実行前に、Oracle8i Enterprise Edition の拡張機能を使用するアプリケーションを変更し、それらの拡張機能を使用しないようにしてください。

表領域

この項では、表領域に関連する非互換性の削除について説明します。

トランスポートされた表領域の削除

表領域をダウングレードしようとしているデータベースに移動、または表領域をこのデータベースから他のデータベースに移動するのにトランスポータブル表領域機能を使用していた場合、ダウングレードする前に次のステップを実行してください。

1. 次の SQL 文を発行し、データベースにつながれたトランスポートされた表領域を識別します。

```
SELECT tablespace_name, plugged_in
FROM dba_tablespaces
WHERE plugged_in = 'YES';
```

2. SQL 文によって示された各トランスポートされた表領域を削除または移動します。

トランスポートされた表領域にデータを保存する必要がない場合は、その表領域を削除してください。データを保存する必要がある場合には、現行のデータベースからデータをエクスポートし、ダウングレード後にそれをインポートするか、ダウングレードする前に表領域を他のデータベースにトランスポートしてください。

3. DBMS_TTS.DOWNGRADE を実行します。

```
EXECUTE dbms_tts.downgrade;
```

DBMS_TTS.DOWNGRADE プロシージャは、トランスポータブル表領域機能が使用したシステム表領域にある一時表を削除します。

ローカル管理の表領域の使用中止

リリース 8.1 では、ローカル管理の表領域をサポートします。ダウングレードする前に、ローカル管理のすべての表領域をディクショナリ表領域に変換する必要があります。

ローカル管理の表領域を識別するには、次の SQL 文を入力します。

```
SELECT tablespace_name, extent_management
FROM dba_tablespaces
WHERE extent_management = 'LOCAL';
```

一覧表示されたすべての表領域で、DBMS_ADMIN.TABLESPACE_MIGRATE_FROM_LOCAL を実行します。たとえば、TS_1 という表領域が示されている場合、次の SQL 文を入力して TS_1 をディクショナリの表領域に変換します。

```
EXECUTE dbms_admin.tablespace_migrate_from_local('ts_1');
```

スキーマ・オブジェクト

この項では、スキーマ・オブジェクトに関連する非互換性の削除について説明します。

一時表の削除

ダウングレードする前に、すべての一時表を削除します。既存の一時表領域を識別するために、次の SQL 文を発行します。

```
SELECT owner, table_name FROM dba_tables
WHERE temporary = 'Y' AND
       table_name NOT LIKE 'RUPD$%' AND
       table_name NOT LIKE 'ATEMPTAB$';
```

一覧表示されたすべての表を削除します。

索引および索引構成表上におけるキー圧縮の中断

ダウングレードする前に、データベースのキー圧縮がある索引および索引構成表の使用を中断してください。キー圧縮のある既存索引および索引構成表を識別するために、次の SQL 文を発行します。

```
SELECT index_name, index_type, table_owner, table_name
FROM dba_indexes WHERE compression = 'ENABLED';
```

表示された各索引に対して、ALTER INDEX ... REBUILD NOCOMPRESS 文を発行します。たとえば、キー圧縮のある I_JOB という索引がある場合、次の SQL 文を入力します。

```
ALTER INDEX i_job REBUILD NOCOMPRESS;
```

表示されたすべての索引構成表に対して ALTER TABLE ... MOVE NOCOMPRESS 文を発行します。たとえば、キー圧縮のある IOT_ITEM という索引がある場合、次の SQL 文を入力します。

```
ALTER TABLE iot_item MOVE NOCOMPRESS;
```

注意： ALTER TABLE ... MOVE NOCOMPRESS 文は、索引構成表として格納されたネストした格納表には使用できません。ネストした表の各列に対しては、ネストした表の列に対するヒープ格納表を指定し列を削除するか、CREATE TABLE ... AS SELECT 文で親表を再作成してください。親表はネストした表の列を持つ表です。

LOB および索引構成表の VARRAY の使用の中断

ダウングレードする前に、データベース内の LOB または VARRAY のあるすべての索引構成表を削除します。LOB のある既存の索引構成表を識別するために、次の SQL 文を発行します。

```
SELECT column_name, t.owner, t.table_name
FROM dba_lobs l, dba_tables t
WHERE l.table_name = t.table_name
AND l.owner = t.owner
AND t.iot_type = 'IOT';
```

VARRAY のある既存の索引構成表を識別するために、次の SQL 文を発行します。

```
SELECT v.parent_table_column, t.owner, t.table_name
FROM dba_varrays v, dba_tables t
WHERE v.parent_table_name = t.table_name
AND v.owner = t.owner
AND t.iot_type = 'IOT';
```

SQL 文によって一覧表示された表内のデータを保存する必要がある場合は、表を削除します。ただし、テーブル内のデータを保存する必要がある場合は、各表に対して次のステップを実行します。

1. LOB または VARRAY のある索引構成表から全列を選択し、索引構成されていない新しい表を作成してください。

たとえば、次のように定義された、LOBIOT という名前の LOB のある索引構成表があると仮定します。

```
CREATE TABLE lobiot (a INT, b CLOB, c INT PRIMARY KEY) ORGANIZATION INDEX;
```

LOBIOT 内のデータを使用して、NIOTD2 という名前の索引構成されていない表を作成するために、次の SQL 文を発行します。

```
CREATE TABLE niotd2 (a,b,c PRIMARY KEY) AS SELECT * FROM lobiot;
```

2. 新しい表が正しく機能していると確認して、LOB または VARRAY のある索引構成表を削除します。
3. 新しい表を元の名前に改名してください。

索引構成表のすべての 2 次索引の削除

ダウングレードする前に、データベース内にある索引構成表の 2 次索引をすべて削除します。索引構成表の既存の 2 次索引を識別するために、次の SQL 文を発行します。

```
SELECT index_name, i.owner, t.table_name
FROM dba_indexes i, dba_tables t
WHERE i.index_type = 'NORMAL'
      AND i.table_name = t.table_name
      AND t.owner = i.table_owner
      AND t.iot_type = 'IOT';
```

一覧表示された索引を削除してください。

未使用および部分的に削除された列の削除

ダウングレードする前に、未使用列および部分的に削除された列をすべて削除します。

未使用の列の削除 データベースの表内に未使用列がある場合、ダウングレードできません。未使用列を含む表を識別するために、次の SQL 文を発行します。

```
SELECT * FROM dba_unused_col_tabs;
```

テーブル内の未使用の列すべてを削除するには、ALTER TABLE ... DROP UNUSED COLUMNS コマンドを使用します。一覧表示された各表に対して、このコマンドを実行します。たとえば、次のコマンドを入力して CUSTOMERS という表内の未使用の列すべてを削除します。

```
ALTER TABLE customers DROP UNUSED COLUMNS;
```

部分的に削除された列の削除 データベースの表内に部分的に削除された列がある場合、ダウングレードできません。部分的に削除された列を含む表を識別するために、次の SQL 文を発行します。

```
SELECT * FROM dba_partial_drop_tabs;
```

テーブル内の部分的に削除された列すべてを削除するには、ALTER TABLE ... DROP COLUMNS CONTINUE コマンドを使用します。一覧表示された各表に対して、このコマンドを実行します。たとえば、CUSTOMERS という名前の表内にある部分的に削除された列すべてを削除するには、次のコマンドを入力します。

```
ALTER TABLE customers DROP COLUMNS CONTINUE;
```


全単一表ハッシュ・クラスタの削除

ダウングレードする前に、単一表ハッシュ・クラスタをすべて削除してください。単一表のみのハッシュ・クラスタをチェックするには、次の SQL 文を発行します。

```
SELECT cluster_name, single_table FROM dba_clusters
       WHERE single_table='Y';
```

一覧表示されたクラスタをすべて削除します。

互換性のないマテリアライズド・ビューの削除

次の SQL 文を発行して、リリース 8.0 と互換性のないマテリアライズド・ビューを識別します。

```
SELECT mv.owner, mv.name
       FROM dba_snapshots mv, dba_mview_analysis mva
       WHERE mva.owner = mv.owner
       AND mva.mview_name = mv.name;
```

一覧表示されたマテリアライズド・ビューをすべて削除します。たとえば、SCOTT に所有される、MV_1 という名前のマテリアライズド・ビューが表示された場合、このマテリアライズド・ビューを削除するために次の SQL 文を発行します。

```
DROP MATERIALIZED VIEW scott.mv_1;
```

ダウングレード中に変更または削除されるマテリアライズド・ビューの識別

次の項では、12-43 ページの「データベースのダウングレード」に記述されているダウングレード・プロセス中に、変更または削除されるマテリアライズド・ビューの識別方法について説明します。

注意： マテリアライズド・ビューは、スナップショットと同義語です。

REFRESH ON COMMIT モードを REFRESH ON DEMAND モードに変更 リリース 8.1 では、マテリアライズド・ビューに REFRESH ON COMMIT モードを使用できます。ただし、リリース 8.0 では、このモードを使用できません。REFRESH ON COMMIT モードのマテリアライズド・ビューを識別するには、次の SQL 文を発行します。

```
SELECT owner, name, refresh_mode
       FROM dba_snapshots
       WHERE refresh_mode = 'COMMIT';
```

一覧表示されたマテリアライズド・ビューはすべて REFRESH ON COMMIT モードです。ダウングレード時に、これらのマテリアライズド・ビューは自動的に REFRESH ON DEMAND モードに変更されます。

ダウングレード後に使用不可能な FAST REFRESH モード 結合または GROUP BY 句（一括問合せ）を使用するマテリアライズド・ビューは、ダウングレード後には FAST REFRESH モードを使用できません。

NEVER REFRESH モードのマテリアライズド・ビューの削除 リリース 8.1 では、マテリアライズド・ビューに NEVER REFRESH モードを使用することができます。ただし、リリース 8.0 では、このモードを使用できません。NEVER REFRESH モードのマテリアライズド・ビューを識別するには、次の SQL 文を発行します。

```
SELECT owner, name, type
FROM dba_snapshots
WHERE type = 'NEVER';
```

一覧表示されたマテリアライズド・ビューはすべて NEVER REFRESH モードです。ダウングレード時に、これらのマテリアライズド・ビューは自動的に削除されます。

PREBUILT TABLE 句によって作成されたマテリアライズド・ビューの削除 リリース 8.1 では、PREBUILT TABLE 句を使用してマテリアライズド・ビューを作成できます。ただし、リリース 8.0 では、これらのビューをサポートしていません。ダウングレード時に PREBUILT TABLE 句で作成したビューは、自動的に削除されます。

ビューなしで作成されたマテリアライズド・ビュー ダウングレード時にビューなしで作成されたマテリアライズド・ビューは自動的に削除されます。

BUILT DEFERRED 句で作成されたマテリアライズド・ビューのリフレッシュ ダウングレード時に、BUILT DEFERRED 句で作成したすべてのビューについて自動的に完全リフレッシュが行われます。

相互参照ビューおよびリリース 8.0.4 以前へのダウングレード

相互参照ビューがあり、リリース 8.0.4 以前のリリースにダウングレードしようとしている場合、これらのビューを削除する必要があります。相互参照ビューがない場合、またはリリース 8.0.5 以降にダウングレードする場合、この項は省略してください。

注意： ダウングレードする前または後に、相互参照ビューを削除できません。相互参照ビューはダウングレード操作に影響しません。

リリース 8.0.3 では、相互参照ビューをサポートしていません。リリース 8.0.3 にダウングレードする場合は、相互参照ビューをすべて削除してください。

相互参照ビューはリリース 8.0.4 以降でサポートされます。ただし、リリース 8.0.4 にダウングレードする場合にも、これらのビューの削除は必要です。ダウングレード終了後、削除した相互参照ビューをリリース 8.0.4 に再作成できます。バグ #662863 のため、この処置は必要です。このバグはリリース 8.0.4 にありますが、リリース 8.0.5 以降では訂正されています。

相互参照ビューとは、MAKE_REF 演算子を介してオブジェクト・ビューが互いに参照するビューのことです。次の相互参照ビューの例では、HUSBAND 型および WIFE 型はお互いに参照し、オブジェクト・ビューは MAKE_REF 演算子で作成されています。

```
CREATE TYPE husband
/

CREATE TYPE wife AS object
    (id2 NUMBER,
     name2 CHAR(10),
     salary number,
     buddy2 REF husband)
/

CREATE OR replace TYPE husband AS object
    (id NUMBER,
     name CHAR(10),
     buddy REF wife)
/

CREATE TABLE husbandtab
    (id NUMBER,
     name CHAR(10),
     buddy NUMBER);

CREATE TABLE wifetab
    (id2 NUMBER,
     name2 CHAR(10),
     salary NUMBER,
     buddy2 NUMBER);
```

```
CREATE VIEW husbandview OF husband
  WITH object OID(id) AS
  SELECT id, name, NULL FROM husbandtab;

CREATE VIEW wifeview OF wife WITH object OID(id2) AS
  SELECT id2, name2, salary,
  MAKE_REF(husbandview, buddy2)
  FROM wifetab;

CREATE OR replace VIEW husbandview
  OF husband WITH object OID(id) AS
  SELECT id, name, MAKE_REF(wifeview, buddy)
  FROM husbandtab;
```

無効になる可能性のあるビットマップ索引のチェック

リリース 8.1 にはビットマップ索引の保護があります。これらは、ビットマップ索引が不本意に無効にされないように保護します。

参照： 詳細は、8-25 ページの「[ビットマップ索引の保護](#)」を参照してください。

12-43 ページの「[データベースのダウングレード](#)」に記述されているダウングレード・プロセスでリリース 8.0 にダウングレードするとき、この新機能が保護するビットマップ索引は自動的に無効になります。ダウングレード・プロセス中に無効になる可能性のある索引を一覧表示させるには、次の SQL 文を発行します。

```
SELECT o.name INDEX_NAME, u.name INDEX_OWNER
  FROM sys.user$ u, sys.obj$ o, sys.ind$ i, sys.tab$ t
 WHERE t.obj# = o.obj#
 AND i.bo# = t.obj# AND t.spare1 > 32767
 AND i.type# = 2 AND o.owner# = u.user#;
```

ファンクション索引の削除

データベースにファンクション索引がある場合、ダウングレードできません。ファンクション索引を識別するには、次の SQL 文を発行します。

```
SELECT DISTINCT index_owner, index_name FROM dba_ind_columns
WHERE column_name IS NULL;
```

ファンクション索引をすべて削除するために、DROP INDEX 文を使用します。たとえば、FUNCIN1 というファンクション索引を削除するには、次の SQL 文を発行します。

```
DROP INDEX funcin1;
```

この文を一覧表示された各ファンクション索引に発行します。

拡張索引作成機能の解除

リリース 8.1 は拡張索引作成機能をサポートします。この機能によって、ドメイン索引、索引タイプおよび演算子の作成が可能になります。ダウングレードする前に、これらのオブジェクトを削除する必要があります。

ドメイン索引の識別および削除 ドメイン索引を識別するために、次の SQL 文を発行します。

```
SELECT owner, index_name, index_type
FROM dba_indexes
WHERE index_type = 'DOMAIN';
```

一覧表示されたドメイン索引を削除します。

索引タイプの識別および削除 索引タイプを識別するために、次の SQL 文を発行します。

```
SELECT owner, indextype_name FROM dba_indextypes;
```

一覧表示された索引タイプを削除するには、DROP INDEXTYPE を使用します。たとえば、USER2 が所有する IX_TYPE という索引タイプが表示された場合、索引タイプを削除するために、次の SQL 文を発行します。

```
DROP INDEXTYPE user2.ix_type;
```

演算子の識別および削除 演算子を識別するために、次の SQL 文を発行します。

```
SELECT owner, operator_name FROM dba_operators;
```

一覧表示された演算子を削除するには、DROP OPERATOR を使用します。たとえば、USER3 が所有する OP1 という演算子が表示された場合、演算子を削除するために、次の SQL 文を発行します。

```
DROP OPERATOR user3.op1;
```

全ディメンションの削除

ダウングレードする前に、ディメンションをすべて削除する必要があります。リリース 8.0 では、ディメンションをサポートしません。

削除しなければならないディメンションを識別するために、次の SQL 文を発行します。

```
SELECT * FROM dba_dimensions;
```

この SQL 文によって一覧表示されたディメンションを削除してください。

パーティション化

この項では、リリース 8.1 のパーティション化機能の使用禁止について説明します。

パーティション化された索引構成表の使用の中断

ダウングレードする前に、データベース内にあるパーティション化された索引構成表をすべて削除します。パーティション化された索引構成表を識別するために、次の SQL 文を発行します。

```
SELECT table_name, tablespace_name, iot_type, partitioned  
FROM dba_tables WHERE partitioned = 'YES' AND iot_type = 'IOT';
```

一覧表示された表内のデータを保存する必要がない場合は、表を削除します。ただし、表のデータを保存する必要がある場合は、表に対して次のステップを実行します。

1. 表のパーティション 2 次索引をすべて削除します。
2. パーティション化された索引構成表から行をすべて選択し、索引構成でもパーティションでもない新しい表を作成します。

たとえば、次のように定義された、PIOT という名前のパーティション索引構成表があると仮定します。

```
CREATE TABLE piot (a int, b int, c int, d int, e int,  
    PRIMARY KEY (d,e)) ORGANIZATION INDEX  
    PARTITION BY RANGE (d)  
    (  
        PARTITION itp1 VALUES LESS THAN (15),  
        PARTITION itp2 VALUES LESS THAN (30),  
        PARTITION itp3 VALUES LESS THAN (MAXVALUE)  
    );
```

次の SQL 文を発行して、PIOT のデータを使用する IOT という名前の非パーティション索引構成表を作成してください。

```
CREATE TABLE iot (a, b, c, d, e,  
    PRIMARY KEY (d,e)) ORGANIZATION INDEX  
    AS SELECT * FROM piot;
```

または、索引構成なしでパーティションを保存する場合、次の SQL 文を発行して、PIOT のデータを使用する PAR という名前の索引構成ではないパーティション表を作成してください。

```
CREATE TABLE par (a, b, c, d, e,  
    PRIMARY KEY (d,e)) PARTITION BY RANGE (d)  
    (  
        PARTITION itp1 VALUES LESS THAN (15),  
        PARTITION itp2 VALUES LESS THAN (30),  
        PARTITION itp3 VALUES LESS THAN (MAXVALUE)  
    ) AS SELECT * FROM piot;
```

3. 新しい表が正しく機能していることを確認して、パーティション索引構成表を削除してください。
4. 新しい表を元の名前に改名してください。

パーティション・オブジェクト表の使用の中断

リリース 8.1 は、オブジェクト表のパーティション化および次のユーザー・タイプの表をサポートします。

- ネストした表
- オブジェクト
- REF
- VARRAY

ダウングレードする前に、パーティション・オブジェクト表をすべて削除します。パーティション・オブジェクト表すべてを識別するには、次の SQL 文を発行します。

```
SELECT UNIQUE t.table_name, t.owner
  FROM dba_part_tables t, dba_tab_columns c
 WHERE t.table_name = c.table_name
 AND c.data_type IN
      (SELECT type_name
       FROM dba_types
       WHERE predefined = 'NO');
```

一覧表示された表内のデータを保存する必要がない場合は、表を削除します。ただし、一覧表示された 1 つ以上の表内のデータを保存する必要がある場合は、CREATE TABLE ... AS SELECT 文を使用して、表のデータを非パーティション表にコピーします。

たとえば、SQL 文によって OBP1 という表が表示され、この表のデータを保存したい場合は、次のステップを実行します。

1. 次の SQL 文を発行して、TEMP_OBP1 という表を作成します。

```
CREATE TABLE temp_obp1 AS SELECT * FROM obp1;
```

2. 表 TEMP_OBP1 にデータがあり、正しく機能していることを確認して、OBP1 を削除します。

```
DROP TABLE obp1;
```

3. TEMP_OBP1 を OBP1 に改名します。

```
ALTER TABLE temp_obp1 RENAME TO obp1;
```


コンポジット・メソッドを使用するパーティション表使用の中断

リリース 8.1 は、コンポジット・メソッドおよび RANGE 以外の非コンポジット・メソッドを使用するパーティション表の作成をサポートします。データベースにそのような表がある場合、次の処置の 1 つを行う必要があります。

- これらの表内のデータを保存する必要がない場合は、表を削除します。
- データの保存が必要な場合は、データを非パーティション表にコピーするか、RANGE がパーティション化した表にデータをコピーします。他のオプションは、EXCHANGE を使用して（ハッシュ・メソッドを使用してパーティション化した表の）パーティション、または（コンポジット・メソッドを使用してパーティション化した表の）サブ・パーティションを非パーティション表へ交換します。

コンポジット・メソッド、および RANGE 以外の非コンポジット・メソッドでパーティション化した表をリストで示すには、次の SQL 文を発行します。

```
SELECT owner, table_name FROM dba_part_tables
WHERE partitioning_type != 'RANGE' or SUBPARTITIONING_TYPE != 'NONE';
```

データ型

この項では、リリース 8.1 以降でのみ有効なデータ型を使用禁止にする方法について説明します。

ユニバーサル ROWID データ型の使用をすべて削除

UROWID（ユニバーサル ROWID）データ型の使用をすべて削除するには、次の項のステップを実行します。

ALL UROWID 列の削除 UROWID データ型列のある表すべてを一覧表示させるには、次の SQL 文を発行します。

```
SELECT owner, table_name, column_name FROM dba_tab_columns
WHERE data_type = 'UROWID' ORDER BY owner, table_name;
```

このコマンドの結果、一覧表示された各表の UROWID データ型列を削除するか、または表全体を削除してください。

UROWID 引数のストアド・プロシージャをすべて削除 UROWID データ型の引数がある表すべてを一覧表示させるには、次の SQL 文を発行します。

```
SELECT owner, object_name, package_name, argument_name
FROM all_arguments
WHERE data_type = 'UROWID' AND package_name != 'STANDARD'
ORDER BY owner, object_name, package_name;
```

表示された各プロシージャを削除するか、または引数を ROWID データ型に変更します。

既存の連鎖行および例外表の削除 リリース 8.1 では、UROWID データ型によって、連鎖行および例外に対して単一表の使用が可能になります。ただし、この機能はリリース 8.0 のデータベースではサポートされません。したがって、連鎖行および例外の両方に対して複数の表を作成する必要があります。これは通常表すべてに対して最低 1 つの表、および各索引構成表に対して最低 1 つの表が必要なためです。

CH_ROWS という連鎖行表をダウングレードするために、次のステップを実行します。

1. 既存の CH_ROWS 表を削除します。
2. 通常表の連鎖行を格納するために、utlchain.sql スクリプトを使用して CH_ROWS 表を再作成します。
3. DBMS_IOT.BUILD_CHAIN_ROWS_TABLE プロシージャを使用して、各索引構成表に対する単一連鎖表を作成します。

EXC_TB という例外表をダウングレードするために、次のステップを完了してください。

1. 既存 EXC_TB 表を削除します。
2. 通常表の例外を格納するために、utlexcp.sql スクリプトを使用して、EXC_TB 表を再作成します。
3. DBMS_IOT.BUILD_EXCEPTIONS_TABLE プロシージャを使用して、各索引構成表に対する例外表を作成します。

リリース 8.1LOB 機能の使用中断

リリース 8.1 は、新しいLOB 機能のいくつかをサポートします。ダウングレードする前に、次の項に記述されている処置を実行して、これらの新しい機能の使用を中断してください。

可変幅のキャラクタ・セットのあるデータベースの表から CLOB および NCLOB を削除

データベースが可変幅のキャラクタ・セットを使用している場合、次のステップを完了して CLOB および NCLOB 列をすべて削除します。固定幅のキャラクタ・セットを使用するデータベースの場合は、このプロシージャを完了する必要はありません。

1. 次の SQL 文を発行して、LOB 列を持つ表をすべて一覧表示します。

```
SELECT owner, table_name, column_name FROM dba_lobs
WHERE dba_lobs.owner != 'SYSTEM'
AND table_name NOT IN ('KOTAD$', 'KOTMD$', 'KOTTB$', 'KOTTD$');
```

2. 一覧表示された各表に DESC を実行して、データベースに CLOB 列または NCLOB 列があるか確認します。DESC を実行するときには、TYPE 列をチェックします。
3. CLOB および NCLOB 列を削除するか、その列のある各表を削除します。

注意： SYS が作成した表を慎重にチェックし、バージョン 8 に必要な（名前に \$ シンボルのある）表を削除しないように気を付けてください。削除してはいけない表の例は次のとおりです。kotad\$,kotmd\$,kottb\$, および kotttd\$。

4. アドバンスド・レプリケーションを使用する場合は、次のステップを実行します。アドバンスド・レプリケーションを使用しない場合は、これらのステップは必要ありません。

- a. SQL*Plus を使用して、遅延トランザクション・キューのレプリケーションを送信します。

```
DECLARE
    rc number;
BEGIN
    rc := dbms_defer_sys.push();
END;
```

- b. 次の SQL 文を発行して、遅延 LOB ビューを削除します。

```
DROP VIEW deflob;
```

参照： アドバンスド・レプリケーション・ステップの完了の詳細は、『Oracle8i レプリケーション・ガイド』を参照してください。

パーティション表からの LOB 列の削除 ダウングレードする前に、パーティション表から LOB 列をすべて削除します。パーティション表内に LOB 列があるかどうか確認するために、次の SQL 文を発行してください。

```
SELECT table_name, lob_name FROM dba_part_lob;
```

パーティション表に LOB データを保存する必要がない場合は、LOB 列を削除してください。ただし、パーティション表に LOB データを保存する必要がある場合には、ALTER TABLE ... EXCHANGE PARTITION コマンドを使用してください。このコマンドは次の例で説明するように、データを非パーティション表に移動させます。

LOB 列を含む既存のパーティション表があり、その LOB 列には、リリース 8.1 からリリース 8.0 にダウングレードする前に保存するデータがすでに含まれていると仮定します。パーティション表には次の定義があります。

```
CREATE TABLE part_lob_table (part_id NUMBER, part_blob_col BLOB)
  PARTITION BY RANGE (part_id) (
    PARTITION p1 VALUES LESS THAN (10) TABLESPACE ts1,
    PARTITION p2 VALUES LESS THAN (20) TABLESPACE ts2)
  TABLESPACE tsx;
```

LOB データを非パーティション表に移動させるために、次のステップを実行してください。

1. 次に類似のコマンドを発行し、LOB 列のある非パーティション表を作成します。

```
CREATE TABLE lob_table_p1 (id NUMBER, blob_col BLOB);
```

```
CREATE TABLE lob_table_p2 (id NUMBER, blob_col BLOB);
```

パーティション LOB 表内の各パーティションに、表を 1 つ作成します。ただし、これらの新しい非パーティション表にはデータを挿入しないでください。

2. EXCHANGE を使用して、PART_BLOB_COL 列からのパーティション表の LOB データと、BLOB_COL 列の非パーティション表の LOB データを交換します。

```
ALTER TABLE part_lob_table  
    EXCHANGE PARTITION p1 WITH TABLE lob_table_p1;
```

```
ALTER TABLE part_lob_table  
    EXCHANGE PARTITION p2 WITH TABLE lob_table_p2;
```

このコマンドは、パーティション表の LOB 列 PART_BLOB_COL からのデータを、各非パーティション表の LOB 列 BLOB_COL に移動させます。

パーティション表のすべての LOB データを非パーティション表に移動させた後、データベースのダウングレードおよびデータの保存ができます。

ユーザー定義データ型

この項では、ユーザー定義データ型に関するリリース 8.1 の機能の使用禁止について説明します。

ユーザー定義データ型のリリース 8.0 形式への変換

リリース 8.1 は、ユーザー定義データ型の新しい形式をサポートします。新しい形式によって、リリース 8.0 で使用される形式に対するパフォーマンスが大幅に向上します。

データベースをリリース 8.0 にダウングレードするとき、ユーザー定義データ型をリリース 8.0 形式に変換する必要があります。ただし、使用しているリリース 8.1 データベースに新しい形式のユーザー定義データ型がない場合、次の変換ステップを行う必要はありません。

8.1 互換性レベルのユーザー定義型を識別するために、次の SQL 文を発行します。

```
SELECT u.name AS USER_NAME, o.name AS TABLE_NAME, c.name AS COLUMN_NAME
FROM sys.user$ u, sys.obj$ o, sys.tab$ t, sys.col$ c, sys.coltype$ ct
WHERE bitand(ct.flags, 128) != 128 AND
      o.obj# = c.obj# and o.obj# = ct.obj# and t.obj# = o.obj# and
      c.intcol# = ct.intcol# AND
      bitand(t.property, 8192) = 0 AND
      u.user# = o.owner# AND
      o.type# = 2 AND
      bitand(c.property, 32) = 0 AND
      (c.type# = 123 OR
       (c.type# = 121 and bitand(c.property, 8) = 0) OR
       (c.type# = 122 and exists
        (SELECT * FROM sys.ntab$ n1, sys.col$ c1, sys.coltype$ ct1
         WHERE n1.obj# = c.obj# AND n1.intcol# = c.intcol# AND
              n1.ntab# = ct1.obj# and bitand(ct1.flags, 128) = 0 AND
              ct1.obj# = c1.obj# and ct1.intcol# = c1.intcol# AND
              bitand(c1.property, 8) = 0)));
```

