

Oracle8*i*

for Linux Intel 管理者リファレンス

リリース 8.1.7

2001 年 1 月

部品番号 : J02813-01

ORACLE®

Oracle8i for Linux Intel 管理者リファレンス , リリース 8.1.7

部品番号 : J02813-01

原本名 : Oracle8i Administrator's Reference, Release 3 (8.1.7) for Linux Intel

原本部品番号 : A85344-01

原本著者 : Pallavi Bhowmik

原本協力者 : Corinne Arne, Caroline Johnston, Tom Leah-Martin, Lori Meyers, Sally Norton, Severine Tymon, Joycelyn Wee, Doug Williams, Sanjib Das, Danny Fields, Prashanth Joshi, Malik Kalfane, Amit Khare, Ravi Kuntamukkala, Allen Lui, Srinivas Maturi, Rajesh Parasuraman, Vijayalakshmi Sekuboyina, Keshav Sharma, Ajay Srivastava, Binoy Sukumaran, Karen Thrift

Copyright © 1996, 2000, Oracle Corporation. All rights reserved.

Printed in Japan.

制限付権利の説明

プログラム（ソフトウェアおよびドキュメントを含む）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。

当プログラムのリバース・エンジニアリング等は禁止されています。

このドキュメントの情報は、予告なしに変更されることがあります。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を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 Notice が適用されます。

Restricted Rights Notice

Programs delivered subject to the DOD FAR Supplement are "commercial computer software" and use, duplication, and disclosure of the Programs, including documentation, 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 52.227-19, Commercial Computer Software - Restricted Rights (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 | vii |
| 用途 | viii |
| 対象読者 | viii |
| Oracle8i Workgroup Server および Oracle8i Enterprise Edition | viii |
| このマニュアルで使用する表記規則 | viii |
| コマンドの構文 | ix |
| インストールしたドキュメントの表示 | ix |
| Oracle Information Navigator | x |
| 参照マニュアル | x |
| Oracle サービスおよびサポート | xi |
| 1 Oracle8i の管理 | |
| 概要 | 1-2 |
| 環境変数 | 1-2 |
| Oracle8i 環境変数 | 1-3 |
| UNIX 環境変数 | 1-5 |
| 共通の環境設定 | 1-7 |
| 現行セッションでの環境変数値の設定およびエクスポート | 1-8 |
| システム時刻の設定 | 1-8 |
| ネットワーク実行ファイル | 1-9 |
| システム・グローバル領域 | 1-10 |
| SGA のサイズの計算 | 1-11 |
| Oracle8i のメモリー要件および使用量 | 1-12 |
| サーバー・リソースの制限 | 1-13 |
| データベースの制限 | 1-14 |
| 特殊アカウントおよび特殊グループ | 1-15 |

| | |
|-----------------------------------|------|
| セキュリティ | 1-16 |
| グループとセキュリティ | 1-16 |
| Server Manager コマンドのセキュリティ | 1-17 |
| データベース・ファイルのセキュリティ | 1-17 |
| リモート・パスワード | 1-18 |
| initsid.ora ファイルのカスタマイズ | 1-20 |
| 埋込み PL/SQL ゲートウェイ | 1-22 |
| 概要 | 1-22 |
| 埋込み PL/SQL ゲートウェイのインストール | 1-23 |
| Oracle HTTP Server | 1-24 |
| Oracle HTTP Server のステータス | 1-25 |
| Oracle HTTP Server のログ・ファイル | 1-25 |
| デモンストレーション・ファイル | 1-26 |
| SQL*Loader のデモンストレーション | 1-26 |
| SQL*Loader の管理 | 1-27 |
| PL/SQL デモンストレーション | 1-28 |
| データベースの例 | 1-30 |

2 Oracle8i のチューニング

| | |
|--|-----|
| チューニングの重要性 | 2-2 |
| Linux のツール | 2-2 |
| vmstat | 2-2 |
| swapon | 2-3 |
| free | 2-3 |
| SQL スクリプト | 2-4 |
| メモリー管理のチューニング | 2-4 |
| 十分なスワップ領域の割当て | 2-4 |
| ページングの制御 | 2-5 |
| 単一共有メモリー・セグメントでの SGA の確保 | 2-5 |
| ディスク I/O のチューニング | 2-5 |
| データベース・ライターをチューニングして書込み帯域幅を大きくする | 2-5 |
| 適切なファイル・システム・タイプの選択 | 2-6 |
| ディスク・パフォーマンスの監視 | 2-7 |
| ディスクのパフォーマンスに関する問題 | 2-7 |
| CPU の使用状況のチューニング | 2-7 |
| Oracle ユーザーおよびプロセスの優先順位をすべて同じにする | 2-7 |

| | |
|---|------|
| 大量のエクスポート / インポートおよび SQL*Loader ジョブでのシングルタスク・リンクの使用 ... | 2-7 |
| Oracle リソースの競合のチューニング | 2-8 |
| UNIX カーネル・パラメータのチューニング | 2-8 |
| ブロック・サイズおよびファイル・サイズのチューニング | 2-8 |
| Oracle ブロック・サイズの指定 | 2-8 |
| Linux バッファ・キャッシュ・サイズのチューニング | 2-9 |
| トレース・ファイルおよびアラート・ファイルの使用 | 2-10 |
| トレース・ファイルの名前 | 2-10 |
| アラート・ファイル | 2-10 |

3 SQL*Plus の管理

| | |
|---|-----|
| SQL*Plus の管理 | 3-2 |
| 設定ファイルの使用 | 3-2 |
| PRODUCT_USER_PROFILE 表の使用 | 3-3 |
| デモンストレーション表の使用 | 3-3 |
| ヘルプ機能の使用 | 3-4 |
| SQL*Plus の使用 | 3-5 |
| SQL*Plus からのシステム・エディタの使用 | 3-5 |
| SQL*Plus からのオペレーティング・システム・コマンドの実行 | 3-6 |
| SQL*Plus への割込み | 3-6 |
| SPOOL コマンドの使用 | 3-6 |
| SQL*Plus の制限事項 | 3-7 |
| ウィンドウのサイズ変更 | 3-7 |
| リターン・コード | 3-7 |

4 Oracle プリコンパイラおよび Oracle コール・インタフェースの使用

| | |
|---------------------------------|-----|
| Oracle プリコンパイラの概要 | 4-2 |
| プリコンパイラ構成ファイル | 4-2 |
| すべてのプリコンパイラに共通の問題 | 4-3 |
| 静的および動的リンク | 4-3 |
| Oracle 共有ライブラリ | 4-4 |
| Pro*C/C++ プリコンパイラ | 4-4 |
| Pro*C/C++ デモ・プログラム | 4-4 |
| Pro*C/C++ ユーザー・プログラム | 4-6 |
| Oracle コール・インタフェース | 4-7 |

| | |
|------------------------|------|
| OCI デモ・プログラム | 4-7 |
| OCI ユーザー・プログラム | 4-8 |
| カスタム Make ファイル | 4-9 |
| 未定義シンボルの修正 | 4-9 |
| マルチスレッド・アプリケーション | 4-10 |
| シグナル・ハンドラの使用法 | 4-10 |
| XA 機能 | 4-13 |

5 Net8 の構成

| | |
|---|------|
| 主な Net8 製品および特徴 | 5-2 |
| Net8 ファイルおよびユーティリティ | 5-2 |
| Oracle Connection Manager | 5-3 |
| サーバー・モデル | 5-4 |
| Oracle Names | 5-5 |
| Net8 プロトコルのサポート | 5-5 |
| ADDRESS の指定 | 5-5 |
| BEQ プロトコル | 5-6 |
| IPC プロトコル | 5-8 |
| RAW プロトコル | 5-9 |
| TCP/IP プロトコル | 5-9 |
| TCP/IP の ADDRESS 指定 | 5-9 |
| Oracle Enterprise Manager | 5-10 |
| Oracle SNMP での Oracle Intelligent Agent の構成 | 5-11 |
| Oracle Advanced Security | 5-13 |
| 参照ドキュメント | 5-13 |

6 Oracle データ・オプション・デモの実行

| | |
|----------------------------------|-----|
| Oracle8i <i>interMedia</i> | 6-2 |
| Text | 6-2 |
| Audio、Video および Image | 6-2 |
| Locator | 6-6 |
| Oracle8i Time Series デモ | 6-8 |
| Oracle8i Spatial | 6-9 |

A Optimal Flexible Architecture

Optimal Flexible Architecture (OFA) A-2

 OFA に準拠したデータベースの特長 A-2

UNIX で適用される OFA A-4

 マウント・ポイント A-4

 ディレクトリの名前 A-5

 ファイルの名前 A-6

 表領域の名前 A-8

 OFA 構造に基づいた Oracle ファイル A-9

 OFA ファイルのマッピング A-9

 ディレクトリ構造 A-11

索引

はじめに

用途

このマニュアルおよび『Oracle 8i for Linux Intel インストレーション・ガイド』は、Linux Intel システムでの Oracle8i リリース 8.1.7 のインストールおよび構成方法を記載しています。このマニュアルの内容は、Oracle8i 関連マニュアル・セットの製品情報を補足しています。

対象読者

このマニュアルは、Linux Intel システムで Oracle8i リリース 8.1.7 を管理する方を対象としています。

Oracle8i Workgroup Server および Oracle8i Enterprise Edition

特に指示がない限り、このマニュアルで説明している特長や機能は、Oracle8i Workgroup Server および Oracle8i Enterprise Edition に共通です。

このマニュアルで使用する表記規則

| | |
|----------|--|
| 固定幅フォント | 固定幅フォントは、UNIX コマンド、ディレクトリ名、ユーザー名、パス名およびファイル名を表します。 |
| 大カッコ [] | 大カッコで囲まれた語は、キーの名前を表します（たとえば、[Return] キーを押す）。ただし、コマンド構文で大カッコを使用する場合は意味が異なります。 |
| 大文字 | 大文字は、SQL の予約語、初期化パラメータおよび環境変数を表します。 |

コマンドの構文

UNIX のコマンド構文は固定幅フォントで表し、Bourne シェルの使用を前提としています。UNIX コマンド例の先頭の \$ 文字はプロンプトで、入力する必要はありません。UNIX では大文字と小文字が区別されるため、このマニュアルの表記規則は、他の Oracle マニュアルで使用されている表記規則と多少異なります。

| | |
|------------|--|
| バックスラッシュ \ | バックスラッシュは、コマンドが 1 行に入りきらない場合に使用します。このマニュアルで記載しているとおりに入力する（バックスラッシュを付ける）か、またはバックスラッシュを付けずに 1 行で入力します。 <pre>dd if=/dev/rdisk/c0t1d0s6 of=/dev/rst0 bs=10b \ count=10000</pre> |
| 中カッコ {} | 中カッコは、必ず選択する項目を表します。 <pre>.DEFINE {macro1}</pre> |
| 大カッコ [] | 大カッコは、任意に選択する項目を表します。 <pre>cvtcrt termname [outfile]</pre> <p>ただし、本文で大カッコを使用する場合は意味が異なります。</p> |
| 省略記号 ... | 省略記号は、同じ項目を任意の回数だけ繰り返すことを表します。 <pre>CHKVAL fieldname value1 value2 ... valueN</pre> |
| イタリック体 | イタリック体は、変数を表します。変数の箇所を適切な値に置き換えてください。 <pre>library_name</pre> |
| 縦棒線 | 縦棒線は、中カッコまたは大カッコで囲まれている選択項目を表します。 <pre>SIZE filesize [K M]</pre> |

インストールしたドキュメントの表示

Oracle8i Linux Intel のドキュメントには、このマニュアルおよび『Oracle 8i for Linux Intel インストレーション・ガイド』があります。ドキュメントは、HTML および PDF (Adobe Portable Document Format) を表示するには Acrobat Reader が必要) 形式でインストールできます。Linux 固有のドキュメント・ファイルは、Oracle8i CD-ROM からインストールできます。Oracle8i 関連ドキュメント・ファイルは、CD-ROM 版の Oracle8i 関連マニュアル・セットからインストールできます。ドキュメント・ファイルの位置は、次の規則に従って決定されます。

- 環境変数に ORACLE_DOC が定義されている場合、定義されたディレクトリにファイルがインストールされます。
- ORACLE_DOC は定義されていないが、ORACLE_BASE が定義されている場合、ファイルは \$ORACLE_BASE/doc ディレクトリにインストールされます。

- 環境変数に ORACLE_DOC または ORACLE_BASE のどちらも定義されていない場合、ファイルは \$ORACLE_HOME/doc ディレクトリにインストールされます。

ドキュメントを参照するには、index.htm または products.htm のいずれかをブラウザで表示します（後者は、フレーム表示可能なブラウザが不要です）。紙マニュアルが必要な場合は、PDF ファイルを印刷してください。

Oracle 製品のドキュメント

Oracle8i 製品のドキュメントは、CD-ROM 版の Oracle8i 関連マニュアル・セットにあります。CD-ROM のドキュメントの表示およびインストール方法については、CD-ROM の最上位ディレクトリにある README ファイルを参照してください。

Oracle Information Navigator

Oracle Information Navigator は、Oracle オンライン・ドキュメントとともに提供される Java ベースの検索およびナビゲーション・ユーティリティです。Java 対応のブラウザを使用して、CD-ROM の最上位層にある index.htm ファイルをオープンすると、Information Navigator が自動的に起動されます。Information Navigator は Oracle 製品ドキュメントに使用でき、CD-ROM またはインストールしたファイルのどちらからでもドキュメントを読むことができます。

参照マニュアル

リレーショナル・データベース管理システム関連の概念または用語に慣れていない場合は、インストールを始める前に、『Oracle8i 概要』の第 1 章をお読みください。

本番データベース・システム用のシステム管理およびチューニングの詳細は、次のマニュアルを参照してください。

- 『Oracle 8i for Linux Intel インストレーション・ガイド』
- 『Oracle8i 管理者ガイド』
- 『Oracle8i Net8 管理者ガイド』
- 『Oracle8i パフォーマンスのための設計およびチューニング』

前回のリリースの Oracle Server からの移行およびアップグレードの詳細は、『Oracle8i 移行ガイド』を参照してください。

Oracle Internet Directory のインストールについては、『Oracle Internet Directory 管理者ガイド』を参照してください。

Oracle サービスおよびサポート

オラクル社は、企業システム・ソリューションをサポートする、Oracle Education コース、Oracle Consulting サービス、Oracle Worldwide Customer Support などの幅広いサービスを提供しています。Oracle 製品およびグローバル・サービスの詳細は、インターネットで次の URL を参照してください。

<http://www.oracle.com>

ここでは、選択されたサービスの URL を示します。

Oracle サポート・サービス

テクニカル・サポートの情報は、次の URL に示されています。

<http://www.oracle.com/support>

テンプレートをご使用になると、電話をかける前に、問題に関する情報を用意することができます。CSI 番号（該当する場合）、または詳しい連絡先情報（特別なプロジェクト情報がある場合は、それも含む）も必要です。

オラクル社カスタマ・サポート・センター

オラクル社カスタマ・サポート・センターの連絡先は、次の URL に示されています。

<http://www.oracle.com/support/>

教育およびトレーニング

トレーニング情報およびスケジュールについては、次の URL から参照できます。

<http://education.oracle.com>

Oracle8i の管理

この章では、Linux Intel で Oracle8i を管理する方法について説明します。この章では、次の項目について説明します。

- [概要](#)
- [環境変数](#)
- [ネットワーク実行ファイル](#)
- [システム・グローバル領域](#)
- [Oracle8i のメモリー要件および使用量](#)
- [サーバー・リソースの制限](#)
- [データベースの制限](#)
- [特殊アカウントおよび特殊グループ](#)
- [セキュリティ](#)
- [埋込み PL/SQL ゲートウェイ](#)
- [Oracle HTTP Server](#)
- [デモンストレーション・ファイル](#)

概要

Oracle8i で作業するには、環境変数、パラメータ、メモリーおよびユーザー設定を構築する必要があります。この章では、Linux Intel の様々な設定について説明します。

Oracle8i のファイルおよびプログラムでは、疑問符「?」は ORACLE_HOME の値を表します。たとえば、Oracle8i では、次の SQL 文中の疑問符を ORACLE_HOME のフルパス名として使用します。

```
alter tablespace TEMP add datafile '?/dbs/dbs2.dbf' size 2M
```

アットマーク「@」は、\$ORACLE_SID を表します。たとえば、ファイルが現行のインスタンスに属していることを指定する場合、次のように入力します。

```
alter tablespace tablespace_name add datafile 'dbsfile@.dbf'
```

環境変数

この項では、通常使用される Oracle8i および UNIX 環境変数について説明します。

現行の環境変数の値を表示するには、env コマンドを使用します。たとえば、ORACLE_SID の値を表示するには、次のように入力します。

```
$ env | grep ORACLE_SID
```

注意： コマンド env は、環境にエクスポートされたものを表示するために使用してください。Bourne シェルおよび Korn シェルでは、エクスポートせずに値を設定できます。

これらの環境変数のいくつかは、Oracle8i をインストールする前に設定しておく必要があります。このような環境変数については、『Oracle8i for Linux Intel インストレーション・ガイド』を参照してください。

Oracle8i 環境変数

表 1-1 に、Oracle8i 環境変数の機能、構文および例を示します。

表 1-1 UNIX における Oracle8i 環境変数

| 環境変数 | 項目 | 定義 |
|--------------|----|---|
| EPC_DISABLED | 機能 | Oracle TRACE を使用禁止にします。 |
| | 構文 | true または false |
| NLS_LANG | 機能 | クライアント環境の言語、地域およびキャラクタ・セットを指定します。NLS_LANG で設定するキャラクタ・セットは、端末または端末エミュレータのキャラクタ・セットと一致する必要があります。NLS_LANG で設定されたキャラクタ・セットが、データベース側のキャラクタ・セットと異なる場合は、キャラクタ・セットを自動的に変換します。 設定値の詳細は、『Oracle8i NLS ガイド』を参照してください。 |
| | 構文 | <i>language_territory.characterset</i> |
| | 例 | <i>french_france.we8dec</i> |
| ORA_NLS33 | 機能 | 言語、地域、キャラクタ・セットおよび言語の定義ファイルが保存されているディレクトリを示します。 |
| | 例 | <i>\$ORACLE_HOME/ocommon/nls/admin/data</i> |
| ORACLE_BASE | 機能 | Optimal Flexible Architecture (OFA) に準拠した Oracle データベースのディレクトリ構造の基本となるディレクトリを指定します。 |
| | 構文 | <i>directory_path</i> |
| | 例 | <i>/u01/app/oracle</i> |
| ORACLE_HOME | 機能 | Oracle ソフトウェアがあるディレクトリを指定します。 |
| | 構文 | <i>directory_path</i> |
| | 例 | <i>\$ORACLE_BASE/product/8.1.7</i> |

表 1-1 UNIX における Oracle8i 環境変数（続き）

| 環境変数 | 項目 | 定義 |
|--------------|-----|---|
| ORACLE_PATH | 機能 | *.sql、*.frm、*.rpt などの Oracle アプリケーションが使用するファイルの検索パス名を指定します。そのファイルのフルパス名が指定されていない場合、またはそのファイルが現行のディレクトリにない場合、Oracle アプリケーションは、ファイルを見つけるために、ORACLE_PATH を使用します。 |
| | 構文 | ディレクトリをコロンで区切ったリスト <i>directory1:directory2:directory3</i> |
| | 例 | /u01/oracle/adhoc/8.1.7/bin:. 注意: 最後にピリオドを付けることによって、検索パスに現行の作業ディレクトリが追加されます。 |
| ORACLE_SID | 機能 | Oracle システム識別子（Oracle System Identifier: SID）を指定します。 |
| | 構文 | 英字で始まる英数文字列を指定します。8 文字以内で指定することをお薦めします。詳細は、『Oracle8i for Linux Intel インストール・ガイド』を参照してください。 |
| | 例 | SAL1 |
| ORACLE_TRACE | 機能 | インストール時の Bourne シェル・スクリプトのトレース状態を切り替えます。T を設定すると、多くの Oracle シェル・スクリプトは set -x フラグがオンの状態で実行されます。 |
| | 許容値 | T または T 以外 |
| ORAENV_ASK | 機能 | coraenv または oraenv が、ORACLE_SID または ORACLE_HOME を入力するためのプロンプトを表示するかどうかを指定します。NO を設定すると、プロンプトは表示されません。NO 以外を設定するとプロンプトが表示されます。 |
| | 構文 | 文字列 |
| | 許容値 | NO または NO 以外 |
| SQLPATH | 機能 | SQL*Plus が login.sql ファイルを検索するディレクトリまたはディレクトリのリストを設定します。 |
| | 構文 | ディレクトリをコロンで区切ったリスト <i>directory1:directory2:directory3</i> |
| | 例 | /home:/home/oracle:/u01/oracle |