一覧表示されたユーザー定義データ型をダウングレードするために、次のステップを実行してください。

1. リリース 8.1 のエクスポート・ユーティリティを使用して、8.1 互換性レベルのユーザー定義型を含むリリース 8.1 データベースの部分をエクスポートします。
2. 8.1 互換性レベルのユーザー定義データ型を含むリリース 8.1 データベースの部分を削除します。
3. 12-43 ページの「[データベースのダウングレード](#)」にある手順で、データベースをダウングレードします。
4. リリース 8.1 のインポート・ユーティリティを使用して、エクスポートしたファイルをダウングレードしたデータベースにインポートします。

参照： エクスポート / インポートの方法の詳細は、『Oracle8i ユーティリティ・ガイド』も参照してください。

ユーザー定義オブジェクト識別子のある表の削除

リリース 8.1 はユーザー定義オブジェクト識別子 (OID) をサポートします。Oracle のデフォルト・メカニズムのかわりに、この機能によって、ユーザー自身のオブジェクト識別子を指定できます。ダウングレードする前に、ユーザー定義オブジェクト識別子のある表およびユーザー定義オブジェクト識別子に基づいた REF 列のある表をすべて削除してください。

ユーザー定義オブジェクト識別子のある表を識別するには、次の SQL 文を発行します。

```
SELECT owner, table_name FROM dba_object_tables
       WHERE object_id_type = 'USER-DEFINED';
```

一覧表示されたすべての表を削除します。

ユーザー定義オブジェクト識別子に基づいた REF 列のある表を識別するには、次の SQL 文を発行します。

```
SELECT owner, table_name, column_name FROM dba_refs
       WHERE object_id_type = 'USER-DEFINED';
```

一覧表示されたすべての表を削除します。

リリース 8.1 ネストした表の機能の使用中断

ダウングレード前に、次のリリース 8.1 のネストした表の機能の使用を中断してください。

- コレクション・ロケータ
- 索引構成表にネストした表のデータ
- ネストした表のユーザー指定領域句

これらの機能を 1 つ以上使用する表を識別するには、次の SQL 文を発行します。

```
SELECT owner, parent_table_name
       FROM dba_nested_tables
       WHERE storage_spec LIKE '%USER_SPECIFIED%'
          OR return_type LIKE '%LOCATOR%';
```

一覧表示された表をすべて削除します。

リリース 8.1 VARRAY 機能の使用中断

ダウングレード前に、VARRAY を LOB として格納するための記憶領域パラメータの指定の使用を中断してください。VARRAY を LOB として格納するための記憶領域パラメータを使用する表を識別するために、次の SQL 文を発行してください。

```
SELECT owner, parent_table_name
       FROM dba_varrays
       WHERE storage_spec LIKE '%USER_SPECIFIED%';
```

一覧表示された表をすべて削除します。

相互参照型のリリース 8.0.4 または以前へのダウングレード

相互参照型を使用している場合、リリース 8.0.3.0 または 8.0.4.0 へのダウングレードはサポートされません。相互参照型を使用している場合、ダウングレードに関する 2 つのオプションがあります。

- 相互参照型の削除。12-43 ページの「[データベースのダウングレード](#)」に示すプロシージャを使用して、リリース 8.0.3.0 および 8.0.4.0 へのダウングレードは、サポートされます。
- リリース 8.0.3.0 または 8.0.4.0 ではなく、リリース 8.0.4.1 以降へのダウングレード。このオプションを選択する場合、オラクル社に連絡して 8.0.4.1 以降のリリースを入手してください。

次の SQL 文は、相互参照型の例について説明します。

```
CREATE TYPE manager
/

CREATE TYPE employee AS OBJECT
    (empno NUMBER, ename VARCHAR2(20), mgr REF manager)
/

CREATE OR REPLACE TYPE manager AS OBJECT
    (dept NUMBER, empno REF employee)
/
```

注意： リリース 8.0.3.0 および 8.0.4.0 に存在するが、リリース 8.0.4.0 以降では修正されているバグ #629468 のため、相互参照型の存在はパッケージ STANDARD のコンパイルがループに入り、エラー ORA-01000 最大オープン・カーソルを超過で終了する原因になります。PL/SQL コードのコンパイルにはパッケージ STANDARD が必要で、ダウングレード操作中に実行します。バグがある場合、ダウングレードは正しく行われません。

SQL および PL/SQL

リリース 8.1 は、SQL および PL/SQL への変更および追加を導入します。リリース 8.1 以降で有効なスクリプトまたはストアード・プロシージャの SQL または PL/SQL コードを現在使用している場合、ダウングレードする前にこのコードを削除してください。リリース 8.0 データベースで、このコードをコンパイルまたは実行しようとすると、エラーが発生します。

参照： 新しい SQL および PL/SQL 機能の詳細は、『Oracle8i SQL リファレンス』および『PL/SQL ユーザーズ・ガイドおよびリファレンス』を参照してください。このマニュアルの[付録 E「新しい内部データ型と SQL 関数」](#)も参照してください。

次の項では、SQL および PL/SQL のダウングレード問題について説明します。これらの項で説明のある処置を行うと、SQL スクリプトおよびストアード・プロシージャでのコンパイルおよび実行中のエラーを回避できます。必須ではありませんが、オラクル社ではダウングレード前にこれらの処置を行うことをお勧めします。

C コール指定の削除

ダウングレードする前に、C コール指定として定義されたストアード・プロシージャを削除してください。

起動権利句の削除

SQL コードで起動権利句を使用する場合、ダウングレード前に削除してください。起動権利句には、AUTHID 句および SQL_NAME_RESOLVE 句があります。

PL/SQL のネイティブ動的 SQL の削除

ネイティブ動的 SQL の PL/SQL プログラムは、PL/SQL リリース 8.1 以前のリリースでコンパイル時エラーの原因になることがあります。プログラムを正常にリリース 8.0 にコンパイルするために、ダウングレードする前にネイティブ動的 SQL 構文をすべて削除します。

PL/SQL における一括バインドの削除

一括バインド機能を持つ PL/SQL プログラムは、PL/SQL リリース 8.1 以前のリリースでコンパイル時エラーの原因になることがあります。一括バインド機能は新しい構文および処理方法を定義します。したがって、この機能を持つプログラムを削除するか、可能であれば、スカラー・バインドの使用に変更する必要があります。一括バインド機能を持つ PL/SQL 文には、次のキーワード内の 1 つ以上が含まれます。

- FORALL
- BULK COLLECT INTO
- BULK_ROWCOUNT

PL/SQL における UROWID データ型の削除

PL/SQL コードにおいて、変数として UROWID データ型を使用している場合、ダウングレードする前にこの変数を削除してください。

PL/SQL における NOCOPY パラメータ・パス・モード参照の削除

NOCOPY モードを持つ PL/SQL プログラムは、PL/SQL リリース 8.1 以前のリリースでコンパイル時にエラーの原因になることがあります。プログラムを正常にリリース 8.0 コンパイルするため、ダウングレードする前に NOCOPY 参照をすべて削除します。NOCOPY 参照を削除するとき、変更エイリアシングおよび例外処理方法が許容されているかを確認してください。

Java

リリース 8.0 では Java はサポートされません。ダウングレードする前に、データベースにおける Java オブジェクトをすべて削除する必要があります。utljavarm.sql スクリプトは、Java オブジェクトをすべて削除します。utljavarm.sql スクリプトによって削除された Java オブジェクトを識別するには、次の SQL 文を発行してください。

```
SELECT object_name, owner FROM all_objects WHERE object_type LIKE 'JAVA%';
```

utljavarm.sql スクリプトを実行するために、次のステップを実行します。

1. システム・プロンプトで、\$ORACLE_HOME/rdbms/admin ディレクトリに変更します。
2. SQL*Plus を起動し、SYS 権限を持つユーザーとして接続します。

3. 次のように入力します。

```
SQL> SPOOL utljavaout.log
SQL> @utljavarm.sql
SQL> SPOOL OFF
```

スプール・ファイルをチェックして、文が正しく実行されたか確認します。

アドバンスト・キューイング (AQ)

次の作業を実行し、キュー表におけるリリース 8.1AQ 機能を使用禁止にしてください。

1. すべての非永続キューおよびキュー表の削除
2. リリース 8.1 互換性キュー表の識別
3. リリース 8.1 互換性キュー表から非互換性の削除
4. キュー表のダウングレード
5. キュー表のエクスポートおよびダウングレード後のインポート
6. リリース 8.0.3 にダウングレードする場合の伝播の削除

これらのステップについては、次の項で説明します。

参照： この項で説明する処置の完了の詳細は、『Oracle8i アプリケーション開発者ガイド アドバンスト・キューイング』を参照してください。

作業 1: すべての非永続キューおよびキュー表の削除

非永続キューを使用する場合、それを持つキューおよびキュー表を削除する必要があります。非永続キューを持つスキーマ（ユーザー）には、スキーマの非永続キューすべてを含むキュー表が 1 つまたは 2 つある可能性があります。次の SQL 文を入力して、非永続キューのあるキュー表の存在をチェックします。

```
SELECT owner, queue_table FROM dba_queue_tables
WHERE queue_table = 'AQ$_MEM_MC' OR queue_table = 'AQ$_MEM_SC';
```

SQL 文が戻したすべてのキュー表に対して、DBMS_AQADM.DRO_QUEUE_TABLE プロシージャ（フォース・パラメータを TRUE に設定）で、非永続キューおよび相応するキュー表を削除します。これはコマンドの一例です。

```
EXECUTE dbms_aqadm.drop_queue_table (  
    queue_table => 'SCOTT.AQ$_MEM_MC',  
    force => TRUE);
```

作業 2: リリース 8.1 互換性キュー表の識別

キュー表のうちいずれかがリリース 8.1 互換性である場合、その表のダウングレードが必要です。次の SQL 文を入力して、キュー表の互換性をチェックしてください。

```
SELECT owner, queue_table, compatible FROM dba_queue_tables  
WHERE compatible LIKE '8.1%';
```

一覧表示されたキュー表はリリース 8.1 互換性で、リリース 8.0 と非互換のため、ダウングレード前に削除が必要です。リリース 8.1 互換性のキュー表のリストを印刷してください。12-31 ページの「[作業 4: キュー表のダウングレード](#)」に達したら、これらのキュー表をダウングレードしてください。

注意： この問合せでは、リリース 8.1 互換性だったが、その後リリース 8.0 互換性にダウングレードしたキュー表は表示されません。ただし、そのようなキュー表がある場合、リリース 8.0 にダウングレードする前に削除する必要があります。これらのキュー表については、12-31 ページの「[作業 5: キュー表のエクスポートおよびダウングレード後のインポート](#)」の説明に従ってください。

作業 3: リリース 8.1 互換性キュー表から非互換性の削除

多くの非互換性がキュー表内にある可能性があります。リリース 8.1 で使用可能だがリリース 8.0 では使用できない特定の機能を使用すると、これらの非互換性が発生します。

次の項では、リリース 8.1 機能に基づいたこれら非互換性の削除について説明します。

ルールベース・サブスクリプション AQ\$queue_table_name_r ビューで、リリース 8.1 のルールベース・サブスクリプション機能を持つキューを識別してください。作業 2 で一覧表示されたリリース 8.1 のキュー表すべてをチェックをします。たとえば、キューが QTABLE3 という名前であれば、次の SQL 文を発行してルールベース・サブスクライバがあるかどうかチェックします。

```
SELECT * FROM aq$htable3_r;
```

注意： SQL 文を発行し、エラー「ORA-04063: 'aq\$queue_table_name_r' にエラーがあります。」が表示された場合、キュー表にはルール・サブスクライバのキューはありません。

ルールベース・サブスクライバを削除するか、または DBMS_AQADM.ALTER_SUSCRIBER プロシージャで、各ルールベース・サブスクライバのルールを NULL に変更してください。たとえば、[表 12-1](#) に表示される値を持つ AQ.MSG_QUEUE というキューに対するサブスクライバがあると仮定します。

表 12-1 サンプル・サブスクライバ値

パラメータ	値
name	SUBSCRIBER1
address	AQ2.MSG_QUEUE2@LONDON
protocol	NULL
rule	'PRIORITY = 1'

このサブスクライバのルールを NULL に変更する方法は 2 つあります。PLS/SQL ブロックを使用する方法と SQL 文を使う方法です。[例 12-1](#) では PL/SQL ブロックの、[例 12-2](#) では SQL 文の説明をしています。

例 12-1 ルールを NULL に変更する PL/SQL ブロック

```
DECLARE
    subscriber sys.aq$_agent;
BEGIN
    subscriber := sys.aq$_agent ('SUBSCRIBER1', 'aq2.msg_queue2@london', null);
    dbms_aqadm.alter_subscriber
        (queue_name => 'aq.msg_queue',
         subscriber => subscriber,
         rule => NULL);
END;
```

例 12-2 ルールを NULL に変更する SQL 文

```
EXECUTE dbms_aqadm.alter_subscriber (  
    'aq.msg_queue',  
    sys.aq$_agent ('SUBSCRIBER1', 'aq2.msg_queue2@london', NULL),  
    NULL);
```

参照： DBMS_AQUADM.ALTER_SUBSCRIBER プロシージャの詳細は、『Oracle8i アプリケーション開発者ガイド アドバンスド・キューイング』を参照してください。

オブジェクト・レベルおよびシステム・レベル権限 DBMS_AQADAM パッケージにある次のプロシージャのどれかを使用した場合には、オブジェクト・レベルおよびシステム・レベル権限を使用していることになります。

- GRANT_ACCESS_PRIVILEGES
- REVOKE_ACCESS_PRIVILEGES
- GRANT_SYSTEM_PRIVILEGES
- REVOKE_SYSTEM_PRIVILEGES

これらのプロシージャのいずれかを使用した場合、ダウングレードする前に、オブジェクト・レベルおよびシステム・レベル権限をすべて取り消す必要があります。

次の SQL 文を発行して、オブジェクト・レベル権限を識別してください。

```
SELECT owner, table_name, privilege  
FROM dba_tab_privs WHERE privilege LIKE '%QUEUE%';
```

DBMS_AQADM.REVOKE_ACCESS_PRIVILEGES プロシージャを使用して、PRIVILEGE 列に ENQUEUE または DEQUEUE を持つ各権限を取り消してください。

次の SQL 文を発行して、オブジェクト・レベル権限を識別してください。

```
SELECT * FROM dba_sys_privs WHERE privilege LIKE '%QUEUE%';
```

DBMS_AQUADM.REVOKE_SYSTEM_PRIVILEGE プロシージャを使用して、PRIVILEGE 列に 1 つでも次の型を持つ各権限を取り消してください。

- MANAGE ANY QUEUE
- ENQUEUE ANY QUEUE
- DEQUEUE ANY QUEUE

作業 4: キュー表のダウングレード

次のステップを完了して、リリース 8.0 と非互換だった各キュー表をダウングレードしてください。12-28 ページの「[作業 2: リリース 8.1 互換性キュー表の識別](#)」で、非互換性キュー表の一覧表示を印刷したはずです。

1. ダウングレードする前に、DBMS_AQADM.DISABLE_PROPAGATION_SCHEDULE を使用して、キュー表における全キューの伝播スケジュールを使用禁止にします。
2. 非互換性キュー表をダウングレードして、リリース 8.0 互換性に戻します。

DBMS_AQADM.MIGRATE_QUEUE_TABLE プロシージャを実行し、COMPATIBLE 設定に 8.0 と指定して、キュー表をダウングレードします。次の例は、このプロシージャの実行について説明します。

```
EXECUTE dbms_aqadm.migrate_queue_table (  
    queue_table => 'sys.tkaqqtdef',  
    compatible => '8.0');
```

注意: キュー表では、キュー表の所有者のみ DBMS_AQADM.MIGRATE_QUEUE_TABLE プロシージャを実行できます。

作業 5: キュー表のエクスポートおよびダウングレード後のインポート

次のステップを完了して、非互換性キュー表をエクスポートし、ダウングレードしてインポートし直してください。12-28 ページの「[作業 2: リリース 8.1 互換性キュー表の識別](#)」で、非互換性キュー表の一覧表示を印刷したはずです。

1. リリース 8.1 エクスポート・ユーティリティを使用して、非互換性キュー表をすべてエクスポートします。
2. エクスポートの完了したキュー表をすべて削除します。
3. この項の残りの指示に従って、データベースをダウングレードします。
4. データベースをリリース 8.0 にダウングレードしたら、リリース 8.1 インポート・ユーティリティを使用してエクスポート・ファイルをデータベースにインポートします。

残りの非互換性キュー表にも同じ作業を繰り返してください。

参照: キュー表のエクスポート / インポートの詳細は、『Oracle8i アプリケーション開発者ガイド アドバンスド・キューイング』を参照してください。

作業 6: リリース 8.0.3 にダウングレードする場合の伝播の削除

アドバンスド・キューイング・オプションにおいてメッセージ伝播を使用しており、リリース 8.0.3 にダウングレードする場合は、伝播を削除してください。メッセージ伝播を使用していない場合、またはリリース 8.0.4 以降にダウングレードする場合は、この作業は必要ありません。

次のステップを完了して、伝播を削除してください。

1. 次の SQL 文を発行して、複数コンシューマ・キューを識別してください。

```
SELECT owner, queue_table FROM dba_queue_tables
WHERE recipients = 'MULTIPLE';
```

この問合せの結果を保存してください。複数コンシューマ・キューのみ伝播をサポートします。この問合せが行を戻さない場合、伝播は使用されておらず、このプロシージャの残りのステップは必要ありません。

2. ステップ 1 の問合せによって一覧表示されたキュー表に次の SQL 文を実行して、複数コンシューマ・キューのどれかが伝播機能を利用しているかどうか確認してください。

次の SQL 文では、例として SCOTT.OTABLE1 キュー表を使用します。

```
SELECT unique(q_name) FROM scott.qtable1 a
WHERE EXISTS (SELECT consumer
FROM the (SELECT cast(history as sys.aq$_dequeue_history_t)
FROM scott.qtable1 b
WHERE a.msgid = b.msgid)
WHERE consumer like 'AQ$_%');
```

各キュー表において、SQL 文が戻したキューをすべて削除する必要があります。

注意： 使用しているすべてのキュー表を一覧表示させるには、DBA_QUEUE_TABLES 静的データ・ディクショナリ・ビューを問い合わせます。

3. DBA_QUEUE_SCHEDULES の静的データ・ディクショナリ・ビューをチェックして、キューに対して伝播スケジュールがあるか確認してください。DBMS_AQADM.UNSCHEDULE_PROPAGATION プロシージャを使用して、ビュー内で選択した各キューの伝播スケジュールを解除する必要があります。
4. リモート・サブスクライバをすべて排除します。リモート・サブスクライバには、指定されたアドレス・フィールドまたはプロトコル・フィールドがあります。DBMS_AQADM.QUEUE_SUBSCRIBERS プロシージャを実行して、非 NULL アドレス・フィールドまたは非 NULL プロトコル・フィールドを持つサブスクライバをチェックしてください。これらのサブスクライバをすべて削除します。

プロシージャおよびパッケージ

この項では、プロシージャおよびパッケージに関するリリース 8.1 機能の使用禁止について説明します。

SET_SESSION_LONG_LONGOPS プロシージャに対する構文変更

リリース 8.1 は DBMS_APPLICATION_INFO.SET_SESSION_LONGOPS プロシージャに変更をもたらします。アプリケーションがこのプロシージャを使用し、リリース 8.1 構文に準拠するように変更された場合、ダウングレードするリリースに準拠するようにアプリケーションを変更してください。構文の詳細は、ダウングレードするリリース内の dbmsapin.sql ファイルを参照してください。

UTL_REF パッケージ

リリース 8.0.3 にダウングレードする場合は、UTL_REF パッケージの使用を中断してください。このパッケージはリリース 8.0.3 では使用できません。

リリース 8.0.4 にダウングレードする場合、UTL_REF パッケージはダウングレード・プロセス中に自動的に削除されます。パッケージが削除されるのは、UTL_REF パッケージが標準リリース 8.0.4 インストールの一部ではないためです。このパッケージを続けて使用するには、リリース 8.0.4 にダウングレードした後、パッケージを手動で再インストールする必要があります。

注意： リリース 8.0.5 以降にダウングレードする場合、UTL_REF パッケージへの処置は必要ありません。このパッケージはリリース 8.0.5 以降の標準インストールの一部であり、ダウングレード・プロセス中に自動的に保存されます。

DBMS_REPAIR パッケージ

リリース 8.1 は DBMS_REPAIR パッケージをサポートします。ダウングレードする前に、全オブジェクトのスキップ・コラプト属性が使用禁止になっているかどうかを確認してください。

次の SQL 文を発行して、スキップ・コラプト属性が使用可能になっているオブジェクトを識別してください。

```
SELECT owner, table_name from dba_tables where skip_corrupt = 'ENABLED';
```

選択した各表のスキップ・コラプト属性を消去してください。たとえば、SCOTT が所有する TB_5 という表に、次のように入力してください。

```
EXECUTE DBMS_REPAIR.SKIP_CORRUPT_BLOCKS (schema_name => 'SCOTT',
object_name => 'TB_5', flags => DBMS_REPAIR.NOSKIP_FLAG);
```

制約およびトリガー

この項では、制約およびトリガーに関する非互換性の削除について説明します。

DISABLE VALIDATE 制約の使用の中断

リリース 8.1 は DISABLE VALIDATE 制約状態をサポートします。ダウングレードする前に、DISABLE VALIDATE 制約をすべて削除するか無効にする必要があります。

次の SQL 文を発行して、DISABLE VALIDATE 制約を識別してください。

```
SELECT constraint_name, status, validated
FROM dba_constraints
WHERE status = 'DISABLED'
AND validated = 'VALIDATED';
```

ALTER TABLE コマンドの DROP 句を使用して、一覧表示されたすべての制約を削除します。または、ALTER TABLE コマンドの DISABLE 句を使用して、一覧表示されたすべての制約を無効にします。

ネストした表のビュー列におけるトリガーの削除

リリース 8.1 はネストした表のビュー列におけるトリガーの作成をサポートします。ダウングレードする前に、これらのトリガーをすべて削除する必要があります。

次の SQL 文を入力して、ビュー列におけるネストした表のトリガーを識別してください。

```
SELECT trigger_name, table_name, column_name
FROM dba_triggers
WHERE column_name IS NOT NULL;
```

TRIGGER コマンドで、一覧表示されたトリガーをすべて削除します。

非互換トリガーの削除

リリース 8.1 ではトリガーを拡張し、データベースのイベント発行をサポートします。ダウングレードする前に、リリース 8.0 と互換性のないトリガーをすべて削除する必要があります。

次の SQL 文を発行して、ダウングレード・プロセス中に削除しなければならないトリガーを識別してください。

```
SELECT trigger_name, base_object_type, action_type
FROM dba_triggers
WHERE base_object_type
IN ('DATABASE', 'SCHEMA')
OR action_type = 'CALL';
```

SCHEMA および DATABASE におけるトリガーを、リリース 8.0 で互換にすることはできません。これらのトリガーは削除してください。ただし、CALL トリガーについてはダウングレード・グレード中に保存可能です。BEGIN ... END ブロックで CALL 文をラップして、CALL トリガーをリリース 8.0 で互換にしてください。

この項では、Oracle オプティマイザに関する非互換性の削除について説明します。

拡張オプティマイザ

リリース 8.1 は、拡張オプティマイザをサポートします。ダウングレードする前に、結合をすべて削除して拡張オプティマイザの使用を中断します。次の SQL 文を入力して、結合を識別します。

```
SELECT object_owner, object_name, column_name, object_type
FROM dba_associations;
```

一覧表示された各結合に、FORCE オプションを使用して DISASSOCIATE STATISTICS コマンドを実行してください。たとえば、先行の SQL 文発行のときに次の出力を受信すると仮定します。

OBJECT_OWNER	OBJECT_NAME	COLUMN_NAME	OBJECT_TYPE
-----	-----	-----	-----
SYS	TYPE1		TYPE
TKOQEX	TKOQ_TAB1	A	COLUMN

2 rows selected.

一覧表示されたオブジェクト型に対応する適切な DISASSOCIATE STATISTICS 文を発行してください。オブジェクト型が TYPE および COLUMN の場合は、次の SQL 文を発行して一覧表示された結合を削除してください。

```
DISASSOCIATE STATISTICS FROM types sys.type1 FORCE;
```

```
DISASSOCIATE STATISTICS FROM columns tkoqex.tkoq_tab1.a FORCE;
```

参照:DISASSOCIATE STATISTICS コマンドの詳細は、『Oracle8i SQL リファレンス』を参照してください。

オブティマイザ計画の安定性

リリース 8.1 はオブティマイザ計画の安定性をサポートします。この機能によって、CREATE OUTLINE 文を使用してストアド・アウトラインを作成できます。リリース 8.0 では、ストアド・アウトラインをサポートしません。

次の SQL 文を発行して、ストアド・アウトラインを識別してください。

```
SELECT owner, name FROM dba_outlines;
```

この SQL 文によって一覧表示されたアウトラインを削除してください。

セキュリティ

この項では、データベースのセキュリティに関する非互換性の削除について説明します。

全アプリケーション・コンテキストの削除

リリース 8.1 の新しい機能では、アプリケーション・コンテキストを指定できます。ダウングレードする前に、アプリケーション・コンテキストをすべて削除します。次の SQL 文を発行して、アプリケーション・コンテキストを識別してください。

```
SELECT * FROM dba_context;
```

DROP CONTEXT 文を使用して、この SQL 文によって一覧表示されたアプリケーション・コンテキストをすべて削除します。

全ユーザー定義セキュリティ・ポリシーの削除

ファイン・グレイン・アクセス・コントロールとは、リリース 8.1 の新機能で、ユーザー定義セキュリティ・ポリシーの作成が可能になります。ダウングレードする前に、ユーザー定義セキュリティ・ポリシーをすべて削除します。

次の SQL 文を発行して、ユーザー定義セキュリティ・ポリシーを識別してください。

```
SELECT object_owner, object_name, policy_name
       FROM dba_policies;
```

DBMS_RLS.DROP_POLICY プロシージャを使用して、この SQL 文によって一覧表示されたポリシーをすべて削除します。

プロキシ・ユーザーの CONNECT THROUGH 権限の取消し

リリース 8.1n-tier 認証および認可機能は、リリース 8.0 では使用できません。したがって、プロキシ・ユーザーに CONNECT THROUGH 権限がある場合、これらの権限を取り消す必要があります。

次の SQL 文を発行して、プロキシ・ユーザーを一覧表示させます。

```
SELECT * FROM proxy_users;
```

ALTER USER ... REVOKE CONNECT THROUGH 文を発行して、CONNECT THROUGH 権限を取り消してください。たとえば、次の文はプロキシ・ユーザー APPSERVER1 の権限を取り消し、ユーザー JANE として接続します。

```
ALTER USER jane REVOKE CONNECT THROUGH appserver1;
```

データベース・バックアップおよび回復

この項では、ダウングレードでバックアップを回復させる責任について説明します。

Oracle メディア管理 API およびプロキシ・コピー

Oracle メディア管理 API のバージョン 2 はプロキシ・コピー機能をサポートしますが、データベースをリリース 8.0 にダウングレードするとサポートしなくなります。したがって、リリース 8.0 を使用して、プロキシ・コピーをサポートするバージョン 2 ソフトウェア・バックアップ・テープ (SBT)・レイヤーを使用して作成されたリリース 8.1 プロキシ・バックアップは復元できません。

リリース 8.0 にダウングレードしたデータベースをリリース 8.1 データベースに復元する必要がある場合、ダウングレードする前に、プロキシ・コピーをオフにしてバックアップを作成します。これはプロキシ・コピーをオフにすることによって、リリース 8.0 のバックアップを復元できるためです。使用しているメディア・マネージャが Oracle メディア管理 API バージョン 2 のみサポートする場合、リリース 8.0 で使用するバージョン 1.1 SBT レイヤーの取得も必要です。

旧アーカイブ・ログの宛先パラメータへの変更

リリース 8.1 の新しいアーカイブ・ログの宛先パラメータ (LOG_ARCHIVE_DEST n および LOG_ARCHIVE_DEST_STATE n) を使用した場合、ダウングレードする前に、それを旧アーカイブ・ログの宛先パラメータ (LOG_ARCHIVE_DEST および LOG_ARCHIVE_DUPLEX_DEST) に戻してください。

参照： 詳細は、B-14 ページの「[旧アーカイブ・ログの宛先のパラメータへの変更](#)」を参照してください。

分散データベース

この項では、分散データベースに関する非互換性の削除について説明します。

アドバンスド・レプリケーション環境のダウングレードに対する準備

アドバンスド・レプリケーションを使用している場合、次の項にある処置を実行して、ダウングレードするためのアドバンスド・レプリケーション環境を準備してください。

参照： これらの処置については、『Oracle8i レプリケーション・ガイド』を参照してください。

アドバンスド・レプリケーション環境における非互換性の削除 レプリケーション環境のデータベースがリリース 8.1 プロトコルを使用する RPC をダウングレード中のデータベースへ送る場合は、ダウングレードの前にすべての遅延 RPC を更新するかまたは削除してください。次に示すコールのいずれかで GENERATE_80_COMPATIBLE に一度でも FALSE が設定されていた場合は、データベースの RPC はリリース 8.1 のプロトコルを使用します。

- GENERATE_REPLICATION_SUPPORT (すべてのマスター・サイトに影響)
- CREATE_SNAPSHOT_REOBJECT (ローカルのスナップショット・サイトに影響)
- CREATE_SNAPSHOT_SUPPORT (ローカルのスナップショット・サイトに影響)

遅延 RPC を更新するかまたは削除するには、次のステップを実行してください。

1. ダウングレード元のデータベースが 1 つ以上のオブジェクト・グループに対するマスター・サイトで、なおかつマスター定義サイトがリリース 8.1 以降のリリースにある場合、次のステップを実行します。
 - a. 各オブジェクトグループを停止します。
 - b. 各マスター・サイトを各オブジェクトグループの top flavor に変換します。
 - c. GENERATE_REPLICATION_SUPPORT コール中の GENERATE_80_COMPATIBLE を TRUE に設定して、レプリケート表用にレプリケーション・サポートを再生成します。
 - d. 停止中のレプリケーション・オブジェクトグループのマスター・アクティビティを再開します。
 - e. 各マスター・サイトのスナップショット・サイトをそれぞれオブジェクト・グループの top flavor に変換します。
2. ダウングレード元のデータベースが 1 つ以上のスナップショット・サイト（リリース 8.1 以降のリリースにあるスナップショット・サイト）のマスター・サイトの場合、次のステップを実行します。
 - a. GENERATE_SNAPSHOT_SUPPORT コール中の GENERATE_80_COMPATIBLE を TRUE に設定して、このサイトにあるすべてのスナップショット用にサポートを再生成します。
 - b. リリース 8.1 以降のリリースからキューを送信します。
3. ダウングレード元のデータベースに対するエラー・キューのエラーを更新または削除するために、DBMS_DEFER_SYS.EXECUTE_ERROR プロシージャまたは DBMS_DEFER_SYS.DELETE_ERROR プロシージャを使用します。

データベース・オブジェクト・グループが Top Flavor にあるかどうかの確認 ダウングレード元のデータベースのオブジェクト・グループは、ダウングレードする前に top flavor である必要があります。top flavor がないオブジェクト・グループを見つけるためには、次の SQL 文を発行します。

```
SELECT gname, fname FROM dba_repgroup
WHERE fname IS NOT NULL;
```

もしデータベースのオブジェクト・グループが表示されたら、ダウングレードする前にそのオブジェクト・グループの flavor を top flavor に変更します。オブジェクト・グループの FNAME が NULL になっているときはそのオブジェクト・グループは表示されません。この場合はオブジェクト・グループはすでに top flavor にあり、操作は必要ありません。

一時更新可能スナップショット・ログの削除 次のような SQL 文を発行して、一時更新可能スナップショット・ログがあるかどうか確認します。

```
SELECT owner, table_name FROM dba_tables
WHERE temporary='Y' AND
table_name LIKE 'RUPD$%';
```

行が戻された場合には、一時更新可能スナップショット・ログがデータベースに存在します。一時更新可能スナップショット・ログを削除するために、次の PL/SQL ブロックを実行します。

```
DECLARE
    sql_cur  BINARY_INTEGER;
    dummy    BINARY_INTEGER;
    new_flag BINARY_INTEGER;

    CURSOR mv_logs IS
        SELECT '''||mowner||'."'||temp_log||''' temp_log,
               flag, mowner, master
        FROM mlog$ m
        WHERE temp_log IS NOT NULL
        FOR UPDATE;
BEGIN
    sql_cur := dbms_sql.open_cursor;
    FOR alog IN mv_logs LOOP
        new_flag := alog.flag;
        IF dbms_ijklbit(new_flag, 64) THEN ---KKZLOGTUPS
            new_flag := new_flag - 64;
        END IF;

        BEGIN
            dbms_sql.parse(sql_cur, 'DROP TABLE ' || alog.temp_log, dbms_sql.v7);
            dummy := dbms_sql.execute(sql_cur);

            UPDATE mlog$ m
            SET flag = new_flag, temp_log = NULL
            WHERE m.mowner = alog.mowner AND m.master = alog.master;
        EXCEPTION WHEN others THEN
            NULL; --- Ignore the error
        END;
    END LOOP;
    dbms_sql.close_cursor(sql_cur);
    COMMIT;
```



```
EXCEPTION WHEN others THEN
  IF dbms_sql.is_open(sql_cur) THEN
    dbms_sql.close_cursor(sql_cur);
  END IF;
  RAISE;
END;
/
```

スナップショットにおける非互換性の識別 スナップショットとマテリアライズド・ビューは同義語です。

参照： マテリアライズド・ビューにおける非互換性の識別の詳細は、12-9 ページの「[ダウングレード中に変更または削除されるマテリアライズド・ビューの識別](#)」を参照してください。

Net8

この項では、Net8 に関する非互換性の削除について説明します。

サービス命名使用の中断

Net8 のサービス命名はリリース 8.1 ではサポートされていますが、リリース 8.0 ではサポートされていません。サービス命名を使用するのをやめるには、次の操作を行います。

- `initSID.ora` ファイルの `SERVICE_NAMES` パラメータおよび `INSTANCE_NAME` パラメータを削除します。
- `tnsnames.ora` ファイルの `SERVICE_NAME` パラメータを削除します。
- 接続記述子の `SID` パラメータを使用します。
- リスナーによって提供されるインスタンスの情報を手動で設定するためには、`listener.ora` ファイルの `SID_LIST` パラメータを使用します。

参照： 接続記述子の `SID` パラメータの使用については、ダウングレードするリリースの Net8 または SQL*Net ドキュメントで該当するものを参照してください。

データベース互換性のリセット

ダウングレード先のリリースとの非互換性をすべて削除した後、次のステップを実行して、データベースの互換性レベルを（以前のリリースに対する互換性に）リセットします。

1. ダウングレード先のリリース以降のリリースに追加された初期化パラメータを使用している場合は、そのパラメータを `initSID.ora` ファイルから削除してください。

参照： 各リリースに追加されたパラメータの表示については、[付録 B「初期化パラメータの変更」](#)を参照してください。

2. SQL*Plus を起動し、ユーザーとして SYSDBA 権限に接続します。
3. STARTUP を使用してデータベースを起動します。
4. ALTER DATABASE RESET COMPATIBILITY を実行します。

```
SQL> ALTER DATABASE RESET COMPATIBILITY;
```

参照： 詳細は、8-7 ページの「[ALTER DATABASE RESET COMPATIBILITY について](#)」を参照してください。

5. SHUTDOWN IMMEDIATE を実行します。

```
SQL> SHUTDOWN IMMEDIATE
```

注意： ステップを完了したら、ダウングレード先のリリース以降に設定された COMPATIBLE を使用してデータベースをオープンしないでください。

6. `initSID.ora` ファイルにダウングレード先のリリースに合致する COMPATIBLE パラメータを設定してください。

たとえば、リリース 8.0.5 にダウングレードする場合には、COMPATIBLE パラメータを次のように設定してください。

```
COMPATIBLE=8.0.5
```

7. データベースをオープンして、COMPATIBLE パラメータで指定したリリースと互換性のあることを確認してください。

非互換性がまだ存在する場合、データベースをオープンできません。その場合は、COMPATIBLE パラメータを高いリリース、たとえば 8.0.1 に再設定してください。そして、非互換性を削除し、データベースの互換性を再び再設定してください。12-43 ページの「[データベースのダウングレード](#)」で説明するダウングレード・プロセスを続行する前に、ダウングレード先のデータベースの非互換性をすべて削除する必要があります。

参照： 非互換性の削除の詳細は、12-2 ページの「[非互換性の削除](#)」を参照してください。

データベースのダウングレード

この項のダウングレードのステップを行う前に、データベースがダウングレード先のリリースに互換性のあることを確認してください。まだ非互換性を削除していない場合は、12-2 ページの「[非互換性の削除](#)」を参照してください。

リリース 8.1 データベースをそれ以前のリリースへダウングレードするときは、次のステップを実行してください。

1. 次のファイルを \$ORACLE_HOME/rdbms/admin ディレクトリから、システムの一時ディレクトリのような Oracle ホーム以外のディレクトリへコピーします。

- catlg803.sql
- utlip.sql
- utlrp.sql

これらのファイルの新しい場所を覚えておいてください。ダウングレード・プロセスにおいて、これらのファイルが必要になることがあります。

2. システム・プロンプトで、\$ORACLE_HOME/rdbms/admin ディレクトリに変更します。

3. NOLOG オプションを使用して SQL*Plus を起動します。

```
sqlplus /nolog
```

注意： ダウングレード手順のこの部分では、SQL*Plus を使用する必要があります。Server Manager は使用しないでください。

4. データベース・インスタンスへ接続します。

```
SQL> CONNECT INTERNAL
```

5. データベースが停止した場合、STARTUP を使用してデータベースを起動します。

```
SQL> STARTUP
```

PFILE オプションを使用して、使用する `inittsid.ora` ファイルの場所を指定できます。

6. 成功したかどうかを後で確認するために、結果をログ・ファイルにスプールするようにシステムを設定します。

```
SQL> SPOOL catoutd.log
```

スクリーンで実行するスクリプトの出力を見たい場合、SET ECHO ON 文も発行できます。

```
SQL> SET ECHO ON
```

7. `old_release` がダウングレード先のリリースを参照する場合、`dold_release.sql` を実行します。正しいスクリプトを選択するには、[表 12-2](#) を参照してください。各スクリプトによって、「ダウングレード先」"Downgrading To" 列で指定したリリースに直接ダウングレードできます。

スクリプトを実行するには、次のように入力します。

```
SQL> @dold_release.sql
```

表 12-2 ダウングレード・スクリプト

ダウングレード先	実行するスクリプト
8.1.4	d0801040.sql
8.1.3	d0801030.sql
8.1.2	未サポート
8.1.1	未サポート
8.0.5	d0800050.sql
8.0.4S	未サポート
8.0.4	d0800040.sql
8.0.3	d0800030.sql
8.0.2	未サポート
8.0.1	未サポート

注意： ダウングレード先のリリースが[表 12-2](#)に含まれていない場合、実行する正しいダウングレード・スクリプトについては、新しいインストールの README ファイルを参照してください。

スクリプトの実行についての注意は、次のとおりです。

- ダウングレード元のリリースに付属するスクリプトのバージョンを使用してください。
- ダウングレード元のリリースの環境にあるスクリプトを実行してください。

- ダウングレードが複数のリリースにわたる場合でも、実行するスクリプトは1つです。たとえば、リリース 8.0.3 にダウングレードする場合、d0800030.sql のみ実行します。
- 相互参照型を使用している場合、リリース 8.0.3 および 8.0.4.0 へのダウングレードはサポートされません。詳細は、12-24 ページの「[相互参照型のリリース 8.0.4 または以前へのダウングレード](#)」を参照してください。

スクリプトまたは残りのステップ内のスクリプトのいずれか 1 つを実行中に問題が発生した場合、問題の原因を訂正し、スクリプトを戻します。この章に説明のあるスクリプトを、必要に応じて何度でも戻せます。

参照： スクリプト実行時に探すエラーの型の詳細は、1-3 ページの「[スクリプトの実行](#)」を参照してください。

8. SHUTDOWN IMMEDIATE を実行し、SQL*Plus を終了します。

```
SQL> SHUTDOWN IMMEDIATE
SQL> EXIT
```

Oracle Parallel Server を使用している場合、インスタンスをすべて停止します。

9. 次の環境変数がダウングレード先のリリースのディレクトリを指定するよう変更してください。

- ORACLE_HOME
- PATH
- ORA_NLS
- ORACLE_BASE
- LD_LIBRARY_PATH (UNIX の場合)
- ORACLE_PATH (設定されている場合)

注意： これらの環境変数には、ご使用のオペレーティング・システムに適用されないものもあります。

参照： ご使用のオペレーティング・システムにおけるその他の重要な環境変数の詳細は、オペレーティング・システム固有の Oracle8i のインストール・ドキュメントを参照してください。

10. ダウングレード先のリリースのインストール・メディアを使用して、そのリリースをインストールしてください。

たとえば、リリース 8.0.5 にダウングレードする場合、リリース 8.0.5 のインストール・メディアを使用して Oracle のリリース 8.0.5 配布をインストールします。

また、リリース 8.0 にダウングレードする場合は、リリース 8.1 とは別の Oracle ホームのリリース 8.0 ソフトウェアをインストールする必要があります。ただし、前の 8.1 リリースにダウングレードする場合は、この制限は適用されません。新しいリリースを同じ Oracle ホームにインストールしたければ、それは可能です。

注意： インストールは、オペレーティング・システム固有のものです。インストールの指示については、オペレーティング・システム固有のインストール・ドキュメントおよび使用しているオペレーティング・システムの README ファイルを参照してください。

11. 別の Oracle ホームを使用している場合に、ダウングレード元の Oracle ホーム内に `initSID.ora` ファイルが常駐する場合、その `initSID.ora` ファイルを Oracle ホーム以外の場所にコピーします。リリース 8.1 では、`initSID.ora` ファイルのデフォルトの場所は `$ORACLE_BASE/admin/SID/pfile` で、ここでは `SID` が Oracle インスタンス ID です。前回までのリリースでは、デフォルト場所は UNIX では `$ORACLE_HOME/dbs`、Windows NT では `$ORACLE_HOME\database` でした。`initSID.ora` ファイルは任意の場所に格納できますが、ダウングレード元のリリースの Oracle ホームには常駐することができません。

`initSID.ora` ファイルに `IFILE` (インクルード・ファイル) エントリがあり、`IFILE` エントリで指定されたファイルがダウングレード元のデータベースの Oracle ホーム内に常駐する場合、`IFILE` エントリが指定したファイルを、Oracle ホーム以外の場所にコピーしてください。`IFILE` エントリ内で指定されたファイルには、追加の初期化パラメータがあります。このファイルをコピーしたら、`initSID.ora` ファイルを編集して新しい場所を指定します。

注意： Oracle Parallel Server では、すべてのノードにこのステップを実行してください。また、`PARALLEL_SERVER` 初期化パラメータを `FALSE` に設定してください。ダウングレード操作終了後、`TRUE` に戻せます。

12. 次のファイルを \$ORACLE_HOME/rdbms/admin ディレクトリにコピーします。

- catlg803.sql
- utlip.sql
- utlrp.sql

ステップ 1 で、これらのファイルを Oracle ホーム以外のディレクトリにコピーしています。

13. システム・プロンプトで、\$ORACLE_HOME/rdbms/admin ディレクトリに移動します。

14. Server Manager を起動し、CONNECT INTERNAL を実行します。

```
SVRMGR> CONNECT INTERNAL
```

注意： ダウングレード手順の残りでは、SQL*Plus を使用する必要があります。SQL*Plus は使用しないでください。

15. STARTUP を実行します。

```
SVRMGR> STARTUP
```

PFILE オプションを使用して、使用する `inittsid.ora` ファイルの場所を指定できます。

16. 次の条件のどちらか 1 つが当てはまる場合、このステップを実行します。

- 使用するインストレーションが文字サイズ (32 ビットと 64 ビットの切換え) の変更を必要とする場合。
- リリース 8.0 にダウングレードする場合。

前述の条件に当てはまらない場合、このステップはオプションです (必要ないのに実行すると無効になります)。

utlip.sql を実行します。

```
SVRMGR> @utlip.sql
```

UTLIP.SQL スクリプトは一定のディクショナリ表を変更し、データベースが必要とする形式で次の再コンパイルが起こるように既存の PL/SQL モジュールをすべて無効にします。また PL/SQL コンパイルに必要な STANDARD および DBMS_STANDARD パッケージを、再ロードします。

17. ダウングレード先のリリースによって、catalog.sql スクリプトまたは catlg803.sql スクリプトのどちらかを実行します。これらのスクリプトの両方は実行できません。

リリース 8.0.4 以降のリリースにダウングレードする場合、catalog.sql を実行します。

```
SVRMGR> @catalog.sql
```

リリース 8.0.3 にダウングレードする場合、catlg803.sql を実行します。

```
SVRMGR> @catlg803.sql
```

注意： リリース 8.0.3 に存在し、リリース 8.0.4 以降で修正されたバグ #571546 があるため、リリース 8.0.3 にダウングレードした後に catalog.sql を実行しないでください。パッケージ STANDARD を再作成すると、このバグが起こります。catalog.sql がパッケージ STANDARD を再作成するため、Oracle はリリース 8.0.3 の catalog.sql スクリプトのすべての機能を効率的に行う新しいスクリプト (catlg803.sql) を提供しました。ただし、このスクリプトにはバグ #571546 について説明した問題に対処するための追加ステップが含まれていません。

18. catproc.sql を実行します。

```
SVRMGR> @catproc.sql
```

19. Oracle システムにアドバンスド・レプリケーションをインストールしている場合は、ダウングレード先のリリースに付属する次のカタログ・スクリプトを実行します。

```
SVRMGR> @catrep.sql
```

20. Oracle システムに Parallel Server をインストールしている場合は、ダウングレード先のリリースに付属する次のカタログ・スクリプトを実行します。

```
SVRMGR> @catparr.sql
```

21. utlrlp.sql を実行します。このステップはオプションで、文字サイズに変更がある場合でも実行できます。

```
SVRMGR> @utlrlp.sql
```

utlrlp.sql スクリプトは、以前パッケージ、プロシージャ、タイプなどの INVALID 状態にあった既存の PL/SQL モジュールをすべて再コンパイルします。これらの処置はオプションですが、再コンパイルのコストは将来よりはインストール時に掛かります。

オラクル社は utlrlp.sql の実行をお勧めします。

22. スクリプト結果のログ・ファイルへのスプーリングをオフにします。

```
SVRMGR> SPOOL OFF;
```

次にスプール・ファイルをチェックして、パッケージとプロシージャのコンパイルが成功したかどうかを検証してください。ステップ 6 でスプールに名前を付けましたが、その名前は catoutd.log でした。このファイル内に問題があれば訂正してください。

SET ECHO ON を指定した場合、ここで SET ECHO OFF にできます。

```
SVRMGR> SET ECHO OFF;
```

23. 12-10 ページの「[相互参照ビューおよびリリース 8.0.4 以前へのダウングレード](#)」の指示を実行する間に相互参照ビューを削除し、リリース 8.0.4 にダウングレードした場合は、ここでこれらのビューを再作成してください。

注意： リリース 8.0.3 にリリースした場合には、これらのビューの再作成はできません。これは相互参照ビューはリリース 8.0.3 ではサポートされないためです。

データベースがダウングレードされました。次の項で説明する手順を実行し、固有のコンポーネントのダウングレードを終了してください。

アドバンスド・レプリケーション・サポートの再生成

アドバンスド・レプリケーションを使用している場合、アドバンスド・レプリケーション・サポートを再生成する前に、12-38 ページの「[アドバンスド・レプリケーション環境のダウングレードに対する準備](#)」で説明した手順が完了しているかを確認してください。次に下記の処置を実行してください。ただし、ダウングレードしたデータベースがマスター・サイトかスナップショット・サイトかによって処置は異なります。

マスター・サイト

ダウングレードしたデータベースが、1 つ以上のオブジェクト・グループのマスター・サイトの場合、次のステップを実行してアドバンスド・レプリケーション・サポートを再生成します。

1. 各オブジェクト・グループを停止します。
2. グループ内の各レプリケート表にレプリケーション・サポートを生成します。
3. オブジェクト・グループにマスター・アクティビティを再開します。マスター定義サイトがリリース 8.1 にある場合、`GENERATE_REPLICATION_SUPPORT` コールの `GENERATE_80_COMPATIBLE=>TRUE` を指定してください。

スナップショット・サイト

ダウングレードしたデータベースがスナップショット・サイトの場合、各更新可能スナップショットにレプリケーション・サポートを生成してください。

参照： レプリケーション・サポートの生成の詳細は、『Oracle8i レプリケーション・ガイド』を参照してください。

リリース 8.0.4 での UTL_REF パッケージの再インストール

リリース 8.0.4 にダウングレードし、ダウングレードする前に UTL_REF パッケージを使用していた場合は、UTL_REF パッケージを再インストールしてください。このパッケージはダウングレード・プロセス中に、自動的に削除されました。これはこのパッケージはリリース 8.0.4 の標準インストールの一部ではないためです。

注意： リリース 8.0.5 以降にダウングレードした場合、処置は必要ありません。UTL_REF パッケージはダウングレード・プロセス中に保存されました。リリース 8.0.3 にダウングレードした場合、UTL_REF パッケージは使用できません。これは、UTL_REF パッケージがリリース 8.0.3 では無効なためです。

リリース 8.0.3 でのリカバリ・マネージャ・パッケージの再インストール

リリース 8.0.3 にダウングレードし、ダウングレード前に Recovery Manager (RMAN) のリリース 8.0.4 以降を使用していた場合、リカバリ・カタログ・データベースに次のリリース 8.0.3 のパッケージを再インストールしてください。

- dbmsrman.sql および prvtmnu.plb
- dbmsrvct.sql および prvtrvct.plb

注意： リリース 8.0.4 以降にダウングレードした場合、またはリリース 8.0.3 にダウングレードする前に Recovery Manager リリース 8.0.4 以降を使用しなかった場合、これらのパッケージを再インストールする必要はありません。

13

Oracle7 へのダウングレード

この章で説明する内容は、Oracle リリース 8 以降のインストール環境だけに適用されます。ダウングレードとは、Oracle データベースを同じバージョンの前のリリースへ変換することを意味します。たとえば、データベースをリリース 8.1 をリリース 8.0 に変換するときなどです。ダウングレードは、Oracle データベースを前のバージョンに変換することも意味します。たとえば、データベースを Oracle8i から Oracle7 に変換するときなどです。

この章では、Oracle7 へのダウングレードについて説明します。前の 8.0 リリースまたは 8.1 リリースにダウングレードする場合は、[第 12 章](#)を参照してください。

この章では、次のトピックについて説明します。

- [Oracle8i から Oracle7 へのダウングレードの概要](#)
- [新規または変更データを含まないデータベースのダウングレード](#)
- [新規または変更データを含むデータベースのダウングレード](#)

参照： ダウングレードの一部には、オペレーティング・システム固有の部分があります。ダウングレードの詳細は、オペレーティング・システム固有の Oracle ドキュメントを参照してください。

Oracle8i から Oracle7 へのダウングレードの概要

Oracle8i のデータベースは、Oracle7 のデータベース（リリース 7.3 など）にダウングレードできます。ただし、ダウングレードの方法はあまり多くありません。そのために必要な手順は操作が難しく、時間もかかります。また、Oracle7 で有効ではないバージョン 8 の新しい機能を使用するデータの保存ができないこともあります。

オラクル社では、移行ユーティリティの Oracle8i を使用した Oracle8i から Oracle7 へのダウングレードはサポートしていません。つまり Oracle8i 移行ユーティリティは下位移行をサポートしません。Oracle8i には、特にダウングレードのための他の機能は付属していません。

ダウングレードの手順は、保存する必要がある新規または変更したデータが Oracle8i のデータベースに含まれているかどうかによって決まります。実際に使用している Oracle8i のデータベースに該当する次の手順を使用します。

- [新規または変更データを含まないデータベースのダウングレード](#)
- [新規または変更データを含むデータベースのダウングレード](#)

新規または変更データを含まないデータベースのダウングレード

保存する必要がある新規または変更したデータが Oracle8i のデータベースにまだ入力されていない場合は、以前の Oracle7 ソース・データベースの完全なバックアップを復元し、それを再オープンできます。前の Oracle7 のデータベースで使用した初期化パラメータを復元したか確認してください。

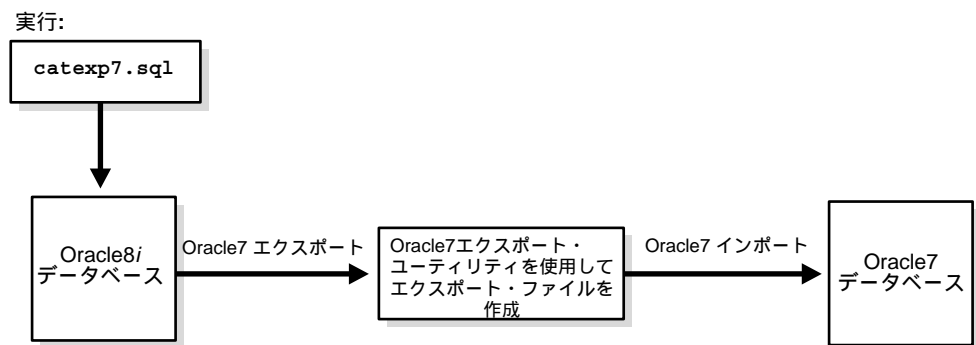
参照： バックアップした Oracle7 データベースの復元の詳細は、『Oracle7 Server 管理者ガイド』を参照してください。

前の段落で説明した方法では、リリース 8.1 のデータベースの新規または変更したデータがなくなります。リリース 8.1 のデータベースに保存しなくてはならない新規または変更したデータがある場合、13-3 ページの「[新規または変更データを含むデータベースのダウングレード](#)」で説明する手順を参照してください。

新規または変更データを含むデータベースのダウングレード

Oracle8i のデータベースに保存しなくてはならない新規または変更したデータがある場合、[図 13-1](#) に示す手順を参照してください。

図 13-1 Oracle7 へのダウングレード



次のステップで、手順を詳しく説明します。

1. Oracle8i のデータベース上の `catexp7.sql` のスクリプトを実行します。このスクリプトは `$ORACLE_HOME/rdbms/admin` のディレクトリにあります。
2. Oracle7 のエクスポート・ユーティリティを使用して、新規または変更したデータのあ
る Oracle8i のデータベースの一部をエクスポートします。

参照： Oracle8i のデータベースからの Oracle7 のエクスポートの実行の詳細は、『Oracle8i ユーティリティ・ガイド』を参照してください。

3. 以前の Oracle7 のデータベースの完全なバックアップを復元し、以前の初期化パラメータが復元結果に含まれていることを確認します。

参照： バックアップした Oracle7 データベースの復元の詳細は、『Oracle7 Server 管理者ガイド』を参照してください。

4. 復元した Oracle7 のデータベースをオープンします。
5. Oracle7 のインポート・ユーティリティを使用して、以前 Oracle8i のデータベースからエクスポートしたファイルを、復元した Oracle7 データベースへインポートします。

ダウングレードの代替方法

Oracle8i のデータベースから Oracle7 のデータベースへ表データを送る方法として、他にもいくつかの方法が利用できます。Oracle7 へデータを戻すためのこれらの方法は、Oracle8i を使用して更新した表の数が少ない場合であれば、比較的簡単に行えます。ただし、表の全データベースをコピーすることは、時間のかかる複雑なタスクであるため、Oracle8i を使用して多くの表を更新する前に、Oracle7 へ戻す必要があるかどうか判断する必要があります。

Oracle8i のデータベースを Oracle7 へダウングレードする方法として、次のような代替方法を利用できます。

- SQL*Plus を使用して新しい Oracle8i のデータから Oracle 形式でないテキスト・ファイルを作成し、次に SQL*Loader を使用してそのデータを Oracle7 のデータベースにロードし直します。
- SQL*Plus の COPY コマンドを使用して、Oracle8i のデータベース表から以前の Oracle7 のデータベース表へ新しいデータをコピーします。

参照： COPY コマンドの詳細は、『SQL*Plus ユーザーズ・ガイドおよびリファレンス』を参照してください。

- まだ以前の Oracle7 のデータベースを使用している場合は、データを追加または変更した表を作成します (CREATE TABLE ... AS SELECT を使用)。データベース・リンクを用いた Oracle7 のデータベースからの分散問合せによって、Oracle8i のデータベース表内のデータを選択します。

参照： CREATE TABLE コマンドの AS 句の詳細は、『Oracle8i SQL リファレンス』を参照してください。

移行問題のトラブルシューティング

この付録では、移行ユーティリティまたは Oracle Data Migration Assistant 使用の際に、移行プロセス中に起こりうる共通の問題に対する訂正措置について説明します。移行プロセスの説明は、移行ユーティリティの場合は第 3 章「[移行ユーティリティによる移行](#)」、Oracle Data Migration Assistant の場合は第 4 章「[Oracle Data Migration Assistant による移行](#)」を参照してください。

この付録では、次のトピックについて説明します。

- [移行ユーティリティおよび Oracle Data Migration Assistant の使用に関する問題](#)
- [ALTER DATABASE CONVERT コマンドにおける問題](#)

移行ユーティリティおよび Oracle Data Migration Assistant の使用に関する問題

移行ユーティリティまたは Oracle Data Migration Assistant の実行の際、次のような問題が起こる可能性があります。

- [一般的な移行問題](#)
- [移行ユーティリティのエラー](#)

一般的な移行問題

移行ユーティリティ、または Oracle Data Migration Assistant 実行の際に起こりうる一般的な移行問題の原因は、ユーザーのデータベース・システムの構成にあります。ユーティリティまたは Assistant がデータベースの移行に必要な処置実行中に、ユーザーの Oracle ソフトウェアによってエラーが発生します。通常、エラーが起こった場合には、ユーティリティまたは Assistant は動作を中断し、1 つ以上のエラー・メッセージを表示します。

ユーティリティまたは Assistant の実行中に次のような問題が起こった場合には、提案された処置を実行し、ユーティリティまたは Assistant を再実行してください。

注意： Oracle Data Migration Assistant を使用する場合は、Assistant の再実行前に、Oracle7 データベースのバックアップを復元してください。

SYSTEM 表領域の領域不足

この問題は、次と類似するエラー・メッセージを表示することがあります。

ORA-01652: 一時セグメントを拡張できません (473 分、表領域: SYSTEM)