表 1-1 UNIX における Oracle8i 環境変数（続き）

| 環境変数 | 項目 | 定義 |
|-----------|-----|---|
| TNS_ADMIN | 機能 | Net8 構成ファイルがあるディレクトリを設定します。 |
| | 構文 | <i>directory_path</i> |
| | 許容値 | 任意のディレクトリ。詳細は、『Oracle8i for Linux Intel インストール・ガイド』を参照してください。 |
| | 例 | \$ORACLE_HOME/network/admin |
| TWO_TASK | 機能 | tnsnames.ora ファイルで定義されている、Net8 接続文字列記述子のデフォルトの別名を設定します。 |
| | 構文 | 使用可能なネットワーク別名 |
| | 許容値 | tnsnames.ora ファイルで定義されている有効な Oracle Net8 別名 |
| | 例 | PRODDB_TCP |

注意： 環境変数に、Oracle Server プロセスと同じ名前（arch、pmon、dbwr など）は設定しないでください。

UNIX 環境変数

表 1-2 に、Oracle8i で使用する UNIX 環境変数の構文および例を示します。

表 1-2 Oracle8i で使用する UNIX 環境変数

| 環境変数 | 項目 | 定義 |
|-----------|----|---|
| CLASSPATH | 機能 | Java 機能に使用されます。この環境変数は、様々な Java アプリケーションで異なります。詳細は、ご使用の Java アプリケーションの製品ドキュメントを参照してください。 |
| | 構文 | <i>directory_path</i> |
| | 例 | デフォルトの設定はありません。CLASSPATH には、次の項目を含める必要があります。 JRE_Location、\$ORACLE_HOME/product/jlib JRE_Location は、\$ORACLE_HOME/JRE のように定義されます。 |

表 1-2 Oracle8iで使用する UNIX 環境変数（続き）

| 環境変数 | 項目 | 定義 |
|----------------------|-----|--|
| DISPLAY | 機能 | X ベースのツールで使⽤します。入出力に使用するディスプレイ・デバイスを指定します。詳細は、各ベンダーの X-Window のドキュメントを参照してください。 |
| | 構文 | <i>hostname:display</i> <i>hostname</i> はマシン名 (IP アドレスまたは別名)、 <i>display</i> はモニター番号です。モニターが 1 つの場合、番号は 0 です。 |
| | 例 | 135.287.222.12:0 bambi:0 |
| HOME | 機能 | ユーザーのホーム・ディレクトリを指定します。 |
| LANG または LANGUAGE | 機能 | メッセージなどを出力するためにオペレーティング・システムが使用する言語およびキャラクタ・セットを指定します。詳細は、オペレーティング・システムのドキュメントおよび『Oracle8i for Linux Intel インストレーション・ガイド』を参照してください。 |
| LD_LIBRARY_PATH | 機能 | 共有オブジェクト・ライブラリを検索するため、共有ライブラリ・ローダーが実行時に使⽤します。詳細は、ld の man ページを参照してください。 |
| | 構文 | ディレクトリをコロンで区切ったリスト <i>directory1:directory2:directory3</i> |
| | 例 | /usr/dt/lib:\$ORACLE_HOME/lib |
| PATH | 機能 | シェルが実行プログラムの場所を見つけるために使⽤します。 \$ORACLE_HOME/bin が含まれている必要があります。 |
| | 構文 | ディレクトリをコロンで区切ったリスト <i>directory1:directory2:directory3</i> |
| | 例 | /bin:/usr/bin:/usr/local/bin: /usr/bin/X11:\$ORACLE_HOME/bin:\$HOME/bin. 注意: 最後にピリオドを付けることによって、検索パスに現行の作業ディレクトリが追加されます。 |
| PRINTER | 機能 | デフォルト・プリンタを指定します。 |
| | 構文 | <i>printer_name</i> |
| | 例 | docqms |
| SHELL | 機能 | ホスト・コマンドで使用するコマンド・インタプリタを指定します。 |
| | 構文 | <i>shell_path</i> |
| | 許容値 | /bin/sh、/bin/csh、/bin/ksh または Linux Intel に付属のその他のコマンド・インタプリタ |

表 1-2 Oracle8i で使用する UNIX 環境変数（続き）

| 環境変数 | 項目 | 定義 |
|--------------|----|--|
| TERM | 例 | /bin/sh |
| | 機能 | Oracle Toolkit II キャラクタ・モード・ツールおよび UNIX ツールが端末タイプを判断する場合に使用します。 |
| TMPDIR | 例 | vt100 |
| | 機能 | テンポラリ・ディスク・ファイル用のデフォルト・ディレクトリを指定します。設定すると、テンポラリ・ファイルを作成するツールは、このディレクトリにテンポラリ・ファイルを作成します。 |
| | 構文 | <i>directory_path</i> |
| XENVIRONMENT | 例 | /u02/oracle/tmp |
| | 機能 | X-Window システムのリソース定義を含むファイルを指定します。詳細は、X-Window のドキュメントを参照してください。 |

共通の環境設定

Oracle8i では、データベース管理者（DBA）がすべてのユーザーに共通の環境を設定できます。共通の環境を設定することによって、システム管理者とデータベース管理者が Oracle8i システムを変更しやすくなります。

oraenv コマンド・ファイル

oraenv コマンド・ファイルは、インストール時に作成されます。このファイルには、Oracle 環境変数の値および次の情報が含まれています。

- データベースの変更に必要なユーザー・アカウントを更新するための主な方法
- Oracle8i データベース間を移動するメカニズム

たとえば、開発システムに対してデータベースの追加および削除を頻繁に行う場合や、同一のシステムにインストールされている複数の異なる Oracle データベース間をユーザーが移動する場合があります。oraenv によって、それぞれのユーザー・プロファイルが oraenv コマンド・ファイルを呼び出します。

注意： C シェルの場合は、oraenv のかわりに coraenv コマンドを使用します。

ローカル bin ディレクトリ

oraenv および dbhome を、Oracle ソフトウェアのホーム・ディレクトリではなくローカル bin ディレクトリに入れると、すべてのユーザーがそれらのファイルにアクセスできるようになります。また、これによって、パスが別の ORACLE_HOME を示すように変更した場合でも、oraenv は引き続き動作することが保証されます。ローカル bin ディレクトリは、次のインストールで実行される root.sh で指定します。Linux の場合、ローカル bin ディレクトリのデフォルト位置は /usr/local/bin です。

データベース間の移動

あるデータベースまたはインスタンスから別のデータベースまたはインスタンスへ移動するには、oraenv ルーチン呼び出して、プロンプトに対して移動先のデータベースの *sid* を指定します。oraenv コマンド・ファイルのフルパス名を必ず入力してください。次に例を示します。

次に例を示します。

```
$ . /usr/local/bin/oraenv
ORACLE_SID= [default]? sid
```

現行セッションでの環境変数値の設定およびエクスポート

コマンド `env` は、環境にエクスポートされたものを表示するために使用してください。Bourne シェルおよび Korn シェルでは、エクスポートせずに値を設定できます。

Bourne または Korn シェルの場合、次のように入力します。

```
$ ORACLE_SID=test
$ export ORACLE_SID
```

C シェルの場合、次のように入力します。

```
% setenv ORACLE_SID test
```

この場合、*test* は環境変数 ORACLE_SID に設定する値です。

システム時刻の設定

環境変数 TZ には、タイム・ゾーンを設定します。これによって、時間を夏時間に変更したり、別のタイム・ゾーンにしたりできます。調整した時刻は、ファイルのタイムスタンプを決定したり、date コマンドの出力を生成したり、現在の SYSDATE の値を調べたりする際に使用します。

できるだけ、TZ 値は変更しないでください。GMT+24 などの異なる TZ 値を使用すると、トランザクションが記録された日が変わってしまいます。その場合、SYSDATE を使用する Oracle アプリケーション（Oracle Financials など）が影響を受けます。この問題を回避するために、表の順序を決定する場合は、日付列ではなく順序番号を使用してください。

ネットワーク実行ファイル

\$ORACLE_HOME/bin ディレクトリ内のシェル・スクリプトの再リンクによって、製品実行ファイルを手動で再リンクできます。再リンクは、オペレーティング・システム・パッチの適用後またはオペレーティング・システムのアップグレード後に必要となります。

再リンク・スクリプトは、ORACLE_HOME にインストールされているものに基づいて Oracle 製品実行ファイルの再リンクを手動で実行します。

再リンクするには、次のように入力します。

```
$ relink parameter
```

表 1-3 再リンク・スクリプト・パラメータ

| パラメータ | 値 |
|------------|---|
| all | インストールされたものすべて |
| oracle | Oracle データベースの実行ファイルのみ |
| network | net_client、net_server、nau、cman、cnames |
| client | net_client、otrace、plssql、client_sharedlib |
| interMedia | ctx、ordimg、ordaud、md |
| precomp | インストールされたすべてのプリコンパイラ |
| utilities | ユーティリティ |
| oemagent | oemagent、odg |

注意： データベースを再リンクする場合は、Oracle Intelligent Agent および ORACLE_HOME にあるその他の Oracle プログラムを停止してください。

システム・グローバル領域

システム・グローバル領域（SGA）は、共有メモリー内に常駐する Oracle 構造体です。この構造体には、静的データ構造体、ロックおよびデータ・バッファが含まれています。各 Oracle プロセスが SGA 全体をアドレス指定するために、十分な共有メモリーが必要です。

1 つの共有メモリー・セグメントの最大サイズは、Linux の SHMMAX パラメータで指定します。SHMMAX の推奨値は、システムにインストールされた実メモリーの半分です。

SGA のサイズが共有メモリー・セグメントの最大サイズ（SHMMAX）を超える場合、Oracle8i は、要求された SGA サイズになるように、連続セグメントをさらに連結しようとします。SHMSEG は、プロセスが接続できるセグメントの最大数です。

注意： SHMMAX がデータベースの SGA サイズより小さい場合、緊密共有メモリー（ISM）によって問題が発生する場合があります。

SGA のサイズは、次の `init.ora` パラメータによって決まります。

- DB_BLOCK_BUFFERS
- DB_BLOCK_SIZE
- SORT_AREA_SIZE
- SHARED_POOL_SIZE
- JAVA_POOL_SIZE

これらのパラメータの値は、十分注意して設定してください。値が大きすぎると、マシンの物理メモリーに対する共有メモリーの割合が大きくなりすぎて、パフォーマンスが低下します。

SGA のサイズの計算

次のいずれかの方法で、SGA サイズを判断できます。

- インスタンスあたりの SGA の適切なサイズは、次の式で計算します。

$$(\text{DB_BLOCK_BUFFERS} \times \text{DB_BLOCK_SIZE}) + \text{SORT_AREA_SIZE} + \text{SHARED_POOL_SIZE} + \text{LOG_BUFFER} + \text{JAVA_POOL_SIZE}$$
- 実行中のデータベースの SGA サイズを表示するには、SQL*Plus の `show sga` コマンドを使用します。結果はバイト単位で表示されます。
- データベース・システムを起動するときに、SGA のサイズを決定します。ヘッダー「Total System Global Area」の横に SGA サイズが表示されます。

SGA を連結するアドレスは、SGA 内のデータベース・バッファやユーザーのアプリケーション・データ領域のカーソルに使用できる仮想アドレス領域の大きさに影響します。次に、SGA の再配置方法を示します。

1. 共有メモリー・セグメントを連結するための有効な仮想アドレス範囲を決定します。次のように入力して、Oracle8i の今回のリリースに含まれている `tstshm` 実行ファイルを使用します。

```
$ tstshm
```

`tstshm` 表示では、「Lowest shared memory address」および「Highest shared memory address」の行が有効なアドレス範囲を示しています。

2. `tstshm` の「Segment boundaries」出力を調べて、共有メモリー・セグメントを連結できる有効な仮想アドレス範囲を調べます。
3. 次のように入力して、`$ORACLE_HOME/rdbms/lib` ディレクトリに移動し、`genksms` を実行して `ksms.s` ファイルを生成します。

```
$ cd $ORACLE_HOME/rdbms/lib
$ $ORACLE_HOME/bin/genksms -b sgabeg > ksms.s
```

この場合、`sgabeg` は SGA の開始アドレス（デフォルトは `0x50000000`）で、手順 2 で調べた範囲内で設定します。

`sgabeg` の値を小さくすると、次のエラー・メッセージが表示される場合があります。（string には文字列が入ります）。

```
ORA-4030: out of process memory when trying to allocate %s bytes (%s,%s)
```

または

```
ORA-7324: smpall: malloc error while allocating pga.
```

いずれかのメッセージが表示された場合、PGA が必要とする領域にまで開始アドレスを下げた可能性があります。sgabeg を上げて、再試行します。

4. 既存の Oracle データベースを停止します。
5. 次のように入力して、\$ORACLE_HOME/rdbms/lib ディレクトリで Oracle 実行ファイルを作成し直します。

```
$ make -f ins_rdbms.mk ksms.o
$ make -f ins_rdbms.mk ioracle
```

ioracle を使用して、次の処理を行います。

- 古い実行ファイル (oracle0) のバックアップを取ります。
- 新しい Oracle 実行ファイルに正しい権限を割り当てます。
- 新しい実行ファイルを \$ORACLE_HOME/bin ディレクトリに移動します。

その結果、sgabeg で指定するアドレスに SGA をロードする新しい Oracle カーネルが作成されます。

参照： データベースでの Java の使用が SGA の計算に与える影響については、\$ORACLE_HOME/javavm/doc の README ファイルを参照してください。

Oracle8i のメモリー要件および使用量

システム上のユーザー数を決定するために、メモリー要件および使用量を計算します。これは、物理メモリー要件およびスワップ領域要件を決定する場合にも役立ちます。次の式を使用して、メモリー要件を見積もります。

$$\begin{aligned}
 &<\text{Oracle 実行ファイルのテキストのサイズ}> \\
 + &<\text{SGA のサイズ}> \\
 + &n \times (\\
 &\quad <\text{Tool 実行ファイルのプライベート・データ・セクションのサイズ}> \\
 &\quad + <\text{Oracle 実行ファイルの未初期化データ・セクションのサイズ}> \\
 &\quad + <\text{スタック用に 8192 バイト}> \\
 &\quad + <\text{プロセス・ユーザー領域用に 2048 バイト}>)
 \end{aligned}$$

この場合、 n はバックグラウンド・プロセスの数を示します。

各クライアント / サーバー接続に対して、次の式を使用して仮想メモリー要件を見積もります。

$$\begin{aligned}
 &<\text{Oracle 実行ファイルのデータ・セクションのサイズ}> \\
 + &<\text{Oracle 実行ファイルの未初期化データ・セクションのサイズ}> \\
 + &<\text{スタック用に 8192 バイト}>
 \end{aligned}$$

- + <プロセス・ユーザー領域用に 2048 バイト>
- + <アプリケーションに必要なカーソル領域>

実行ファイルのテキスト・サイズ、プライベート・データ・セクションのサイズおよび未初期化データ・セクションのサイズ（または DSS）を見積もるには、size コマンドを使用します。Oracle 実行ファイル・テキストは共有されているため、プログラムが何度起動されても、プログラム・テキストがカウントされるのは 1 回のみです。

実行中のデータベースにユーザーが接続しているときに、実際の Oracle 物理メモリー（バックグラウンド・プロセスおよびシャドウ・プロセス）の使用量を計算するには、pmap コマンドを使用します。pmon プロセスに対しては、共有セクション（read/write/exec/shared および read/exec で示される）を合計します。pmon を含む各シャドウ・プロセスおよびバックグラウンド・プロセスには、プライベート・セクション（read/write/exec で示される）を合計します。バックグラウンド・プロセス名は、ora_ で始まり、SID で終わります。シャドウ・プロセス名は、oracleSID で始まります。

ps -elf コマンドは使用しないでください。このコマンドでは、SZ 列が、表示される各プロセスのメモリーの共有部分を繰り返すため、Oracle が使用しているメモリーの量が実際よりも多く表示されます。

参照： ps コマンドで使用できるスイッチのリストについては、Linux Intel のドキュメントを参照してください。

ps コマンドは、プロセス・サイズをページで返します。システムのパッケージ・サイズは、アーキテクチャによって異なります。pagesize コマンドを使用して、サイズが 4096 バイトと 8192 バイトのどちらかであることを確認します。各プロセスについて、SZ 値とページ・サイズを掛けます。

Oracle 実行ファイルのテキスト・サイズと、システムで実行中のその他のすべての Oracle Tool 実行ファイルのテキスト・サイズを、その小計に加算します。実行ファイルが何度起動されても、実行ファイルのサイズがカウントされるのは 1 回のみです。

サーバー・リソースの制限

Linux では、親プロセスからリソースの制限を継承します（オペレーティング・システムのドキュメントの getrlimit (2) を参照）。この制限は、ユーザー・プロセスに対して実行される Oracle8i のシャドウ・プロセスに適用されます。Linux のデフォルトのリソース制限は、どの Oracle8i のシャドウ・プロセスまたはバックグラウンド・プロセスに対しても十分高い値です。この制限を低くすると、Oracle8i システムに影響が出る場合があります。詳細は、Linux のシステム管理者にお問い合わせください。

Oracle ユーザー用に設定したディスクの割当て制限によって、Oracle8i システムの操作性が悪くなる場合があります。ディスクの割当て制限を設定する前に、Oracle8i のデータベース管理者および Linux のシステム管理者にお問い合わせください。

データベースの制限

表 1-4 に、CREATE DATABASE 文または CREATE CONTROL FILE 文のパラメータの最大値およびデフォルト値を示します。

注意： それらのパラメータ間の相互依存によって、正当な値に影響を与える場合があります。

表 1-4 CREATE DATABASE 文および CREATE CONTROL FILE 文のパラメータ

| パラメータ | デフォルト値 | 最大値 |
|---------------|--------|-------|
| MAXDATAFILES | 30 | 65534 |
| MAXINSTANCES | 1 | 63 |
| MAXLOGFILES | 16 | 255 |
| MAXLOGHISTORY | 100 | 65534 |
| MAXLOGMEMBERS | 2 | 5 |

特殊アカウントおよび特殊グループ

Oracle Server では、特殊アカウントが必要です。表 1-5 に、特殊 UNIX アカウントを示します。表 1-6 に、特殊 Oracle Server アカウントを示します。表 1-7 に、特殊グループ・アカウントを示します。

表 1-5 特殊 UNIX アカウント

| | |
|--------|---|
| oracle | Oracle ソフトウェア所有者は、Oracle8i ソフトウェアを所有するアカウントを示します。このメンテナンス・アカウントには、データベースを作成、起動、停止、または INTERNAL として接続するための DBA 権限が必要です。Oracle ソフトウェア所有者は、スーパーユーザーであってはいけません。 |
| root | root ユーザーは、最高の権限（スーパーユーザー権限）が与えられた特殊 UNIX アカウントです。このアカウントを使用することによって、UNIX カーネルを構成したり、ネットワーク・ソフトウェアを構成およびインストールしたりできます。また、ユーザー・アカウントおよびグループを作成できます。 |

表 1-6 特殊 Oracle Server アカウント

| | |
|--------|---|
| SYS | インストール時に自動的に作成され、DBA 権限が与えられた標準的な Oracle8i アカウントです。SYS アカウントは、データ・ディクショナリのすべてのベース表を所有します。このアカウントは DBA が使用します。 |
| SYSTEM | インストール時に自動的に作成され、DBA 権限が与えられた標準的な Oracle8i アカウントです。SYSTEM ユーザーで追加の表またはビューを作成できます。DBA は SYSTEM としてログインし、データベースを監視またはメンテナンスします。 |

表 1-7 特殊グループ・アカウント

| | |
|---------------|---|
| dba グループ | Oracle ソフトウェア所有者は、dba グループの唯一の必須メンバーです。dba グループには、他の UNIX ユーザーを追加できます。このグループのメンバーは、SQL*Plus の特殊権限機能にアクセスできます。アカウントが dba グループのメンバーでない場合、INTERNAL として接続するためにパスワードを入力するか、または SQL*Plus のその他の管理機能へのアクセス権限を取得する必要があります。デフォルトの OSDBA グループは、dba です。 |
| oinstall グループ | ORACLE_HOME では、Oracle8i をインストールするすべてのユーザーは同じ UNIX グループに属している必要があります。マシン上のすべての ORACLE_HOME は、OUI Inventory を共有します。また、グループの書き込み権限もあります。プライマリ・グループとして oinstall でインストールすることをお勧めします。 |
| oper グループ | オプションの UNIX グループです。メンバーには、データベースの OPERATOR 権限が与えられます。DBA 権限の一部を制限したものが OPERATOR 権限です。 |
| root グループ | root グループのメンバーになれるのは root ユーザーのみです。 |