移行処理を正常に行うため、SYSTEM 表領域にデータ・ファイルを新規追加し、新規データ・ファイルに十分な領域を割り当てます。

移行処理中に、一時表領域に領域がなくなることもあります。その場合は、移行処理を正常に行うために、一時表領域にデータ・ファイルを新規追加し、新規データ・ファイルに十分な領域を割り当ててください。

参照： 移行ユーティリティを使用する場合は、3-6 ページの「領域要件」および 3-18 ページのステップ 8 を参照してください。Oracle Data Migration Assistant を使用する場合は、4-4 ページの「領域要件」および 4-10 ページのステップ 9 を参照してください。

これらの項では、SYSTEM 表領域の領域要件および有効領域増加のためのデータ・ファイルの新規追加に関する詳細を説明します。

不適切な AUDIT_TRAIL パラメータ設定

この問題は、以下と類似のエラー・メッセージを表示することがあります。

ORA-00604: 再帰 SQL レベル : *num* でエラーが発生しました。

ORA-01552: SYSTEM 表領域でない表領域 : '*name*' にシステム・ロールバック・セグメントは使用できません。

ORA-02002: 監査証跡への書込み中にエラーが発生しました。

これらのエラーは、次の条件下でのみ起こります。

- 初期化パラメータ AUDIT_TRAIL が、DB または TRUE に設定されている場合
- SYS.AUD\$ 表が SYSTEM 以外の表領域にある場合

この問題を回避するには、次のステップを実行します。

1. データベースが起動されている場合は、停止します。
2. 次の方法で、*initSID.ora* の AUDIT_TRAIL パラメータを設定します。

```
AUDIT_TRAIL = NONE
```

3. 移行ユーティリティの再実行、または Oracle7 のバックアップを復元し、Oracle Data Migration Assistant を戻します。

SYSTEM ロールバック・セグメントの OPTIMAL 設定

この問題は、以下と類似のエラー・メッセージを表示することがあります。

ORA-01562: ロールバック・セグメント番号 : (id = 0) を拡張できません。

移行ユーティリティおよび Oracle Data Migration Assistant は、SYSTEM ロールバック・セグメント以外をオフラインにし、SYSTEM ロールバック・セグメントの現在割り当てられたエクステント数に MAXEXTENTS を設定し、サイズを固定します。この処置には、ユーティリティまたは Assistant の領域管理表処理中、エクステント割当てといった、領域操作を防ぐ役割があります。

SYSTEM ロールバック・セグメントに OPTIMAL 設定がある場合、エクステントは、そのデータがアクティブ・トランザクションに不必要であれば、動的に割り当てられません。動的に割当ては、SYSTEM ロールバック・セグメントが固定された場合、現在割り当てられたエクステントを小さくすることがあります。したがって、移行処理中に領域管理表のあるトランザクションを処理するには、SYSTEM ロールバック・セグメントは小さすぎることがあります。

参照： 移行ユーティリティを使用の場合は、3-11 ページのステップ 7 を参照してください。Oracle Data Migration Assistant を使用の場合は、4-9 ページのステップ 7 を参照してください。これらの項には、OPTIMAL 設定および必要時の再設定のチェック方法の説明があります。

小さい MULTIPLIER オプション設定

この問題は、以下と類似のエラー・メッセージを表示することがあります。

ORA-01632: 最大エクステント : (%s) に達しました (索引 : %s.%s)。

移行ユーティリティは、15 という低いデフォルト値を MULTIPLIER オプションに使用します。問題を回避するために、MULTIPLIER オプションの値を大きくします。

移行ユーティリティをコマンド行から実行する場合には、次を入力して MULTIPLIER オプションを 30 まで大きくしてください。

```
mig MULTIPLIER=30
```

ただし、Oracle Data Migration Assistant を使用して移行ユーティリティを実行する場合には、Oracle7 のバックアップを復元し、Assistant を再実行してください。Assistant のカスタム移行オプションを選択します。MULTIPLIER 値を求めたら、デフォルト値 15 以上の値を入力します。

参照： MULTIPLIER オプションの詳細は、3-14 ページの「[移行ユーティリティ・コマンド行オプションの検討](#)」を参照してください。

移行ユーティリティのエラー

移行ユーティリティは、移行処理中にエラー・メッセージと情報メッセージを表示することがあります。この項では、移行ユーティリティの使用中表示されるエラーについて説明します。また、それぞれのエラーについて、考えられる原因とエラー訂正のための処置を示します。情報メッセージも示していますが、訂正措置は必要ありません。

Oracle Data Migration Assistant を使用の場合、移行ユーティリティのメッセージは、ログ・ファイルに記録します。ログ・ファイルのアクセス方法については、Oracle Data Migration Assistant のオンライン・ヘルプを参照してください。Oracle Data Migration Assistant、および薦められた処置実行中の移行ユーティリティを含むメッセージを使用の場合も、Assistant に戻してください。

移行時に DBA のファイル番号ビットの圧縮はできません。

原因：移行ユーティリティがデータ・ブロック・アドレスで使用するファイル番号ビットの数を減らそうとしました。

処置：オラクル社カスタマ・サポートに連絡してください。

レコードが *number* バイトを超えているため変換ファイルを作成できません。

原因：内部エラーが発生しました。Oracle7 の制御ファイルから有効な変換ファイルを作成できませんでした。

処置：Oracle7 の制御ファイルが壊れていないかを確認し、問題を解決して移行ユーティリティを再実行してください。

CHECK_ONLY - V8 カタログ領域要件だけを見積もる (デフォルト =FALSE)

原因：これは、CHECK_ONLY コマンド行引数に関する情報メッセージです。

処置：ユーザーの処置は不要です。

CHECK_ONLY と NO_SPACE_CHECK は、相互排他オプションです。

原因：これら 2 つの相互排他のコマンド行オプションが移行ユーティリティに渡されました。

処置：どちらか 1 つだけを使用して、移行ユーティリティを再実行してください。

クライアント `nls_characterset` がサーバー `nls_characterset` と一致しない -NLS_LANG 環境変数設定をチェック。

原因： NLS_LANG キャラクタ・セットは、PROPSS のキャラクタ・セットとは合致しません。

処置： PROPSS のデータベース・キャラクタ・セットおよび NLS_LANG 環境変数の合致をチェックしてください。

コマンド行の引数：(*string*) は TRUE または FALSE でなければなりません。

原因： コマンド行引数に TRUE または FALSE 以外の値が入力されました。

処置： コマンド行引数の構文をチェックし、文を訂正して操作をもう一度やり直してください。

コマンド行の引数が < キーワード >=< 値 > で指定されていません。(*string*)

原因： コマンド行引数の指定方法が間違っています。

処置： コマンド行引数の構文をチェックし、文を訂正して操作をもう一度やり直してください。

コマンド行の引数：

原因： この情報メッセージは、コマンド行引数を表示しています。

処置： ユーザーの処置は不要です。

コマンド名が見つかりません。(*string*)

原因： 内部エラーが発生しました。migrate.psq スクリプトが壊れている可能性があります。

処置： 移行ユーティリティ、migrate.bsq およびターゲット Oracle8i ソフトウェアのバージョンの互換性と、migrate.bsq 内の破損をチェックします。問題があれば修正し、移行ユーティリティに戻ります。

コマンドが CMD (ARG1,ARG2, ...) で指定されていません。

原因： 内部エラーが発生しました。migrate.psq スクリプトが壊れている可能性があります。

処置： 移行ユーティリティ、migrate.bsq およびターゲット Oracle8i ソフトウェアのバージョンの互換性と、migrate.bsq 内の破損をチェックします。問題があれば修正し、移行ユーティリティに戻ります。

コマンドが COPYLONG (U1,T1,C1,U2,T2,C2,K1<,K2>) で指定されていません。

原因: 内部エラーが発生しました。migrate.psq スクリプトが壊れている可能性があります。

処置: 移行ユーティリティ、migrate.bsq およびターゲット Oracle7 ソフトウェアのバージョンの互換性と、migrate.bsq 内の破損をチェックします。問題があれば修正し、移行ユーティリティに戻ります。

c_file#_block# に対する number バイトの単一の連続エクステントが見つかりません。

原因: SYSTEM 表領域に連続したエクステントが十分ありません。

処置: SYSTEM 表領域に空き領域を追加し、移行ユーティリティを再実行してください。

c_ts# に対する number バイトの単一の連続エクステントが見つかりません。

原因: SYSTEM 表領域に連続したエクステントが十分ありません。

処置: SYSTEM 表領域に空き領域を追加し、移行ユーティリティを再実行してください。

i_file#_block# に対する number バイトの単一の連続エクステントが見つかりません。

原因: SYSTEM 表領域に連続したエクステントが十分ありません。

処置: SYSTEM 表領域に空き領域を追加し、移行ユーティリティを再実行してください。

i_ts# に対する number バイトの単一の連続エクステントが見つかりません。

原因: SYSTEM 表領域に連続したエクステントが十分ありません。

処置: SYSTEM 表領域に空き領域を追加し、移行ユーティリティを再実行してください。

論理名: name を変換できません。

原因: 内部エラーが発生しました。

処置: 論理名が正しく定義されているかをチェックして、移行ユーティリティを再実行してください。

現行バージョン: str-- データベースはバージョン 7.1 以降でなければなりません。

原因: 現行のデータベースが、Oracle7 リリース 7.1 より前のバージョンです。

処置: 現行のデータベースを、使用しているオペレーティング・システムでの移行ユーティリティがサポートするリリースに移行またはアップグレードしてください。その後、移行ユーティリティを再実行してください。使用しているオペレーティング・システムでの移行ユーティリティでサポートされるリリースについては、オペレーティング・システム固有の Oracle ドキュメントを参照してください。

列: *name* に対するデータ型は LONG 型でなければなりません。

原因: 内部エラーが発生しました。migrate.psq スクリプトが壊れている可能性があります。

処置: 移行ユーティリティ、migrate.bsq およびターゲット Oracle7 ソフトウェアのバージョンの互換性と、migrate.bsq 内の破損をチェックします。問題があれば修正し、移行ユーティリティに戻ります。

不整合な状態でデータ・ファイルが見つかりました。(内部エラー) --filename

原因: 内部エラーが発生しました。データ・ファイルが不整合な状態になっています。

処置: オラクル社カスタマ・サポートに連絡してください。

データ・ファイルはオフラインですが表領域がオンラインです。--datafile

原因: 表領域はオンラインですが、表領域内のデータ・ファイルがオフラインです。データ・ファイルと表領域の両方がオンラインまたは通常オフラインになるまで、移行を行うことはできません。

処置: 移行を再実行する前に、メディア回復を行ってデータ・ファイルをオンラインにしてください。

DBNAME- 現在のデータベース名 (init.ora 内の db_name)

原因: これは、DBNAME コマンド行引数に関する情報メッセージです。

処置: ユーザーの処置は不要です。

ディクショナリ定数が見つかりません。-name

原因: 内部エラーが発生しました。migrate.psq スクリプトが壊れている可能性があります。

処置: 移行ユーティリティ、migrate.bsq およびターゲット Oracle8i ソフトウェアのバージョンの互換性と、migrate.bsq 内の破損をチェックします。問題があれば修正し、移行ユーティリティに戻ります。

移行前にすべての遅延トランザクションをプッシュしてください。

原因: SYSTEM.DEF\$_CALL、DEF\$_CALLDEST または DEF\$_ERROR に、エントリが存在します。

処置: エントリが SYSTEM.DEF\$_CALL の中にある場合は、SYSTEM.DEF\$_CALL が空になるまですべての遅延トランザクションをプッシュしてください。エントリが SYSTEM.DEF\$_ERROR の中にある場合は、ローカル・キューが空になるまでその中のエラーを解決および再実行してください。移行ユーティリティを再実行してください。

slgtd のコール中にエラーが発生しました。

原因：slgtd から現在の時間を取得する際のエラーであり、内部エラーです。移行ユーティリティが壊れている可能性があります。

処置：移行ユーティリティ、migrate.bsq およびターゲット Oracle8i ソフトウェアのバージョンの互換性と、migrate.bsq 内の破損をチェックします。問題があれば修正し、移行ユーティリティに戻ります。

ファイル: *name* のクローズ中にエラーが発生しました。

原因：内部エラーが発生しました。データをディスクに書き込めませんでした。

処置：ファイルのアクセス権限が正しいかを確認してください。ファイル書込みに十分な領域や割当てが確保されているかを確認してください。また、ディスクが破損していないかを確認してください。問題を解決して移行ユーティリティを再実行してください。

object に必要な領域はおよそ *number* ブロックです。

原因：この情報メッセージの中で、移行ユーティリティはオブジェクトに必要な領域を表示します。

処置：ユーザーの処置は不要です。

ファイル: *filename* は大きすぎるため DBA に変換できません。

原因：内部エラーが発生しました。DBA 変換を行うには、ファイル名が大きすぎます。

処置：オラクル社カスタマ・サポートに連絡してください。

ファイル・ヘッダーを *number* バイト中に入れることができません。

原因：内部エラーが発生しました。

処置：制御ファイルが壊れていないかを確認し、問題を解決して移行ユーティリティを再実行してください。

制御ファイルの固定領域を *number* バイト中に入れることができません。

原因：内部エラーが発生しました。

処置：制御ファイルが壊れていないかを確認し、問題を解決して移行ユーティリティを再実行してください。

NULL の SQL 文が見つかりました。

原因：内部エラーが発生しました。migrate.psq スクリプトが壊れている可能性があります。

処置：移行ユーティリティ、migrate.bsq およびターゲット Oracle8i ソフトウェアのバージョンの互換性と、migrate.bsq 内の破損をチェックします。問題があれば修正し、移行ユーティリティに戻ります。

SYSTEM 表領域内の空き領域は *number* ブロックです。

原因：この情報メッセージは、SYSTEM 表領域内にある空き領域の大きさを通知します。

処置：ユーザーの処置は不要です。

空き領域: *number*

原因：この情報メッセージは、SYSTEM 表領域内にある空き領域の大きさを通知します。

処置：ユーザーの処置は不要です。

書込みが不完全です。

原因：内部エラーが発生しました。データをディスクに書き込めませんでした。

処置：ファイルのアクセス権限が正しいかを確認してください。ファイル書込みに十分な領域や割当てが確保されているかを確認してください。また、ディスクが破損していないかを確認してください。問題を解決して移行ユーティリティを再実行してください。

新規ディクショナリ用の領域が不足しています。(*number* バイト必要、*number* 検出)

原因：新しいデータ・ディクショナリ情報用の領域が SYSTEM 表領域に不足しています。

処置：SYSTEM 表領域に必要とされる領域を加えてから、移行ユーティリティを再実行してください。

無効な NLS_NCHAR の値が指定されました。

原因：コマンド行に指定した NLS_NCHAR 値が無効です。

処置：コマンド行に指定した NLS_NCHAR 値を訂正し、移行ユーティリティを再実行してください。

移行処理不可。データベースのブロック長 *number* が 2k (Oracle8 の最小ブロック長) 未満。

原因：既存のデータベースのブロックサイズが 2K バイトに達していません。

処置：Oracle7 のデータベースのブロックサイズが最小でも 2K バイトあることを確認してください。Oracle7 のデータベースの再作成を検討することもできます。その後、移行ユーティリティを再実行してください。

データ・ファイルがオンラインで表領域がオフラインの状態に移行処理は不可 --*datafile*。

原因：表領域はオフラインですが、表領域内のデータ・ファイルがオンラインです。データ・ファイルと表領域の両方がオンラインまたは通常オフラインになるまで、移行を行うことはできません。

処置：データ・ファイルのオンライン状態が表領域のオンライン状態と同じであることを確認し、移行ユーティリティを再実行してください。

アクティブ・トランザクションまたは未解決の取消しがあるオフライン表領域のために移行を進められません。

原因: 移行ユーティリティがデータベースを移行しようとしたときに、1 つまたは複数の表領域がオフラインで未処理の保存取消しがありました。

処置: 移行ユーティリティを使用している場合は、3-9 ページのステップ 4 に戻り、全オフラインの表領域がきれいにオフラインできたことを確認してください。Oracle Data Migration Assistant を使用している場合は、4-7 ページのステップ 4 に戻り、全オフラインの表領域がきれいにオフラインできたことを確認してください。その後、Oracle8i 移行ユーティリティを再実行してください。

データベースをマウントしています ...

原因: これは情報メッセージです。移行ユーティリティが Oracle7 のデータベースをマウントしています。

処置: ユーザーの処置は不要です。

MULTIPLIER - seg\$/uet\$ クラスタ索引サイズの増加ファクタ (デフォルト =15)

原因: これは、MULTIPLIER コマンド行設定に関する情報メッセージであり、移行ユーティリティが表示します。

処置: ユーザーの処置は不要です。

MULTIPLIER の値は、最小で 2 です。

原因: コマンド行の中で Oracle8i_file#_block# の初期サイズを指定する MULTIPLIER 値が、2 よりも小さな値です。

処置: MULTIPLIER 値を 2 以上の値に変更し、移行ユーティリティを再実行してください。

NEW_DBNAME *name* が長すぎます - 最大長は 8 文字です。

原因: 指定した新しいデータベース名の長さが 8 文字を超えています。

処置: 新しいデータベースの名前を 8 文字以下に変更し、移行ユーティリティを再実行してください。

NEW_DBNAME - 新規データベース名 (最大 8 文字)

原因: この情報メッセージは、NEW_DBNAME コマンド行引数に関する情報を表示しています。

処置: ユーザーの処置は不要です。

NLS_NCHAR - nchar キャラクタ・セット値を指定

原因: この情報メッセージは、NLS_NCHAR コマンド行引数に関する情報を表示しています。

処置: ユーザーの処置は不要です。

NO_SPACE_CHECK - 領域チェックを実行しない (デフォルト=FALSE)

原因: これは、NO_SPACE_CHECK コマンド行引数に関する情報メッセージです。

処置: ユーザーの処置は不要です。ただし、このオプションを指定して移行ユーティリティを実行する前に、十分な領域があるかを確認してください。

変換ファイルの作成中に処理された tablespace/datafile の番号が正しくありません。

原因: 変換ファイルの作成時に内部エラーが発生しました。

処置: オラクル社カスタマ・サポートに連絡してください。

データベースをオープンしています ...

原因: これは情報メッセージです。移行ユーティリティが Oracle7 のデータベースをオープンしています。

処置: ユーザーの処置は不要です。

環境変数 ORA_NLS33 が未設定か、正しく設定されていません。

原因: ORA_NLS33 環境変数が NLS データ・ファイルを指定していません。

処置: ORA_NLS33 環境変数が正しいファイルを指定するように設定し、移行ユーティリティを再実行してください。

ORA- 数値

原因: 移行ユーティリティが ORA エラーを受け取りましたが、メッセージ・テキストを取り出せません。

処置: Oracle エラー番号に応じた適切な処置を行ってください (『Oracle8i エラー・メッセージ』を参照)。

パラメータ・バッファがオーバーフローしました。

原因: 初期化パラメータ・ファイルが大きすぎて、バッファに格納できません。

処置: 廃止されたパラメータなどを削除してパラメータ・ファイルのサイズを小さくし、移行ユーティリティを再実行してください。

パラメータ・ファイルが *number* バイトを超えています。

原因: Oracle7 のデータベース用のパラメータ・ファイルが所定の最大サイズを超えています。

処置: 可能ならば、廃止されたパラメータを除去することによってパラメータ・ファイルのサイズを縮小してください。それ以外の場合は、オラクル社カスタマ・サポートに連絡してください。

PFILE - 代替 *init.ora* を使用する。

原因: これは、PFILE コマンド行引数に関する情報を表示する情報メッセージです。

処置: ユーザーの処置は不要です。

ファイル: *name* でシーク・エラーが発生しました。

原因: ファイル *name* の読取り時に内部エラーが発生しました。

処置: ファイルやディスクが壊れていないかを確認してください。移行ユーティリティを再実行する前に、問題を解決してください。

***number* バイト要求されましたが *number* バイトしか読み込まれませんでした。**

原因: 制御ファイルの読取りに問題があります。

処置: 制御ファイルが壊れていないかを確認し、問題を解決して移行ユーティリティを再実行してください。

データベースを強制停止します ...

原因: 内部エラーが発生しました。

処置: 停止の原因を通知するために追加エラー・メッセージが表示されます。そのメッセージで提案される処置を行ってください。

データベースを停止しています ...

原因: これは情報メッセージです。移行ユーティリティが Oracle7 のデータベースを停止しています。

処置: ユーザーの処置は不要です。

SPOOL - ファイルへの出力をスプールする。

原因: これは、SPOOL コマンド行引数に関する情報を表示する情報メッセージです。

処置: ユーザーの処置は不要です。

データベースを起動しています ...

原因: これは情報メッセージです。移行ユーティリティが Oracle7 インスタンスを起動しています。

処置: ユーザーの処置は不要です。

文字列引数が長すぎます。最大長は *number* です。

原因: 移行ユーティリティに渡したコマンド行引数の文字列が最大サイズを超えています。

処置: コマンド行引数の文字列を短縮し、移行ユーティリティを再実行してください。

データ・ファイルの表領域が通常オフラインではありません。--tablespace

原因: 表領域が IMMEDIATE または TEMPORARY を使用してオフラインにされました。

処置: 表領域をオンラインにしてから、NORMAL を使用してオフラインにするか削除してください。その後、移行ユーティリティを再実行してください。

コマンドの引数が多すぎます。(最大 *number*)

原因: コマンド行に指定した引数の数が多すぎます。

処置: コマンドの構文をチェックして、コマンド行に適切な数のオプションを指定してください。

LONG をコピーするのにバッファ領域を割り当てられません。

原因: 移行ユーティリティが、データベースの LONG 列にコピーするためのバッファにメモリーを割り当てられません。

処置: 移行ユーティリティに使用できる十分なコンピュータ・リソースがあるかを確認して、移行ユーティリティを再実行してください。

ファイル: *name* をオープンできません。

原因: 内部エラーが発生しました。あるいは、Oracle8i 移行ユーティリティを起動したときにファイルが所定の位置に存在しませんでした。

処置: ファイルが存在するか、ファイルのオープンと読取りの権限が Oracle にあるかを確認してください。または、ファイルやファイルが存在するディスクが壊れていないかを確認してください。問題を解決して移行ユーティリティを再実行してください。

ファイル: *name* を読み込みません。

原因: 内部エラーが発生しました。あるいは、移行ユーティリティを起動したときにファイルが所定の位置に存在しませんでした。

処置: ファイルが存在するか、ファイルのオープンと読取りの権限が Oracle にあるかを確認してください。または、ファイルやファイルが存在するディスクが壊れていないかを確認してください。問題を解決して、移行ユーティリティを再実行してください。

ファイル: *name* を書き込みません。

原因: 内部エラーが発生しました。

処置: ファイルの書き込み権限が Oracle にあるかを確認してください。または、ファイルを書き込むディスクが破損していないかを確認してください。問題を解決してから、移行ユーティリティを再実行してください。

V8 カタログの必要領域: *number*

原因: これは、移行ユーティリティを正常に実行するために SYSTEM 表領域に必要な空領域の大きさを表示する情報メッセージです。

処置: 移行ユーティリティを実行する前に、追加の空領域がどの程度あるかを確認してください。

ALTER DATABASE CONVERT コマンドにおける問題

移行ユーティリティ実行後、移行プロセス中に ALTER DATABASE CONVERT を発行すると、この項で記述した問題の内の 1 つが起こる可能性があります。通常、変換は中断し、1 つまたは複数のエラー・メッセージが表示されます。ALTER DATABASE CONVERT コマンド実行中に、次の問題の内の 1 つが起こった場合、提案した処置を実行して問題を回避してください。

注意： Assistant が ALTER DATABASE CONVERT コマンドをバックで自動的に実行し、問題を起こす条件を回避するため、Oracle Data Migration Assistant を使用すれば、これらの問題は回避できます。したがって、これらの問題は移行ユーティリティを使用の場合にのみ現れます。

Oracle7 制御ファイルの存在

この問題は、次のエラー・メッセージを戻すことがあります。

ORA-00200: 制御ファイルを作成できませんでした。
ORA-00202: 制御ファイル: *name*
ORA-27038: skgfrcre: ファイルが存在しています。

旧 Oracle7 制御ファイルは、ALTER DATABASE CONVERT コマンドを発行する前に、改名または削除が必要です。

参照： 3-22 ページ「[Oracle8i の環境での移行ステップ](#)」のステップ 2 を参照してください。

NOMOUNT 以外のモードで起動するデータベース

この問題は、次のエラー・メッセージを戻すことがあります。

ORA-00227: 制御ファイル (ブロック : block num, 番号 : blocks num) で破損ブロックが検出されました。

ORA-00202: 制御ファイル : '%s'

旧 Oracle7 制御ファイルは、ALTER DATABASE CONVERT コマンドを発行する前に、改名または削除が必要です。ALTER DATABASE CONVERT コマンドを発行する場合、データベースを NOMOUNT モード以外のモードで起動するのではなく、MOUNT モードで起動してください。

参照： 3-22 ページの「[Oracle8i の環境での移行ステップ](#)」および 3-26 ページのステップ 2 およびステップ 10 を参照してください。

変換ファイルが見付からない

この問題は、次のエラー・メッセージを戻すことがあります。

ORA-00404: 変換ファイルが見つかりません : *name*

ORA-27037: ファイル・ステータスを取得できません。

移行ユーティリティが生成した変換ファイル (UNIX の `convsid.dbf` および WindowsNT の `convert.ora`) は、所定の位置にありませんでした。所定の位置とは、UNIX においては Oracle8i 環境の `$ORACLE_HOME/dbs` ディレクトリ、WindowsNT においては Oracle8i 環境の `$ORACLE_HOME\database` ディレクトリのことです。ALTER DATABASE CONVERT コマンドを発行する前に、コンバートファイルをこの所定の位置に移動させる必要があります。

参照： 3-23 ページの「[Oracle8i の環境での移行ステップ](#)」のステップ 3 を参照してください。

REMOTE_LOGIN_PASSWORDFILE から EXCLUSIVE へ初期化パラメータの設定

この問題は、次のエラー・メッセージを戻すことがあります。

ORA-00600: 内部エラー・コード, 引数: [kzsrsdn: 1], [32]

これらのエラーは、次の条件下でのみ起こります。

- データベースがパスワード・ファイルを使用している。
- パスワード・ファイルが、正しいディレクトリに移動しなかった。正しいディレクトリとは、UNIX においては Oracle8i 環境の \$ORACLE_HOME/dbs、Windows NT においては Oracle8i 環境の \$ORACLE_HOME\database のことです。
- REMOTE_LOGIN_PASSWORDFILE 初期化パラメータが、initSID.ora ファイルの EXCLUSIVE に設定されている。

移行を続行するため、次のステップを実行します。

1. データベースの停止
2. REMOTE_LOGIN_PASSWORDFILE を、initSID.ora ファイルの NONE に設定します。

```
REMOTE_LOGIN_PASSWORDFILE = NONE
```

3. 次の SQL コマンドを入力して、データベースのマウントを起動します。

```
SVRMGR> STARTUP MOUNT
```

initSID.ora ファイルの位置を指定するために、PFILE オプションを使用することがあります。

4. ALTER DATABASE OPEN RESETLOGS コマンドを発行します。

```
SVRMGR> ALTER DATABASE OPEN RESETLOGS;
```

5. 3-26 ページのステップ 13 を起動して、移行プロセスを続行します。

既存のパスワードはすでに有効ではないため、使用することはできません。Oracle8i でパスワード・ファイルを使用する場合は、パスワード・ファイルを再作成し、それをユーザーの間で定着させます。必ず REMOTE_LOGIN_PASSWORDFILE を正確に設定してください。

データベース名の不合致

この問題は、次のエラー・メッセージを戻すことがあります。

ORA-01103: 制御ファイルのデータベース名: *name* が *name* ではありません。

データベース名に不合致があります。次の位置に、1 つまたは複数の不合致があります。

- *init_{sid}.ora* ファイルの DB_NAME パラメータが指定したデータベース名は、*conv_{sid}.dbf* ファイル名のデータベース名と合致しません。
- ORACLE_SID 環境変数が設定した Oracle8i インスタンス ID は、*conv_{sid}.dbf* のファイル名のデータベース名と合致しません。

注意: この問題は、UNIX オペレーティング・システムでのみ発生します。この問題は、Windows NT では発生しません。

問題を訂正するために、正しいデータベース名が次の各位置で指定されているか確認してください。

- ORACLE_SID 環境変数
- *init_{sid}.ora* ファイルの DB_NAME パラメータです。
- *conv_{sid}.dbf* ファイル名の一部の *sid*

たとえば、*init_{sid}.ora* ファイルの ORACLE_SID 環境変数および DB_NAME パラメータ両方を DB1 に設定すれば、*conv_{sid}.dbf* ファイル名は次のとおりになります。

convDB1.dbf

ALTER DATABASE CONVERT コマンドの再実行

この問題は、次のエラー・メッセージを戻すことがあります。

ORA-01122: データベース・ファイル: *name* の照合検査でエラーが発生しました。

ORA-01110: データ・ファイル: *name: str*

ORA-01202: このファイルの作成日時が一致しません。

これらのエラーは、以前に ALTER DATABASE CONVERT を発行したが、失敗したことを意味します。これらのエラーが発生した場合には、ALTER DATABASE OPEN RESETLOGS コマンドを発行することによって、移行プロセスの次のステップに進むことができます。ただし、問題が発生した場合、移行プロセスを始める前に作成したバックアップを復元し、それを使用して移行を初めから再開します。第 3 章の最初から開始します。ただし、第 2 章にある移行前の処理を必ず実行してください。