セキュリティ

Oracle8i では、UNIX オペレーティング・システムのいくつかの機能を使用し、ユーザーに安全性の高い環境を提供します。その機能には、ファイル所有権、グループ・アカウント、および実行時にそのユーザー ID を変更するプログラム機能が含まれます。

Oracle8i の 2 タスク構造によって、ユーザー・プログラムと Oracle プログラム間で作業（およびアドレス領域）を分割し、セキュリティを高めることができます。すべてのデータベース・アクセスは、このシャドウ・プロセスおよび Oracle プログラムへの特殊権限によって行うことができます。

参照： セキュリティ問題の詳細は、『Oracle8i 管理者ガイド』を参照してください。

グループとセキュリティ

Oracle プログラムは、セキュリティを高めるために 2 つのグループに分けられます。すべてのユーザー（UNIX 用語では、other）がアクセスできる実行プログラムおよび DBA 専用の実行プログラムです。次のようにして、セキュリティを高めることをお勧めします。

oracle アカウントのプライマリ・グループは、oinstall グループである必要があります。oracle アカウントは、dba グループをセカンダリ・グループとする必要があります。DBA 権限を必要とするユーザー・アカウントは dba グループに属することができますが、oinstall グループに属することができるのは、oracle アカウントのみです。

Server Manager コマンドのセキュリティ

Server Manager へのアクセスを制限することをお勧めします。STARTUP、SHUTDOWN および CONNECT INTERNAL のシステム権限にアクセスできるのは、Oracle ソフトウェア所有者および dba グループのメンバーのみにしてください。

警告： システム権限が与えられた文は、正しく使用しないとデータベースを破壊する可能性があります。dba グループ以外のユーザーでも、パスワードを知っていれば、INTERNAL として接続できるので注意してください。

データベース・ファイルのセキュリティ

Oracle8i のインストールに使用するユーザー ID は、データベース・ファイルの所有者である必要があります。デフォルトのユーザー ID は、Oracle ソフトウェア所有者です。これらのファイルの認可を、読取りおよび書込み権限は所有者のみに、読取り専用権限はグループや他のユーザーに与えられるように設定します。

Oracle ソフトウェア所有者は、データベース・ファイルを含むディレクトリを所有する必要があります。セキュリティを高めるために、グループおよび他のユーザーの読取り権限を取り消してください。

保護されているデータベース・ファイルにアクセスするには、Oracle プログラムでそのセット・ユーザー ID (setuid) ビットをオンにする必要があります。

Oracle Universal Installer は、Oracle 実行ファイルの権限を自動的に次のように設定します。

```
-rwsr-s--x 1 oracle dba 443578 Mar 10 23:03 oracle
```

ユーザーの実行フィールドにある「s」は、Oracle プログラムを実行する場合に、そのプログラムを起動した実際のユーザー ID に関係なく、そのプログラムの事実上のユーザー ID は、oracle になることを示しています。

これを手動で設定する必要がある場合は、次のように入力します。

```
$ chmod 6751 $ORACLE_HOME/bin/oracle
```

リモート・パスワード

オペレーティング・システム・アカウントがなくても、パーソナル・コンピュータ（PC）などのリモート・マシンからデータベースを管理できます。この場合、ユーザーは、orapwd ユーティリティで作成および管理される Oracle8i パスワード・ファイルを使用して検証されます。オペレーティング・システム・アカウントをサポートするシステムでは、パスワード・ファイル検証も行うことができます。

ローカル・パスワード・ファイルは \$ORACLE_HOME/dbs ディレクトリにあり、1つのデータベースのユーザー名およびパスワード情報が入っています。1つのマシンに複数の \$ORACLE_HOME ディレクトリがある場合、それぞれにパスワード・ファイルがあります。データベースで、パスワード・ファイルを使用可能にするには、initlsid.ora パラメータ remote_login_passwordfile を EXCLUSIVE に設定します。

リモート PC からデータベースへのアクセス

Oracle8i パスワード・ファイルがあれば、DBA 権限を持つネットワーク上の PC ユーザーは、このデータベースに INTERNAL としてアクセスできます。権限を持つユーザーがデータベースに対して DBA 機能を実行する場合は、SQL*Plus コマンドに dba ユーザーのパスワードを追加して入力します。次に例を示します。

```
SQL> connect internal/dba_password@alias as {sysdba|sysoper}
```

リモート認証

表 1-8 に、安全性の低いプロトコルを使用したリモート接続の動作を制御する initlsid.ora ファイルのパラメータを示します。

表 1-8 リモート接続を制御するパラメータ

| | |
|-------------------|-----------------------------|
| REMOTE_OS_AUTHENT | ops\$ 接続を使用可能または使用禁止にします。 |
| OS_AUTHENT_PREFIX | ops\$ アカウントによって使用されます。 |
| REMOTE_OS_ROLES | リモート接続でロールを使用可能または使用禁止にします。 |

参照： リソースの制限の詳細は、オペレーティング・システムのドキュメントの「getrlimit(2)」を参照してください。

orapwd の実行

orapwd ユーティリティは \$ORACLE_HOME/bin にあり、Oracle ソフトウェア所有者が実行します。orapwd のコマンド構文は、次のとおりです。

```
$ orapwd file=filename password=password entries=max_users
```

表 1-9 に、orapwd の構文を示します。

表 1-9 orapwd の実行構文

| | |
|------------------|--|
| <i>filename</i> | パスワード情報が書き込まれているファイル名です。ファイル名は <i>orapwsid</i> で、フルパス名を指定します。その内容は暗号化されていて、ユーザーには読めません。このパラメータの入力は必須です。 |
| <i>password</i> | INTERNAL および SYS 用に指定する初期パスワードです。このパスワードは、データベースの作成後に ALTER USER 文を使用して変更できます。このパラメータの入力は必須です。 |
| <i>max_users</i> | データベースに SYSDBA または SYSOPER として接続できるユーザーの最大数です。このパスワード・ファイルを EXCLUSIVE にする必要がある場合のみ、このパラメータの入力は必須です。 <i>max_users</i> には、必要であると予想される数より大きい値を設定してください。ユーザー数がこの値より多くなる場合、新しいパスワード・ファイルを作成する必要があります。 |

orapwd の例

```
$ orapwd file=/u01/app/oracle/product/8.1.7/dbs/orapwV817  
password=V817pw entries=30
```

参照： セキュリティおよびパスワードの詳細は、『Oracle8i 管理者ガイド』を参照してください。

initsid.ora ファイルのカスタマイズ

デフォルトの initsid.ora ファイルは、Oracle8i ソフトウェアとともに提供されます。Oracle Universal Installer (OUI) は、\$ORACLE_BASE/admin/sid/pfile ディレクトリ中にファイルを作成します。このファイルを変更することによって、Oracle8i のインストールをカスタマイズできます。initsid.ora のサンプル・ファイルは、\$ORACLE_HOME/dbs ディレクトリにあります。

表 1-10 に、Linux での初期化パラメータのデフォルト値を示します。initsid.ora ファイルのこれらのパラメータに別の値を設定しなければ、すべての Oracle8i インスタンスでこれらの値が使用されます。initsid.ora ファイルには、デフォルトとは異なる値のパラメータのみを設定してください。

システムでの現在のパラメータ値を表示するには、SQL*Plus コマンドの SHOW PARAMETERS を使用します。

表 1-10 初期化パラメータ

| パラメータ | デフォルト値 | 許容値 |
|-------------------------------|--------------------|--|
| BACKGROUND_DUMP_DEST | ?/rdbms/log | 有効なディレクトリ名 |
| BITMAP_MERGE_AREA_SIZE | 1048576 | 65536 ～無制限 |
| COMMIT_POINT_STRENGTH | 1 | 0 ～ 255 |
| CONTROL_FILES | ?/dbs/cntrl@.dbf | 有効なファイル名 |
| CREATE_BITMAP_AREA_SIZE | 8388608 | 65536 ～無制限 |
| DB_BLOCK_BUFFERS | 48MB のバッファ | 50KB ～無制限 |
| DB_BLOCK_SIZE | 2048 | 2KB ～ 16KB |
| DB_FILES | 200 | 1 ～ 2000000 |
| DB_FILE_DIRECT_IO_COUNT | 64 (最大値は 1048576) | 0 ～ 1048576/ ブロック・サイズ |
| DB_FILE_MULTIBLOCK_READ_COUNT | 8 | 1 ～次の小さい方の値 (DB_BLOCK_BUFFERS/4、 1048576/DB_BLOCK_SIZE) |
| DISTRIBUTED_TRANSACTIONS | TRANSACTIONS の 1/4 | 0 ～無制限 |
| HASH_AREA_SIZE | 2 × SORT_AREA_SIZE | 0 ～無制限 |
| HASH_MULTIBLOCK_IO_COUNT | 0 (セルフ・チューニング) | 0 ～次の小さい方の値 (127、 DB_BLOCK_BUFFERS/4、 1048576/DB_BLOCK_SIZE) |
| JAVA_POOL_SIZE | 20000000 | 1000000 ～ 1000000000 |
| LOCK_SGA | FALSE | TRUE、FALSE |

表 1-10 初期化パラメータ（続き）

| パラメータ | デフォルト値 | 許容値 |
|-------------------------------|---|----------------------------------|
| LOG_ARCHIVE_DEST | null | 有効なディレクトリ名 |
| LOG_ARCHIVE_FORMAT | "%t_%s.dbf" | 有効なファイル名 |
| LOG_BUFFER | 次の大きい方の値（512KB、128KB × CPU_COUNT） | 66560 ～ 無制限 |
| LOG_CHECKPOINT_INTERVAL | 0 | 0 ～ 無制限 |
| MTS_MAX_DISPATCHERS | 5 | MTS_DISPATCHERS 値と PROCESSES 値の間 |
| MTS_MAX_SERVERS | MTS_SERVERS > 20 の場合は 2 × MTS_SERVERS、それ以外の場合は 20 | MTS_SERVERS 値と PROCESSES 値の間 |
| MTS_SERVERS | MTS_DISPATCHERS が指定されている場合は 1、それ以外の場合は 0 | 1 と PROCESSES 値の間 |
| MTS_LISTENER_ADDRESS | ADDRESS=address | |
| NLS_LANGUAGE | AMERICAN | 有効な言語名 |
| NLS_TERRITORY | AMERICA | 有効な地域名 |
| OBJECT_CACHE_MAX_SIZE_PERCENT | 10 | 0 ～ 無制限 |
| OBJECT_CACHE_OPTIMAL_SIZE | 100KB | 10KB ～ 無制限 |
| OPEN_CURSORS | 50 | 1 ～ 無制限 |
| OS_AUTHENT_PREFIX | ops\$ | 任意の文字列 |
| PROCESSES | PARALLEL_AUTOMATIC_TUNING でない場合は 30 | 6 ～ 無制限 |
| SHARED_POOL_SIZE | 64 ビットの場合 64MB、32 ビットの場合は 8MB | 300000 ～ 無制限 |
| SORT_AREA_SIZE | 65536 | 0 ～ 無制限 |

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』、『Oracle8i 管理者ガイド』および『Oracle8i パフォーマンスのための設計およびチューニング』を参照してください。

埋込み PL/SQL ゲートウェイ

埋込み PL/SQL ゲートウェイは、PL/SQL ベースのデータベース・アプリケーションを Web 上で展開するための固有のサポートを提供するために Oracle8i サーバーに埋め込まれたゲートウェイです。埋込み PL/SQL ゲートウェイは、Oracle Servlet Engine (OSE) サーブレットとして実装されており、OSE および OSE をサポートする Apache モジュール、`mod_ose` の存在および構成に依存します。次に、ゲートウェイのインストールおよび構成方法について説明します。

概要

Apache モジュール `mod_ose` および `mod_plsql` は、PL/SQL ベースの Web アプリケーションをサポートしています。

`mod_ose` モジュールは、Oracle8i インスタンスで実行する OSE の要求ルーターとして機能します。`mod_ose` のルーティング機能によって、ステートフル OSE アプリケーションはステートフルな要求をルーティングし、中間層を介して指定された OSE/Oracle8i インスタンスに戻すことができます。埋込み PL/SQL ゲートウェイは、Oracle8i サーバーで実行する OSE サーブレットとして実装されているため、ステートフルおよびステートレスな PL/SQL Web アプリケーションに対応できます。ステートフル PL/SQL Web アプリケーションでは、そのデータベース・セッション（たとえば、パッケージおよびトランザクション）の状態がすべて、要求間で保存されます。

`mod_plsql` モジュールは、PL/SQL ゲートウェイに与えられた名前です。PL/SQL ゲートウェイは、中間層サーバーにある Apache モジュール内で実行し、バックエンド Oracle Server の PL/SQL プロシージャを、OCI を使用して実行します。現在、`mod_plsql` モジュールでは、ステートレス PL/SQL Web アプリケーションのみをサポートしています。

参照： PL/SQL ベースの Web アプリケーションの開発については、PL/SQL ゲートウェイの共通ドキュメントである、『Oracle9i Application Server PL/SQL ゲートウェイの使用』を参照してください。

埋込み PL/SQL ゲートウェイのインストール

すべての OSE サブレットと同様、埋込み PL/SQL ゲートウェイはロードおよび公開される必要があります。次に、埋込み PL/SQL ゲートウェイ・サブレットをロードおよび公開する方法について示します。

1. サブレットをロードするには、sys として SQL*Plus に接続し、SQL スクリプト initplgs を実行します。

```
$ SQL>@rdbms/admin/initplgs.sql
```

2. 埋込み PL/SQL ゲートウェイ・サブレットの名前は、oracle.plsql.web.PLSQLGatewayServlet です。サブレットを公開するには、次のコマンドを実行します。

```
% $ORACLE_HOME/jis/bin/unix/sess_sh -s http://<OSE machine name>:<OSE port \
number> -u sys/change_on_install -c "publishservlet -virtualpath \
pls/*/webdomains/contexts/default plsGateway \
SYS:oracle.plsql.web.PLSQLGatewayServlet"
```

これによって、ゲートウェイ・サブレットがデフォルト・コンテキストとともに、plsGateway として公開されます。サブレットは、仮想パス /pls を使用して接続できます。ゲートウェイ・サブレットにアクセスする URL の例は、次のとおりです。

```
http://dlsun240/pls/dadname/hello_world
```

参照： サブレットの使用および公開については、『Oracle8i Oracle Servlet Engine ユーザーズ・ガイド』を参照してください

Oracle PL/SQL 埋込みゲートウェイの構成

Apache Server または mod_ose の構成手順および埋込み PL/SQL ゲートウェイについては、このマニュアルでは説明していません。

参照： Apache Server または mod_ose の構成については、『Oracle8i Oracle Servlet Engine ユーザーズ・ガイド』を参照してください。

Oracle HTTP Server

Oracle HTTP Server は、Apache ベースです。サーバーの管理タスクは、サーバーが実行されているローカル・マシンに接続する必要があります。場合によっては、root での接続が必要です。

Oracle HTTP Server は、ポート 7777 で自動的に起動します。バイナリのサーバー名は、httpd です。次のコマンドを使用してサーバーが実行中であることを確認してください。

```
ps -edaf | grep httpd
```

Oracle HTTP Server の開始および停止

構成に変更が加えられた場合、サーバーを再起動する必要があります。使用可能な SSL を使用してサーバーを再起動するには、root ユーザーである必要があります。次のようにサーバーを再起動します。

```
cd $ORACLE_HOME/Apache/Apache/bin
su root
./apachectl {start|startssl|graceful|stop}
```

注意： 使用可能な SSL を使用してサーバーを再起動すると、デフォルト・ポートは 80 および 443 になります。

デフォルトの初期静的ページへのアクセス

デフォルトの初期静的ページには、Apache のオンライン・ドキュメントおよび各コンポーネントのデモへのリンクが含まれます。初期静的ページにアクセスするには、ご使用のインターネット・ブラウザで、次のいずれかの URL をロードします。

- サーバーに使用可能な SSL がない場合は、次の URL をロードします。

```
http://<ServerName>:7777/
```

- サーバーに使用可能な SSL がある場合は、次の URL をロードします。

```
http://<ServerName>/
```

ServerName が Apache Server で構成されている場合、構成ファイルは、httpd.conf です。次のコマンドを使用して、構成ファイルの位置を確認します。

```
grep ServerName $ORACLE_HOME/Apache/Apache/conf/httpd.conf
```

Oracle HTTP Server のステータス

複数のステータス・ページが使用可能です。デフォルトのサーバー構成ファイルでは、セキュリティのため、サーバー・ステータスが使用禁止になっています。これらを使用可能にするには、適切な構成ファイルを編集し、サーバーを再起動します。

構成ファイル `$ORACLE_HOME/Apache/Apache/conf/httpd.conf` の場合は、次の URL を使用します。

```
http://<ServerName>/server-status  
http://<ServerName>/server-info  
http://<ServerName>/perl-status
```

構成ファイル `$ORACLE_HOME/Apache/Jserv/etc/conf/jserv.conf` の場合は、次の URL を使用します。

```
http://<ServerName>/jserv
```

Oracle HTTP Server のログ・ファイル

多くのログ・ファイルがサーバーによって生成されます。サーバーが正常に動作していることを確認するために、定期的にログ・ファイルをチェックすることは重要です。デフォルトでは、構成ファイルのエラー・ログ・レベルは、warn に設定されています。デフォルトのエラー・レベルは、適切な構成ファイルを編集し、サーバーを再起動することによって変更できます。

```
$ORACLE_HOME/Apache/Apache/logs/error_log  
$ORACLE_HOME/Apache/Apache/logs/ssl_engine_log  
$ORACLE_HOME/Apache/Jserv/logs/jserv.log  
$ORACLE_HOME/Apache/Jserv/logs/mod_jserv.log
```

デモンストレーション・ファイル

SQL*Loader のデモンストレーション

デモンストレーション・ファイルは、Oracle8i に付属しています。SQL*Loader のデモンストレーションは、次の順番で実行してください。

表 1-11 SQL*Loader のデモンストレーション・ファイル

| | | | |
|---------|---------|---------|---------|
| ulcase1 | ulcase3 | ulcase5 | ulcase7 |
| ulcase2 | ulcase4 | ulcase6 | |

デモンストレーションの作成と実行

デモンストレーションは、ユーザー scott/tiger として実行する必要があります。次の要件を確認してください。

- ユーザー scott/tiger に CONNECT 権限と RESOURCE 権限がある
- 空の EMP 表および DEPT 表がある
- 1. 実行するデモンストレーションの ulcasen.sql スクリプトを実行します。
\$ sqlplus scott/tiger @ulcasen.sql
- 2. デモンストレーション・データをオブジェクトにロードします。
\$ sqlldr scott/tiger ulcasen.ctl
- ulcase2 デモンストレーションに対して、ulcase2.sql スクリプトを実行する必要はありません。
- ulcase6 デモンストレーションに対して、ulcase6.sql スクリプトを実行し、コマンドラインに次のように入力します。
\$ sqlldr scott/tiger ulcase6 DIRECT=true
- ulcase7 デモンストレーションに対して、ulcase7.sql スクリプトを実行し、コマンドラインに次のように入力します。
\$ sqlldr scott/tiger ulcase7

デモンストレーションを実行した後、ulcase7e.sql を実行して、挿入トリガーおよびグローバル変数パッケージを削除します。

SQL*Loader の管理

Oracle8i には、SQL*Loader 機能が組み込まれています。デモンストレーション・ファイルおよびメッセージ・ファイルは、rdbms ディレクトリにあります。

SQL*Loader の制御ファイルには、次のファイル処理オプションの文字列が追加されています。デフォルトは `str` で、引数は指定されていません。

```
[ "str" | "fix n" | "var n" ]
```

表 1-12 ファイル処理オプション

| | |
|------------------|--|
| <code>str</code> | 改行文字で終了するレコードを、一度に 1 レコードずつ読み込むストリームを指定します。(デフォルト) |
| <code>fix</code> | ファイルが <i>n</i> バイトの固定長レコードで構成されていることを示します。 <i>n</i> は整数です。 |
| <code>var</code> | ファイルが <i>n</i> バイトの可変長レコードで構成されていることを示します。 <i>n</i> は整数です。レコードの最初の 5 文字を指定します。 |

ファイル処理オプションを選択しないと、情報はデフォルトでレコード・ストリーム (`str`) として処理されます。`fix` モードではレコード終了記号をスキャンする必要がないため、デフォルトの `str` モードより高いパフォーマンスが得られます。

固定長レコードでの改行

各レコードが改行で終了する固定長レコードを含むファイルを読むために `fix` オプションを使用する場合、レコード長に改行文字の長さ (1 文字) を含めて、SQL*Loader に指定してください。