データ・ファイルのバージョン整合性の問題

この問題は、次のエラー・メッセージを戻すことがあります。

ORA-01122: データベース・ファイル : *name* の照合検査でエラーが発生しました。

ORA-01110: データ・ファイル : *name*: *str*

ORA-01211: Oracle7 データ・ファイルは Oracle8 に移行していません。

移行ユーティリティは、Oracle7 の環境のデータベースに最後にアクセスするユーティリティでなければなりません。エラー・メッセージの中で指摘されたデータ・ファイルは、移行ユーティリティを実行する前にバックアップしたもの、または移行ユーティリティを実行した後に Oracle7 でオープンされたデータベースのいずれかです。移行ユーティリティの実行の際のデータ・ファイルのみ、Oracle8i を使用してアクセスできます。

データ・ファイルのバージョンを確実に統合するには、データベースを Oracle8i へ変換するときに、ディクショナリ内のシステム変更番号 (SCN)、変換ファイルおよびファイル・ヘッダーのすべてに一貫性がなければなりません。移行ユーティリティを実行した後で Oracle7 でデータベースを開くと、SCN のチェックが、ALTER DATABASE CONVERT コマンドを発行するときに失敗します。

この問題を回避するには、次のステップを実行します。

1. データベースを停止します。
2. ALTER DATABASE CONVERT を使用して作成された制御ファイルを別の名前に改名します。
3. STARTUP NOMOUNT コマンドを発行する直前に保存した Oracle7 の制御ファイルのコピーを復元します。

保存した Oracle7 の制御ファイルがない場合、移行プロセスの起動前に作成したバックアップを復元します。

4. 移行処理を初めから開始し、移行ユーティリティの実行後、Oracle7 環境でデータベースを起動しないでください。第 3 章の初めから開始します。

B

初期化パラメータの変更

バージョン 8 は、新規初期化パラメータを `init.ora` ファイルでできるようにサポートしています。バージョン 8 では、一部の初期化パラメータは改名または廃止されています。この付録では、バージョン 8 内の新規、改名済みおよび廃止されたパラメータを示し、特定の初期化パラメータの互換性の問題について説明します。

この付録では、次のトピックについて説明します。

- [バージョン 8 で追加された初期化パラメータ](#)
- [バージョン 8 で改名された初期化パラメータ](#)
- [バージョン 8 で廃止された初期化パラメータ](#)
- [初期化パラメータの互換性の問題](#)

参照： この付録に示す新規および変更初期化パラメータの詳細は、『Oracle8i リファレンス・マニュアル』を参照してください。

注意： この付録に示すパラメータの一部は、オペレーティング・システム固有のものです。これら初期化パラメータの詳細は、オペレーティング・システムに固有の Oracle ドキュメントを参照してください。

バージョン 8 で追加された初期化パラメータ

この項で示す初期化パラメータは、バージョン 8 で新規に追加されたものです。

リリース 8.0 で追加された初期化パラメータ

次のパラメータはリリース 8.0 に追加されたものです。

ALLOW_PARTIAL_SN_RESULTS	ALWAYS_SEMI_JOIN
AQ_TM_PROCESSES	ARCH_IO_SLAVES
BACKUP_DISK_IO_SLAVES	BACKUP_TAPE_IO_SLAVES
BUFFER_POOL_KEEP	BUFFER_POOL_RECYCLE
COMPLEX_VIEW_MERGING	CONTROL_FILE_RECORD_KEEP_TIME
DB_BLOCK_MAX_DIRTY_TARGET	DB_FILE_DIRECT_IO_COUNT
DB_FILE_NAME_CONVERT	DB_WRITER_ROCESSES
DBWR_IO_SLAVES	DISK_ASYNC_IO
FAST_FULL_SCAN_ENABLED	FREEZE_DB_FOR_FAST_INSTANCE_RECOVERY
GC_DEFER_TIME	GC_LATCHES
HI_SHARED_MEMORY_ADDRESS	INSTANCE_GROUPS
LARGE_POOL_MIN_ALLOC	LARGE_POOL_SIZE
LGWR_IO_SLAVES	LM_LOCKS
LM_PROCS	LM_RESS
LOCAL_LISTENER	LOCK_NAME_SPACE
LOCK_SGA	LOCK_SGA_AREAS
LOG_ARCHIVE_DUPLEX_DEST	LOG_ARCHIVE_MIN_SUCCEED_DEST
LOG_FILE_NAME_CONVERT	MTS_RATE_LOG_SIZE
MTS_RATE_SCALE	NLS_CALENDAR
O7_DICTIONARY_ACCESSIBILITY	OBJECT_CACHE_MAX_SIZE_PERCENT
OBJECT_CACHE_OPTIMAL_SIZE	OGMS_HOME
OPEN_LINKS_PER_INSTANCE	OPS_ADMIN_GROUP
OPTIMIZER_FEATURES_ENABLE	OPTIMIZER_INDEX_CACHING
OPTIMIZER_INDEX_COST_ADJ	OPTIMIZER_MAX_PERMUTATIONS
PARALLEL_ADAPTIVE_MULTI_USER	PARALLEL_BROADCAST_ENABLED

PARALLEL_EXECUTION_MESSAGE_SIZE	PARALLEL_INSTANCE_GROUP
PARALLEL_MIN_MESSAGE_POOL	PARALLEL_SERVER
PARALLEL_TRANSACTION_RESOURCE_TIMEOUT	PLSQL_V2_COMPATIBILITY
PUSH_JOIN_PREDICATE	READ_ONLY_OPEN_DELAYED
REPLICATION_DEPENDENCY_TRACKING	SERIAL_REUSE
SESSION_MAX_OPEN_FILES	SHARED_MEMORY_ADDRESS
STAR_TRANSFORMATION_ENABLED	TAPE_ASYNC_IO
TIMED_OS_STATISTICS	TRANSACTION_AUDITING
USE_INDIRECT_DATA_BUFFERS	

リリース 8.1 で追加された初期化パラメータ

次のパラメータはリリース 8.1 に追加されたものです。

DB_BLOCK_CHECKING	ENT_DOMAIN_NAME
FAST_START_IO_TARGET	FAST_START_PARALLEL_ROLLBACK
HS_AUTOREGISTER	INSTANCE_NAME
JAVA_POOL_SIZE	LOG_ARCHIVE_DEST_ <i>n</i>
LOG_ARCHIVE_DEST_STATE_ <i>n</i>	LOG_ARCHIVE_MAX_PROCESSES
NLS_COMP	NLS_DUAL_CURRENCY
PARALLEL_AUTOMATIC_TUNING	PARALLEL_SERVER_INSTANCES
PARALLEL_THREADS_PER_CPU	PLSQL_LOAD_WITHOUT_RECOMPILE
QUERY_REWRITE_ENABLED	QUERY_REWRITE_INTEGRITY
RESOURCE_MANAGER_PLAN	SERVICE_NAME
SORT_MULTIBLOCK_READ_COUNT	STANDBY_ARCHIVE_DEST

バージョン 8 で改名された初期化パラメータ

この項で示す初期化パラメータは、バージョン 8 で改名されたものです。

リリース 8.0 で改名された初期化パラメータ

次の初期化パラメータはリリース 8.0 で改名されたものです。

表 B-1 リリース 8.0 で改名された初期化パラメータ

リリース 8.0 以前の名前	リリース 8.0 の名前
ASYNC_READ	DISK_ASYNCH_IO
ASYNC_WRITE	DISK_ASYNCH_IO
CCF_IO_SIZE *	DB_FILE_DIRECT_IO_COUNT *
DB_FILE_STANDBY_NAME_CONVERT	DB_FILE_NAME_CONVERT
DB_WRITERS	DBWR_IO_SLAVES
LOG_FILE_STANDBY_NAME_CONVERT	LOG_FILE_NAME_CONVERT
SNAPSHOT_REFRESH_INTERVAL	JOB_QUEUE_INTERVAL

*CCF_IO_SIZE (バイト) および DB_FILE_DIRECT_IO_COUNT (データベース・ブロック) に対して、ユニットは異なります。

リリース 8.1.4 で改名された初期化パラメータ

次の初期化パラメータはリリース 8.1.4 で改名されたものです。

表 B-2 リリース 8.1.4 で改名された初期化パラメータ

リリース 8.1.3 の名前	リリース 8.1.4 以降の名前
MVIEW_REWRITE_ENABLED	QUERY_REWRITE_ENABLED
REWRITE_INTEGRITY	QUERY_REWRITE_INTEGRITY

リリース 8.1.5 で改名された初期化パラメータ

次の初期化パラメータはリリース 8.1.5 で改名されたものです。

表 B-3 リリース 8.1.5 で改名された初期化パラメータ

リリース 8.1.3 および 8.1.4 の名前	リリース 8.1.5 以降の名前
NLS_UNION_CURRENCY	NLS_DUAL_CURRENCY
PARALLEL_RRANSACTION_RECOVERY	FAST_START_PARALLEL_ROLLBACK

バージョン 8 で廃止された初期化パラメータ

この項が示す初期化パラメータはバージョン 8 で廃止されたものです。

リリース 8.0 で廃止された初期化パラメータ

次の初期化パラメータは、リリース 8.0 で廃止されたもので、リリース 8.0 以降での使用はできません。

CHECKPOINT_PROCESS	FAST_CACHE_FLUSH
GC_DB_LOCKS	GC_FREELIST_GROUPS
GC_ROLLBACK_SEGMENTS	GC_SAVE_ROLLBACK_LOCKS
GC_SEGMENTS	GC_TABLESPACES
IO_TIMEOUT	INIT_SQL_FILES
IPQ_ADDRESS	IPQ_NET
LM_DOMAINS	LM_NON_FAULT_TOLERANT
MLS_LABEL_FORMAT	OPTIMIZER_PARALLEL_PASS
PARALLEL_DEFAULT_MAX_SCANS	PARALLEL_DEFAULT_SCAN_SIZE
POST_WAIT_DEVICE	SEQUENCE_CACHE_HASH_BUCKETS
UNLIMITED_ROLLBACK_SEGMENTS	USE_IPQ
USE_POST_WAIT_DRIVER	USE_READV
USE_SIGIO	V733_PLANS_ENABLED

注意： 1 つまたは複数の廃止された初期化パラメータを使用して、リリース 8.0 またはそれ以降のデータベースを起動するとエラーになり、データベースを起動できません。

リリース 8.1 で廃止された初期化パラメータ

データベース管理を単純化するため、かなりの数の初期化パラメータがリリース 8.1 で廃止されました。次の初期化パラメータは、リリース 8.1 で廃止されたもので、リリース 8.1 以降での使用はできません。

ALLOW_PARTIAL_SN_RESULTS	ARCH_IO_SLAVES
B_TREE_BITMAP_PLANS	BACKUP_DISK_IO_SLAVES
CACHE_SIZE_THRESHOLD	CLEANUP_ROLLBACK_ENTRIES
CLOSE_CACHED_OPEN_CURSORES	COMPATIBLE_NO_RECOVERY
COMPLEX_VIEW_MERGING	DB_BLOCK_CHECKPOINT_BATCH
DB_BLOCK_LRU_EXTENDED_STATISTICS	DB_BLOCK_LRU_STATISTICS
DB_FILE_SIMULTANEOUS_WRITES	DELAYED_LOGGING_BLOCK_CLEANOUTS
DISCRETE_TRANSACTIONS_ENABLED	DISTRIBUTED_LOCK_TIMEOUT
DISTRIBUTED_RECOVERY_CONNECTION_HOLD_TIME	FAST_FULL_SCAN_ENABLED
FREEZE_DB_FOR_FAST_INSTANCE_RECOVERY	GC_LATCHES
GC_LCK_PROCS	JOB_QUEUE_KEEP_CONNECTIONS
LARGE_POOL_MIN_ALLOC	LGWR_IO_SLAVES
LOCK_SGA_AREAS	LOG_ARCHIVE_BUFFER_SIZE
LOG_ARCHIVE_BUFFERS	LOG_BLOCK_CHECKSUM
LOG_FILES	LOG_SIMULTANEOUS_COPIES
LOG_SMALL_ENTRY_MAZ_SIZE	MAX_TRANSACTION_BRANCHES
MTS_LISTENER_ADDRESS	MTS_MULTIPLE_LISTENERS
MTS_RATE_LOG_SIZE	MTS_RATE_SCALE
MTS_SERVIVE	OGMS_HOME
OPS_ADMIN_GROUP	PARALLEL_DEFAULT_MAX_INSTANCES
PARALLEL_MIN_MESSAGE_POOL	PARALLEL_SERVER_IDLE_TIME
PARALLEL_TRANSACTION_RESOURCE_TIMEOUT	PUSH_JOIN_PREDICATE
REDUCE_ALARM	ROW_CACHE_CURSORS
SEQUENCE_CACHE_ENTRIES	SEQUENCE_CACHE_HASH_BUCKETS
SHARED_POOL_RESERVED_MIN_ALLOC	SNAPSHOT_REFRESH_KEEP_CONNECTIONS
SNAPSHOT_REFRESH_PROCESSES	SORT_DIRECT_WRITES
SORT_READ_FAC	SORT_SPACEMAP_SIZE
SORT_WRITE_BUFFER_SIZE	SORT_WRITE_BUFFERS
SPIN_COUNT	TEMPORARY_TABLE_LOCKS

TEXT_ENABLE

USE_ISM

注意： 1 つまたは複数の廃止された初期化パラメータを使用すると、リリース 8.1 データベースを起動できますが、警告が戻され、アラート・ログに記録されます。

初期化パラメータの互換性の問題

この章の前半で示した新規、変更および廃止の初期化パラメータの一覧では、異なる Oracle のリリースにおける初期化パラメータの違いを示しました。ただし、特定の初期化パラメータの変更は、データベースの互換性の問題を引き出す可能性があるため、特別な注意が必要です。この項では、これらパラメータの変更について説明します。

LOG_CHECKPOINT_TIMEOUT の新規デフォルト値

LOG_CHECKPOINT_TIMEOUT の初期化パラメータをリリース 8.1.5 で起動すると、新規のデフォルト値が設定されます。以前のリリースではデフォルト値が 0 秒でしたが、リリース 8.1.5 以降でのデフォルト値は 1800 秒です。詳細は、『Oracle8i リファレンス・マニュアル』を参照してください。

データ・ディクショナリの保護

O7_DICTIONARY_ACCESSIBILITY 初期化パラメータは、Oracle7 のデータ・ディクショナリの動作を継承するためのスイッチです。これは一時的な機能であり、将来の Oracle8i リリースでは提供しない予定です。

DML_LOCKS パラメータ

Oracle8i システムでは、通常、DDL 操作の実行中に Oracle7 のシステムで必要とする以上の DML ロックを使用します。ただし、Oracle7 の DML_LOCKS パラメータのデフォルト設定は Oracle8i のシステムにも適しており、DML を非常に多く使用するアプリケーションにも適しています。

DML_LOCKS のデフォルト値はトランザクションの数の倍数であり、ロールバック・セグメントの数を基に計算されます。ただし、Oracle8i ではロールバック・セグメント当たりの使用トランザクション数が Oracle7 よりも少なくなっています。したがって、DML_LOCKS の Oracle8i でのデフォルト値が低くなっています。ある極端なテスト条件では、Oracle8i のシステムは (Oracle7 の) DML ロック制限を超え、DML_LOCKS パラメータ値を増やす必要があります。

また、TRANSACTION_PER_ROLLBACK_SEGMENT パラメータ設定の調整が必要になる場合があります。それはオペレーティング・システム固有の設定によって異なります。この変更についての情報メッセージは、データベースの起動捜査中に表示されます。

DB_DOMAIN パラメータ

リリース 8.1 から、DB_DOMAIN 初期化パラメータが解除されると、このパラメータはデフォルトで NULL に設定されます。Oracle の以前のリリースでは、デフォルト設定は次のように設定されていました。

```
.WORLD
```

DB_DOMAIN の NULL 設定は、環境によってはデータベース接続の問題の原因になることがあります。リリース 8.1 への移行またはアップグレードを実行する前に、`init.ora` ファイルの DB_DOMAIN パラメータを次のどれか 1 つに設定することを確認してください。

- WORLD
- 環境に有効なドメイン設定

DB_DOMAIN を既存のデータベースに設定していない場合、リリース 8.1 に移行またはアップグレードする前に、WORLD に設定してください。

既存のデータベースの環境に有効なドメインに DB_DOMAIN を設定した場合、リリース 8.1 に移行またはアップグレードする前に、その設定を `init.ora` ファイルに保存してください。

大規模プールから割り当てられたパラレル実行

リリース 8.1 から、新規の初期化パラメータである PARALLEL_AUTOMATIC_TUNING が TRUE に設定されるごとに、パラレル実行のメッセージ・バッファを大規模プールから割り当てることができます。以前のリリースでは、共有プールから割り当てられていました。リリース 8.1 に移行またはアップグレードし、PARALLEL_AUTOMATIC_TUNING の設定に TRUE を選択すると、次の初期化パラメータの変更によって問題を回避できます。

- SHARED_POOL_SIZE
- LARGE_POOL_SIZE

通常、問題を回避するためには、SHARED_POOL_SIZE の設定を縮小し LARGE_POOL_SIZE の設定を拡大しますが、かわりに SHARED_POOL_SIZE の設定を縮小し、LARGE_POOL_SIZE の設定を Oracle に計算させます。PARALLEL_AUTOMATIC_TUNING を TRUE に設定し、LARGE_POOL_SIZE を設定しない場合にのみ、Oracle がデフォルトの LARGE_POOL_SIZE の計算をします。

その計算は次の初期化パラメータの設定に基づいて行います。

- PARALLEL_MAX_SERVERS
- PARALLEL_THREADS_PER_CPU
- PARALLEL_SERVER_INSTANCES
- MTS_DISPATCHERS
- DBWR_IO_SLAVES

PARALLEL_AUTOMATIC_TUNING を設定していないまたは FALSE と設定している、および LARGE_POOL_SIZE を設定していない場合、LARGE_POOL_SIZE のデフォルトに対する値は 0 になります。

注意： PARALLEL_AUTOMATIC_TUNING を TRUE に設定すると、COMPATIBLE のパラメータが 8.1.0 以下に設定してある場合でも新規の動作を行います。

参照： PARALLEL_AUTOMATIC_TUNING の初期化パラメータのその他影響の詳細は、『Oracle8i リファレンス・マニュアル』および『Oracle8i チューニング』を参照してください。

次の方法は、リリース 8.1 への移行またはアップグレードの際の、さまざまな初期化パラメータ設定が原因となる動作を示します。

無変更のパラメータ設定の保存

全設定からのパラメータ変更はありません。

表 B-4 無変更のパラメータ設定の保存

パラメータ	設定
PARALLEL_AUTOMATIC_TUNING	解除 (FALSE へ標準設定)
SHARED_POOL_SIZE	パラレル実行に必要な領域も含めて大きい値に設定
LARGE_POOL_SIZE	解除 (ゼロへ標準設定)

これらの設定方法が最も一般的です。この場合、共有プールでのパラレル実行に必要な領域をすでに占めています。

PARALLEL_AUTOMATIC_TUNING の使用

パラメータが前回の設定から次の設定に変更されました。

表 B-5 PARALLEL_AUTOMATIC_TUNING の使用

パラメータ	設定
PARALLEL_AUTOMATIC_TUNING	TRUE に設定
SHARED_POOL_SIZE	パラレル実行を除く、全クライアントを占める小さい値に設定
LARGE_POOL_SIZE	解除 (パラレル実行に必要な領域も含めて大きい値に標準設定)

この場合、パラレル実行は自動計算に基づいて、大規模プールからバッファを割り当てます。バッファの割当てはより効率的で、失敗すると、共有プールのクライアントから孤立します。

PARALLEL_AUTOMATIC_TUNING の使用および LARGE_POOL_SIZE の設定

パラメータが前回の設定から次の設定に変更されました。

表 B-6 PARALLEL_AUTOMATIC_TUNING の使用および LARGE_POOL_SIZE の設定

パラメータ	設定
PARALLEL_AUTOMATIC_TUNING	TRUE に設定
SHARED_POOL_SIZE	パラレル実行を除く、全クライアントを占める小さい値に設定
LARGE_POOL_SIZE	パラレル実行に必要な領域も含めて大きい値に設定

この場合、パラレル実行は大規模プールからのバッファを割り当てます。LARGE_POOL_SIZE の解除の初期テスト終了後、LARGE_POOL_SIZE に対する標準計算が大規模プールへの要件を反映しなかったことを決めます。したがって、LARGE_POOL_SIZE の設定は手動になります。LARGE_POOL_SIZE を正しく設定するとバッファの割当てはより効率的になり、失敗すると、共有プールのクライアントから孤立します。

SHARED_POOL_SIZE 無変更での PARALLEL_AUTOMATIC_TUNING の使用

パラメータが前回の設定から次の設定に変更されました。

表 B-7 SHARED_POOL_SIZE 無変更での PARALLEL_AUTOMATIC_TUNING の使用

パラメータ	設定
PARALLEL_AUTOMATIC_TUNING	TRUE への設定
SHARED_POOL_SIZE	パラレル実行に必要な領域も含めて大きい値に設定します。
LARGE_POOL_SIZE	解除（パラレル実行に必要な領域も含めて大きい値に標準設定します）。

この場合、大規模プールからのバッファを割り当てますが、SHARED_POOL_SIZE の変更がなかったため SGA は不必要に大きくなり、パフォーマンスの妨げになる可能性があります。したがって、適切な SHARED_POOL_SIZE および LARGE_POOL_SIZE の設定をしてから、PARALLEL_AUTOMATIC_TUNING を TRUE に設定してください。

アーカイブ・ログの宛先のパラメータ

リリース 8.1 は、新規アーカイブ・ログの宛先のパラメータをサポートします。リリース 8.1 への移行またはアップグレードの後、旧リリース 8.1 前のパラメータ (LOG_ARCHIVE_DEST および LOG_ARCHIVE_DUPLEX_DEST) から新規リリース 8.1 のパラメータ (LOG_ARCHIVE_DEST_n および LOG_ARCHIVE_DEST_STATE_n) への大幅な変換が可能になります。旧パラメータへの大幅な変換も可能になります。

新規アーカイブ・ログの宛先のパラメータへの変更

新規アーカイブ先、対応する状態およびオプションを決めたら、旧アーカイブ・ログの宛先のパラメータから新規アーカイブ・ログの宛先のパラメータへ変更するため、次のステップを実行してください。

1. ALTER SYSTEM を使用して、LOG_ARCHIVE_MIN_SUCCEED_DEST を 1 に設定します。
2. ALTER SYSTEM を使用して、LOG_ARCHIVE_DUPLEX_DEST を NULL に設定します。
3. ALTER SYSTEM を使用して、LOG_ARCHIVE_DEST を NULL に設定します。
4. ALTER SYSTEM を使用して、LOG_ARCHIVE_DEST_STATE_n のパラメータを指示どおりに "defer" または "enable" に設定します。enable はデフォルトですが、オラクル社では状態を各宛先に明確に設定することをお勧めします。
5. ALTER SYSTEM を使用して、ローカルの宛先を指定する値に、最低でも 1 つの LOG_ARCHIVE_DEST_n を設定します。
6. ALTER SYSTEM を使用して、指定どおりに、その他の LOG_ARCHIVE_DEST_n パラメータを設定します。
7. ALTER SYSTEM を使用して、LOG_ARCHIVE_MIN_SUCCEED_DEST を指定した値に設定します。

たとえば、次の 2 つの宛先があると仮定します。

- /oracle/dbs/arclog
- /backup/dbs/arclog

両方の宛先が固定されます (最小宛先件数は 2) 新規の宛先は次のとおりです。

- /oracle/dbs/arclog (local)
- standby1 (a standby database)
- /backup/dbs/arclog
- /backup2/dbs/arclog

最初の宛先である standby および backup のどちらかが固定されます（最小宛先件数は 3）。これらの仮定をもとに、次の SQL 文を発行して旧アーカイブ・ログの宛先を新規に変更してください。

```
ALTER SYSTEM SET LOG_ARCHIVE_MIN_SUCCEED_DEST = 1;

ALTER SYSTEM SET LOG_ARCHIVE_DUPLEX_DEST = ' ';

ALTER SYSTEM SET LOG_ARCHIVE_DEST = ' ';

ALTER SYSTEM SET LOG_ARCHIVE_DEST_STATE_1 = 'enable';
ALTER SYSTEM SET LOG_ARCHIVE_DEST_STATE_2 = 'enable';
ALTER SYSTEM SET LOG_ARCHIVE_DEST_STATE_3 = 'enable';
ALTER SYSTEM SET LOG_ARCHIVE_DEST_STATE_4 = 'enable';

ALTER SYSTEM SET LOG_ARCHIVE_DEST_1 = 'LOCATION=/oracle/dbs/arclog MANDATORY';
ALTER SYSTEM SET LOG_ARCHIVE_DEST_2 = 'SERVICE=standby1 MANDATORY';
ALTER SYSTEM SET LOG_ARCHIVE_DEST_3 = 'LOCATION=/backup/dbs/arclog OPTIONAL';
ALTER SYSTEM SET LOG_ARCHIVE_DEST_4 = 'LOCATION=/backup2/dbs/arclog OPTIONAL';
ALTER SYSTEM SET LOG_ARCHIVE_MIN_SUCCEED_DEST = 3;
```

旧アーカイブ・ログの宛先のパラメータへの変更

次のステップを実行して、旧アーカイブ・ログの宛先のパラメータに戻してください。

1. ALTER SYSTEM を使用して、LOG_ARCHIVE_MIN_SUCCEED_DEST を 1 に設定します。
2. ALTER SYSTEM を使用して、指定どおりに、全 LOG_ARCHIVE_DEST_n パラメータを NULL に設定します。
3. ALTER SYSTEM を使用して、ローカルの宛先を指定する値に LOG_ARCHIVE_DEST のパラメータを設定します。
4. ALTER SYSTEM を使用して、指定どおりに、LOG_ARCHIVE_MIN_SUCCEED_DEST に設定します。
5. ALTER SYSTEM を使用して、LOG_ARCHIVE_MIN_SUCCEED_DEST を指定した値に設定します。

たとえば、次の 2 つの宛先があると仮定します。

- /oracle/dbs/arclog (LOG_ARCHIVE_DEST_1)
- /backup/dbs/arclog (LOG_ARCHIVE_DEST_4)

宛先は両方とも固定されています。新規宛先および最小件数は同じです。

これらの仮定をもとに、次の SQL 文を発行して新規アーカイブ・ログの宛先を旧アーカイブ・ログの宛先に変更してください。

```
ALTER SYSTEM SET LOG_ARCHIVE_MIN_SUCCEED_DEST = 1;

ALTER SYSTEM SET LOG_ARCHIVE_DEST_4 = ' ';
ALTER SYSTEM SET LOG_ARCHIVE_DEST_1 = ' ';

ALTER SYSTEM SET LOG_ARCHIVE_DEST = '/oracle/dbs/arclog';

ALTER SYSTEM SET LOG_ARCHIVE_DUPLEX_DEST = '/backup/dbs/arclog';

ALTER SYSTEM SET LOG_ARCHIVE_MIN_SUCCEED_DEST = 2;
```

注意： リリース 8.1 からリリース 8.0 へダウングレードする場合、`init.ora` ファイルの新規パラメータを旧パラメータと置き換えて、旧アーカイブ・ログの宛先のパラメータをもとに戻します。ただし、動的方法は使用しないでください。

トランザクション・パラメータ中に起こる可能性のあるエラー

この項の前半で説明した手順にそってアーカイブ先のパラメータを変更する場合、アーカイブが使用可能になると、ログ・ファイル中に次のエラー・メッセージが表示されることがあります。

- アラート・ログで - Archiving not possible: 使用可能な宛先がありません。
- トレース・ログで - ARCH: INCOMPLETE、使用可能な宛先がありません。

アーカイブが使用禁止の場合、これらのエラーは起こりません。有効なアーカイブ先がない場合、作業中にエラーは発生しなくなります。ただし、パラメータの変更が完了すればエラーは発生しなくなります。これらのエラーを回避するため、変換中にアーカイブを使用禁止にはいけません。

静的データ・ディクショナリ・ビューの変更

バージョン 8 は新規の静的データ・ディクショナリ・ビューをサポートします。バージョン 8 では、一部の静的データ・ディクショナリ・ビューは変更または廃止されています。この付録では、バージョン 8 の新規の静的データ・ディクショナリ・ビュー、変更された静的データ・ディクショナリ・ビュー、および廃止された静的データ・ディクショナリ・ビューを示します。

この付録では、次のトピックについて説明します。

- [バージョン 8 で追加された静的データ・ディクショナリ・ビュー](#)
- [バージョン 8 で追加された列と静的データ・ディクショナリ・ビュー](#)
- [バージョン 8 で削除された列と静的データ・ディクショナリ・ビュー](#)
- [バージョン 8 で改名された列と静的データ・ディクショナリ・ビュー](#)
- [NULL を戻す可能性のある列と静的データ・ディクショナリ・ビュー](#)
- [バージョン 8 で廃止された静的データ・ディクショナリ・ビュー](#)

参照： この付録で示す新規のビューあるいは変更されたビューの詳細、およびこれらのビューにある列の詳細は、『Oracle8i リファレンス・マニュアル』を参照してください。