次に例を示します。

```
AAA [改行]  
BBB [改行]  
CCC [改行]
```

改行文字も含めるため、`fix 3` ではなく `fix 4` を指定します。

固定長レコードのファイルの最後のレコードが改行文字で終了しない場合は、その他のレコードも改行文字で終了しないでください。同様に、最後のレコードを改行文字で終了する場合は、すべてのレコードを改行文字で終了してください。

注意： `vi` などの特定のテキスト・エディタを使用すると、ファイルの最後のレコードは自動的に改行文字で終了します。この場合、ファイル内のその他のレコードが改行文字で終了していないと、不整合が発生します。

改行文字の削除

改行文字をロードせずに固定長レコードから改行文字を削除するには、制御ファイルで position(x:y) 関数を使用します。たとえば、次のように入力して、4 桁目の改行文字を削除します。

```
load data
infile xyz.dat "fix 4"
into table abc
( dept position(01:03) char )
```

これによって、固定長レコードの 4 桁目にある改行文字は削除されます。

PL/SQL デモンストレーション

PL/SQL には、ロードできるサンプル・プログラムが多数あります。デモンストレーション・ファイルおよびメッセージ・ファイルは、rdbms ディレクトリにあります。サンプル・プログラムが機能するには、Oracle8i データベースをオープンし、マウントする必要があります。

- 1. SQL*Plus を起動し、ユーザー / パスワードを scott/tiger で接続します。

```
$ cd $ORACLE_HOME/plsql/demo
$ sqlplus scott/tiger
```

- 2. デモンストレーションをロードするために、SQL*Plus から exampbld.sql を起動します。

```
SQL> @exampbld
```

注意： デモンストレーションの作成は、必要な権限を持つ oracle アカウントで行ってください。また、デモンストレーションの実行も、同じアカウントで行ってください。

表 1-13 に、カーネルのデモンストレーションを示します。

表 1-13 カーネルのデモンストレーション

| | | | |
|-------------|------------|-------------|-------------|
| examp1.sql | examp5.sql | examp11.sql | sample1.sql |
| examp2.sql | examp6.sql | examp12.sql | sample2.sql |
| examp3.sql | examp7.sql | examp13.sql | sample3.sql |
| examp4.sql | examp8.sql | examp14.sql | sample4.sql |
| extproc.sql | | | |

PL/SQL デモンストレーションを実行するには、そのデモンストレーションを作成するときに使用したユーザー / パスワードで SQL*Plus を起動し、データベースに接続します。デモンストレーション名の前にアットマーク「@」または start を入力し、デモンストレーションを起動します。たとえば、examp1 のデモンストレーションを起動するには、次のように入力します。

```
$ sqlplus scott/tiger
SQL> @examp1
```

プリコンパイラ PL/SQL デモンストレーションを作成するには、次のように入力します。

```
$ cd $ORACLE_HOME/plsql/demo
$ make -f demo_plsql.mk demos
```

表 1-14 に、プリコンパイラのデモンストレーションを示します。

表 1-14 プリコンパイラのデモンストレーション

| examp9.pc | examp10.pc | sample5.pc | sample6.pc |
|-----------|------------|------------|------------|
|-----------|------------|------------|------------|

1 つのデモンストレーションを作成する場合は、make コマンドにそのデモンストレーション名を引数として指定します。たとえば、examp9.pc 実行ファイルを作成する場合、次のように入力します。

```
$ make -f demo_plsql.mk examp9
```

現行のシェルから examp9 デモンストレーションを起動するには、次のように入力します。

```
$ ./examp9
```

extproc デモンストレーションを実行するには、まず、ファイル tnsnames.ora に次の行を追加します。

```
(DESCRIPTION=(ADDRESS=(PROTOCOL=ipc) (KEY=plsf)) (CONNECT_DATA=(SID=extproc)))
```

次に、listener.ora ファイルに次の行を追加します。

```
SC=( (SID_NAME=extproc) (ORACLE_HOME=/u01/app/oracle/product/8.1.7) (PROGRAM=extproc))
```

SQL*Plus セッションで次のように入力します。

```
SQL> connect system/manager
Connected.
SQL> grant create library to scott;
Grant succeeded.
SQL> connect scott/tiger
Connected.
SQL> create library demolib as '$ORACLE_HOME/plsql/demo/extproc.so';
Library created.
```

最後に、テストを実行します。

```
SQL> connect scott/tiger
Connected.
SQL> @extproc
```

データベースの例

次の例では、ローカル bin ディレクトリが /usr/local/bin で、本番データベースが PROD であることを想定しています。また、oraenv が実行された後、ORAENV_ASK はデフォルトの YES にリセットされています。これによって、次に oraenv が実行されたときに、別の ORACLE_SID を入力するプロンプトが表示されます。

注意： ORACLE_SID を入力するプロンプトが起動時に表示されないようにするには、環境変数 ORAENV_ASK を NO に設定します。

Oracle Database Configuration Assistant を使用するかわりに、データベースを手動で作成した場合、システム構成が /etc/oratab ファイルに反映されていることを確認してください。

各サーバーのインスタンスに、エントリを次の形式で追加します。

```
ORACLE_SID:ORACLE_HOME:{Y|N}
```

Y または N は、dbstart および dbshut スクリプトをアクティブにするかどうかを示します。Oracle Database Configuration Assistant は、作成する各データベースにエントリを自動的に追加します。

シングル・インスタンスの例

Bourne シェルまたは Korn シェルの場合、.profile ファイルに次の行があります。

```
. local_bin_directory/oraenv
```

この行を次の行に置き換えます。

```
PATH=${PATH}:/usr/local/bin
ORACLE_SID=PROD
export PATH ORACLE_SID
ORAENV_ASK=NO
. oraenv
ORAENV_ASK=
```

C シェルの場合、`.cshrc` ファイルに次の行があります。

```
source local_bin_directory/coraenv
```

この行を次の行に置き換えます。

```
setenv PATH ${PATH}:/usr/local/bin
setenv ORACLE_SID PROD
setenv ORAENV_ASK NO
source /usr/local/bin/coraenv
unset ORAENV_ASK
```

複数インスタンスの例

複数インスタンスの場合は、起動時に `sid` を定義します。

Bourne シェルまたは Korn シェルの場合、次のように入力します。

```
#!/usr/bin/sh
echo "The SIDs on this machine are:"
cat /etc/oratab | awk -F: '{print $1}' | grep -v "#"
ORAENV_ASK="YES"
. /usr/local/bin/oraenv
```

C シェルの場合、次のように入力します。

```
#!/usr/bin/csh
echo "The SIDs on this machine are:"
cat /etc/oratab | awk -F: '{print $1}' | grep -v "#"
set ORAENV_ASK="YES"
source /usr/local/bin/coraenv
```

Oracle8i のチューニング

- チューニングの重要性
- Linux のツール
- SQL スクリプト
- メモリー管理のチューニング
- ディスク I/O のチューニング
- ディスク・パフォーマンスの監視
- CPU の使用状況のチューニング
- Oracle リソースの競合のチューニング
- ブロック・サイズおよびファイル・サイズのチューニング
- Linux バッファ・キャッシュ・サイズのチューニング
- トレース・ファイルおよびアラート・ファイルの使用

チューニングの重要性

Oracle8i は、高度に最適化できるソフトウェア製品です。チューニングを頻繁に行うことで、システム・パフォーマンスが最適化され、データのボトルネックの発生を防ぎます。この章の説明は、シングル・プロセッサ・システムを対象としています。パフォーマンス・チューニングに関するヒントの多くは、パラレル・オプションおよび Oracle8i で使用可能な機能を使用する場合にも適用できます。

システムのチューニングを始める前に、次の項の「[Linux のツール](#)」で説明するツールを使用して、通常の動作を監視します。

参照：『Oracle8i パフォーマンスのための設計およびチューニング』

Linux のツール

Linux には、データベースのパフォーマンスを評価し、データベースの要件を決定するパフォーマンス監視ツールが準備されています。これらのツールは、Oracle プロセスの統計だけでなく、システム全体の CPU の使用状況、割込み、スワップ、ページングおよびコンテキスト・スイッチングについての統計情報も提供します。

参照： Linux のツールについては、オペレーティング・システムのドキュメントを参照してください。

vmstat

vmstat ユーティリティは、Linux におけるプロセス、仮想メモリー、ディスク、ページングおよび CPU のアクティビティを表示します。表示内容は、コマンドで切り換えます。次のコマンドは、システム・アクティビティのサマリーを 5 秒ごとに 8 回表示します。

```
$ vmstat 5 8
```

[図 2-1](#) に、vmstat コマンドの出力例を示します。

図 2-1 vmstat コマンドの出力例

| procs | | | memory | | | | swap | | io | | system | | | cpu | |
|-------|---|---|--------|--------|-------|--------|------|----|----|----|--------|----|----|-----|-----|
| r | b | w | swpd | free | buff | cache | si | so | bi | bo | in | cs | us | sy | id |
| 0 | 0 | 0 | 6548 | 234812 | 17696 | 482620 | 0 | 0 | 2 | 0 | 29 | 3 | 0 | 0 | 10 |
| 0 | 0 | 0 | 6548 | 234812 | 17696 | 482620 | 0 | 0 | 0 | 0 | 106 | 8 | 0 | 0 | 100 |
| 0 | 0 | 0 | 6548 | 234812 | 17696 | 482620 | 0 | 0 | 0 | 1 | 107 | 7 | 0 | 0 | 100 |
| 0 | 0 | 0 | 6548 | 234812 | 17696 | 482620 | 0 | 0 | 0 | 0 | 113 | 7 | 0 | 0 | 100 |
| 0 | 0 | 0 | 6548 | 234812 | 17696 | 482620 | 0 | 0 | 0 | 0 | 123 | 7 | 0 | 0 | 100 |
| 0 | 0 | 0 | 6548 | 234812 | 17696 | 482620 | 0 | 0 | 0 | 0 | 121 | 7 | 0 | 0 | 100 |
| 0 | 0 | 0 | 6548 | 234812 | 17696 | 482620 | 0 | 0 | 0 | 0 | 123 | 6 | 0 | 0 | 100 |
| 0 | 0 | 0 | 6548 | 234812 | 17696 | 482620 | 0 | 0 | 0 | 0 | 124 | 8 | 0 | 0 | 100 |

w 列 (procs の下) は、スワップ・アウトされた (ディスクに書き込まれた) プロセスの数を示します。値が 0 以外の場合は、スワッピングが発生し、システムがメモリー不足になっています。si 列および so 列は、それぞれ 1 秒あたりのスワップ・インおよびスワップ・アウトの回数を示します。スワップ・アウト (so 列) は、常に 0 にしてください。

swapon

swapon -s ユーティリティは、デバイスのスワップ領域の使用状況の概要を表示します。スワップ領域が不足すると、システムがハングアップしたり、応答時間が長くなったりする場合があります。図 2-2 に、swap -l コマンドの出力例を示します。

図 2-2 swap -l コマンドの出力例

| Filename | Type | Size | Used | Priority |
|-----------|-----------|---------|------|----------|
| /dev/sda7 | Partition | 2096440 | 2416 | -1 |

free

free ユーティリティは、スワップ領域の使用状況を表示します。スワップ領域が不足すると、システムがハングアップしたり、応答時間が長くなったりする場合があります。

SQL スクリプト

SQL スクリプト `utlbstat.sql` および `utlestat.sql` は、Oracle データベースのパフォーマンスを監視し、システム・グローバル領域（SGA）データ構造をチューニングします。Linux では、スクリプトは `$ORACLE_HOME/rdbms/admin/` に保存されています。

参照： これらのスクリプトの詳細は、『Oracle8i パフォーマンスのための設計およびチューニング』を参照してください。

メモリー管理のチューニング

メモリー・チューニング・プロセスでは、ページングおよびスワップ領域をチューニングし、使用可能なメモリーを判断します。

Oracle バッファ・マネージャによって、アクセス頻度の高いデータをキャッシュに長く保存できます。バッファ・マネージャを監視してバッファ・キャッシュをチューニングすると、Oracle のパフォーマンスはかなり向上する場合があります。各システムの Oracle バッファ・サイズの最適値は、システム全体の負荷と他のアプリケーションとを比較した場合の Oracle の優先順位によって異なります。

十分なスワップ領域の割当て

スワッピングは Linux で大きなオーバーヘッドが発生する原因となるため、最小限に抑える必要があります。スワッピングの状況を確認するには、Linux では、`swapon -s` または `vmstat` を使用します。

システムでスワッピングが行われている場合、メモリーを節約するために次の処理を行います。

- 必要以上にシステム・デーモン（daemon）・プロセスまたはアプリケーション・プロセスを実行しないようにします。
- データベース・バッファの数を減らし、使用可能なメモリーを増やします。
- Linux ファイル・バッファの数を減らします（特に RAW デバイスを使用する場合）。

Linux では、`swapon -s` を使用して、使用中のスワップ領域を確認します。システムにスワップ領域を追加するには、`swapon -a` を使用します。詳細は、Linux Intel のドキュメントを参照してください。

スワップ領域は、はじめはシステムの RAM 容量の 2 ～ 4 倍に設定します。Developer/2000、Oracle Applications または Oracle InterOffice を使用する場合は、もう少し高い値に設定します。スワップ領域の使用状況を見ながら、必要に応じて値を高くしてください。

ページングの制御

プログラムを実行するためにプログラム全体がメモリーに常駐する必要はないため、ページングはスワッピングほど深刻な問題ではありません。少しぐらいのページアウトでは、システムのパフォーマンスにほとんど影響はありません。

大量のページングを検出するには、高速応答時またはアイドル時の測定値と、低速応答時の測定値を比較します。

ページングを監視するには、`vmstat` を使用します。

- ページアウト・アクティビティが空きリストに追加されます。この値は 0 である必要があります。

システムで大量のページアウト・アクティビティが常に発生している場合は、次の方法で解決してください。

- メモリーを増やします。
- 一部の作業を別のシステムに移します。
- カーネルで使用するメモリーを少なく設定します。

単一共有メモリー・セグメントでの SGA の確保

十分な共有メモリーがなければ、データベースを起動できません。共有メモリーを増やすために、UNIX カーネルを再構成できます。

参照： SGA の詳細は、第 1 章の「[システム・グローバル領域](#)」および『Oracle8i for Linux Intel インストレーション・ガイド』の第 2 章の「Oracle8i に UNIX カーネルを構成する」を参照してください。

ディスク I/O のチューニング

I/O ボトルネックは、最も簡単に識別できるパフォーマンスの問題です。使用可能なディスク全体で I/O を均等に分散させて、ディスクへのアクセス時間が短くなるようにしてください。小規模なデータベースおよび Parallel Query を使用しないデータベースでは、異なるデータ・ファイルと表領域を使用可能なディスクに分散させてください。

データベース・ライターをチューニングして書込み帯域幅を大きくする

Oracle では、データベース・ライター（DBWR）のアクティビティがボトルネックになることを避けるために、I/O スレーブを使用した次の方法が用意されています。

I/O スレーブ

I/O スレーブは、I/O の実行のみを目的とするプロセスです。I/O スレーブは Oracle7 の複数 DBWR にかわる機能です（実際は、複数 DBWR を汎化したもので、他のプロセスでも同様に実行できます）。また、非同期 I/O が使用可能かどうかに関係なく動作します。I/O スレーブは、LARGE_POOL_SIZE が設定されている場合はそこから割り当てられます。そうでない場合は、共有メモリー・バッファから割り当てられます。I/O スレーブには、初期化パラメータ・セットが付属しています。このパラメータ・セットによって、スレーブの動作を制御できます。表 2-1 に、I/O スレーブの初期化パラメータを示します。

表 2-1 I/O スレーブの初期化パラメータ

| パラメータ | 許容値 | デフォルト値 |
|-----------------------|------------|--------|
| BACKUP_TAPE_IO_SLAVES | TRUE/FALSE | FALSE |
| DBWR_IO_SLAVES | 0 - 999 | 0 |
| DB_WRITER_PROCESSES | 1-10 | 1 |

各プロセス・タイプの I/O スレーブ数のデフォルトは 0 であるため、特別に設定しない限り、I/O スレーブは実行されません。

DB_WRITER_PROCESSES（Oracle7 の DB_WRITERS パラメータの後継）には、インスタンスに対するデータベース・ライター・プロセス数の初期値を指定します。DBWR_IO_SLAVES を使用する場合、DB_WRITER_PROCESSES の設定に関係なく、使用するデータベース・ライター・プロセスの数は 1 つのみです。

適切なファイル・システム・タイプの選択

Linux では、ファイル・システムを選択できます。ファイル・システムの特性はそれぞれ異なり、それらがデータにアクセスする方法は、データベース・パフォーマンスに大きな影響を与えます。代表的なファイル・システムは、次のとおりです。

- ext2: Linux 用の拡張ファイル・システム 2
- RAW デバイス: ファイル・システムなし

ファイル・システムとアプリケーションには必ずしも互換性があるとは限りません。パフォーマンスを向上させるために ext2 が選択される場合が多くありますが、選択したファイル・システムによってパフォーマンスは異なります。

ディスク・パフォーマンスの監視

vmstat を使用して、ブロック IO アクティビティを監視します。これにより、ブロック・デバイスに対してブロックの送受信が行われます。

ディスクのパフォーマンスに関する問題

Oracle ブロック・サイズは、ディスクのブロック・サイズと同じか、またはディスクのブロック・サイズの倍数にする必要があります。

データベース・ファイルに対してファイル・システムを使用する前に、パーティションでファイル・システム・チェックを行います。次に、クリーンで断片化されていないファイル・システムを新しく作成します。ディスクの I/O はできるだけ均等に分散し、ログ・ファイルをデータベース・ファイルから分離します。

CPU の使用状況のチューニング

Oracle ユーザーおよびプロセスの優先順位をすべて同じにする

Oracle は、すべてのユーザーおよびバックグラウンド・プロセスを同じ優先順位で操作するように設計されています。優先順位を変更すると、競合が起きたり応答時間に影響が出たりする場合があります。

たとえば、ログ・ライター・プロセス（LGWR）の優先順位を低くすると、LGWR の実行頻度は低くなり、ボトルネックになります。逆に、LGWR の優先順位を高くすると、ユーザー・プロセスの応答時間は長くなってしまいます。

大量のエクスポート / インポートおよび SQL*Loader ジョブでのシングルタスク・リンクの使用

大量のデータをユーザーと Oracle8i 間で（たとえば、エクスポート / インポートを使用して）転送する必要がある場合、シングルタスク・アーキテクチャを使用すると効果的です。シングルタスク・インポート（impst）、エクスポート（expst）および SQL*Loader（sqlldrst）実行ファイルを作成する場合、\$ORACLE_HOME/rdbms/lib ディレクトリにある ins_rdbms.mk という Make ファイルを使用します。

次の例では、impst、expst および sqlldrst の実行ファイルが作成されます。

```
$ cd $ORACLE_HOME/rdbms/lib
$ make -f ins_rdbms.mk singletask
```

注意： Oracle 実行ファイルをシングルタスクとしてリンクすると、ユーザー・プロセスは SGA 全体に直接アクセスできるようになります。さらに、シングルタスクを実行するには、多くのメモリーが必要です。これは、Oracle 実行ファイル・テキストがフロントエンド・プロセスとバックグラウンド・プロセス間で共有されなくなるためです。

Oracle リソースの競合のチューニング

UNIX カーネル・パラメータのチューニング

UNIX カーネルををできるだけ小さくすることによって、パフォーマンスが向上します。UNIX カーネルでは、通常、事前に物理的な RAM が割り当てられるため、Oracle プロセスなど他のプロセスで利用できるメモリーが少なくなります。

メモリーが割り当てられているビデオ・ドライバ、ネットワーク・ドライバおよびディスク・ドライバを検索してください。それらのドライバは、削除できる場合がほとんどです。それによって、他のプロセスで利用できるメモリーが増えます。

注意： 必ず UNIX カーネルのバックアップを取っておいてください。バックアップ方法の詳細は、ハードウェア・ベンダーのドキュメントを参照してください。

ブロック・サイズおよびファイル・サイズのチューニング

注意： ブロック・サイズを変更する場合は、新しくデータベースを作成する必要があります。新しいデータベースにデータを転送する前にブロック・サイズを変更してみて、最適な構成を判断してください。

Oracle ブロック・サイズの指定

Linux では、デフォルトの Oracle ブロック・サイズは 2KB、最大ブロック・サイズは 16KB です。実際のブロック・サイズは、16KB 以下で 2KB の倍数の値を設定します。

アプリケーションによって異なりますが、通常、最適なブロック・サイズはデフォルトのサイズです。異なる Oracle ブロック・サイズでデータベースを作成する場合は、データベースを作成する前に `initSID.ora` ファイルに次の行を追加します。

```
db_block_size=new_block_size
```

注意： `db_block_size` の値は、特定のタイプの Oracle ファイルの最大サイズを決定します。

Linux バッファ・キャッシュ・サイズのチューニング

RAW デバイスを最大限活用するには、Oracle8i バッファ・キャッシュのサイズを調整し、メモリーに制限がある場合は、Linux バッファ・キャッシュのサイズも調整します。

Linux バッファ・キャッシュは、オペレーティング・システムで提供されます。バッファ・キャッシュには、メモリーからディスク、またはディスクからメモリーに転送されるメモリー内のデータ・ブロックを保存します。

Oracle8i バッファ・キャッシュは、Oracle データベース・バッファを保存するための領域です。Oracle8i では RAW デバイスを使用できるため、Linux バッファ・キャッシュを使用する必要はありません。

RAW デバイスに移動するときは、Oracle8i バッファ・キャッシュのサイズを大きくしてください。システムで利用できるメモリーに制限がある場合は、それに応じて Linux バッファ・キャッシュのサイズを小さくしてください。init`sid`.ora ファイル内の `db_block_buffers` パラメータを変更し、インスタンスを再起動することによって、Oracle8i バッファ・キャッシュを大きくまたは小さくできます。

キャッシュ・サイズの調整

- キャッシュ・ヒット率が向上する限り、Oracle8i キャッシュ・サイズを大きくします。
- スワッピングおよびページング・アクティビティが増える場合は、キャッシュ・サイズを小さくします。

トレース・ファイルおよびアラート・ファイルの使用

この項では、操作上の問題を診断して解決するために Oracle8i が作成するトレース・ファイル（またはダンプ・ファイル） およびアラート・ファイルについて説明します。

トレース・ファイルの名前

トレース・ファイル名の形式は、`processname_sid_unixpid.trc` です。表 2-4 に、詳細を示します。

表 2-2 トレース・ファイル名の構文

| | |
|--------------------|---|
| <i>processname</i> | トレース・ファイルの作成元である Oracle8i プロセスを示す 3 ～ 4 文字のプロセス名です（たとえば、pmon、dbwr、ora、reco など）。 |
| <i>sid</i> | インスタンスのシステム識別子です。 |
| <i>unixpid</i> | UNIX プロセス ID 番号です。 |
| <i>.trc</i> | すべてのトレース・ファイル名に付くファイル名の拡張子です。 |

たとえば、`lgwr_TEST_1237.trc` などがあります。

アラート・ファイル

`alert_sid.log` ファイルは、データベースと関連付けられており、`initSID.ora` ファイルのパラメータに指定されているディレクトリにあります。
BACKGROUND_DUMP_DEST デフォルトのディレクトリは、`$ORACLE_HOME/rdbms/log` です。

SQL*Plus の管理

この章では、SQL*Plus の次の機能について説明します。

- [SQL*Plus の管理](#)
- [SQL*Plus の使用](#)
- [SQL*Plus の制限事項](#)

SQL*Plus の管理

この項では、SQL*Plus の管理方法について説明します。

設定ファイルの使用

SQL*Plus を起動すると、Site Profile の設定ファイル `glogin.sql` および User Profile の設定ファイル `login.sql` が実行されます。

Site Profile ファイルの使用

グローバルな Site Profile は、`$ORACLE_HOME/sqlplus/admin/glogin.sql` です。SQL*Plus をインストールすると、デフォルトの Site Profile は `$ORACLE_HOME/sqlplus/admin` に保存されます。Site Profile がすでに存在する場合は、上書きされます。SQL*Plus を削除すると、既存の Site Profile も削除されます。

User Profile ファイルの使用

User Profile は、`login.sql` です。SQL*Plus は現行のディレクトリを最初に検索し、次に指定したディレクトリを `login.sql` が検出されるまで検索します。環境変数 `SQLPATH` には、ファイルを検索するディレクトリのリストをコロンで区切って設定します。

たとえば、現在のディレクトリが `/u02/oracle` で、`SQLPATH` が `/home:/home/oracle:/u01/oracle` に設定されている場合、SQL*Plus は次の順番で `login.sql` を検索します。

1. `/u02/oracle` (現行のディレクトリ)
2. `/home`
3. `/home/oracle`
4. `/u01/oracle`

`login.sql` に設定されるオプションは、`glogin.sql` に設定されているオプションを上書きします。

参照： Profile ファイルの詳細は、『Oracle8i 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。

PRODUCT_USER_PROFILE 表の使用

Typical インストール時に、PRODUCT_USER_PROFILE 表が自動的に作成されます。この表は、指定した SQL および SQL*Plus コマンドを使用禁止にする場合に使用します。この表を再作成するには、SYSTEM スキーマの \$ORACLE_HOME/sqlplus/admin/pupbld.sql スクリプトを実行します。

たとえば、次のように入力します。

```
$ sqlplus system/manager
SQL> @?/sqlplus/admin/pupbld.sql
```

SQL*Plus は、通常は疑問符「?」が指定されている場所に \$ORACLE_HOME の値を使用します。

デモンストレーション表の使用

SQL*Plus には、テスト用のデモンストレーション表が用意されています。

Typical インストールの実行

Typical インストールの場合、ユーザー SCOTT およびデモンストレーション表が自動的に作成されます。

デモンストレーション表の手動作成

デモンストレーション表を作成するには、SQL スクリプト

\$ORACLE_HOME/sqlplus/demo/demobld.sql を実行します。SQL*Plus では、任意のユーザー名を使用して demobld.sql ファイルを実行し、スキーマにデモンストレーション表を作成できます。たとえば、次のように入力します。

```
$ sqlplus scott/tiger
SQL> @?/sqlplus/demo/demobld.sql
```

また、次のようにシェル・スクリプト \$ORACLE_HOME/bin/demobld を使用して、\$ORACLE_HOME/sqlplus/demo/demobld.sql を実行することもできます。

```
$ demobld scott tiger
```

デモンストレーション表の削除

デモンストレーション表を削除するには、SQL スクリプト

\$ORACLE_HOME/sqlplus/demo/demobld.sql を使用します。SQL*Plus では、任意のユーザー名を使用して、ユーザーのスキーマからデモンストレーション表を削除できます。たとえば、次のように入力します。

```
$ sqlplus scott/tiger
SQL> @?/sqlplus/demo/demodrop.sql
```

また、次のようにシェル・スクリプト \$ORACLE_HOME/bin/demodrop を使用して、\$ORACLE_HOME/sqlplus/demo/demobld.sql を実行することもできます。

```
$ demodrop scott tiger
```

注意： SQL スクリプト demobld.sql および demodrop.sql はどちらも、EMP、DEPT、BONUS、SALGRADE および DUMMY 表を削除します。これらと同じ名前を持つ表がスキーマに存在しないことを確認してから、どちらかのスクリプトを実行してください。そうしないと、表のデータが失われます。

ヘルプ機能の使用

この項では、ヘルプ機能の使用方法について説明します。

Typical インストールの実行

Typical インストール、または Database Configuration Assistant のオプションで、事前に作成したデータ・ファイルを含む初期データベースをコピーする場合、SQL*Plus は自動的にヘルプ機能をインストールします。

Database Configuration Assistant の使用

データベースを作成する場合、Oracle Database Configuration Assistant を使用してヘルプ表を作成できます。

ヘルプ機能の手動インストール

シェル・スクリプト \$ORACLE_HOME/bin/helpins を使用すると、ヘルプ機能を手動でインストールできます。スクリプトを実行する前に、環境変数 SYSTEM_PASS に、SYSTEM スキーマ名およびパスワードを設定してください。たとえば、次のように入力します。

```
$ setenv SYSTEM_PASS SYSTEM/MANAGER
$ helpins
```

環境変数 `SYSTEM_PASS` が設定されていない場合、`helpins` から、`SYSTEM` パスワードの入力を求めるプロンプトが表示され、`SYSTEM` スキーマにヘルプ・データがロードされます。ヘルプ機能をインストールするには、`$ORACLE_HOME/sqlplus/help/helpbld.sql helpus.sql` を実行することもできます。`SQL*Plus` では、システム・ユーザーはファイル `helpbld.sql` およびその引数 `helpus.sql` を実行して、ヘルプ機能表を作成できます。たとえば、次のように入力します。

```
$ sqlplus system/manager
SQL> @?/sqlplus/admin/help/helpbld.sql helpus.sql
```

注意： シェル・スクリプト `helpins` および `SQL*Plus` スクリプト `helpbld.sql` は、新しい表を作成する前に既存のヘルプ機能表を削除します。

`SQL*Plus` で `$ORACLE_HOME/sqlplus/help/helpbld.sql` を実行して、スキーマにあるヘルプ機能表を手動で削除することもできます。たとえば、次のように入力します。

```
$ sqlplus system/manager
SQL> @?/sqlplus/admin/help/helpdrop.sql
```

参照： ヘルプ機能の詳細は、『*Oracle8i 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。

SQL*Plus の使用

この項では、`SQL*Plus` の使用方法について説明します。

SQL*Plus からのシステム・エディタの使用

`SQL*Plus` プロンプトで `ed` または `edit` コマンドを入力すると、`ed`、`emacs`、`ned`、`vi` などのオペレーティング・システム・エディタが起動されます。起動するには、環境変数 `PATH` にエディタのディレクトリが含まれている必要があります。

エディタを起動すると、現行の `SQL` バッファはエディタに入れられます。エディタを終了すると、変更された `SQL` バッファが `SQL*Plus` に戻されます。

`SQL*Plus` の `_editor` 変数の定義によって、エディタを起動するように指定できます。この変数は、`glogin.sql` または `login.sql` に設定するか、または `SQL*Plus` セッション中に入力します。たとえば、デフォルト・エディタを `vi` に設定するには、次のように入力します。

```
SQL> DEFINE _editor=vi
```

`_editor` 変数が設定されていない場合は、環境変数 `EDITOR` または `VISUAL` のどちらかの値が使用されます。環境変数 `EDITOR` および `VISUAL` が両方設定されている場合は、環境変数 `EDITOR` の値が優先されます。`_editor`、`EDITOR` および `VISUAL` が設定されていない場合は、デフォルト・エディタは `ed` になります。

エディタを起動すると、SQL*Plus はエディタにテキストを渡すためにテンポラリ・ファイル `afiedt.buf` を使用します。`SET EDITFILE` を使用すると、このファイルを改名できます。たとえば、次のように入力します。

```
SQL> SET EDITFILE /tmp/myfile.sql
```

SQL*Plus は、テンポラリ・ファイルを削除しません。

SQL*Plus からのオペレーティング・システム・コマンドの実行

SQL*Plus プロンプトの後に最初の文字として `HOST` コマンドまたは感嘆符 (!) を使用すると、後続の文字がサブシェルに渡されます。オペレーティング・システム・コマンドを実行する際に使用するシェルは、環境変数 `SHELL` によって設定されます。デフォルトのシェルは、`/bin/sh` (`sh`) です。シェルが実行できない場合は、エラー・メッセージが表示されます。

`HOST` コマンドまたは ! を入力することによって、SQL*Plus を終了しなくても、オペレーティング・システム・コマンドを実行できます。たとえば、1 つのコマンドを実行するには、次のように入力します。

```
SQL> ! command
```

この場合、`command` は実行するオペレーティング・システム・コマンドを表します。

複数のオペレーティング・システム・コマンドを実行するには、`HOST` コマンドまたは ! を入力した後に [Enter] キーを押します。コマンドの実行後、制御は SQL*Plus に戻ります。

SQL*Plus への割込み

SQL*Plus を実行している間、[Ctrl]+[c] キーを押すと、スクロール中のレコード表示を停止し、SQL 文を終了できます。

SPOOL コマンドの使用

`SPOOL` コマンドで生成されるファイルのデフォルトの拡張子は、`.lis` です。拡張子を変更するには、ピリオド (.) を含めたスプール・ファイル名を指定します。たとえば、次のように入力します。

```
SQL> SPOOL query.lis
```

SQL*Plus の制限事項

この項では、SQL*Plus の制限事項を説明します。

ウィンドウのサイズ変更

SQL*Plus の LINESIZE および PAGESIZE のデフォルト値によって、ウィンドウ・サイズが自動的に調整されるわけではありません。

リターン・コード

UNIX のリターン・コードは 1 バイトですが、Oracle エラー・コードを返すには 1 バイトでは不十分です。リターン・コードの範囲は、0 ～ 255 です。

Oracle プリコンパイラおよび Oracle コール・インタフェースの使用

この章では、次の Oracle プリコンパイラおよび Oracle コール・インタフェースの機能について説明します。

- [Oracle プリコンパイラの概要](#)
- [Pro*C/C++ プリコンパイラ](#)
- [Oracle コール・インタフェース](#)
- [カスタム Make ファイル](#)
- [未定義シンボルの修正](#)
- [マルチスレッド・アプリケーション](#)
- [XA 機能](#)

Oracle プリコンパイラの概要

Oracle プリコンパイラとは、Oracle データベースの SQL 文と高水準言語で書かれたプログラムを組み合わせるアプリケーション設計ツールです。Oracle プリコンパイラは、ANSI SQL と互換性があり、Oracle8i やその他の ANSI SQL データベース管理システムで実行するオープンでカスタマイズされたアプリケーションを開発するために使用します。Oracle プリコンパイラのデモンストレーションを実行するには、Oracle8i をインストールしておく必要があります。

参照： Oracle プリコンパイラおよびインタフェースの概要については、『Oracle8i アプリケーション開発者ガイド』を参照してください。

プリコンパイラ構成ファイル

プリコンパイラ・システム構成ファイルは、\$ORACLE_HOME/precomp/admin にあります。[表 4-1](#) に、プリコンパイラ・システム構成ファイルを示します。

表 4-1 システム構成ファイル

| 製品 | 構成ファイル |
|------------------------------|------------|
| Pro*C/C++ 8.1.7 | pcscfg.cfg |
| Object Type Translator 8.1.7 | ottcfg.cfg |

[表 4-2](#) に、プリコンパイラ README ファイルの位置を示します。README ファイルには、前回のリリースからの変更情報が記載されています。

表 4-2 Oracle 製品の README ファイルの保存場所

| 製品 | README ファイル |
|-----------|--|
| Pro*C/C++ | \$ORACLE_HOME/precomp/doc/proc2/readme.doc |

すべてのプリコンパイラに共通の問題

大文字から小文字への変換

C 言語以外では、コンパイラによって大文字の関数またはサブプログラム名を小文字に変換します。これによって、「No such user exit」エラー・メッセージが表示される場合があります。このエラー・メッセージが表示された場合、オプション・ファイルの関数またはサブプログラム名の大文字 / 小文字が、iapxtb 表の文字と一致しているかどうかを確認してください。

ベンダー提供のデバッガ・プログラム

ベンダー提供のデバッガとプリコンパイラに互換性がない場合があります。デバッガで動作するプログラムが、オペレーティング・システムでは動作しない可能性があります。

ireclen および oreclen の値

ireclen および oreclen パラメータに最大値はありません。

静的および動的リンク

プリコンパイラおよび OCI アプリケーションは、静的または動的に Oracle ライブラリにリンクできます。静的リンクの場合、アプリケーション全体のライブラリおよびオブジェクトは、1つの実行ファイル・プログラムにリンクされます。そのため、アプリケーションの実行ファイルのサイズは非常に大きくなります。

動的リンクの場合、実行コードは一部分のみが実行ファイル・プログラムに存在し、他の部分は実行時にアプリケーションと動的にリンクされるライブラリに存在します。実行時にリンクされるライブラリを、動的ライブラリまたは共有ライブラリといいます。動的リンクのメリットは、次のとおりです。

- 必要なディスク領域が少なくて済む: 複数のアプリケーションまたは同一アプリケーションの呼出しで、同じ動的ライブラリを使用できます。
- 必要なメイン・メモリーが少なくて済む: 同一の動的ライブラリ・イメージをメイン・メモリーに1回のみロードすれば、複数のアプリケーションで共有できます。

Oracle 共有ライブラリ

Oracle 共有ライブラリは、\$ORACLE_HOME/lib ディレクトリにある libclntsh.so です。オラクル社が提供している Make ファイル demo_product.mk を使用してアプリケーションをリンクする場合、デフォルトで Oracle 共有ライブラリが使用されます。

実行ファイルの起動時に次のエラーが発生する場合があります。

```
$ sample1
sample1: error in loading shared libraries: libclntsh.so.8.0: cannot open shared
object file: No such file on directory
```

このエラー・メッセージが表示された場合、次のように入力して、LD_LIBRARY_PATH を設定します。

```
$ set LD_LIBRARY_PATH $ORACLE_HOME/lib
```

Oracle 共有ライブラリは、インストール時に自動的に作成されます。Oracle 共有ライブラリを再作成する必要がある場合は、Oracle 共有ライブラリを使用している SQL*Plus、Recovery Manager などのすべてのクライアント・アプリケーションを終了します。Oracle ユーザーとしてログインし、次のように入力します。

```
$ cd $ORACLE_HOME/rdbms/lib
$ make -f ins_rdbms.mk libclntsh.so
```

Pro*C/C++ プリコンパイラ

Pro*C/C++ を使用する前に、オペレーティング・システムの適切なバージョンのコンパイラが正しくインストールされていることを確認してください。

参照： オペレーティング・システムの要件については、『Oracle8i for Linux Intel インストレーション・ガイド』を参照してください。

プリコンパイラおよびインタフェース機能の詳細は、『Oracle8i Pro*C/C++ 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++ デモ・プログラム

デモ・プログラムは、Pro*C/C++ プリコンパイラの様々な機能を参照するために提供されています。デモ・プログラムには、C、C++ および Object プログラムの3種類があります。Object プログラムのデモ・プログラムでは、Oracle8i Object の新機能が表示されます。すべてのデモ・プログラムは、\$ORACLE_HOME/precomp/demo/proc ディレクトリにあります。プログラムでは、\$ORACLE_HOME/sqlplus/demo/demobld.sql で作成したデモンストレーション表が SCOTT スキーマにあり、パスワードは TIGER であると仮定しています。デフォルトでは、すべてのプログラムは、クライアント共有ライブラリ \$ORACLE_HOME/lib/libclntsh.so に動的にリンクされます。

Make ファイル `demo_proc.mk` は、`$ORACLE_HOME/precomp/demo/proc/` ディレクトリにあり、デモ・プログラムの作成に必要です。

たとえば、デモ・プログラム `sample1` をプリコンパイル、コンパイルおよびリンクする場合は、次のように入力します。

```
$ make -f demo_proc.mk sample1
```

Pro*C/C++ の C デモ・プログラムを作成するには、次のように入力します。

```
$ make -f demo_proc.mk samples
```

Pro*C/C++ の C++ デモ・プログラムを作成するには、次のように入力します。

```
$ make -f demo_proc.mk cppsamples
```

Pro*C/C++ の Object デモ・プログラムを作成するには、次のように入力します。

```
$ make -f demo_proc.mk object_samples
```

デモ・プログラムによっては、`$ORACLE_HOME/precomp/demo/sql` 内の SQL スクリプトを実行する必要があります。スクリプトを実行しない場合、実行を要求するメッセージが表示されます。このようなデモ・プログラムを作成し、それに対応する SQL スクリプトを実行するには、コマンドラインに `make` マクロ引数 `RUNSQL=run` を追加する必要があります。たとえば、`calldemo` デモ・プログラムを作成し、必要な `$ORACLE_HOME/precomp/demo/sql/calldemo.sql` スクリプトを実行する場合は、次のように入力します。

```
$ make -f demo_proc.mk calldemo RUNSQL=run
```

すべての Object デモ・プログラムを作成し、必要な SQL スクリプトをすべて実行するには、次のように入力します。

```
$ make -f demo_proc.mk object_samples RUNSQL=run
```

注意： Make ファイル `demo_proc.mk` では、環境変数 `ORA_CLIENT_LIB` が使用されなくなりました。

参照： SQL*Plus を使用したデモ・プログラムの作成については、[3-3 ページの「デモンストレーション表の使用」](#)を参照してください。

Pro*C/C++ ユーザー・プログラム

\$ORACLE_HOME/precomp/demo/proc/ ディレクトリにある Make ファイル `demo_proc.mk` を使用して、プログラムを作成できます。`demo_proc.mk` でプログラムをリンクするコマンドの構文は、次のとおりです。

```
$ make -f demo_proc.mk target OBJS="objfile1 objfile2 ..." EXE=exename
```

この場合、次のような意味になります。

- `target` は、Make ファイルが使用するターゲットです（たとえば、`build`）。
- `objfilen` は、プログラムにリンクするオブジェクト・ファイルです。
- `EXE` は、実行プログラムです。

たとえば、Pro*C/C++ ソース `myprog.pc` からプログラム `myprog` を作成する場合、実行ファイルのソースとタイプに応じて、次のいずれかのコマンドを入力します。