バージョン 8 で追加された静的データ・ディクショナリ・ビュー

この項で示す静的データ・ディクショナリ・ビューは、バージョン 8 で新しく追加されました。

リリース 8.0 で追加された静的データ・ディクショナリ・ビュー

リリース 8.0 では、次の静的データ・ディクショナリ・ビューが追加されました。

ALL_ALL_TABLES	ALL_COLL_TYPES
ALL_DIRECTORIES	ALL_IND_PARTITIONS
ALL_LIBRARIES	ALL_LOBS
ALL_METHOD_PARAMS	ALL_METHOD_RESULTS
ALL_NESTED_TABLES	ALL_OBJECT_TABLES
ALL_OBJECTS	ALL_PART_COL_STATISTICS
ALL_PART_HISTOGRAMS	ALL_PART_INDEXES
ALL_PART_KEY_COLUMNS	ALL_PART_TABLES
ALL_PROBE_OBJECTS	ALL_REFS
ALL_REGISTERED_SNAPSHOTS	ALL_REPCOLUMN
ALL_REPGENOBJECTS	ALL_SNAPSHOT_LOGS
ALL_SNAPSHOT_REFRESH_TIMES	ALL_TAB_COL_STATISTICS
ALL_TAB_HISTOGRAMS	ALL_TAB_PARTITIONS
ALL_TYPE_ATTRS	ALL_TYPE_METHODS
ALL_TYPES	DBA_ALL_TABLES
DBA_COLL_TYPES	DBA_DIRECTORIES
DBA_IND_PARTITIONS	DBA_LIBRARIES
DBA_LOBS	DBA_METHOD_PARAMS
DBA_METHOD_RESULTS	DBA_NESTED_TABLES
DBA_OBJECT_TABLES	DBA_PART_COL_STATISTICS
DBA_PART_HISTOGRAMS	DBA_PART_INDEXES
DBA_PART_KEY_COLUMNS	DBA_PART_TABLES
DBA_PENDING_TRANSACTIONS	DBA_QUEUE_SCHEDULES
DBA_QUEUE_TABLES	DBA_QUEUES

DBA_REFS	DBA_REGISTERED_SNAPSHOT_GROUPS
DBA_REGISTERED_SNAPSHOTS	DBA_REPCOLUMN
DBA_REPGENOBJECTS	DBA_SNAPSHOT_LOG_FILTER_COLS
DBA_SNAPSHOT_REFRESH_TIMES	DBA_TAB_COL_STATISTICS
DBA_TAB_HISTOGRAMS	DBA_TAB_PARTITIONS
DBA_TYPE_ATTRS	DBA_TYPE_METHODS
DBA_TYPES	DEFLOB
DEFPROPAGATOR	FILEXT\$
HS_ALL_CAPS	HS_ALL_DD
HS_ALL_INITS	HS_BASE_CAPS
HS_BASE_DD	HS_CLASS_CAPS
HS_CLASS_DD	HS_CLASS_INIT
HS_EXTERNAL_OBJECT_PRIVILEGES	HS_EXTERNAL_OBJECTS
HS_EXTERNAL_USER_PRIVILEGES	HS_FDS_CLASS
HS_FDS_INST	HS_INST_CAPS
HS_INST_DD	HS_INST_INIT
TRUSTED_SERVERS	TS_PITR_CHECK
TS_PITR_OBJECTS_TO_BE_DROPPED	USER_ALL_TABLES
USER_COLL_TYPES	USER_IND_PARTITIONS
USER_LIBRARIES	USER_LOBS
USER_METHOD_PARAMS	USER_METHOD_RESULTS
USER_NESTED_TABLES	USER_OBJECT_TABLES
USER_PART_COL_STATISTICS	USER_PART_HISTOGRAMS
USER_PART_INDEXES	USER_PART_KEY_COLUMNS
USER_PART_TABLES	USER_PASSWORD_LIMITS
USER_QUEUE_TABLES	USER_QUEUES
USER_REFS	USER_REGISTERED_SNAPSHOTS
USER_REPCOLUMN	USER_REPGENOBJECTS
USER_SNAPSHOT_REFRESH_TIMES	USER_TAB_COL_STATISTICS
USER_TAB_HISTOGRAMS	USER_TAB_PARTITIONS
USER_TYPE_ATTRS	USER_TYPE_METHODS
USER_TYPES	

リリース 8.1 で追加された静的データ・ディクショナリ・ビュー

リリース 8.1 では、次の静的データ・ディクショナリ・ビューが追加されました。

ALL_ASSOCIATIONS	ALL_CONTEXT
ALL_DIM_ATTRIBUTES	ALL_DIM_CHILD_OF
ALL_DIM_HIERARCHIES	ALL_DIM_JOIN_KEY
ALL_DIM_LEVEL_KEY	ALL_DIM_LEVELS
ALL_DIMENSIONS	ALL_IND_EXPRESSIONS
ALL_IND_SUBPARTITIONS	ALL_INDEXTYPE_OPERATORS
ALL_INDEXTYPES	ALL_INTERNAL_TRIGGERS
ALL_LOB_PARTITIONS	ALL_LOB_SUBPARTITIONS
ALL_MVIEW_AGGREGATES	ALL_MVIEW_ANALYSIS
ALL_MVIEW_DETAIL_RELATIONS	ALL_MVIEW_JOINS
ALL_MVIEW_KEYS	ALL_OPANCILLARY
ALL_OPARGUMENTS	ALL_OPBINDINGS
ALL_OPERATORS	ALL_PART_LOBS
ALL_PARTIAL_DROP_TABS	ALL_POLICIES
ALL_QUEUE_TABLES	ALL_QUEUES
ALL_REFRESH_DEPENDENCIES	ALL_REPCAT_REFRESH_TEMPLATES
ALL_REPCAT_TEMPLATE_OBJECTS	ALL_REPCAT_TEMPLATE_PARMS
ALL_REPCAT_TEMPLATE_SITES	ALL_REPCAT_USER_AUTHORIZATIONS
ALL_REPCAT_USER_PARM_VALUES	ALL_SUBPART_COL_STATISTICS
ALL_SUBPART_HISTOGRAMS	ALL_SUBPART_KEY_COLUMNS
ALL_SUMDELTA	ALL_SUMMARIES
ALL_SUMMARY_AGGREGATES	ALL_SUMMARY_DETAIL_TABLES
ALL_SUMMARY_JOINS	ALL_SUMMARY_KEYS
ALL_TAB_MODIFICATIONS	ALL_TAB_SUBPARTITIONS
ALL_UNUSED_COL_TABS	ALL_USTATS
ALL_VARRAYS	DATABASE_COMPATIBLE_LEVEL
DBA_ASSOCIATIONS	DBA_CONTEXT
DBA_DIM_ATTRIBUTES	DBA_DIM_CHILD_OF
DBA_DIM_HIERARCHIES	DBA_DIM_JOIN_KEY

DBA_DIM_LEVEL_KEY	DBA_DIM_LEVELS
DBA_DIMENSIONS	DBA_IND_EXPRESSIONS
DBA_IND_SUBPARTITIONS	DBA_INDEXTYPE_OPERATORS
DBA_INDEXTYPES	DBA_INTERNAL_TRIGGERS
DBA_LOB_PARTITIONS	DBA_LOB_SUBPARTITIONS
DBA_MVIEW_AGGREGATES	DBA_MVIEW_ANALYSIS
DBA_MVIEW_DETAIL_RELATIONS	DBA_MVIEW_JOINS
DBA_MVIEW_KEYS	DBA_OPANCILLARY
DBA_OPARGUMENTS	DBA_OPBINDINGS
DBA_OPERATORS	DBA_OUTLINE_HINTS
DBA_OUTLINES	DBA_PART_LOBS
DBA_PARTIAL_DROP_TABS	DBA_POLICIES
DBA_REPCAT_REFRESH_TEMPLATES	DBA_REPCAT_TEMPLATE_OBJECTS
DBA_REPCAT_TEMPLATE_PARS	DBA_REPCAT_TEMPLATE_SITES
DBA_REPCAT_USER_AUTHORIZATIONS	DBA_REPCAT_USER_PARM_VALUES
DBA_RSRC_CONSUMER_GROUP_PRIVS	DBA_RSRC_CONSUMER_GROUPS
DBA_RSRC_MANAGER_SYSTEM_PRIVS	DBA_RSRC_PLAN_DIRECTIVES
DBA_RSRC_PLANS	DBA_RULESETS
DBA_SUBPART_COL_STATISTICS	DBA_SUBPART_HISTOGRAMS
DBA_SUBPART_KEY_COLUMNS	DBA_SUMMARIES
DBA_SUMMARY_DETAIL_TABLES	DBA_TAB_MODIFICATIONS
DBA_TAB_SUBPARTITIONS	DBA_TEMP_FILES
DBA_UNUSED_COL_TABS	DBA_USTATS
DBA_VARRAYS	PLUGGABLE_SET_CHECK
PROXY_USERS	QUEUE_PRIVILEGES
SESSION_CONTEXT	STRADDLING_TS_OBJECTS
UNI_PLUGGABLE_SET_CHECK	USER_ASSOCIATIONS
USER_DIM_ATTRIBUTES	USER_DIM_CHILD_OF
USER_DIM_HIERARCHIES	USER_DIM_JOIN_KEY
USER_DIM_LEVEL_KEY	USER_DIM_LEVELS
USER_DIMENSIONS	USER_IND_EXPRESSIONS
USER_IND_SUBPARTITIONS	USER_INDEXTYPE_OPERATORS

USER_INDEXTYPES	USER_INTERNAL_TRIGGERS
USER_LOB_PARTITIONS	USER_LOB_SUBPARTITIONS
USER_MVIEW_AGGREGATES	USER_MVIEW_ANALYSIS
USER_MVIEW_DETAIL_RELATIONS	USER_MVIEW_JOINS
USER_MVIEW_KEYS	USER_OPANCILLARY
USER_OPARGUMENTS	USER_OPBINDINGS
USER_OPERATORS	USER_OUTLINE_HINTS
USER_OUTLINES	USER_PART_LOBS
USER_PARTIAL_DROP_TABS	USER_POLICIES
USER_QUEUE_SCHEDULES	USER_REPCAT_REFRESH_TEMPLATES
USER_REPCAT_TEMPLATE_OBJECTS	USER_REPCAT_TEMPLATE_PARMs
USER_REPCAT_TEMPLATE_SITES	USER_REPCAT_USER_AUTHORIZATION
USER_REPCAT_USER_PARM_VALUES	USER_RSRC_CONSUMER_GROUP_PRIVS
USER_RSRC_MANAGER_SYSTEM_PRIVS	USER_RULESETS
USER_SUBPART_COL_STATISTICS	USER_SUBPART_HISTOGRAMS
USER_SUBPART_KEY_COLUMNS	USER_SUMMARIES
USER_SUMMARY_DETAIL_TABLES	USER_TAB_MODIFICATIONS
USER_TAB_SUBPARTITIONS	USER_UNUSED_COL_TABS
USER_USTATS	USER_VARRAYS

バージョン 8 で追加された列と静的データ・ディクショナリ・ビュー

新しい列は、次の項で示す静的データ・ディクショナリ・ビューへ追加されました。

参照： これらのビューの詳細は『Oracle8i リファレンス・マニュアル』を参照してください。

リリース 8.0 で追加された列と静的データ・ディクショナリ・ビュー

リリース 8.0 で、新しい列は次の静的データ・ディクショナリ・ビューに追加されました。

ALL_ARGUMENTS	ALL_CLUSTERS
ALL_CONSTRAINTS	ALL_DEPENDENCIES
ALL_INDEXES	ALL_OBJECTS
ALL_REFRESH	ALL_REFRESH_CHILDREN
ALL_REPOBJECT	ALL_REPPRIORITY
ALL_REPSHEMA	ALL_REPSITES
ALL_SNAPSHOTS	ALL_TABLES
ALL_TAB_COLUMNS	ALL_UPDATABLE_COLUMNS
ALL_VIEWS	COL
DBA_CLUSTERS	DBA_CONSTRAINTS
DBA_DATA_FILES	DBA_DEPENDENCIES
DBA_EXTENTS	DBA_FREE_SPACE
DBA_INDEXES	DBA_OBJECTS
DBA_OBJ_AUDIT_OPTS	DBA_PROFILES
DBA_REFRESH	DBA_REFRESH_CHILDREN
DBA_REPOBJECT	DBA_REPPRIORITY
DBA_RGROUP	DBA_ROLLBACK_SEGS
DBA_SEGMENTS	DBA_SNAPSHOTS
DBA_SNAPSHOT_LOGS	DBA_TABLES
DBA_TABLESPACES	DBA_TAB_COLUMNS
DBA_UPDATABLE_COLUMNS	DBA_USERS
DBA_VIEWS	DEFERROR
DEFTRANDEST	FILE_PING

バージョン 8 で追加された列と静的データ・ディクショナリ・ビュー

INDEX_STATS	SYS_OBJECTS
USER_ARGUMENTS	USER_CLUSTERS
USER_CONSTRAINTS	USER_DEPENDENCIES
USER_EXTENTS	USER_FREE_SPACE
USER_INDEXES	USER_OBJECTS
USER_OBJ_AUDIT_OPTS	USER_REFRESH
USER_REFRESH_CHILDREN	USER_REPOBJECT
USER_REPPRIORITY	USER_REPSHEMA
USER_REPSITES	USER_SEGMENTS
USER_SNAPSHOTS	USER_SNAPSHOT_LOGS
USER_TAB_COLUMNS	USER_UPDATABLE_COLUMNS
USER_TABLES	USER_TABLESPACES
USER_USERS	USER_VIEWS

リリース 8.1 で追加された列と静的データ・ディクショナリ・ビュー

リリース 8.1 で、新しい列は、次の静的データ・ディクショナリ・ビューに追加されました。

ALL_ALL_TABLES	ALL_CLUSTERS
ALL_COLL_TYPES	ALL_CONSTRAINTS
ALL_IND_PARTITIONS	ALL_NESTED_TABLES
ALL_INDEXES	ALL_IND_COLUMNS
ALL_IND_PARTITIONS	ALL_IND_SUBPARTITIONS
ALL_MVIEW_ANALYSIS	ALL_OBJECTS
ALL_OBJECT_TABLES	ALL_PART_COL_STATISTICS
ALL_PART_INDEXES	ALL_PART_TABLES
ALL_PROBE_OBJECTS	ALL_REFS
ALL_REPCAT	ALL_REPCAT_TEMPLATE_SITES
ALL_REPCOLUMN	ALL_REPGROUP
ALL_REPOBJECT	ALL_SNAPSHOTS
ALL_TAB_COL_STATISTICS	ALL_TAB_PARTITIONS
ALL_TABLES	ALL_TAB_COLUMNS
ALL_TRIGGERS	DBA_ALL_TABLES
DBA_CLUSTERS	DBA_COLL_TYPES
DBA_CONSTRAINTS	DBA_CONTEXT
DBA_DATA_FILES	DBA_IND_PARTITIONS
DBA_JOBS	DBA_INDEXES
DBA_IND_COLUMNS	DBA_IND_PARTITIONS
DBA_IND_SUBPARTITIONS	DBA_JOBS_RUNNING
DBA_MVIEW_ANALYSIS	DBA_NESTED_TABLES
DBA_OBJECTS	DBA_OBJECT_TABLES
DBA_OUTLINE_HINTS	DBA_PART_COL_STATISTICS
DBA_PART_INDEXES	DBA_PART_TABLES
DBA_PRIV_AUDIT_OPTS	DBA_QUEUE_SCHEDULES
DBA_QUEUE_TABLES	DBA_REFS
DBA_REGISTERED_SNAPSHOT_GROUPS	DBA_REPCAT
DBA_REPCAT_TEMPLATE_SITES	DBA_REPCOLUMN

DBA_REPGROUP	DBA_REPOBJECT
DBA_SNAPSHOTS	DBA_STMT_AUDIT_OPTS
DBA_TAB_COLUMNS	DBA_TAB_COL_STATISTICS
DBA_TAB_PARTITIONS	DBA_TRIGGERS
DBA_TABLES	DBA_TABLESPACES
DBA_USERS	INDEX_STATS
REPCAT_REPCAT	REPCAT_REPOBJECTS
USER_ALL_TABLES	USER_CLUSTERS
USER_COLL_TYPES	USER_CONSTRAINTS
USER_IND_PARTITIONS	USER_IND_SUBPARTITIONS
USER_JOBS	USER_INDEXES
USER_IND_COLUMNS	USER_MVIEW_ANALYSIS
USER_NESTED_TABLES	USER_OBJECTS
USER_OBJECT_TABLES	USER_OUTLINE_HINTS
USER_PART_COL_STATISTICS	USER_PART_INDEXES
USER_PART_TABLES	USER_QUEUE_TABLES
USER_REFS	USER_REPCAT
USER_REPCAT_TEMPLATE_SITES	USER_REPCOLUMN
USER_REPGROUP	USER_REPOBJECT
USER_SNAPSHOTS	USER_TABLES
USER_TAB_COL_STATISTICS	USER_TAB_PARTITIONS
USER_TABLESPACES	USER_TAB_COLUMNS
USER_TRIGGERS	USER_USERS

バージョン 8 で削除された列と静的データ・ディクショナリ・ビュー

次の項で示す列は、バージョン 8 で削除されました。アプリケーションが、次に示す列を 1 つ以上必要とする場合、それに応じてアプリケーションを変更します。

リリース 8.0 で削除された列と静的データ・ディクショナリ・ビュー

リリース 8.0 では、表 C-1 で示す列は削除されました。

表 C-1 リリース 8.0 で削除された列と静的データ・ディクショナリ・ビュー

静的データ・ディクショナリ・ビュー	削除された列
DEFCALLDEST	DEFERRED_TRAN_DB
DEFERROR	DEFERRED_TRAN_DB ERROR_TIME
DEFTRAN	COMMIT_COMMENT DEFERRED_TRAN_DB DESTINATION_LIST ORIGIN_TRAN_DB ORIGIN_TRAN_ID ORIGIN_USER
DEFTRANDEST	DEFERRED_TRAN_DB

リリース 8.1 で削除された列と静的データ・ディクショナリ・ビュー

リリース 8.1 では、表 C-2 で示す列は削除されました。

表 C-2 リリース 8.1 で削除された列と静的データ・ディクショナリ・ビュー

静的データ・ディクショナリ・ビュー	削除された列
DBA_AUDIT_OBJECT USER_AUDIT_OBJECT	OBJECT_LABEL SESSION_LABEL
DBA_AUDIT_SESSION USER_AUDIT_SESSION	SESSION_LABEL
DBA_AUDIT_STATEMENT USER_AUDIT_STATEMENT	SESSION_LABEL
DBA_AUDIT_TRAIL USER_AUDIT_TRAIL	OBJECT_LABEL SESSION_LABEL
DBA_CONTEXT	ATTRIBUTE
ALL_IND_COLUMNS DBA_IND_COLUMNS USER_IND_COLUMNS	COLUMN_EXPRESSION
ALL_JOBS DBA_JOBS USER_JOBS	CLEARANCE_HI CLEARANCE_LO CURRENT_SESSION_LABEL
ALL_REFS DBA_REFS USER_REFS	HAS_REFERENTIAL_CONS REFERENTIAL_CONS_NAME

バージョン 8 で改名された列と静的データ・ディクショナリ・ビュー

次の項で示す列は、バージョン 8 で改名されました。アプリケーションが、次に示す列を 1 つ以上必要とする場合、それに応じてアプリケーションを変更します。

リリース 8.0 で改名された列と静的データ・ディクショナリ・ビュー

リリース 8.0 では、表 C-3 で示す列は改名されました。

表 C-3 リリース 8.0 で改名された列と静的データ・ディクショナリ・ビュー

静的データ・ディクショナリ・ビュー	Oracle7 の列名	リリース 8.0 の列名
DBA_RCHILD	TYPE	TYPE#
DEFSCCHEDULE	LAST_ERROR	LAST_ERROR_NUMBER
	LAST_MSG	LAST_ERROR_MESSAGE

注意： リリース 8.1 には、改名された列を持つ静的データ・ディクショナリ・ビューはありません。

NULL を戻す可能性のある列と静的データ・ディクショナリ・ビュー

リリース 8.1 から、表 C-4 に示す静的データ・ディクショナリ・ビューの列は、NULL を戻す可能性があります。つまり、以前のリリースでは、これらの列は NULL を戻すことはできません。アプリケーションが、以下に示す 1 つ以上の列に対して、NULL 以外の値を必要とする場合、それに応じてアプリケーションを変更する。

表 C-4 リリース 8.1 で NULL を戻す可能性のある列

静的データ・ディクショナリ・ビュー	列	解説
DBA_DATA_FILES	AUTOEXTENSIBLE BLOCKS BYTES INCREMENT_BY MAXBLOCKS MAXBYTES	データ・ファイルがオフラインの場合は判読不可能なので、これらの列は NULL を戻します。
ALL_IND_COLUMNS DBA_IND_COLUMNS USER_IND_COLUMNS	COLUMN_NAME	索引が列についてではなく機能についての場合、この列は NULL を戻しません。この場合、示すべき列がありません。
ALL_IND_PARTITIONS DBA_IND_PARTITIONS USER_IND_PARTITIONS	INITIAL_EXTENT MAX_EXTENT MIN_EXTENT NEXT_EXTENT PCT_INCREASE	索引がコンポジット・メソッドを使用して区分される場合や、デフォルト値が区分に対して特定されていない場合、これらの列は NULL を戻します。
ALL_OBJECT_TABLES DBA_OBJECT_TABLES USER_OBJECT_TABLES	TABLESPACE_NAME	オブジェクト表がパーティション化されている場合、あるいは一時表の場合、この列は NULL を戻します。

表 C-4 リリース 8.1 で NULL を戻す可能性のある列

静的データ・ディクショナリ・ビュー	列	解説
ALL_SEGMENTS DBA_SEGMENTS USER_SEGMENTS	BLOCKS BYTES EXTENTS NEXT_EXTENT PCT_INCREASE	ファイルがオフラインであるためセグメント・ヘッダーが判読不可能の場合、あるいは他に問題がある場合、BLOCKS の列、BYTES の列そして EXTENTS の列は NULL を戻します。 セグメントを格納する表領域が局部的に管理され、AUTOALLOCATE オプションを使用する場合、NEXT_EXTENT や PCT_INCREASE 列は NULL を戻します。それは、システムが範囲を選び、アルゴリズムは NEXT_EXTENT や PCT_INCREASE の点において説明できないからです。
ALL_TAB_PARTITIONS DBA_TAB_PARTITIONS USER_TAB_PARTITIONS	INITIAL_EXTENT MAX_EXTENT MIN_EXTENT NEXT_EXTENT PCT_INCREASE	索引がコンポジット・メソッドを使用してパーティション化される場合や、パーティション化について、デフォルト値が特定されていない場合、これらの列は NULL を戻します。
ALL_TABLESPACES DBA_TABLESPACES USER_TABLESPACES	NEXT_EXTENT PCT_INCREASE	表領域が局部的に管理され、AUTOALLOCATE オプションを使用する場合、これらの列は NULL を戻します。それは、システムが範囲を選び、アルゴリズムは NEXT_EXTENT や PCT_INCREASE の点において説明できないからです。
ALL_TRIGGERS DBA_TRIGGERS USER_TRIGGERS	TABLE_NAME	トリガーがシステム・トリガーの場合、この列は NULL を戻します。この場合、トリガーのオブジェクト基本型は、TABLE や VIEW になるかわりに、SCHEMA や DATABASE になります。

バージョン 8 で廃止された静的データ・ディクショナリ・ビュー

この項で静的データ・ディクショナリ・ビューは、バージョン 8 で廃止されます。

リリース 8.0 で廃止された静的データ・ディクショナリ・ビュー

リリース 8.0 では、次の静的データ・ディクショナリ・ビューは廃止になり、リリース 8.0 以降では使用できません。

ALL_HISTOGRAMS

DBA_HISTOGRAMS

DEFCALL

USER_HISTOGRAMS

リリース 8.1 で廃止された静的データ・ディクショナリ・ビュー

リリース 8.1 では、次の静的データ・ディクショナリ・ビューは廃止になり、リリース 8.1 以降では使用できません。

ALL_LABELS

動的パフォーマンス・ビューの変更

バージョン 8 は新規の動的パフォーマンス・ビュー（VS ビュー）をサポートします。バージョン 8 では、一部の動的パフォーマンス・ビューは変更または廃止されています。この付録は、バージョン 8 の新規の動的パフォーマンス・ビュー、変更された動的パフォーマンス・ビュー、および廃止された動的パフォーマンス・ビューを示します。

この付録では、次のトピックについて説明します。

- [バージョン 8 で追加された動的パフォーマンス・ビュー](#)
- [バージョン 8 で追加された列と動的パフォーマンス・ビュー](#)
- [リリース 8.1 で削除された列と動的パフォーマンス・ビュー](#)
- [バージョン 8 で廃止された動的パフォーマンス・ビュー](#)
- [動的パフォーマンス・ビューの日付列](#)

参照： この付録で示す新規のビューあるいは変更されたビューの詳細、およびこれらのビューにある列の詳細は、『Oracle8i リファレンス・マニュアル』を参照してください。

バージョン 8 で追加された動的パフォーマンス・ビュー

この項で示す動的パフォーマンス・ビューは、バージョン 8 で新しく追加されました。

リリース 8.0 で追加された動的パフォーマンス・ビュー

リリース 8.0 では、次の動的パフォーマンス・ビューが追加されました。

GV\$ACCESS	GV\$ACTIVE_INSTANCES
GV\$AQ	GV\$ARCHIVE
GV\$ARCHIVE_DEST	GV\$ARCHIVED_LOG
GV\$BACKUP	GV\$BACKUP_CORRUPTION
GV\$BACKUP_DATAFILE	GV\$BACKUP_DEVICE
GV\$BACKUP_PIECE	GV\$BACKUP_REDOLOG
GV\$BACKUP_SET	GV\$BGPROCESS
GV\$BH	GV\$BUFFER_POOL
GV\$CACHE	GV\$CIRCUIT
GV\$CLASS_PING	GV\$COMPATIBILITY
GV\$COMPATSEG	GV\$CONTROLFILE
GV\$CONTROLFILE_RECORD_SECTION	GV\$COPY_CORRUPTION
GV\$CURRENT_BUCKET	GV\$DATABASE
GV\$DATAFILE	GV\$DATAFILE_COPY
GV\$DATAFILE_HEADER	GV\$DB_OBJECT_CACHE
GV\$DB_PIPES	GV\$DBFILE
GV\$DBLINK	GV\$DELETED_OBJECT
GV\$DISPATCHER	GV\$DISPATCHER_RATE
GV\$DLM_CONVERT_LOCAL	GV\$DLM_CONVERT_REMOTE
GV\$DLM_LATCH	GV\$DLM_LOCKS
GV\$DLM_MISC	GV\$ENABLEDPRIVS
GV\$ENQUEUE_LOCK	GV\$EVENT_NAME
GV\$EXECUTION	GV\$FALSE_PING
GV\$FILE_PING	GV\$FILESTAT
GV\$FIXED_TABLE	GV\$FIXED_VIEW_DEFINITION
GV\$GLOBAL_TRANSACTION	GV\$INDEXED_FIXED_COLUMN

GV\$INSTANCE	GV\$LATCH
GV\$LATCH_CHILDREN	GV\$LATCH_MISSES
GV\$LATCH_PARENT	GV\$LATCHHOLDER
GV\$LATCHNAME	GV\$LIBRARYCACHE
GV\$LICENSE	GV\$LOADCSTAT
GV\$LOADPSTAT	GV\$LOADTSTAT
GV\$LOCK	GV\$LOCK_ACTIVITY
GV\$LOCK_ELEMENT	GV\$LOCKED_OBJECT
GV\$LOCKS_WITH_COLLISIONS	GV\$LOG
GV\$LOG_HISTORY	GV\$LOGFILE
GV\$LOGHIST	GV\$MLS_PARAMETERS
GV\$MTS	GV\$MYSTAT
GV\$NLS_PARAMETERS	GV\$NLS_VALID_VALUES
GV\$OBJECT_DEPENDENCY	GV\$OFFLINE_RANGE
GV\$OPEN_CURSOR	GV\$OPTION
GV\$PARAMETER	GV\$PING
GV\$PQ_SESSTAT	GV\$PQ_SLAVE
GV\$PQ_SYSSTAT	GV\$PQ_TQSTAT
GV\$PROCESS	GV\$PWFILERS
GV\$QUEUE	GV\$RECENT_BUCKET
GV\$RECOVER_FILE	GV\$RECOVERY_FILE_STATUS
GV\$RECOVERY_LOG	GV\$RECOVERY_PROGRESS
GV\$RECOVERY_STATUS	GV\$REQDIST
GV\$RESOURCE	GV\$RESOURCE_LIMIT
GV\$ROLLSTA	GV\$ROWCACHE
GV\$ROWCACHE_PARENT	GV\$ROWCACHE_SUBORDINATE
GV\$SESS_IO	GV\$SESSION
GV\$SESSION_CONNECT_INFO	GV\$SESSION_CURSOR_CACHE
GV\$SESSION_EVENT	GV\$SESSION_LONGOPS
GV\$SESSION_OBJECT_CACHE	GV\$SESSION_WAIT
GV\$SESSTAT	GV\$SGA
GV\$SGASTAT	GV\$SHARED_POOL_RESERVED