クライアント共有ライブラリに動的にリンクする C ソースの場合、次のように入力します。

```
$ make -f demo_proc.mk build OBJS=myprog.o EXE=myprog
```

静的にリンクする C ソースの場合、次のように入力します。

```
$ make -f demo_proc.mk build_static OBJS=myprog.o EXE=myprog
```

クライアント共有ライブラリに動的にリンクする C++ ソースの場合、次のように入力します。

```
$ make -f demo_proc.mk cppbuild OBJS=myprog.o EXE=myprog
```

静的にリンクする C++ ソースの場合、次のように入力します。

```
$ make -f demo_proc.mk cppbuild_static OBJS=myprog.o EXE=myprog
```

注意： 前述の例では、ファイル `myprog.o` は、コンパイラによって生成されるオブジェクト・ファイルです。

Oracle コール・インタフェース

Oracle コール・インタフェース (OCI) を使用する前に、適切なバージョンの Pro*C/C++ が正しくインストールされていることを確認してください。

参照： ご使用のオペレーティング・システムに必要なバージョンについては、『Oracle8i for Linux Intel インストレーション・ガイド』を参照してください。

OCI デモ・プログラム

デモ・プログラムは、OCI の様々な機能を参照するために提供されています。デモ・プログラムには、C および C++ の 2 種類があります。すべてのデモ・プログラムは、`$ORACLE_HOME/rdbms/demo` にあります。デフォルトでは、すべてのプログラムは、クライアント共有ライブラリ `$ORACLE_HOME/lib/libclntsh.so` に動的にリンクされます。

ほとんどのデモ・プログラムでは、`$ORACLE_HOME/sqlplus/demo/demobld.sql` で作成したデモンストレーション表が SCOTT にあり、パスワードは TIGER であると仮定しています。

`$ORACLE_HOME/rdbms/demo` ディレクトリにある Make ファイル `demo_rdbms.mk` を使用して、デモ・プログラムを作成できます。たとえば、`cdemo1` をコンパイルおよびリンクするには、次のように入力します。

```
$ make -f demo_rdbms.mk cdemo1
```

OCI の C デモ・プログラムを作成するには、次のように入力します。

```
$ make -f demo_rdbms.mk demos
```

OCI の C++ デモ・プログラムを作成するには、次のように入力します。

```
$ make -f demo_rdbms.mk c++demos
```

デモ・プログラムによっては、`$ORACLE_HOME/rdbms/demo` にある SQL スクリプトを実行する必要があります。スクリプトを実行しないと、スクリプトの実行を要求するメッセージが表示されます。ほとんどの場合、SQL スクリプト名はプログラム名と同じで、拡張子は `.sql` です。

参照： SQL*Plus を使用したデモ・プログラムの作成については、[3-3 ページの「デモンストレーション表の使用」](#)を参照してください。

デモ・プログラムの詳細は『Oracle8i コール・インタフェース・プログラマーズ・ガイド』を、各プログラムの詳細はプログラム・ソースを参照してください。

OCI ユーザー・プログラム

\$ORACLE_HOME/rdbms/demo ディレクトリにある Make ファイル demo_rdbms.mk を使用して、プログラムを作成できます。demo_rdbms.mk を使用してユーザー・プログラムをリンクするコマンドの構文は、次のとおりです。

```
$ make -f demo_rdbms.mk target OBJS="objfile1 objfile2 ..." EXE=exename
```

この場合、次のような意味になります。

- target は、Make ファイルが使用するターゲットです（たとえば、build）。
- objfilen は、プログラムにリンクするオブジェクト・ファイルです。
- EXE は、実行プログラムです。

たとえば、C/C++ ソース myprog.c からプログラム myprog を作成する場合、実行ファイルのタイプに応じて、次のいずれかのコマンドを入力します。

C ソースからクライアント共有ライブラリに動的にリンクするプログラム myprog を作成する場合、次のように入力します。

```
$ make -f demo_rdbms.mk build OBJS=myprog.o EXE=myprog
```

C ソースから静的にリンクするプログラム myprog を作成する場合、次のように入力します。

```
$ make -f demo_rdbms.mk build_static OBJS=myprog.o EXE=myprog
```

C++ ソースからクライアント共有ライブラリに動的にリンクするプログラム myprog を作成する場合、次のように入力します。

```
$ make -f demo_rdbms.mk buildc++ OBJS=myprog.o EXE=myprog
```

C++ ソースから静的にリンクするプログラム myprog を作成する場合、次のように入力します。

```
$ make -f demo_rdbms.mk buildc++_static OBJS=myprog.o EXE=myprog
```

注意： 前述の例では、ファイル myprog.o は、コンパイラによって生成されるオブジェクト・ファイルです。

カスタム Make ファイル

オラクル社が提供している Make ファイル `demo_product.mk` を使用してユーザー・プログラムをリンクすることをお勧めします。この Make ファイルを修正する必要がある場合、またはカスタム Make ファイルを使用する場合には、次の制限事項があります。

- Oracle ライブラリの順番は変更しないでください。リンク中にすべてのシンボルを解決するために、Oracle ライブラリはリンク・ラインに 2 回以上追加されます。Oracle ライブラリの順番が重要である理由は、次のとおりです。
 - Oracle ライブラリは、相互に参照し合います。ライブラリ A の関数はライブラリ B の関数を呼び出し、ライブラリ B の関数はライブラリ A の関数を呼び出します。
 - Linux リンカーは、1 パスのリンカーです。リンカーは、リンク・ラインに出現した時点で 1 回のみライブラリを検索します。
- リンク・ラインに独自のライブラリを追加する場合は、リンク・ラインの最初または最後に追加します。ユーザー・ライブラリは、Oracle ライブラリの上に置かないでください。
- `nmake` などの `make` ユーティリティを使用する場合は、マクロおよび接尾辞の処理について、Linux GNU で提供されている `make` ユーティリティとの違いに注意してください。Oracle の Make ファイルは、すでにテスト済で、Linux GNU の `make` ユーティリティでサポートされています。
- Oracle ライブラリの名前および内容は、リリース間で変更されることがあります。必要なライブラリを判断するには、現行のリリースで提供されている Make ファイル `demo_product.mk` を必ず使用してください。

未定義シンボルの修正

オラクル社が提供している `symfind` ユーティリティを使用すると、シンボルが定義されているライブラリまたはオブジェクト・ファイルの場所を調べる場合に役立ちます。プログラムのリンク時に共通するエラーに、未定義シンボルによって、次のようなエラー・メッセージが表示されます。

```
$ make -f demo_proc.mk sample1
sample1.o:In function my_sample1
sample1.o (.text + 0x90f): undefined reference to sqlcex
collect2: ld returned 1 exit status
make: *** [sample1] Error 1
```

```
ld: fatal: Symbol referencing errors. No output written to sample1
```

このエラーは、参照するシンボルの定義をリンカーが検出できなかった場合に発生します。このエラー・メッセージが表示された場合、シンボルが定義されているライブラリまたはオブジェクト・ファイルがリンク・ラインにあるかどうか、およびリンカーが検索しているファイルのディレクトリが正しいかどうかを確認します。

表 4-3 2 タスク通信で使用するシグナル（続き）

| シグナル | 説明 |
|---------|--|
| SIGIO | Net8 プロトコル・メソッドが、ネットワーク・イベントの着信を示すために、SIGIO を使用します。 |
| SIGPIPE | パイプ・ドライバが、通信チャネルでファイルの終わりを検出するために、SIGPIPE を使用します。パイプへの書き込み時に、読取りプロセスが存在していなければ、SIGPIPE シグナルが書き込みプロセスに送信されます。SIGPIPE は、Oracle プロセスおよびユーザー・プロセスが受け取ります。SIGCLD は SIGPIPE と似ていますが、ユーザー・プロセスに適用され、Oracle プロセスには適用されません。 |
| SIGTERM | パイプ・ドライバが、ユーザー側から Oracle プロセスに割込みシグナルを送る場合に SIGTERM を使用します。ユーザーが割込みキー ([Ctrl]+[c]) を押すと、このシグナルが送られます。SIGTERM は、ユーザー・プロセスではなく、Oracle プロセスが受け取ります。 |
| SIGURG | Net8 TCP/IP ドライバが、アウトオブバンド・ブレイクをユーザー・プロセスから Oracle プロセスへ送信する場合に、SIGURG を使用します。 |

シグナルは、Pro*C またはその他のプリコンパイラ・アプリケーションに影響します。Oracle プロセスへの接続時に、SIGCLD（または SIGCHLD）および SIGPIPE に 1 つのシグナル・ハンドラをインストールできます。osnsui() ルーチン呼び出しで設定すると、複数のシグナル・ハンドラを SIGINT 用にインストールできます。SIGINT の場合、osnsui() および osncui() を使用してシグナル受取りルーチンを登録および削除します。その他のシグナルにも、シグナル・ハンドラをインストールできます。Oracle プロセスに接続していない場合は、複数のシグナル・ハンドラをインストールできます。

次に、シグナル・ルーチンおよび受取りルーチンの設定方法を示します。

```
/* user side interrupt set */
word osnsui( /*_ word *handlp, void (*astp), char * ctx, _*/)
/*
** osnsui: Operating System dependent Network Set
**User-side
** Interrupt. Add an interrupt handling procedure
**astp.
** Whenever a user interrupt(such as a ^C) occurs,
**call astp
** with argument ctx. Put in *handlp handle for this
**handler so that it may be cleared with osncui.
** Note that there may be many handlers; each should
** be cleared using osncui. An error code is
**returned if an error occurs.
*/
```

```
/* user side interrupt clear */
word osncui( /*_ word handle _*/ );
/*
** osncui: Operating System dependent Clear User-side
**Interrupt.
** Clear the specified handler. The argument is the
**handle obtained from osnsui. An error code is
** returned if an error occurs.
*/
```

次に、アプリケーション・プログラムで `osnsui()` および `osncui()` を使用する場合のテンプレートを示します。

```
/*
** User interrupt handler template.
*/
void sig_handler()
{
    ...
}

main(argc, argv)
int arc;
char **argv;
{
    int handle, err;
    ...

    /* set up my user interrupt handler */
    if (err = osnsui(&handle, sig_handler, (char *) 0))
    {
        /* if the return value is non-zero, an error has occurred
        Take appropriate action for the error. */
        ...
    }
    ...
    /* clear my interrupt handler */
    if (err = osncui(handle))
    {
        /* if the return value is non-zero, an error has occurred
        Take appropriate action for the error. */
        ...
    }
    ...
}
```

XA 機能

Oracle コール・インタフェースには、XA 機能があります。TP モニター XA アプリケーションを作成する場合、TP モニター・ライブラリ（シンボル `ax_reg` および `ax_unreg` を定義するライブラリ）が、リンク・ラインで Oracle のクライアント共有ライブラリより前に設定されていることを確認してください。このリンク制限は、XA の動的登録（Oracle XA スイッチ `xaoswd`）を使用する場合にのみ必要です。

Oracle8i では、Oracle7 リリース 7.1.6 の XA コールをサポートしていません（Oracle7 リリース 7.3 の XA コールはサポートしています）。そのため、Oracle7 リリース 7.1.6 の XA コールを使用する TP モニター XA アプリケーションは、Oracle8i の XA ライブラリにリンクする必要があります。Oracle8i の XA コールは、共有ライブラリ `$ORACLE_HOME/lib/libclntsh.so` および静的ライブラリ `$ORACLE_HOME/lib/libclient8.a` の両方で定義されています。

Net8 の構成

この章では、Linux Intel での、Net8 の次の機能について説明します。

- [主な Net8 製品および特徴](#)
- [Net8 プロトコルのサポート](#)
- [BEQ プロトコル](#)
- [IPC プロトコル](#)
- [RAW プロトコル](#)
- [TCP/IP プロトコル](#)
- [Oracle Enterprise Manager](#)
- [Oracle SNMP での Oracle Intelligent Agent の構成](#)
- [Oracle Advanced Security](#)
- [参照ドキュメント](#)

主な Net8 製品および特徴

この項では、主な Net8 製品および機能について説明します。

Net8 ファイルおよびユーティリティ

Net8 の構成ファイルの保存場所

Linux では、Net8 構成ファイルのデフォルトのディレクトリは `/etc` です。

Net8 は、グローバル・ファイルを次の順に検索します。

1. 環境変数 `TNS_ADMIN` に設定されたディレクトリ（設定されている場合）
2. `/etc` ディレクトリ
3. `$ORACLE_HOME/network/admin`

次のように入力して、すべてのネットワーク・ユーザーの起動ファイルに環境変数 `TNS_ADMIN` を設定します。

Bourne または Korn シェルの場合、次のように入力します。

```
$ TNS_ADMIN=directory_path
$ export TNS_ADMIN
```

C シェルの場合、次のように入力します。

```
% setenv TNS_ADMIN directory_name
```

システム・レベルの構成ファイルごとに、ローカルのプライベート構成ファイル（ユーザーのホーム・ディレクトリに保存されている）があります。プライベート・ファイルの設定は、システム・レベル・ファイルの設定を上書きします。`sqlnet.ora` のプライベート構成ファイルは、`$HOME/.sqlnet.ora` です。`tnsnames.ora` のプライベート構成ファイルは、`$HOME/.tnsnames.ora` です。これらのファイルの構文は、対応するシステム・ファイルの構文と同じです。

サンプル構成ファイル

`cman.ora`、`listener.ora`、`names.ora`、`sqlnet.ora`、`tnsnames.ora` 構成ファイルの例は、`$ORACLE_HOME/network/admin/samples` にあります。

アダプタ・ユーティリティ

Net8 では、様々なネットワーク・プロトコルおよびネーミング方法をサポートしています。それらは、個々の実行ファイルにリンクされ、ネットワーク・プロトコルと Net8 間のインタフェースを提供します。インストールされている Net8 プロトコルを表示するには、次のように入力します。

```
$ adapters
```

指定した実行ファイルにリンクされているアダプタを表示するには、次のように入力します。

```
$ adapters executable
```

たとえば、次のコマンドを実行すると、Oracle 実行ファイルにリンクされている Net8 プロトコルが表示されます。

```
$ adapters oracle
```

Net8 プロトコルが、次のように表示されます。

```
Net8 Protocol Adapters linked with oracle are:
```

```
BEQ Protocol Adapter
```

```
IPC Protocol Adapter
```

```
TCP/IP Protocol Adapter
```

```
RAW Protocol Adapter
```

```
Net8 Naming Adapters linked with oracle are:
```

```
Oracle TNS Naming Adapter
```

```
Oracle Naming Adapter
```

```
Oracle Advanced Security/Networking Security products linked with oracle are:
```

Oracle Connection Manager

Oracle Connection Manager (OCM) は、クライアント接続要求が次のホップに送られるか、または直接データベース・サーバーに送られる際のルーターです。Connection Manager を介して接続要求を送信するクライアントは、Connection Manager で構成された接続集中化、Net8 アクセス制御または複数プロトコル・サポート機能を利用できます。

Oracle Connection Manager はクライアントから受信した要求をリスニングして、宛先サービスへの接続要求を開始します。Oracle Connection Manager は、Oracle Connection ゲートウェイ・プロセスおよび Oracle Connection Manager 管理プロセスを使用して、これらのタスクを実行します。

OCM には、TCP/IP 環境で指定されたサーバーへのクライアント・アクセスを制御するために使用できる機能も含まれています。あるフィルタ・ルールを設定することによって、次の基準に基づいたサーバーへの特定のクライアント・アクセスを可能にしたり、制限することができます。

- クライアントのソース・ホスト名または IP アドレス

- サーバーの接続先ホスト名または IP アドレス
- 接続先データベース・サービス名

Net8 のファイアウォール・プロキシ

ファイアウォール・ベンダーによっては、Net8 のファイアウォール・プロキシを提供しているものもあります。これは、アプリケーションのプロキシを必要とするファイアウォールにインストールされます。Net8 のファイアウォール・プロキシには、Oracle Connection Manager と同じアクセス制御機能があります。

参照： Oracle Connection Manager の詳細は、『Oracle8i Net8 管理者ガイド』を参照してください。

サーバー・モデル

クライアント / サーバーの Net8 接続は、次の 2 つのモデルを使用して構築されます。

- マルチスレッド・サーバー・モデル：リスナーは、ディスパッチャに接続要求を渡すか、またはディスパッチャを介して、接続するクライアントをリダイレクトします。

マルチスレッド・サーバー（MTS）構成では、クライアントのユーザー・プロセスはディスパッチャに接続します。ディスパッチャは、同時に複数のクライアント接続をサポートできます。各クライアント接続は、仮想サーキットにバインドされます。仮想サーキットは、ディスパッチャがクライアントのデータベース接続の要求および応答に使用する共有メモリーの一部です。この方法によって、サーバー・プロセスの小さなプールで大量のクライアントを処理できるようになります。専用サーバー・モデルに比べて、MTS モデルの大きなメリットは、システム・リソースが少なくすむため、ユーザー数の増加をサポートできることです。

- 専用サーバー・モデル：リスナーは専用サーバーを起動し、専用サーバーに接続要求を渡すか、または専用サーバーにクライアントをリダイレクトします。

専用サーバー・モデルは、クライアントごとに 1 つの単一のサーバー・プロセスがあります。クライアントが専用サーバーに接続するには、リスナーとインスタンスが同じマシンで実行されている必要があります。専用サーバー・プロセスでは、MTS より多くのメモリーが必要です。

参照： マルチスレッド・サーバーの詳細は、『Oracle8i Net8 管理者ガイド』を参照してください。

Oracle Names

Oracle Names は、ネットワーク上のすべてのデータベース・サービスの名前およびアドレスを格納する Oracle 固有のソフトウェアです。サーバーに接続するクライアントは、接続要求を Oracle Names Server に送ります。Oracle Names Server は、ネットワーク・アドレスに名前を変換し、クライアントに情報を戻します。

電話番号を格納するディレクトリ・アシスタンスを使用するコール側と同様、Oracle Names を使用するために構成されたクライアントは、それぞれの接続要求を Oracle Names Server に示します。Oracle Names Server は、クライアントによって提供されるサービス名をネットワーク・アドレスに変換しようとします。Oracle Names Server がネットワーク・アドレスを検索し、クライアントに情報を戻します。クライアントはそのアドレスを使用してサービスに接続します。

参照： Oracle Names の詳細は、『Oracle8i Net8 管理者ガイド』を参照してください。

Net8 プロトコルのサポート

Linux がサポートする Net8 リリース 8.1.7 用のプロトコルは、BEQ プロトコル、IPC プロトコル、RAW プロトコル、および TCP/IP プロトコルです。

TCP/IP プロトコルをインストールする前に、適切なオペレーティング・システム・ソフトウェアがインストールおよび構成されていることを確認してください。BEQ および IPC Net8 プロトコルに固有のオペレーティング・システム要件はありません。

参照： 要件の詳細は、『Oracle8i for Linux Intel インストレーション・ガイド』を参照してください。

ADDRESS の指定

IPC および TCP/IP Net8 プロトコルには、それぞれのプロトコル固有の ADDRESS 指定があります。ADDRESS 指定は、Net8 構成ファイルおよび `initSID.ora` ファイル中のデータベースの初期化パラメータ `MTS_LISTENER_ADDRESS` を設定する場合に必要です。詳細は、この章に記載している各プロトコルの項を参照してください。

表 5-1 に、各プロトコルの ADDRESS 指定を示します。

表 5-1 ADDRESS 指定

| サポート・ プロトコル | ADDRESS 指定 |
|----------------|--|
| BEQ | (ADDRESS= (PROTOCOL = BEQ) (PROGRAM = ORACLE_HOME/bin/oracle) (ARGV0 = oracleORACLE_SID) (ARGS = ' (DESCRIPTION= (LOCAL=YES) (ADDRESS= (PROTOCOL=BEQ))) ') (ENVS = 'ORACLE_HOME=ORACLE_HOME,ORACLE_SID=ORACLE_SID')) |
| IPC | (ADDRESS= (PROTOCOL=IPC) (KEY=key)) |
| RAW | なし |
| TCP/IP | (ADDRESS= (PROTOCOL=TCP) (HOST=hostname) (PORT=port_id)) |

BEQ プロトコル

BEQ プロトコルは、通信メカニズムであり、また、プロセス起動メカニズムでもあります。クライアントおよびサーバーは、同一のマシン上に存在する必要があります。ネット・サービス名を指定（コマンドラインまたはログイン画面でユーザーが直接指定するか、または TWO_TASK などの環境変数で間接的に指定する）しなかった場合、BEQ プロトコルが使用されます。その場合、専用サーバーが使用され、マルチスレッド・サーバーは使用されません。この専用サーバーは、BEQ プロトコルによって自動的に起動された後、サーバー・プロセスが起動されて既存の SGA に割り当てられるのを待ちます。サーバー・プロセスが正常に起動されると、BEQ プロトコルで、UNIX パイプを介してプロセス間通信ができるようになります。