GV\$SHARED_SERVER	GV\$SORT_SEGMENT
GV\$SORT_USAGE	GV\$SQL
GV\$SQL_BIND_DATA	GV\$SQL_BIND_METADATA
GV\$SQL_CURSOR	GV\$SQL_SHARED_MEMORY
GV\$SQLAREA	GV\$SQLTEXT
GV\$SQLTEXT_WITH_NEWLINES	GV\$STATNAME
GV\$SUBCACHE	GV\$SYSSTAT
GV\$SYSTEM_CURSOR_CACHE	GV\$SYSTEM_EVENT
GV\$SYSTEM_PARAMETER	GV\$TABLESPACE
GV\$THREAD	GV\$TIMER
GV\$TRANSACTION	GV\$TRANSACTION_ENQUEUE
GV\$TYPE_SIZE	GV\$VERSION
GV\$WAITSTAT	V\$AQ
V\$ARCHIVE_DEST	V\$ARCHIVED_LOG
V\$BACKUP_CORRUPTION	V\$BACKUP_DATAFILE
V\$BACKUP_DEVICE	V\$BACKUP_PIECE
V\$BACKUP_REDOLOG	V\$BACKUP_SET
V\$BUFFER_POOL	V\$CLASS_PING
V\$CONTROLFILE_RECORD_SECTION	V\$COPY_CORRUPTION
V\$CURRENT_BUCKET	V\$DATAFILE_COPY
V\$DATAFILE_HEADER	V\$DELETED_OBJECT
V\$DISPATCHER_RATE	V\$DLM_CONVERT_LOCAL
V\$DLM_CONVERT_REMOTE	V\$DLM_LATCH
V\$DLM_LOCKS	V\$DLM_MISC
V\$ENQUEUE_LOCK	V\$FILE_PING
V\$GLOBAL_TRANSACTION	V\$LOADPSTAT
V\$OFFLINE_RANGE	V\$RECENT_BUCKET
V\$RECOVERY_PROGRESS	V\$RESOURCE_LIMIT
V\$ROWCACHE_PARENT	V\$ROWCACHE_SUBORDINATE
V\$SESSION_LONGOPS	V\$SESSION_OBJECT_CACHE
V\$SORT_USAGE	V\$SUBCACHE
V\$TABLESPACE	V\$TRANSACTION_ENQUEUE

リリース 8.1 で追加された動的パフォーマンス・ビュー

リリース 8.1 では、次の動的パフォーマンス・ビューが追加されました。

GV\$ARCHIVE_PROCESSES	GV\$BACKUP_ASYNC_IO
GV\$BACKUP_SYNC_IO	GV\$CONTEXT
GV\$DLM_ALL_LOCKS	GV\$DLM_RESS
GV\$FAST_START_SERVERS	GV\$FAST_START_TRANSACTIONS
GV\$GLOBAL_BLOCKED_LOCKS	GV\$HS_AGENT
GV\$HS_SESSION	GV\$INSTANCE_RECOVERY
GV\$LOGMNR_CONTENTS	GV\$LOGMNR_DICTIONARY
GV\$LOGMNR_LOGS	GV\$LOGMNR_PARAMETERS
GV\$PARALLEL_DEGREE_LIMIT_MTH	GV\$PROXY_ARCHIVEDLOG
GV\$PROXY_DATAFILE	GV\$PX_PROCESS
GV\$PX_PROCESS_SYSSTAT	GV\$PX_SESSION
GV\$PX_SESSTAT	GV\$RESERVED_WORDS
GV\$RSRC_CONSUMER_GROUP	GV\$RSRC_CONSUMER_GROUP_CPU_MTH
GV\$RSRC_PLAN	GV\$RSRC_PLAN_CPU_MTH
GV\$TEMP_EXTENT_MAP	GV\$TEMP_EXTENT_POOL
GV\$TEMP_PING	GV\$TEMP_SPACE_HEADER
GV\$TEMPFILE	GV\$TEMPORARY_LOBS
GV\$TEMPSTAT	V\$ARCHIVE_PROCESSES
V\$BACKUP_ASYNC_IO	V\$BACKUP_SYNC_IO
V\$CONTEXT	V\$DLM_ALL_LOCKS
V\$DLM_RESS	V\$FAST_START_SERVERS
V\$FAST_START_TRANSACTIONS	V\$GLOBAL_BLOCKED_LOCKS
V\$HS_AGENT	V\$HS_SESSION
V\$INSTANCE_RECOVERY	V\$LOGMNR_CONTENTS
V\$LOGMNR_DICTIONARY	V\$LOGMNR_LOGS
V\$LOGMNR_PARAMETERS	V\$OBSOLETE_PARAMETER
V\$PARALLEL_DEGREE_LIMIT_MTH	V\$PROXY_ARCHIVEDLOG
V\$PROXY_DATAFILE	V\$PX_PROCESS
V\$PX_PROCESS_SYSSTAT	V\$PX_SESSION

バージョン 8 で追加された動的パフォーマンス・ビュー

V\$PX_SESSTAT	V\$RESERVED_WORDS
V\$RSRC_CONSUMER_GROUP	V\$RSRC_CONSUMER_GROUP_CPU_MTH
V\$RSRC_PLAN	V\$RSRC_PLAN_CPU_MTH
V\$TEMP_EXTENT_MAP	V\$TEMP_EXTENT_POOL
V\$TEMP_PING	V\$TEMP_SPACE_HEADER
V\$TEMPFILE	V\$TEMPORARY_LOBS
V\$TEMPSTAT	

バージョン 8 で改名された動的パフォーマンス・ビュー

この項の動的パフォーマンス・ビューは、バージョン 8 で改名されました。アプリケーションが、次に示すビューを 1 つ以上必要とする場合、それに応じてアプリケーションを変更します。

リリース 8.1 で改名された動的パフォーマンス・ビュー

リリース 8.1 では、次の動的パフォーマンス・ビューが改名されました。

表 D-1 リリース 8.1 で改名された動的パフォーマンス・ビュー

リリース 8.1.3 およびリリース 8.1.4 の名前	リリース 8.1.5 以降の名前
GV\$RECOVERY_SERVERS	GV\$FAST_START_SERVERS
GV\$RECOVERY_TRANSACTIONS	GV\$FAST_START_TRANSACTIONS
GV\$TARGETRBA	GV\$INSTANCE_RECOVERY
V\$RECOVERY_SERVERS	V\$FAST_START_SERVERS
V\$RECOVERY_TRANSACTIONS	V\$FAST_START_TRANSACTIONS
V\$TARGETRBA	V\$INSTANCE_RECOVERY

注意： リリース 8.0. では動的パフォーマンス・ビューは改名されません。

バージョン 8 で追加された列と動的パフォーマンス・ビュー

次の項で示す動的パフォーマンス・ビューに、新しい列が追加されました。

リリース 8.0 で追加された列と動的パフォーマンス・ビュー

リリース 8.0 で、新しい列は次の動的パフォーマンス・ビューに追加されます。

V\$BH	V\$CACHE
V\$CACHE_LOCK	V\$DATABASE
V\$DATAFILE	V\$FALSE_PING
V\$FILESTAT	V\$INSTANCE
V\$LATCH_MISSES	V\$LOADCSTAT
V\$LOADTSTAT	V\$LOG_HISTORY
V\$PING	V\$SESSION
V\$SESSION_EVENT	V\$SGASTAT
V\$SORT_SEGMENT	V\$SQL
V\$THREAD	V\$TRANSACTION

リリース 8.1 で追加された列と動的パフォーマンス・ビュー

リリース 8.1 で、新しい列は次の動的パフォーマンス・ビューに追加されます。

GV\$ARCHIVE_DEST	GV\$AQ
GV\$BACKUP_PIECE	GV\$BH
GV\$CIRCUIT	GV\$DATABASE
GV\$DATAFILE	GV\$DISPATCHER
GV\$DLM_LATCH	GV\$DLM_LOCKS
GV\$INSTANCE	GV\$INSTANCE_RECOVERY
GV\$LOGMNR_CONTENTS	GV\$SESSION
GV\$SESSION_LONGOPS	GV\$SQL
V\$ARCHIVE_DEST	V\$AQ
V\$BACKUP_PIECE	V\$BH
V\$CIRCUIT	V\$DATABASE
V\$DATAFILE	V\$DISPATCHER
V\$DLM_LATCH	V\$DLM_LOCKS
V\$INSTANCE	V\$INSTANCE_RECOVERY
V\$LOGMNR_CONTENTS	V\$SESSION
V\$SESSION_LONGOPS	V\$SQL

注意： GV\$TARGETRBA と V\$TARGETRBA の動的パフォーマンス・ビューは、リリース 8.1.3. に追加しました。リリース 8.1.5 以降では、これらのビューの名前は、それぞれ GV\$INSTANCE_RECOVERY と V\$INSTANCE_RECOVERY に変更されます。先のリストが示すように、リリース 8.1.5 では、新しい名前の下にあるこれらの動的パフォーマンス・ビューに、新しい列が追加されます。

リリース 8.1 で削除された列と動的パフォーマンス・ビュー

リリース 8.1 では、次の項で示す列は削除されました。アプリケーションが、以下に示す列を 1 つ以上必要とする場合、それに応じて、アプリケーションを変更します。

リリース 8.1 で削除された列と動的パフォーマンス・ビュー

リリース 8.1 では、[表 D-2](#) で示す列は削除されました。

表 D-2 リリース 8.1 で削除された列と動的パフォーマンス・ビュー

動的パフォーマンス・ビュー	削除された列
V\$ARCHIVE_DEST	ARCMODE
V\$DLM_LATCH	IMM_GETS LATCH_TYPE TTL_GETS
V\$DLM_LOCKS	RESOURCE_NAME
V\$SESSION_LONGOPS	APPLICATION_DATA_1 APPLICATION_DATA_2 APPLICATION_DATA_3 COMPNAM CURRENT_TIME MSG OBJID OPID STEPID STEPSOFAR STEPTOTAL UPDATE_COUNT

バージョン 8 で廃止された動的パフォーマンス・ビュー

バージョン 8 において、この項にある動的パフォーマンス・ビューは廃止されました。

リリース 8.1 で廃止された動的パフォーマンス・ビュー

リリース 8.1 では、次の動的パフォーマンス・ビューは廃止になり、リリース 8.1 以降では使用できません。

GV\$CURRENT_BUCKET

GV\$RECENT_BUCKET

V\$CURRENT_BUCKET

V\$RECENT_BUCKET

注意： リリース 8.0 では、動的パフォーマンス・ビューは廃止されません。

動的パフォーマンス・ビューの日付列

Oracle7 では、動的パフォーマンス・ビューの日付列はすべて MM/DD/YY HH24:MI:SS という書式の、VARCHAR2(20) 文字列です。バージョン 8 では、新しい日付列はすべて、DATE データ型を使用する実際の DATE 列です。以前の VARCHAR2(20) 文字列とは対照的に、DATE データ型は、次のような利点を提供します。

- すべての日付列が DATE データ型なので、一貫性を確立します。
- SQL および PL/SQL では、日付算術（ソートを含む）を簡易化します。
- NLS_DATE_FORMAT を使用して、日付書式を設定します。
- NLS_DATE_FORMAT を MM/DD/YY HH24:MI:SS へ設定することにより、以前の書式で日付を見せることができます。
- 2 桁の年数値の使用を避け、それによって西暦 2000 年以降の問題が避けることができます。

注意： Oracle7 は、動的パフォーマンス・ビューの VARCHAR(20) データ型を使用して日付を表示しますが、完全には西暦 2000 年問題を避けることができません。Oracle7 は、REDO ログ・ファイルと制御ファイルにある最も近い秒にあわせて時間を格納します。

新しい内部データ型と SQL 関数

この付録では、新規内部データ型とバージョン 8 に追加される新規 SQL 関数を示します。
この付録では、次のトピックについて説明します。

- [新しいリリースで追加された内部データ型](#)
- [新しいリリースで追加された SQL 関数](#)

参照： Oracle の内部データ型および SQL 関数の詳細は、『Oracle8i SQL
リファレンス』を参照してください。

新しいリリースで追加された内部データ型

この項で示す内部データ型は、バージョン 8 で新しく追加されました。これらのデータ型は、SELECT リスト内の関数名として使用される場合があります。ただし、スキーマ (*schema.function*) で修飾されている場合に限りです。次に、その例を示します。

```
select scott.true() ...
```

バージョン 8 では、スキーマ修飾がない場合はエラーを生成しますが、バージョン 7 では、未修飾で使用するエラーを生成せずに機能します。

リリース 8.0 で追加された内部データ型

リリース 8.0 では、次の内部データ型が追加されました。

- NCHAR
- NVARCHAR2
- CLOB
- NCLOB
- BLOB
- BFILE

リリース 8.1 で追加された内部データ型

リリース 8.1 では、次の内部データ型が追加されました。

- UROWID (ユニバーサル ROWID)

新しいリリースで追加された SQL 関数

この項で示す SQL 関数は、バージョン 8 で新しく追加されました。

リリース 8.0 で追加された SQL 関数

リリース 8.0 では、次の SQL 関数が追加されました。

単一行関数

リリース 8.0 では、次の単一行関数が追加されました。

- EMPTY_[B | C]LOB
- BFILENAME
- NLS_CHARSET_DECL_LEN
- NLS_CHARSET_ID
- NLS_CHARSET_NAME

オブジェクト参照関数

リリース 8.0 では、次のオブジェクト参照関数が追加されました。

- DEFEF
- MAKE_REF
- REF
- REFTOHEX
- VALUE

リリース 8.1 で追加された SQL 関数

リリース 8.1 では、次の関数が追加されました。

文字関数

リリース 8.1 では、次の文字関数が追加されました。

- TRIM

変換関数

リリース 8.1 では、次の変換関数が追加されました。

- TO_LOB

集合関数

リリース 8.1 では、次の集合関数が追加されました。

- GROUPING

単一行関数

リリース 8.1 では、次の単一行関数が追加されました。

- SYS_CONTEXT
- SYS_GUID

注意： SYS で始まる関数名は、Oracle が確保しています。関数名は、SYS で始めてはいけません。関数名を SYS で始めると、互換性に問題が生じる可能性があります。

数字

- 2GB を超えるファイル
 - オペレーティング・システム依存性, 8-50
- 2 次索引
 - 索引構成表からの削除, 12-8
- 32 ビットから 64 ビットへの変換「ワード・サイズ」を参照

A

- Ada. 「Ada 用の SQL*Module」を参照
- ALTER DATABASE CONVERT コマンド, 3-4, 3-26
 - 再実行, A-19
- ALTER DATABASE OPEN RESETLOGS コマンド,
3-4, 3-26
- ALTER DATABASE RESET COMPATIBILITY, 8-7
- ALTER TABLE コマンド
 - ビットマップ索引の無効性, 8-25
- ANALYZE TABLE VALIDATE STRUCTURE コマンド
 - リリース 8.1 で変更する, 8-17
- AQ「アドバンスド・キューイング」を参照
- AUDIT_TRAIL 初期化パラメータ
 - 移行, A-3

B

- BUILD DEFERRED 句
 - マテリアライズド・ビューおよびダウングレード,
12-10

C

- CALL 文
 - 互換性, 8-14

- CATALOG5.SQL、Oracle8i で廃止, 8-24
- CATALOG6.SQL、Oracle8i で廃止, 8-24
- CATALOG.SQL スクリプト, 3-4, 3-27, 7-19, 12-49
- CATPROC.SQL スクリプト, 12-49
- CATEXP7.SQL スクリプト, 8-48
 - Oracle7 へのダウングレード, 13-3
- CATNOQUEUE.SQL スクリプト, 7-28
- CATPARR.SQL スクリプト, 3-28, 7-24
- CATREPARR.SQL スクリプト, 12-49
- CATPROC.SQL スクリプト, 3-4, 3-27, 7-19
- CATQUEUE.SQL スクリプト, 7-28
- CATREP.SQL スクリプト, 3-28, 4-17, 7-21, 12-49
- CHECK_ONLY
 - 移行ユーティリティ・オプション, 3-14, 3-18
- CLOB
 - 互換性, 8-25
- COMMIT キーワード
 - Oracle7 と Oracle8i での動作, 8-22
- COMMIT コマンド
 - Server Manager と SQL*Plus との違い, 10-14
- COMPATIBLE 初期化パラメータ, 8-2
 - 8.1.0 設定必要とする機能, 8-9
 - 設定, 8-6
 - 設定の確認, 8-5
 - データベースの構造, 8-3
- CREATE LIBRARY コマンド
 - Server Manager と SQL*Plus との違い, 10-13
- CREATE TABLE
 - 動作の変化
 - Oracle7 から Oracle8i へ, 8-31
- CREATE TYPE コマンド
 - Server Manager と SQL*Plus との違い, 10-13
- C のコール仕様
 - 互換性, 8-14

D

DB_BLOCK_SIZE 初期化パラメータ
移行, 3-6, 4-5

DBMS
ブリコンパイラ・コマンド行オプション, 8-19

DBMS_LOB パッケージ
NOCOPY 構文および互換性, 8-33

DBMS_REPAIR パッケージ
互換性, 8-14, 8-33

DBMS_ROWID パッケージ, 11-3

DBMSLOB.SQL スクリプト
互換性, 8-33

DBNAME
移行ユーティリティ・オプション, 3-14

DEC キーワード
Oracle7 と Oracle8i での動作, 8-22

DEGREE キーワード
PARALLEL 句, 8-35

Developer/2000 アプリケーション
アップグレード, 9-6

DISABLE VALIDATE 制約
互換性, 8-14
ダウングレード, 12-34

DB_DOMAIN の初期化パラメータ
互換性, B-9

DML_LOCKS の初期化パラメータ
互換性, B-8

DTYCHR 型, 11-8

E

EBU
バックアップ管理, 8-40

F

FALSE キーワード
Oracle7 と Oracle8i での動作, 8-22

FAST REFRESH モード
マテリアライズド・ビューおよびダウングレード,
12-10

Forms
Oracle Forms アプリケーションのアップグレード,
9-6

I

INIT.ORA パラメータ「初期化パラメータ」を参照

INSTANCES キーワード
PARALLEL 句からの削除, 8-35

INT キーワード
Oracle7 と Oracle8i での動作, 8-22

J

Java
アップグレード
Java オブジェクトのインポート, 7-36
Java オブジェクトのエクスポート, 7-3
Java オブジェクトの削除, 7-36
互換性, 8-9
ダウングレードのための非互換性の削除, 12-26

L

LARGE_POOL_SIZE 初期化パラメータ
バージョン 8.1 の変更点, 6-3, 7-34

LARGE_POOL_SIZE の初期化パラメータ
パラレル実行の割当て, B-9

LOB
互換性, 8-11, 8-25
索引構成表からの削除, 12-7
ダウングレード, 12-19
パーティション表からの削除, 12-20

M

MIGRATE.BSQ スクリプト, 3-19

MIGRATE ユーザー
回避, 3-10, 4-8

Migration Assistant 「Oracle Data Migration Assistant」
を参照

MULTIPLIER
移行ユーティリティ・オプション, 3-14, A-4

N

NCHAR
Oracle8i での使用, 8-27
移行, 8-27
互換性, 8-27
相互運用性, 8-27

- NCHAR と NLS 環境変数
 - 互換性, 8-28
- NCLOB
 - 互換性, 8-25
- Net8
 - 移行またはアップグレード, 6-4, 8-46
 - サービス命名, 8-47
 - ダウングレード, 12-41
 - 接続ロード・バランシング, 8-47
- NEVER REFRESH モード
 - マテリアライズド・ビューおよびダウングレード, 12-10
- NEW_DBNAME
 - 移行ユーティリティ・オプション, 3-14
- NLS
 - 移行, 8-27
 - 互換性, 8-27
 - 相互運用性, 8-27
- NLS_LANG 環境変数
 - 互換性, 8-28
- NLS_NCHAR
 - 移行ユーティリティ・オプション, 3-14
- NLS および NCHAR 環境変数
 - 互換性, 8-28
- NO_SPACE_CHECK
 - 移行ユーティリティ・オプション, 3-14
- NOCOPY パラメータ引渡しモード
 - 互換性, 8-14
- N-Tier の認証と認可
 - 互換性, 8-13
- NUMERIC キーワード
 - Oracle7 と Oracle8i での動作, 8-22

O

- O7_DICTIONARY_ACCESSIBILITY のパラメータ
 - 互換性, B-8
- O7_DICTIONARY_ACCESSIBILITY パラメータ
 - 互換性, 8-24
- OCI
 - OCIChangePassword コール, 8-37
 - OCISessionBegin コール, 8-37
 - Oracle8i へのアプリケーションのアップグレード, 9-2
 - 互換性, 8-17
 - クライアント・メッセージ, 8-9
 - スレッド・セーフティ, 8-18

- リンク・ライン, 8-18
- 相互運用性, 8-17
 - Oracle7 クライアント, 8-18
 - 共有構造, 8-17
- OFA, 1-5
- OLON コール
 - 廃止, 8-18
- OLQP, 2-17
- OLTP, 2-17
- ORA_NLS32 環境変数
 - 互換性, 8-28
- ORA_NLS33 環境変数, 3-16
 - 互換性, 8-28
- Oracle Data Migration Assistant
 - Oracle Parallel Server
 - アップグレードをサポートしない, 7-6
 - 移行をサポートしない, 2-9, 4-3
 - SYSTEM 表領域に必要な領域, 4-4
 - 移行ユーティリティとの違い, 2-9
 - 概要, 4-3
 - 異なるオペレーティング・システムへの移行, 4-5
 - 実行, 4-16, 7-7, 7-12
 - 使用するキャラクタ・セット, 4-6
 - 長所および短所, 2-8
 - ドキュメント・ロードマップ, 4-2
- Oracle Parallel Server
 - アップグレード, 7-1, 7-24
 - Oracle Data Migration Assistant ではサポートされていない, 7-6
 - 移行, 3-7
 - Oracle Data Migration Assistant でサポートされない, 2-9, 4-3
 - 互換性の要件, 8-34
 - ジョブのインスタンス親和性
 - 互換性, 8-12
 - ダウングレード, 12-49
 - ローリング・アップグレード, 1-5
- Oracle7
 - Oracle8i へのアプリケーションのアップグレード, 9-1
 - 移行のための準備, 3-9, 4-7
 - ダウングレード, 13-1
- Oracle8i
 - 初期化パラメータへの変更, B-1
 - 新規 SQL 関数, E-1
 - 新規内部型, E-1
 - 新機能

- 移行後の追加, 6-7
- Oracle8i の新しい機能
 - 移行後の追加, 6-7
- Oracle コール・インタフェース「OCI」を参照
- Oracle メディア管理 API
 - 互換性, 8-13
 - プロキシコピー要件, 8-44
 - プロキシ・コピーおよびダウングレード, 12-37
- ORLON コール
 - 廃止, 8-18
- OUTLN ユーザー
 - 回避, 3-10, 4-8, 7-4
 - パスワードの変更, 7-35
- OUTLN ユーザーパスワードの変更, 6-4

P

- Parallel Server「Oracle Parallel Server」を参照
- PARALLEL 句
 - DEGREE キーワード, 8-35
 - INSTANCES キーワードの削除, 8-35
- PFILE
 - 移行ユーティリティ・オプション, 3-15
- PL/SQL
 - PLSQL_V2_COMPATIBILITY 初期化パラメータ, 8-20
 - 下位互換性, 8-19
 - ファンクション
 - Oracle8i での非サポート, 8-30
 - 互換性, 8-14, 8-20
 - 相互運用性, 8-20
 - 変数
 - NCHAR および NLS, 8-27
 - モジュール
 - 再コンパイルする, 7-32
- PREBUILT TABLE 句
 - マテリアライズド・ビューおよびダウングレード, 12-10
- Pro*Ada
 - Ada 用 SQL*Module へのアップグレード, 8-19
- Pro*C/C++
 - Oracle7 のインタオペレータ性, 9-5
 - SYSDBA の権限と接続する, 8-18
- Pro*COBOL
 - SYSDBA の権限と接続する, 8-19
- PROPS\$ ビュー
 - NCHAR および NLS, 8-27

- NCHAR キャラクタ・セット, 3-14

R

- R0703040.SQL スクリプト
 - レプリケーション, 3-28, 4-17
- REAL キーワード
 - Oracle7 と Oracle8i での動作, 8-22
- Recovery Manager
 - 8.0.3 にダウングレード
 - パッケージを再インストール, 12-52
 - 互換性, 8-39
 - バックアップ管理, 8-40
- REDO ログ・ファイル
 - 再アーカイブ, 8-43
- REFRESH ON COMMIT モード
 - マテリアライズド・ビューおよびダウングレード, 12-9
- REMOTE_LOGIN_PASSWORDFILE 初期化パラメータ
 - 移行, A-18
- ROWID
 - DBMS_ROWID 互換性パッケージ, 11-3
 - Oracle8i 形式への変換, 11-3
 - 例, 11-5
 - 移行
 - 質問および回答, 11-7
 - 互換性、クライアント・アクセス, 11-7
 - 索引, 11-8
 - スナップショット・リフレッシュ, 11-6

S

- SAVEPOINT キーワード
 - Oracle7 と Oracle8i での動作, 8-22
- Server Manager
 - SQL*Plus との違い
 - COMMIT コマンド, 10-14
 - CREATE LIBRARY コマンド, 10-13
 - CREATE TYPE コマンド, 10-13
 - アンバサンド, 10-12
 - 起動, 10-2
 - 構文, 10-7
 - コマンド, 10-3
 - コメント, 10-7
 - ハイフン継続文字, 10-10
 - ブランク行, 10-10
 - SQL*Plus への移行, 10-1

- SET COMPATIBILITY コマンド
 - SQL*Plus スクリプト, 9-6
- SET_SESSION_LONGOPS プロシージャ
 - 構文の変更, 8-33
- SHARED_POOL_SIZE 初期化パラメータ
 - バージョン 8.1 の変更点, 6-3, 7-34
- SHARED_POOL_SIZE の初期化パラメータ
 - パラレル実行の割当て, B-9
- SPOOL
 - 移行ユーティリティ・オプション, 3-15
- PL/SQL
 - ダウングレードのための非互換性の削除, 12-25
- SQL
 - 互換性, 8-14
 - ダウングレードのための非互換性の削除, 12-25
- SQL*Module
 - Ada 用, 8-19
- SQL*Net
 - Net8 への移行, 6-4, 8-46
 - Oracle8i での使用, 9-2
 - SQL*Net V1
 - Oracle8i では使用できない, 9-2
 - V1 から V2 へのアップグレード, 8-46
 - 再リンク, 9-2
- SQL*Net での再リンク, 9-2
- SQL*Plus
 - Server Manager から SQL*Plus への移行, 10-1
 - Server Manager との違い
 - COMMIT コマンド, 10-14
 - CREATE LIBRARY コマンド, 10-13
 - CREATE TYPE コマンド, 10-13
 - アンパサンド, 10-12
 - 起動, 10-2
 - 構文, 10-7
 - コマンド, 10-3
 - コメント, 10-7
 - ハイフン継続文字, 10-10
 - ブランク行, 10-10
- SQL*Plus スクリプト
 - アップグレード, 9-6
- SQLLIB コール
 - 再リンク, 9-5
- SQL 関数
 - Oracle8i で新規, E-3
- SQL スクリプト
 - Server Manager から SQL*Plus への移行, 10-1
- SYSDBA

- Pro*C/C++ で接続, 8-18
- Pro*COBOL で接続する, 8-19
- SYSTEM 表領域
 - Oracle Data Migration Assistant, 4-4
 - 移行ユーティリティ, 2-6, 3-6
- 領域
 - 移行用の領域不足, A-2
- SYSTEM ロールバック・セグメント OPTIMAL 設定
 - 移行, A-4
- SYS スキーマ
 - ユーザー作成オブジェクト, 8-24

T

- TP モニター, 2-17
- TRUE キーワード
 - Oracle7 と Oracle8i での動作, 8-22
- Trusted Oracle
 - エクスポート / インポート, 5-3
- Tuxedo アプリケーション
 - Oracle8iXA ライブラリとの互換性, 8-17
- TWO_TASK 環境変数, 3-16

U

- U0703040.SQL スクリプト, 3-27
- UROWID
 - 互換性, 8-11
 - 相互運用性, 8-26
 - ダウングレード, 12-17
- UTL_REF パッケージ
 - ダウングレード後の再インストール, 12-51
 - ダウングレード, 12-33
- UTLCHAIN1.SQL スクリプト, 8-31
 - 互換性, 8-14
- UTLCONST.SQL スクリプト, 6-2, 7-33
- UTLDST.SQL, 8-6, 12-4
- UTLEXCPT1.SQL スクリプト, 8-31
 - 互換性, 8-14
- utljavarm.sql スクリプト, 7-36
- UTLRP.SQL スクリプト, 3-28, 4-18, 7-32