BEQ プロトコルが動作するためにリスナーは不要です。これは、プロトコルがクライアント・ツールにリンクされていて、外部から操作しなくてもそのサーバー・プロセスを直接起動するためです。ただし、BEQ プロトコルは、クライアント・プログラムおよび Oracle8i が同一のマシンに常駐する場合にのみ使用できます。BEQ プロトコルは常にインストールされ、すべてのクライアント・ツールおよび Oracle8i サーバーにリンクされています。

ADDRESS のキーワードとその値の組合せが BEQ プロトコルの接続パラメータです。通常、ADDRESS の指定は、接続記述子、構成ファイルなどを構成している内容の一部です。パラメータの入力順に決まりはありません。

```
(ADDRESS =
  (PROTOCOL = BEQ)
  (PROGRAM = ORACLE_HOME/bin/oracle)
  (ARGV0 = oracleORACLE_SID)
  (ARGS = '(DESCRIPTION=(LOCAL=YES) (ADDRESS=(PROTOCOL=BEQ)))')
  (ENVS = 'ORACLE_HOME=ORACLE_HOME,ORACLE_SID=ORACLE_SID')
)
```

表 5-2 に、BEQ プロトコルの接続パラメータの構文を示します。

表 5-2 BEQ プロトコルの接続パラメータの構文

| | |
|----------|---|
| PROTOCOL | 使用するプロトコルを指定します。 値は beq で、大文字または小文字のどちらかで指定します。 |
| PROGRAM | Oracle 実行ファイルのフルパス名を指定します。 |
| ARGV0 | ps コマンドで表示されるプロセスの名前を指定します。推奨値は oracleORACLE_SID です。 |
| ARGS | '(DESCRIPTION=(LOCAL=YES) (ADDRESS=(PROTOCOL=BEQ)))' |
| ENVS | 環境を指定します。ORACLE_HOME には、接続先データベースの ORACLE_HOME ディレクトリをフルパスで指定します。ORACLE_SID には、接続先データベースのシステム識別子を指定します。 |

次に、BEQ ADDRESS の例を示します。

```
(ADDRESS =
  (PROTOCOL = BEQ)
  (PROGRAM = /u01/app/oracle/product/8.1.7/bin/oracle)
  (ARGV0 = oracleV817)
  (ARGS = '(DESCRIPTION=(LOCAL=YES) (ADDRESS=(PROTOCOL=BEQ)))')
  (ENVS = 'ORACLE_HOME=/u01/app/oracle/product/8.1.7,ORACLE_SID=V817')
)
```

IPC プロトコル

IPC プロトコルは、クライアント・プログラムおよび Oracle8i が同一マシンに常駐する場合にのみ使用できるという点が、BEQ プロトコルと似ています。IPC プロトコルと BEQ プロトコルの違いは、IPC プロトコルが専用サーバーおよびマルチスレッド・サーバーで使用できることです。IPC プロトコルが動作するには、リスナーが必要です。IPC プロトコルは常にインストールされ、すべてのクライアント・ツールおよび Oracle8i にリンクされています。

IPC プロトコルでは、Oracle7 リリース 7.1 以降、UNIX での UNIX Domain Socket ファイルの保存場所が変わりました。そのため、Oracle7 リリース 7.1 を Oracle8i と同じマシンにインストールし、2つのインスタンス間で IPC 接続しようとするとう失敗します。この問題を解決するには、IPC ファイルが以前に保存されていたディレクトリ (/var/tmp/o)、および現在保存されているディレクトリ (/var/tmp/.oracle) 間にシンボリック・リンクを作成します。

ADDRESS のキーワードとその値の組合せが IPC プロトコルの接続パラメータです。通常、ADDRESS の指定は、接続記述子、構成ファイルなどを構成している内容の一部です。パラメータの入力順に決まりはありません。

```
(ADDRESS=
  (PROTOCOL=IPC)
  (KEY=key)
)
```

表 5-3 に、IPC プロトコルの接続パラメータの構文を示します。

表 5-3 IPC プロトコルの接続パラメータの構文

| | |
|----------|---|
| PROTOCOL | 使用する IPC プロトコルを指定します。 値は ipc で、大文字または小文字のどちらかで指定します。 |
| KEY | データベースのサービス名またはデータベースの識別子 (ORACLE_SID) を指定します。 |

次に、IPC ADDRESS の例を示します。

```
(ADDRESS=
  (PROTOCOL=IPC)
  (KEY=PROD)
)
```

RAW プロトコル

データがクライアント / サーバー間で転送されると、Net8 は、そのヘッダー情報を各ネットワーク・パケットに追加します。Raw Transport 機能によって、ネットワークを介する各パケットのヘッダー情報を最小限に抑えることができます。

接続後、2 種類の情報（データおよびブレイク・ハンドリング）がネットワーク上に送信されます。接続パケットには、適切に接続するための Net8 のヘッダー情報が必要です。ただし、接続後、すべてのデータ・パケットは Net8 のヘッダー情報から削除され、Net8 のネットワークおよびプロトコル・レイヤーを通過して、直接オペレーティング・システムに渡されます。データが流れるプロトコル・スタック・レイヤーの数が減り、ネットワーク上に送信されるデータのバイト数が減るため、接続のパフォーマンスが向上します。

この機能は、ユーザーが意識しなくても、必要な場合に有効になります。つまり、ヘッダー情報を送信しなくてもよい場合、ヘッダーは削除されます。暗号化および認証では、各情報のパケットと一緒に特定の情報を送信する必要があります。そのため、Raw Transport は有効になりません。

この機能を使用する場合、特別な構成はありません。Net8 では、ユーザーが意識しなくても、Raw Transport モードに切り替える必要があるかどうか判断され、必要な場合には切り替えられます。

TCP/IP プロトコル

リスナー・ポートを定義するネットワーク上の各ノードの `/etc/services` ファイルで、リスナー用のポートを予約してください。ほとんどの場合、ポートは 1521 です。リスナー名とポート番号という形式で入力します。たとえば、次のように入力します。

```
listener      1521/tcp
```

この場合、`listener` は `listener.ora` で定義されているリスナーの名前を示します。複数のリスナーを起動する場合は、複数のポートを予約する必要があります。

TCP/IP の ADDRESS 指定

ADDRESS のキーワードとその値の組合せが TCP/IP プロトコルの接続パラメータです。通常、ADDRESS の指定は、接続記述子、構成ファイルなどを構成している内容の一部です。パラメータの入力順に決まりはありません。

```
(ADDRESS=
  (PROTOCOL=TCP)
  (HOST=hostname)
  (PORT=port_id)
)
```

表 5-4 に、TCP/IP プロトコルの接続パラメータの構文を示します。

表 5-4 TCP/IP プロトコルの接続パラメータの構文

| | |
|----------|---|
| PROTOCOL | 使用するプロトコルを指定します。 値は大文字でも小文字でもかまいません。デフォルトは tcp です。 |
| HOST | ホスト名またはホスト IP アドレスを指定します。 |
| PORT | TCP/IP のポートを指定します。/etc/services ファイルで指定されている番号または名前を指定します。推奨値は 1521 です。 |

次に、MADRID ホストにクライアントを指定する TCP/IP ADDRESS の例を示します。

```
(ADDRESS=
  (PROTOCOL=TCP)
  (HOST=MADRID)
  (PORT=1521)
)
```

最後のフィールドは、PORT=listener というように名前を指定することもできます。

Oracle Enterprise Manager

エージェント・サービスの展開および自動構成

Simple Network Management Protocol (SNMP) システムで統合しない限り、Oracle Intelligent Agent での構成は必要ありません（「[Oracle SNMP での Oracle Intelligent Agent の構成](#)」を参照）。

参照： 詳細は、『Oracle8i Net8 管理者ガイド』を参照してください。

Tcl スクリプトのデバッグ

oratclsh 実行ファイルは、tcl スクリプトをデバックする場合に必要です。oratclsh を実行する前に、環境変数 TCL_LIBRARY に、\$ORACLE_HOME/network/agent/tcl を設定してください。

参照： 詳細は、『Oracle Enterprise Manager 開発者ガイド』を参照してください。

Oracle SNMP での Oracle Intelligent Agent の構成

Oracle Intelligent Agent が動作するために、Simple Network Management Protocol (SNMP) は必要ありませんが、Intelligent Agent を起動する前に Oracle SNMP 用に構成することができます。後述する構成ファイルはすべて、`$ORACLE_HOME/network/snmp/peer` ディレクトリに保存されています。

マスター・エージェントの構成

`CONFIG.master` ファイルで、次の変更を行います。

1. `MANAGER` で始まる行を検索します。
2. `ipaddr` フィールド (130.35.10.210 とコード化されている) を、SNMP 検出メッセージを送信するマシンの IP アドレスまたはホスト名に変更します。

ファイル内の記述に従って、`CONFIG.master` ファイルに他の変更を加えることもできます。

カプセル化プログラムの構成

1. `snmpd.conf` ファイルに次の行を追加します。

```
traphostname_or_IP_address
```

この場合、`hostname_or_IP_address` はローカル・マシンの IP アドレスを表します。

2. `CONFIG.encap` ファイルでは、オプションとしてデフォルト・ファイルのポート番号 1161 の設定を変更できます。ポート番号を変更する場合、`start_peer` スクリプトの `NEW_SNMPD_PORT` に対するポート番号も変更する必要があります。

`NEW_SNMPD_PORT` は、`snmpd` エージェント (ネイティブな Linux Intel SNMP エージェント) がリスニングするポートです。`CONFIG.encap` ファイルで指定されているポートが同じであることを確認してください。`NEW_TRAPD_PORT` は、`snmpd` エージェントがトラップを送信する `PEER` カプセル化プログラムのポートです。

`start_peer` スクリプトの `NEW_SNMPD_PORT` および `NEW_TRAPD_PORT` は、異なるポート番号である必要があります。`NEW_TRAPD_PORT` のポート番号の変更もできます。

start_peer スクリプトの確認

start_peer スクリプトは、次のような行を含みます。

```
SNMPD = snmpd_executable_path
```

ご使用のシステムの snmpd 実行ファイルが、start_peer スクリプトによって示された位置にない場合、snmpd_executable_path が snmpd 実行ファイルの正しい位置を示すように編集します。

SNMP コンポーネントの起動

次の手順を実行して、SNMP コンポーネントを起動します。

1. 次のように入力して、SNMP コンポーネントの master_peer、encap_peer および snmpd が実行されていないことを確認します。

```
$ ps -aef | grep peer
$ ps -aef | grep snmp
```

実行されているコンポーネントがあった場合、次に進む前に root ユーザーでログインして、kill コマンドを使用してプロセスを終了させます。

2. root ユーザーで start_peer スクリプトを実行し、PEER マスター・エージェント、PEER カプセル化プログラム、およびネイティブな Linux Intel SNMP エージェントを起動します。

```
# cd $ORACLE_HOME/network/snmp/peer
# ./start_peer -a
```

注意： ご使用のシステムにネイティブな Linux Intel SNMP エージェントがない場合、PEER カプセル化プログラムは使用できません。マスター・エージェントのみを起動するには、start_peer -m を実行します。

3. 次のように入力して、SNMP コンポーネントが実行されていることを確認します。

```
# ps -aef | grep peer
# ps -aef | grep snmp
```

データベース・サブエージェントの構成および起動

データベース・サブエージェント（Oracle Intelligent Agent）の構成および起動については、『Oracle Enterprise Manager 構成ガイド』を参照してください。

Oracle Advanced Security

.bak ファイル

Oracle Advanced Security のインストール時に、3 つの .bak ファイル（naeet.o.bak、naect.o.bak、および naedhs.o.bak）が作成されます。これらのファイルは、\$ORACLE_HOME/lib に格納されます。これらのファイルは、Oracle Advanced Security の削除（デインストール）中の再リンクに必要なため、削除しないでください。

セキュリティおよびシングル・サインオン

セキュリティおよびシングル・サインオンの構成の詳細は、『Oracle Advanced Security 管理者ガイド』を参照してください。

参照ドキュメント

[表 5-5](#) に、各バンドル製品の README ファイルの保存場所を示します。README ファイルには、前回のリリースからの変更情報が記載されています。

表 5-5 Oracle 製品の README ファイルの保存場所

| 製品 | README ファイル |
|--------------------------|---|
| Net8 | \$ORACLE_HOME/network/doc/README.Net8 |
| Advanced Security | \$ORACLE_HOME/network/doc/README.ASO |
| Oracle Intelligent Agent | \$ORACLE_HOME/network/doc/README.oemagent |

Oracle データ・オプション・デモの実行

- [Oracle8i interMedia](#)
- [Oracle8i Time Series デモ](#)
- [Oracle8i Spatial](#)

Oracle8i *interMedia*

Oracle8i *interMedia* には、次のコンポーネントがあります。

- Text
- Audio、Video および Image
- Locator

Text

Oracle8i には、Text のデモンストレーション（デモ）はありません。ただし、*interMedia* Text には、コード・サンプルが含まれています。次の URL で、ご使用のブラウザを示してください。

`$ORACLE_HOME/ctx/sample/api/index.html`

参照： 詳細は、『Oracle8i *interMedia* Text リファレンス』および『Oracle8i *interMedia* Text 移行ガイド』を参照してください。

Audio、Video および Image

Oracle8i *interMedia* には、次に示すディレクトリにスクリプトおよびサンプル・プログラムが多数あります。

`$ORACLE_HOME/ord/aud/demo/`
`$ORACLE_HOME/ord/img/demo/`
`$ORACLE_HOME/ord/vid/demo/`

サンプル Audio スクリプト

Audio スクリプトは、次のファイルで構成されています。

- `auddemo.sql` - 次に示す Audio オブジェクトの機能を含む Audio デモンストレーションです。
 - *interMedia* オブジェクトをチェックします。
 - デモ内の Audio でサンプル表を作成します。
 - Audio 表に NULL 行を挿入します。
 - アウトラインをチェックします。
 - すべての Audio 属性を直接チェックします。
 - すべての Audio 属性をメソッドを呼び出すことによってチェックします。
 - 2つのファイル（`fpplugins.sql` および `fppluginb.sql`）を使用してユーザー独自の形式のプラグインをインストールします。

- `fplugins.sql` - サポートする任意の形式のプラグインを書き込むためのガイドラインとして使用するデモ形式のプラグイン仕様です。
- `fpluginb.sql` - サポートする任意の形式のプラグインを書き込むためのガイドラインとして使用するデモ形式のプラグイン本体です。

この SQL デモを実行する要件および手順の詳細は、`$ORACLE_HOME/ord/aud/demo` ディレクトリの `README.txt` ファイルを参照してください。

Image の変更または Image インストールのテスト用のサンプル・プログラム

一度 Oracle8i *interMedia* Image をインストールすると、Oracle8i *interMedia* Image デモ・プログラムを実行できます。このプログラムは、インストールが正常に完了したかを確認するためのテストにも使用できます。

この項では、*interMedia* Image デモを作成および実行するために必要な手順を示します。

interMedia Image デモは、`$ORACLE_HOME/ord/img/demo` に保存されています。この場合、`$ORACLE_HOME` は `ORACLE_HOME` ディレクトリです。

デモンストレーション (デモ)・インストールの手順

1. Oracle8i *interMedia* Image デモは、SCOTT/TIGER データベース・ユーザーを使用します。このユーザーが存在しない場合は、作成する必要があります。次のように入力します。

```
% svrmgrl
SVRMGR> connect internal;
SVRMGR> create user SCOTT identified by tiger;
SVRMGR> grant connect,resource to SCOTT;
```

2. `$ORACLE_HOME` が `$ORACLE_HOME` ディレクトリを示すように Image デモ・ディレクトリを作成します。

```
% svrmgrl
SVRMGR> connect internal;
SVRMGR> create or replace directory imgdemodir as '$ORACLE_HOME/ord/img/ demo';
```

3. 次のように入力して、ディレクトリの権限を PUBLIC に与えます。

```
SVRMGR> grant read on directory imgdemodir to public with grant option;
```

4. 必要ならば、imgdemo プログラムを作成します。

```
% cd $ORACLE_HOME/ord/img/demo
% make -f demo_ordimg.mk imgdemo
```

デモの実行

imgdemo ファイルは、Oracle8i *interMedia Image* をプログラム内からどのようにして使用できるかを示すサンプル・プログラムです。デモは C 言語で書かれており、OCI (Oracle Call Interface) を使用して、データベースにアクセスし、Oracle8i *interMedia Image* を実行します。

プログラムは、デモ・ディレクトリ内にビットマップ (BMP) ・イメージがある `imgdemo.dat` で動作します。オプションとして、ファイルがデモと同じディレクトリに常駐する場合は、コマンドラインでイメージ・ファイル名を指定できます。いずれの場合も、一度イメージが Oracle8i *interMedia Image* で処理されると、結果のイメージは `imgdemo.out` ファイルに書き込まれて、指定した共通のレンダリング・ツールで表示されます。

デモが実行されると、デフォルト・データベースの SCOTT/TIGER スキーマ内の IMGDEMOTAB という名前の表が削除および再作成されます。この表は、デモ・データを保持するために使用されます。一度表が作成されると、イメージ・ファイルへの参照が表に挿入されます。次に、データが表にロードされ、ORDImage の `processCopy()` メソッドを使用して JFIF に変換されます。

イメージ・プロパティは、`setProperties()` メソッドを使用してデータベース内に抽出されます。UPDATE コマンドは、`setProperties()` の起動後に発行されます。これは、`setProperties()` の起動がタイプ属性のローカル・コピーを更新するのみなので必須となります。

次に、Oracle8i *interMedia Image* の `process()` メソッドが、データベース内でイメージをカットおよびスケール変更するために使用されます。この次に、変更をコミットする更新が行われます。プログラムは、ピクセル位置 (100,100) から始めて、100 ピクセルの幅、100 ピクセルの高さでイメージの一部をカットします。このサブイメージは、元のサイズの 2 倍の大きさになります。結果のイメージは、現行のディレクトリにある `imgdemo.out` という名前のファイルへ書き込まれます。

例 6-1 コマンドラインでのデモの実行

コマンドラインで `imgdemo` を入力することによって、デモを実行できます。オプションとして、デモ内で異なるイメージを使用できます。そうするには、最初にデモがあるディレクトリにファイルをコピーしてから、`imgdemo` の引数としてコマンドラインでそのファイル名を指定します。

次のコマンドを入力します。

```
$ imgdemo optional-image-filename
```


デモは、その進行状況を示す多くのメッセージを表示します。それとともに、何かが正しく設定されていない場合に発生するエラーも表示します。たとえば、次のようなメッセージが表示されます。

```
Dropping table IMGDEMOTAB...
Creating and populating table IMGDEMOTAB...
Loading data into cartridge...
Modifying image characteristics...
Writing image to file imgdemo.out...
Disconnecting from database...
Logged off and detached from server.
Demo completed successfully.
```

プログラムでエラーが発生した場合、Oracle8i *interMedia Image* ソフトウェアが正しくインストールされない、またはデータベースが起動されない場合があります。プログラムが正常に完了した場合、前述したカットおよびスケールによって処理された元のイメージおよび結果のイメージは共通のイメージ・レンダリング・ツールで表示できます。

サンプル Video スクリプト

Video スクリプトは、次のファイルで構成されています。

- `viddemo.sql` - 次に示す Video オブジェクトの機能を含む Video デモです。
 - *interMedia* オブジェクトをチェックします。
 - デモ内の Video でサンプル表を作成します。
 - Video 表に NULL 行を挿入します。
 - アウトラインをチェックします。
 - すべての Video 属性を直接チェックします。
 - すべての Video 属性をメソッドを呼び出すことによってチェックします。
 - 2つのファイル (`fplugins.sql` および `fpluginb.sql`) を使用してユーザー独自の形式のプラグインをインストールします。
- `fplugins.sql` - サポートする任意の形式のプラグインを書き込むためのガイドラインとして使用するデモ形式のプラグイン仕様です。
- `fpluginb.sql` - サポートする任意の形式のプラグインを書き込むためのガイドラインとして使用するデモ形式のプラグイン本体です。

参照： この SQL デモの実行方法の要件および手順の詳細は、`$ORACLE_HOME/ord/vid/demo` ディレクトリの `README.txt` ファイルを参照してください。

Java デモ

Java デモは、Audio および Video 両方のクライアント側で Java クラスを使用して、ユーザー独自のアプリケーションの作成方法を理解するために役立ちます。これら 2 つのデモでは、Audio および Video オブジェクトがクライアント側で即座に、多数のアクセサ方式が呼ばれます。Audio の Java デモ・ファイルは、`ORACLE_HOME/ord/aud/demo` ディレクトリに保存されています。また、Video の Java デモ・ファイルは、`$ORACLE_HOME/ord/vid/demo` ディレクトリに保存されています。それぞれの Java デモを実行する要件および手順の詳細は、`README.txt` ファイルを参照してください。