V

- VALUES ビュー
 - NCHAR および NLS, 8-27
- VARRAY

互換性, 8-12
非互換性の削除, 12-24

X

XA ライブラリ
互換性, 8-17

あ

アーカイブする
エラー検出動作, 8-43
アーカイブ・ログの宛先のパラメータ
リリース 8.1 の新規, B-13
アクティブ・トランザクション
読取り専用表領域, 8-23
アップグレード
定義, 1-3
パス, 7-2
ダウングレード
定義, 1-3
アップグレード
Java オブジェクトのインポート, 7-36
Java オブジェクトのエクスポート, 7-3
Java オブジェクトの削除, 7-36
Oracle Data Migration Assistant の使用, 7-7
Oracle Forms アプリケーション, 9-6
Oracle Parallel Server, 7-1, 7-24
ローリング・アップグレード, 1-5
OUTLN ユーザー、回避, 7-4
SQL*Plus スクリプト, 9-6
アップグレード処理後, 7-33
アドバンスト・キューイング, 7-26
アドバンスト・レプリケーション, 7-21
アプリケーション, 9-1
キュー表, 7-29
サポートされているリソース, 7-2
手動, 7-13
初期化パラメータ, 7-15
スクリプト, 7-18
CATALOG.SQL, 7-19
CATPARR.SQL, 7-24
CATPROC.SQL, 7-19
CATREP.SQL, 7-21
UTLCONST.SQL スクリプト, 7-33
UTLRP.SQL, 7-32
再実行, 7-19

実行中のエラー, 1-3
ステップ, 7-3
スナップショット, 7-25
特定のコンポーネント, 7-21
パラレル実行, 7-34
プリコンパイラ・アプリケーション, 9-4
ユーザー定義のデータ型, 7-30
リカバリ・カタログ, 7-30
アドバンスト・キューイング
アップグレード, 7-26
オブジェクトおよびシステム・レベル権限の取消
し, 12-30
拡張アドレス・フィールド, 7-26
キュー表のダウングレード, 12-31
互換性, 8-14, 8-32
権限, 8-32
メッセージ・ストリーム, 8-32
ルール・ベースの加入, 8-32
スクリプト
CATNOQUEUE.SQL, 7-28
CATQUEUE.SQL, 7-28
相互運用性, 8-32
送信者の ID 列, 8-32
伝播の削除, 12-32
アドバンスト・レプリケーション
アップグレード, 7-21
移行, 3-7
互換性, 8-13, 8-45
ダウングレード, 12-49
～の後再生成, 12-51
環境の準備, 12-38
アプリケーション
相互運用性, 8-16
プリコンパイラ
相互運用性, 8-18
OCI
互換性, 8-17
相互運用性, 8-17
Oracle8i の物理 ROWID, 11-2
PL/SQL
互換性, 8-20
相互運用性, 8-20
Tuxedo アプリケーション
互換性, 8-17
アップグレード, 9-1
互換性, 8-16
索引構成表

- 互換性, 8-16
- 物理的な ROWID と UROWID, 8-16
- ブリコンパイラ
 - 互換性, 8-18
- アプリケーション開発者
 - 移行における役割, 1-10
- アプリケーション・コンテキスト
 - 互換性, 8-13
 - ダウングレード, 12-36
- 安定性の計画
 - ダウングレード, 12-36

い

異機種間サービス

- エージェント
 - 互換性, 8-45
 - 相互運用性, 8-45
 - マルチスレッド, 8-45

移行

- Oracle Parallel Server
 - ローリング・アップグレード, 1-5
- ALTER DATABASE CONVERT コマンド, 3-4, 3-26
 - 再実行, A-19
- ALTER DATABASE OPEN RESETLOGS コマンド, 3-4, 3-26
- AUDIT_TRAIL 初期化パラメータ, A-3
- MIGRATE.BSQ スクリプト, 3-19
- MIGRATE ユーザー、回避, 3-10, 4-8
- NCHAR および NLS, 8-27
- Oracle Data Migration Assistant, 2-8
- Oracle Parallel Server, 3-7
- Oracle7 ソース・データベースの準備, 3-9, 4-7
- OUTLN ユーザー、回避, 3-10, 4-8
- Server Manager から SQL*Plus へ, 10-1
- SYSTEM ロールバック・セグメント OPTIMAL 設定, A-4
- 新しい管理手順, 6-7
- アドバンスト・レプリケーション, 3-7
- アプリケーション開発者の役割, 1-10
- 移行後のチューニング, 6-6
- 移行処理後, 6-1, 10-1
- 移行の準備, 2-2
- 移行のステップ, 1-6
- 移行ユーティリティ, 2-6
- 一時表領域, A-2

- エクスポート / インポート, 2-10
 - ステップ, 5-2
- オフライン・データファイル, 3-4
- オフラインの表領域, 4-8, 3-10
- キャラクタ・セット, 3-8, 4-6
- 共通の問題の回避, 2-18
- システム要件, 2-15
- 初期化パラメータ, 3-24, 4-10, B-1
- スクリプト
 - CATALOG.SQL, 3-4, 3-27
 - CATPARR.SQL, 3-28
 - CATPROC.SQL, 3-4, 3-27
 - CATREP.SQL, 3-28, 4-17
 - R0703040.SQL, 3-28, 4-17
 - U0703040.SQL, 3-27
 - UTLRP.SQL, 3-28, 4-18
 - 再実行, 3-27, 4-17
 - 実行中のエラー, 1-3
- 制御ファイル, 3-22
- 中止, 3-30, 4-20
- データのコピー, 2-14
- データベース管理者の役割, 1-10
- テスト, 2-19
- テストの結果, 6-6
- トラブルシューティング, A-1
 - ALTER DATABASE CONVERT コマンド, A-16
 - AUDIT_TRAIL 初期化パラメータ, A-3
 - MULTIPLIER オプション, A-4
 - NOMOUNT データベース・スタート・モード, A-17
 - OPTIMAL 設定, A-4
 - Oracle Data Migration Assistant の実行, A-2
 - Oracle7 制御ファイル, A-16
 - SYSTEM 表領域, A-2
 - 移行ユーティリティのエラー・メッセージ, A-5
 - 移行ユーティリティの実行, A-2
 - 一時表領域, A-2
 - データファイルのバージョン整合性, A-20
 - データベース名が合致しない, A-19
 - パスワード・ファイル, A-18
 - 不明変換ファイル, A-17
- 排他パスワード・ファイル, A-18
- バックアップ計画, 2-18
- パラレル実行, 6-3
- ブロック・サイズ最小値, 3-6, 4-5
- 方法の選択, 2-2, 2-3
- メモリー要件, 2-15, 3-6, 4-4

- 用語, 1-2
- 読取り専用表領域, 3-4
- 領域要件, 2-15, 3-6, 4-4
- ロールバック・セグメント, 3-4
- 移行ユーティリティ
 - 長所および短所, 2-6
- 移行
 - SYSTEM 表領域, A-2
- 移行時のエラー「トラブルシューティング」を参照
- 移行ユーティリティ
 - Oracle Data Migration Assistant との違い, 2-9
 - SYSTEM 表領域, 2-6
 - 使用, 3-16
 - MULTIPLIER オプション, A-4
 - SYSTEM 表領域に必要な領域, 3-6
 - エラーとメッセージ, A-5
 - オプション
 - CHECK_ONLY, 3-14, 3-18
 - DBNAME, 3-14
 - MULTIPLIER, 3-14
 - NEW_DBNAME, 3-14
 - NLS_NCHAR, 3-14
 - NO_SPACE_CHECK, 3-14
 - PFILE, 3-15
 - SPOOL, 3-15
 - 概要, 3-3
 - 異なるオペレーティング・システムへの移行, 3-7
 - コマンド行オプション, 3-14
 - 使用するキャラクタ・セット, 3-8
 - ドキュメント・ロードマップ, 3-2
 - 必要な権限, 3-16
- 移行ユーティリティのパラメータ「コマンド行オプション」を参照
- 一時表
 - 互換性, 8-10
 - ダウングレードの削除, 12-6
- 一時表領域
 - 領域
 - 移行用の領域不足, A-2
- 一括バインド
 - 互換性, 8-14
- 移動可能な表領域
 - 互換性, 8-9
- インストール
 - Oracle ソフトウェア リリース 8.1, 4-11, 5-3
- インストールセッション
 - Oracle8.1 の Oracle ソフトウェア, 3-12

- インポート要件
 - 移行, 2-10
 - データ定義変換, 2-13
- インポート・ユーティリティ
 - 移行要件, 5-2
 - 移行, 5-1

え

- エージェント自動登録
 - 互換性, 8-13
- エクスポート / インポート
 - Trusted Oracle, 5-3
 - 移行されたデータベースへの影響, 2-11
 - 移行の基本ステップ, 5-2
 - 移行の時間要件, 2-13
 - 移行の制約, 2-12
 - 移行の利点, 2-12
 - 互換性, 8-48
 - 使用した移行, 5-3
 - スクリプト
 - CATEXP7.SQL, 8-48
 - ダイレクトおよび従来型パス, 8-48
 - 長所および短所, 2-10
 - データ定義変換, 2-13
 - 非互換データ, 8-49
- エクスポート・ユーティリティ
 - 移行, 2-10
 - CATEXP7.SQL を使用する Oracle8i, 13-3
 - 移行, 5-1
 - 移行要件, 5-2
- 演算子
 - 削除, 12-14
- エンタープライズ・ディレクトリ・サービス
 - 相互運用性, 8-38

お

- 大きいファイル
 - オペレーティング・システム依存性, 8-50
- オブジェクト識別子
 - ユーザー定義
 - ダウングレード, 12-22
- オブジェクト表
 - パーティション化
 - ダウングレード, 12-16
- オプション

- 移行ユーティリティ用, 3-14
- 削除, 1-6
- オブティマイザ
 - 計画の安定性
 - 互換性, 8-14
 - ダウングレード, 12-35
- オフライン・データファイル
 - 移行, 3-4
- オペレーティング・システム
 - 異なるオペレーティング・システムへの移行, 3-7, 4-5
- オンライン索引再構築
 - 互換性, 8-10
- オンライン読取り専用表領域
 - 互換性, 8-9

か

- カーソル
 - オープン数, 2-17
- 回復
 - 互換性, 8-13, 8-39
- 拡張アドレス・フィールド
 - アドバンスト・キューイング, 7-26
- 拡張索引作成機能
 - ダウングレード, 12-13
- 型
 - 相互参照
 - ダウングレード, 12-24
- 各国文字キャラクタ・セット
 - Oracle8i, 8-27
- 可変幅のキャラクタ・セット
 - LOB
 - ダウングレード, 12-19
- 環境変数
 - ORA_NLS32, 8-28
 - ORA_NLS33, 3-16, 8-28
 - TWO_TASK, 3-16
 - アップグレードに必要, 7-16
 - 移行に必要, 3-16, 3-21
 - 互換性
 - NCHAR および NLS, 8-28
- 関数
 - SQL
 - Oracle8i で新規, E-3
- 管理手順
 - 移行後の新しい手順, 6-7

き

- キー圧縮
 - ~ の中断, 12-6
- CHARACTER キーワード
 - Oracle7 と Oracle8i での動作, 8-22
- キーワード
 - 動作の違い
 - Oracle7 と Oracle8i, 8-22
- 機能
 - 新機能, 8-9
- 機能に基づいた索引
 - 互換性, 8-10
- 逆方向移行
 - 未サポート, 13-2
- キャラクタ・セット
 - 移行ユーティリティ, 3-8
 - Oracle Data Migration Assistant, 4-6
 - 可変幅
 - CLOB と NCLOB, 8-25
 - データベースの移行, 3-8, 4-6
- キュー表
 - アップグレード, 7-29
 - ダウングレード, 12-31
- 共有構造
 - 相互運用性, 8-17
- 局所的に管理された表領域
 - 互換性, 8-9

く

- 空間
 - 互換性, 8-15
- 具体化ビュー
 - 互換性, 8-13
- クラスタ表
 - データのコピー, 2-15

け

- 計画の安定性
 - 互換性, 8-14
- 結合
 - ダウングレード, 12-35

こ

高速スタート・パラレル・リカバリ

互換性, 8-42

高速スタート・ロールバック

互換性, 8-13, 8-42

SQLLIB コール

Oracle8i に対して動作する, 9-5

互換性, 8-1

8.1.0 互換性レベル

必要とする機能, 8-9

ALTER DATABASE RESET COMPATIBILITY, 8-7

ANALYZE TABLE VALIDATE STRUCTURE コマンド, 8-17

COMPATIBLE 初期化パラメータ, 8-2

CREATE TABLE 動作の変化, 8-31

DBMS_REPAIR パッケージ, 8-14

DISABLE VALIDATE 制約, 8-14

LOB, 8-11, 8-25

CLOB と NCLOB, 8-25

NCHAR および NLS, 8-27

NLS および NCHAR 環境変数, 8-28

OCI, 8-17

スレッド・セーフティ, 8-18

リンク・ライン, 8-18

Oracle Parallel Server

ジョブのインスタンス親和性, 8-12

Oracle メディア管理 API, 8-13

PL/SQL, 8-20

PLSQL_V2_COMPATIBILITY 初期化パラメータ, 8-20

SQL と PL/SQL, 8-14

CALL 文, 8-14

C のコール仕様, 8-14

NOCOPY パラメータ引渡しモード, 8-14

一括バインド, 8-14

システム固有の動的 SQL, 8-14

自立型トランザクション, 8-14

UROWID, 8-11

VARRAY, 8-12

アドバンスド・キューイング, 8-14, 8-32

アドバンスド・レプリケーション, 8-13, 8-45

アプリケーション, 8-16

Tuxedo アプリケーション, 8-17

XA コール, 8-17

索引構成表, 8-16

物理的な ROWID と UROWID, 8-16

アプリケーション・コンテキスト, 8-13

異機種間サービス・エージェント, 8-45

エージェント自動登録, 8-13

エクスポート / インポート, 8-48

オブティマイザ計画の安定性, 8-14

回復, 8-13, 8-39

回復高速スタート・ロールバック, 8-13

拡張可能なオブティマイザ, 8-14

空間, 8-15

具体化ビュー, 8-13

互換性のレベル, 8-5

最適化, 8-34

索引

キー要約, 8-10

サマリー管理, 8-13

視覚イメージの検索, 8-15

従来型パス・エクスポート, 8-48

初期化パラメータ, B-8

スキーマ・オブジェクト, 8-10

一時表, 8-10

オンライン索引再構築, 8-10

機能に基づいた索引, 8-10

コレクション・ロケータ, 8-12

索引, 8-10

索引構成表, 8-10

単一表ハッシュ・クラスタ, 8-10

追加可能な索引作成, 8-10

ネストした表, 8-12

列削除, 8-10

スキーマ・オブジェクト機能に基づいた索引, 8-10

スクリプト

UTLCHAIN1.SQL, 8-14, 8-31

UTLEXCPT1.SQL, 8-14, 8-31

スタンバイ・データベース, 8-41

ダイレクト・パス・エクスポート, 8-48

データベースのリセット, 12-42

データベース・リソース・マネージャ, 8-14

動的パフォーマンス・ビューの日付列, D-11

トリガー, 8-14

ネストした表, 8-29

パーティション化, 8-11

バックアップ, 8-13, 8-39

パッケージ

DBMS_LOB, 8-33

DBMS_REPAIR, 8-33

非互換性の削除, 12-2

表領域, 8-9

- 移動可能な表領域, 8-9
 - オンライン読取り専用表領域, 8-9
 - 局部的に管理された表領域, 8-9
- ファイン・グレイン・アクセス・コントロール, 8-13
- ブリコンパイラ, 8-18
- ユーザー定義オブジェクト識別子, 8-12
- レプリケーション, 8-44
 - N-Tier の認証と認可, 8-13
- コマンド行オプション
 - 移行ユーティリティ, 3-14
- コメント
 - Server Manager と SQL*Plus との違い, 10-7
- コレクション列
 - ユーザー指定の記憶域
 - 互換性, 8-29
- コレクション・ロケータ
 - 互換性, 8-12, 8-29
- コンボジット・メソッド
 - パーティション化およびダウングレード, 12-17

さ

- サービス命名
 - Net8, 8-47
- 最適化
 - 互換性, 8-34
 - オプティマイザ計画の安定性, 8-14
 - 拡張可能なオプティマイザ, 8-14
- 最適フレキシブル・アーキテクチャ「OFA」を参照
- 索引
 - キー要約, 8-10
 - 互換性, 8-10
 - ドメイン
 - 削除, 12-13
 - ビットマップ, 6-3
- 索引構成表
 - ～ LOB の削除, 12-7
 - ～からのパーティションの削除, 12-14
 - 互換性, 8-10
- 索引タイプ
 - 削除, 12-13
- 索引の再作成
 - 物理 ROWID, 11-8
- 削除, 1-6
- サマリー管理
 - 互換性, 8-13

し

- 視覚イメージの検索
 - 互換性, 8-15
- システム固有の動的 SQL
 - 互換性, 8-14
 - 相互運用性, 8-30
- システム要件
 - 移行に必要な, 2-15
- シャドウ・プロセス
 - オープン・カーソル, 2-17
- 初期化パラメータ
 - COMPATIBLE, 8-2
 - LARGE_POOL_SIZE
 - パラレル実行の割当て, B-9
 - Oracle8i で改名, B-4
 - Oracle8i で追加, B-2
 - Oracle8i での変更, B-1
 - Oracle8i で廃止, B-6
 - SHARED_POOL_SIZE
 - パラレル実行の割当て, B-9
 - アーカイブ・ログの宛先
 - 新規への切替え, B-13
 - 互換性, B-8
 - DB_DOMAIN, B-9
 - DML_LOCKS, B-8
 - O7_DICTIONARY_ACCESSIBILITY, B-8
 - リリース 8.1 の調整, 3-24, 4-10, 7-15
- 自立型トランザクション
 - 互換性, 8-14

す

- スキーマ・オブジェクト
 - 互換性, 8-10
- スクリプト
 - Server Manager から SQL*Plus への移行, 10-1
 - アップグレード, 7-18
 - 再実行, 3-27, 4-17, 7-19, 12-46
 - 実行中のエラー, 1-3
 - ダウングレード, 12-45
- スタンバイ・データベース
 - Oracle8i へ移行, 8-41
 - 互換性, 8-41
- ストアド・アウトライン
 - ダウングレード, 12-36
- スナップショット

- アップグレード, 7-25
- スナップショット「マテリアライズド・ビュー」を参照
- スナップショット・リフレッシュ
 - 物理 ROWID, 11-6
- スレッド・セーフティ
 - 互換性, 8-18

せ

- 制御ファイル
 - 移行の改名または削除, 3-22
- 静的データ・ディクショナリ・ビュー
 - Oracle8i で追加, C-2
 - Oracle8i で廃止, 8-24, C-15
 - Oracle8i で変更
 - NULL を戻す可能性のある列, C-14
 - 列を改名, C-13
 - 列を削除, C-11
 - 列を追加, C-7
- セキュリティ・ポリシー
 - ダウングレード, 12-37
 - ファイン・グレイン・アクセス・コントロール, 12-37
- 接続
 - Net8 でのロード・バランシング, 8-47
 - ローカルとリモート, 2-16
 - マルチスレッド共有サーバーの使用, 2-16

そ

- 相互運用性, 8-1, 8-15
 - NCHAR および NLS, 8-27
 - OCI, 8-17
 - Oracle7 クライアント, 8-18
 - 共有構造, 8-17
 - PL/SQL, 8-20
 - UROWID, 8-26
 - アドバンスド・キューイング, 8-32
 - アプリケーション, 8-16
 - 異機種間サービス・エージェント, 8-45
 - システム固有の動的 SQL, 8-30
 - プリコンパイラ, 8-18
 - ユーザー定義データ型, 8-29
- ソース・データベース
 - 定義, 1-3

た

- ターゲット・データベース
 - 定義, 1-3
- ダウングレード
 - CATALOG.SQL, 12-49
 - CATPROC.SQL, 12-49
 - Oracle Parallel Server, 12-49
 - Oracle7, 13-1
 - CATEXP7.SQL スクリプト, 13-3
 - 代替方法, 13-4
 - アドバンスド・レプリケーション, 12-49
 - 再生成, 12-51
 - ~ の手順, 12-43
 - キュー表, 12-31
 - スクリプト, 12-45
 - 再実行, 12-46
 - 実行中のエラー, 1-3
 - データベース互換性のリセット, 12-42
 - 非互換性の削除, 12-2
 - DISABLE VALIDATE 制約, 12-34
 - Java, 12-26
 - LOB, 12-19
 - Net8 サービス命名, 12-41
 - SETSESSION_LONGOPS の構文変更, 12-33
 - SQL および PL/SQL, 12-25
 - UROWID, 12-17
 - UTL_REF パッケージ, 12-33
 - VARRAY, 12-24
 - アドバンスド・レプリケーション, 12-38
 - アプリケーション・コンテキスト, 12-36
 - 安定性の計画, 12-36
 - 一時表, 12-6
 - 演算子, 12-14
 - オブティマイザ, 12-35
 - 拡張索引作成機能, 12-13
 - 結合, 12-35
 - 索引および索引構成表上におけるキー圧縮, 12-6
 - 索引構成表の 2 次索引, 12-8
 - 索引構成表の LOB, 12-7
 - 索引タイプ, 12-13
 - ストアド・アウトライン, 12-36
 - セキュリティ・ポリシー, 12-37
 - 相互参照型, 12-24
 - 相互参照ビュー, 12-10
 - 単一表ハッシュ・クラスタ, 12-9
 - ディメンション, 12-14

- ドメイン索引, 12-13
- トランスポートされた表領域, 12-5
- トリガー, 12-35
- ネストした表, 12-23
- ネストした表のビュー列におけるトリガー, 12-34
- パーティション化, 12-14
- ビットマップ索引の無効, 12-12
- ファイン・グレイン・アクセス・コントロール, 12-37
- ファンクション索引, 12-13
- 部分的に削除された列, 12-8
- マテリアライズド・ビュー, 12-9
- 未使用の列, 12-8
- ユーザー定義オブジェクト識別子, 12-22
- ユーザー定義データ型, 12-21
- 例外表, 12-18
- 連鎖行表, 12-18
- ローカルで管理される表領域, 12-5
- 古い Oracle8i のリリース, 12-1
- ダウングレードのための非互換性の削除, 12-23
- 単一表ハッシュ・クラスタ
 - 互換性, 8-10
 - 削除, 12-9

ち

- 調整
 - 移行後, 6-6

つ

- 追加可能な索引作成
 - 互換性, 8-10

て

- 定義「用語」を参照
- ディメンション
 - ダウングレード, 12-14
- データ型
 - 内部の
 - Oracle8i で新規, E-2
- データ定義変換
 - インポート要件, 2-13
- データ・ディクショナリ
 - 保護, 8-24

- データのコピー
 - 移行方法, 2-14
 - 大きいクラスタ表, 2-15
 - 領域要件, 2-15
- データファイル
 - 移行におけるオフライン, 3-4
- データベース
 - 移行結果のテスト, 6-6
 - 移行後のチューニング, 6-6
 - 移行のバックアップ, 3-20
 - ダウングレード, 12-43
- データベース管理者
 - 移行における役割, 1-10
- データベース・リソース・マネージャ
 - 互換性, 8-14
- テスト
 - EXPLAIN PLAN, 2-22
 - INTO 句, 2-22
 - 移行結果, 6-6
 - 移行処理, 2-22
 - 移行済みのテスト・データベース, 2-23
 - 移行前後, 2-21
 - 移行のアプリケーション, 2-23
 - 移行の機能, 2-20
 - 移行の計画の作成, 2-19
 - 移行の最小限度, 2-19
 - 移行の統合, 2-20
 - 移行のパフォーマンス, 2-20
 - 移行のボリューム / ロード・ストレス, 2-21
- 伝播
 - 削除, 12-32
- テンプファイル
 - データ・ディクショナリ情報, 8-23

と

- 同時アクセス, 2-16
- 同時ユーザー, 2-17
- 動的パフォーマンス・ビュー
 - Oracle8i で改名する, D-7
 - Oracle8i で廃止する, D-11
 - Oracle8i で変更する
 - 列を追加する, D-8
 - Oracle8i に追加する, D-2
- 日付列
 - 互換性, D-11
- ドキュメント

- Oracle Data Migration Assistant のロードマップ , 4-2
- 移行ユーティリティのためのロードマップ , 3-2
- ドメイン索引
 - 削除 , 12-13
- トラブルシューティング
 - 移行 , A-1
 - ALTER DATABASE CONVERT コマンド , A-16
 - AUDIT_TRAIL 初期化パラメータ , A-3
 - MULTIPLIER オプション , A-4
 - NOMOUNT データベース・スタート・モード , A-17
 - OPTIMAL 設定 , A-4
 - Oracle Data Migration Assistant の実行 , A-2
 - Oracle7 制御ファイル , A-16
 - SYSTEM 表領域 , A-2
 - 移行ユーティリティのエラー・メッセージ , A-5
 - 移行ユーティリティの実行 , A-2
 - 一時表領域 , A-2
 - データファイルのバージョン整合性 , A-20
 - データベース名が合致しない , A-19
 - パスワード・ファイル , A-18
 - 不明変換ファイル , A-17
- トランスポートابل表領域
 - トランスポートابل表領域の削除 , 12-5
- トリガー
 - 互換性 , 8-14
 - ダウングレード中に自動的に削除される , 12-35
 - ネストした表のビュー列における
 - ダウングレード , 12-34

な

- 内部データ型
 - Oracle8i で新規 , E-2

ね

- ネストした表
 - 互換性 , 8-12 , 8-29

は

- バージョン 6
 - エクスポート / インポートによる移行 , 5-2
- パーティション化
 - LOB の削除 , 12-20

- 互換性 , 8-11
- ダウングレード , 12-14
- パスワード管理
 - Oracle8i に必要なアプリケーションの変更 , 8-37
 - 相互運用性 , 8-38
 - パスワードの期限切れ , 8-38
- パスワード・ファイル
 - 移行
 - 排他設定 , A-18
- バックアップ
 - EBU , 8-40
 - Recovery Manager および EBU , 8-40
 - 移行後 , 6-2
 - 移行の計画の準備 , 2-18
 - 移行前 , 3-20
 - 互換性 , 8-13 , 8-39
- パラレル実行
 - 大規模プールからの割当て , B-9
 - 問題の回避 , 6-3 , 7-34

ひ

- 非互換性
 - 削除 , 12-2
 - 定義されたシステム , 12-4
- 非互換性、システム定義 , 8-6
- 日付の制約
 - 無効チェック , 6-2 , 7-33
- ビットマップ索引
 - 無効
 - ダウングレード時 , 12-12
 - 無効化
 - 移行時 , 6-3
 - 無効性 , 8-25
- ビュー
 - 相互参照
 - ダウングレード , 12-10
 - 例 , 12-11
- 表領域
 - オフライン表領域の移行 , 3-10 , 4-8
 - 互換性 , 8-9

ふ

- ファイナ・グレイン・アクセス・コントロール
 - 互換性 , 8-13
- 複数バージョン管理 , 8-50

物理 ROWID 上の索引, 11-8
部分的に削除された列
 削除, 12-8
ブリコンパイラ
 Oracle8i へのアプリケーションのアップグレード, 9-2
 PL/SQL 下位互換性, 8-19
 SQLLIB コール
 再リンク, 9-5
 互換性, 8-18
 相互運用性, 8-18
プロキシ・コピー
 要件, 8-44
プロシージャ・オプション
 移行に必要, 3-9, 4-7
ブロック・サイズ
 DB_BLOCK_SIZE 初期化パラメータ, 3-6, 4-5
 移行最小値, 4-5
 移行のための最小値, 3-6

ま

マテリアライズド・ビュー
 ダウングレード, 12-9
 BUILD DEFERRED 句, 12-10
 FAST REFRESH モード, 12-10
 NEVER REFRESH モード, 12-10
 PREBUILT TABLE 句, 12-10
 REFRESH ON COMMIT モード, 12-9
 非互換性の削除, 12-9
マルチスレッド・サーバー
 共有およびローカル / リモート接続, 2-16
 実行要件, 8-46
 共有, 2-17

み

未使用の列
 削除, 12-8

め

メモリー要件
 移行, 2-15, 3-6, 4-4
 同時アクセス, 2-16

ゆ

ユーザー定義オブジェクト識別子
 ダウングレード, 12-22
ユーザー作成オブジェクト
 SYS スキーマ, 8-24
ユーザー定義オブジェクト識別子
 互換性, 8-12
ユーザー定義データ型
 新しい形式, 8-29
 相互運用性, 8-29
 ダウングレード, 12-21
ユーザー定義のデータ型
 アップグレード, 7-30
ユニバーサル ROWID「UROWID」を参照

よ

要件
 エクスポート, 5-2
用語
 移行, 1-2
用語集「用語」を参照
読取り専用表領域
 アクティブ・トランザクション, 8-23
 移行, 3-4
 互換性, 8-23

ら

ラージ・オブジェクト「LOB」を参照

り

リカバリ・カタログ
 PL/SQL パッケージ要件, 8-39
 アップグレード, 7-30
リソース・マネージャ「データベース・リソース・マネージャ」を参照
リモート接続
 マルチスレッド共有サーバー, 2-16
領域要件
 データのコピー, 2-15
 移行, 2-15, 3-6, 4-4

れ

例外表

削除, 12-18

列

部分的に削除された

削除, 12-8

未使用

削除, 12-8

列削除

互換性, 8-10

レプリケーション

互換性, 8-44

連鎖行表

削除, 12-18

ろ

ローカルで管理される表領域

ディクショナリの表領域への変換, 12-5

ロード・バランシング

Net8, 8-47

ローカル接続

マルチスレッド共有サーバー, 2-16

ロールバック・セグメント

移行, 3-4

ロック

DML ロック制限、DML_LOCKS, B-9

わ

ワード・サイズ

変更, 1-4, 7-37