MediaAnnotator

MediaAnnotator プログラムは、Oracle8i *interMedia* CD に含まれていません。その他の使用可能な Oracle ソフトウェアとともに、次の URL で検索できます。

http://technet.oracle.com/software/products/intermedia/software_index.htm

参照： 詳細は、『Oracle8i *interMedia* Audio, Image, Video ユーザーズ・ガイドおよびリファレンス』および『Oracle8i *interMedia* Audio, Image, Video Java Classes 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。

Locator

Oracle8i *interMedia* Locator には、変更および実行できるスクリプトが多数あります。

参照： 詳細は、『Oracle8i *interMedia* Locator 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。

サンプル・スクリプト

Oracle8i *interMedia* Locator のサンプル・スクリプトは、この製品をインストールした後に、次のディレクトリで使用可能となります。

`$ORACLE_HOME/md/demo/geocoder`

これらのスクリプトは、次のファイルで構成されています。

- `geohttp.sql`

このファイルは、2 つに分かれています。1 つは、対話モードでジオコード機能を実行します。もう 1 つは、バッチ・モードでジオコード機能を実行します。

- － 対話モード

ファイルのこの部分のリストについては、『Oracle8i *interMedia* Locator ユーザーズ・ガイドおよびリファレンス』の「GEOCODE1 ファンクション (lastline フィールドを含む)」の例 1 を参照してください。

- バッチ・モード

バッチ・モードで `geohttp.sql` を実行する前に、`nh_cs.sql` ファイルで設定表を更新しておく必要があります。ファイルのこの部分のリストについては、『Oracle8i interMedia Locator ユーザーズ・ガイドおよびリファレンス』の「GEOCODE1 ファンクション (lastline フィールドを含む)」の例 2 または例 3 を参照してください。

- `geoindex.sql`

このファイルには、次のものが含まれています。

- `ESTIMATE_LEVEL` という名前のファンクションです。このファンクションは、100 マイルを超える半径距離を使用しての問合せを、`WITHIN_DISTANCE` 空間ロケータ索引で使用する索引レベルでの見積りを改善します。このファイルのリストについては、『Oracle8i interMedia Locator ユーザーズ・ガイドおよびリファレンス』の「`ESTIMATE_LEVEL`」の例を参照してください。
- `SETUP_LOCATOR_INDEX` という名前のプロシージャ文です。このプロシージャ文は、空間情報が格納されている `cust_table` 表内の空間情報を含む位置列において、セットアップ空間ロケータ索引を作成します。このファイルのリストについては、『Oracle8i interMedia ユーザーズ・ガイドおよびリファレンス』の「`SETUP_LOCATOR_INDEX`」の例を参照してください。

- `geolocate.sql`

このファイルには、ルーチンが含まれています。このルーチンは、関心対象の形状を動的に作成して、オフィスの半径 10 マイルの範囲内でどのくらい格納されているかを検索するために `NH_COMPUTER_STORES` 表に対して問い合わせます。このファイルのリストについては、『Oracle8i interMedia Locator ユーザーズ・ガイドおよびリファレンス』の第 2 章「Locator ファンクション」、「`LOCATOR_WITHIN_DISTANCE`」の例 2 を参照してください。

Oracle8i Time Series デモ

表 6-1 に、Oracle8i Time Series に含まれるデモを示します。この表では、各デモの説明およびそれらのファイルがインストールされるデフォルト・ディレクトリを示します。

デモ・ディレクトリは、\$ORACLE_HOME/ord/ts で検索できます。

表 6-1 Oracle8i Time Series デモ

| 説明 | ディレクトリ |
|--|---------------|
| クイック・スタート・デモは、すばやく簡単に Oracle8i Time Series の使用を開始します（詳細は、『Oracle8i Time Series ユーザーズ・ガイド』の第 1 章を参照）。 | demo/tsquick |
| 使用方法デモは、エンド・ユーザー向けおよび既存の Oracle8i Time Series 機能を使用する製品開発者向けです（詳細は、『Oracle8i Time Series ユーザーズ・ガイド』の第 1 章を参照）。 | demo/usage |
| 電子ユーティリティ・アプリケーションは、15 分間のデータのサマリーをピークおよびオフ・ピークで計算する方法を示します。 | demo/usageutl |
| Time Series データの Java ベース検索は、Oracle8i Time Series Java API のプロトタイプを使用して、Web ブラウザで実行するために設計されています（詳細は、『Oracle8i Time Series ユーザーズ・ガイド』の第 1 章を参照）。 | demo/applet |
| Java コードのサンプル・セグメントは、Time Series を操作して、その結果を印刷します（詳細は、『Oracle8i Time Series ユーザーズ・ガイド』の第 1 章を参照）。 | demo/java |
| 管理ツールを使用して既存の Time Series ディテール表をレトロフィットする手順を示すデモです。また、Time Series ディテール表の複数の修飾子列に対する、Time Series 問合せのサポート方法も示します。 | demo/retrofit |
| 上級開発者デモは、Oracle8i Time Series ファンクションを拡張するためのものです。 | demo/extend |
| OCI デモは、Oracle コール・インタフェースを使用して Oracle8i Time Series ファンクションをコールする方法を示します。 | demo/oci |
| PRO*C/C++ デモは、Oracle Pro*C/C++ プリコンパイラを使用して作成されるアプリケーションでの Oracle8i Time Series ファンクションをコールする方法を示します。 | demo/proc |
| Oracle デモは、Oracle Forms アプリケーションで Oracle8i Time Series ファンクションをコールする方法を示します。 | demo/dev2k |

デモ・ディレクトリにある README.txt ファイルは、デモを説明しています。また、各デモ用のディレクトリには、そのデモの詳細を説明している README.txt ファイルが含まれています。

参照： 詳細は、『Oracle8i Time Series ユーザーズ・ガイド』を参照してください。

Oracle8i Spatial

より詳細な情報は、\$ORACLE_HOME/md/demo/readme.txt を参照してください。

参照： 詳細は、『Oracle8i Spatial 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。

Optimal Flexible Architecture

- Optimal Flexible Architecture (OFA)
- UNIX で適用される OFA

Optimal Flexible Architecture (OFA)

オラクル社では、Oracle8i のインストールおよび構成に対して、Optimal Flexible Architecture (OFA) 標準を適用することをお薦めしています。OFA 標準は、高速で、メンテナンスの必要性を最小限に抑えた信頼性の高いデータベース構成を実現するためのガイドラインです。

OFA には、次の目的があります。

- デバイスのボトルネックやパフォーマンスの低下が発生しないように、ディスク上の大量で複雑なソフトウェアおよびデータを編成します。
- データの損傷を起こす可能性のある管理タスク（ソフトウェア、データのバックアップ機能など）を簡略化します。
- 複数の Oracle データベースでの移動を簡単にします。
- データベースの拡張を適切に管理します。
- データ・ディクショナリの空き領域の断片化を回避し、その他の領域で起きる断片化を分離して、リソースの競合を最小限に抑えます。

OFA に準拠したデータベースの特長

OFA に準拠したデータベースには、次のメリットがあります。

ファイル・システムの編成

ファイル・システムは、管理が容易で、次のスケーラビリティに対応できるように編成されています。

- 既存のデータベースへのデータの追加
- ユーザーの追加
- データベースの作成
- ハードウェアの追加

I/O 負荷の分散

I/O 負荷は十分な数のディスク・ドライブに分散されるため、パフォーマンスのボトルネックが発生しません。

ハードウェア・サポート

ほとんどの場合、Optimal Flexible Architecture (OFA) 標準を活用するための、新しいハードウェアに出資する必要はありません。

ドライブの障害に対する予防措置

複数ドライブにアプリケーションを分散することによって、ドライブに障害が発生した場合でも、アプリケーションへの影響を最小限に抑えられます。

ホーム・ディレクトリの分散

次のものを、複数のディスク・ドライブに分散できます。

- 複数のホーム・ディレクトリ
- 各ホーム・ディレクトリの内容

ログイン・ホーム・ディレクトリの整合性

ログイン・ホーム・ディレクトリを参照するプログラムを変更することなく、ログイン・ホーム・ディレクトリを追加、移動または削除できます。

UNIX ディレクトリ・サブツリーの独立性

ファイルのカテゴリは、独立した UNIX ディレクトリ・サブツリーに分類されています。そのため、あるカテゴリのファイルでの操作によって、別のカテゴリのファイルが受ける影響を最小限に抑えることができます。

アプリケーション・ソフトウェアの同時実行サポート

複数バージョンのアプリケーション・ソフトウェアを同時に実行できます。これによって、ユーザーはアプリケーションの古いバージョンを削除しなくても、新しいリリースのアプリケーションを使用できます。アップグレード後に新しいバージョンへ移行する作業は、管理者にとっては簡単な作業で、ユーザーがその作業を意識する必要はありません。

各データベースの管理情報の区別

データベースごとに管理情報を区別できるため、管理データをわかりやすい構造で編成および保存できます。

データベース・ファイルのネーミング規則

データベース・ファイルには、次のネーミング規則があります。

- 他のファイルと区別しやすい名前にします。
- 2つのデータベースに属するファイルが、どちらのデータベースに属するかを区別できる名前にします。
- 制御ファイル、REDO ログ・ファイルおよびデータ・ファイルであることが識別できる名前にします。
- データ・ファイルと表領域の関係が明確に識別できる名前にします。

表領域の内容の分割

表領域の内容が分割されているため、次のメリットがあります。

- 表領域内の空き領域の断片化を最小限に抑えられます。
- I/O 要求の競合を最小限に抑えられます。
- 管理面の柔軟性が高くなります。

すべてのドライブに分散されている I/O 負荷のチューニング

Oracle データを RAW デバイスに保存しているドライブを含め、すべてのドライブへの I/O 負荷をチューニングします。

UNIX で適用される OFA

データベース・ファイルのネーミング規則に従うことによって、データ管理の様々な問題を回避できます。ここで説明している OFA 規則は、「The OFA Standard: Oracle for Open Systems」に記載されている OFA 推奨事項に対応しています。

マウント・ポイント

マウント・ポイントの作成

Oracle8i のインストールには、少なくとも 2 つのマウント・ポイント（ソフトウェア用に 1 つ、データベース・ファイル用に 1 つ以上）が必要です。推奨される Optimal Flexible Architecture (OFA) を実装する場合、4 つ以上のマウントポイント（ソフトウェア用に 1 つ、データベース・ファイル用に 3 つ以上）が必要です。

マウント・ポイントの構文

すべてのマウント・ポイントは、`/pm` という構文で名前を付けます。この場合、`p` は文字列定数、`m` は各マウント・ポイントを区別するための一意の固定長キー（通常、2 桁の数字）です。たとえば、`/u01` および `/u02`、`/disk01` および `/disk02` などです。

超大規模データベース（VLDB）のマウント・ポイントの名前

1つのアプリケーションが使用するデータベース・ファイルが各ディスク・ドライブにあり、各データベースでI/Oのボトルネックが発生しないほど十分なドライブがある場合、`/q/dm`という構文で、マウント・ポイントに名前を付けます。表 A-1 に、この構文について示します。

表 A-1 マウント・ポイントの名前の構文

| | |
|-----------|--|
| <i>q</i> | Oracle データが保存されていることを示す文字列 |
| <i>dm</i> | 初期化パラメータ DB_NAME の値（単一インスタンス・データベースのインスタンス <i>sid</i> と同じ） |

たとえば、`/u01/oradata/test` および `/u02/oradata/test` という名前のマウント・ポイントは、Oracle テスト・データベースに2つのドライブを割り当てます。

ディレクトリの名前

ホーム・ディレクトリの構文

`/pm/h/u` という構文で、ホーム・ディレクトリに名前を付けます。表 A-2 に、この構文について示します。

表 A-2 ホーム・ディレクトリの名前の構文

| | |
|-----------|---------------|
| <i>pm</i> | マウント・ポイントの名前 |
| <i>h</i> | 標準のディレクトリ名 |
| <i>u</i> | ディレクトリの所有者の名前 |

たとえば、`/u01/app/oracle` は、Oracle Server ソフトウェア所有者のホーム・ディレクトリ（ORACLE_BASE の値で、OUI を使用してインストールする場合のデフォルト値）で、`/u01/app/applmgr` は、Oracle アプリケーション・ソフトウェア所有者のホーム・ディレクトリです。

ホーム・ディレクトリを UNIX ファイル・システム内の同一レベルに置くと、別のマウント・ポイントにあるアプリケーション所有者のログイン・ホーム・ディレクトリの集まりを、決まった形式（`/*app/*` など）で参照できます。

パス名の参照

明示的なパス名は、そのパス名を特別に保存するファイル（`/etc/passwd`、Oracle `oratab` ファイルなど）で参照します。グループのメンバーは、`/etc/group` ファイルで参照します。

ソフトウェア・ディレクトリ

複数バージョンのアプリケーション・ソフトウェアを同時に実行するための OFA 要件を満たすために、各バージョンの Oracle8i ソフトウェアを `/pm/h/product/v` という構文の 1 つのディレクトリに保存します。表 A-3 に、この構文について示します。

表 A-3 Oracle8i ソフトウェアを保存するディレクトリ名の構文

| | |
|----------------|--------------|
| <code>h</code> | 標準のディレクトリ名 |
| <code>v</code> | ソフトウェアのバージョン |

たとえば、`/u01/app/oracle/product/8.1.7` は、Oracle8i Server ファイルが保存されているディレクトリ構造の先頭を示します。環境変数 `ORACLE_HOME` をこのディレクトリに設定します。

ファイルの名前

管理ファイル

管理データの編成を簡略化するために、データベース固有の管理ファイルを `h/admin/d/a/` というサブディレクトリに保存しておくことをお勧めします。この場合、`h` は Oracle ソフトウェア所有者のホーム・ディレクトリ、`d` はデータベース名 (`DB_NAME`)、`a` は表 A-4 に示すデータベース管理ファイルのそれぞれのサブディレクトリを表しています。

表 A-4 データベース管理ファイルのサブディレクトリ

| | |
|----------------------|---|
| <code>adhoc</code> | 指定したデータベースの非定型 SQL スクリプト |
| <code>arch</code> | アーカイブ REDO ログ・ファイル |
| <code>adump</code> | 監査ファイル (<code>configdb_name.ora</code> の <code>AUDIT_FILE_DEST</code> を、この位置に設定します。このサブディレクトリは、定期的に整理してください)。 |
| <code>bdump</code> | バックグラウンド・プロセスのトレース・ファイル |
| <code>cdump</code> | コア・ダンプ・ファイル |
| <code>create</code> | データベースを作成するプログラム |
| <code>exp</code> | データベース・エクスポート・ファイル |
| <code>logbook</code> | データベースの状態および履歴を記録するファイル |
| <code>pfile</code> | インスタンス・パラメータ・ファイル |
| <code>udump</code> | ユーザー SQL トレース・ファイル |

たとえば、`sab` という名前のデータベースの一部である場合、サブディレクトリ `adhoc` のパス名は、`/u01/app/oracle/admin/sab/adhoc/` です。

データベース・ファイル

次のネーミング規則に従ってデータベース・ファイルに名前を付けることによって、これらの識別が簡単になります。

- 制御ファイルの場合、`/pm/q/d/control.ctl`
- REDO ログ・ファイルの場合、`/pm/q/d/redon.log`
- データ・ファイルの場合、`/pm/q/d/tn.dbf`

表 A-5 に、この構文について示します。

表 A-5 データベース・ファイルの名前の構文

| | |
|-----------|---|
| <i>pm</i> | マウント・ポイント名 |
| <i>q</i> | Oracle データを他の Oracle ファイルと区別するための文字列（通常、ORACLE または oradata） |
| <i>d</i> | データベースの DB_NAME |
| <i>t</i> | Oracle 表領域名 |
| <i>n</i> | 2 桁の数字 |

注意： `/pm/q/d` に、データベース *d* に対応付けられた制御ファイル、REDO ログ・ファイル、データ・ファイル以外のファイルは保存しないでください。

たとえば、この規則に従って `/u03/oradata/sab/system01.dbf` という名前のデータ・ファイルを作成すると、そのファイルが属しているデータベースを簡単に判断できます。

要件に応じた個々のセグメント

異なる表領域間で、存在期間、I/O 要求の必要性およびバックアップの頻度別に、セグメントをいくつかのグループに分類します。

Oracle データベースごとに、表 A-6 に示す特殊な表領域を作成します。アプリケーション・セグメントに必要な表領域以外に、これらの表領域が必要です。

表 A-6 特殊な表領域

| | |
|----------------|----------------------------------|
| SYSTEM | データ・ディクショナリ・セグメント |
| TEMP | テンポラリ・セグメント |
| RBS | ロールバック・セグメント |
| USERS | その他のユーザー・セグメント |
| INDX | USERS 表領域のデータに対応付けられた索引 |
| OEM_REPOSITORY | Oracle Enterprise Manager のリポジトリ |
| DRSYS | Oracle <i>interMedia</i> セグメント |

この方法でセグメントを分割すると、ディクショナリ・セグメントが削除されることはなく、また、削除できるその他のセグメントが SYSTEM 表領域に保存されることはありません。これによって、表領域の空き領域が断片化したために SYSTEM 表領域を再作成する必要はなくなります。

アプリケーション・データが保存されている表領域にロールバック・セグメントが保存されることはないため、管理者は、アプリケーションの表領域をオフラインにしてメンテナンスを行うことができます。セグメントは種類別に分割されているので、管理者は、複雑なツールを使用しなくてもデータ拡張の記録および見積りができます。

表領域の名前

表領域の名前は、8 文字以下で指定します。Oracle8i では表領域に 30 文字の名前を付けることができますが、UNIX ファイル名としては、14 文字までに制限されています。データ・ファイル名は、*tn.dbf* という構文で指定することをお勧めします。この場合、*t* は表領域名、*n* は 2 桁の数字からなる文字列です。拡張子と 2 桁の数字からなる文字列の合計が 6 文字になるため、表領域名に使用できるのは、8 文字までになります。

データ・ファイルの名前とそのデータ・ファイルを使用する表領域の関連性がわかるような名前にします。たとえば、General Ledger データと索引を保存するための表領域には、それぞれ GLD および GLX という名前を指定します。

注意： 表領域の名前に「tablespace」という語を連想させる文字列を付ける必要はありません。表領域は、前後関係から識別できるため、名前によってタイプに関する情報を示す必要はありません。

OFA 構造に基づいた Oracle ファイル

表 A-7 に、ファイルのクラスを識別するための構文を示します。

表 A-7 ファイルのクラスを識別するためのディレクトリ構造の構文

| | |
|--------------------------------|-----------------------------|
| /u[0-9][0-9] | ユーザー・データ・ディレクトリ |
| /*/home/* | ユーザー・ホーム・ディレクトリ |
| /*/app/* | ユーザー・アプリケーション・ソフトウェア・ディレクトリ |
| /*/app/applmgr | Oracle apps ソフトウェアのサブツリー |
| /*/app/oracle/product | Oracle Server ソフトウェアのサブツリー |
| /*/app/oracle/product/8.1.7 | Oracle8i リリース 8.1.7 の配布ファイル |
| /*/app/oracle/admin/sab | sab データベースの管理サブツリー |
| /*/app/oracle/admin/sab/arch/* | sab データベースのアーカイブ・ログ・ファイル |
| /*/oradata | Oracle データのディレクトリ |
| /*/oradata/sab/* | sab データベース・ファイル |
| /*/oradata/sab/*.log | sab データベースの REDO ログ・ファイル |

OFA ファイルのマッピング

表 A-8 に、ファイルのマウント・ポイント、アプリケーション、データベース、表領域などが含まれる、OFA に準拠したサンプル・データベースのファイル・マッピングの階層を示します。ファイル名によって、ファイルの種類（制御、ログまたはデータ）がわかります。

表 A-8 OFA インストールの階層型ファイル・マッピング

| | |
|-----------|-----------------------------|
| / | ルート・マウント・ポイント |
| u01/ | Oracle ソフトウェアのマウント・ポイント #1 |
| app/ | app ソフトウェアのサブツリー |
| oracle/ | Oracle ソフトウェア所有者のホーム・ディレクトリ |
| admin/ | データベース管理ファイルのサブツリー |
| TAR/ | サポート・ログのサブツリー |
| db_name1/ | db_name1 データベースの管理サブツリー |
| bdump/ | background_dump_dest |

表 A-8 OFA インストールの階層型ファイル・マッピング (続き)

| | | |
|--|--------------|----------------------------|
| | cdump/ | core_dump_dest |
| | udump/ | user_dump_dest |
| | create/ | データベース作成 SQL スクリプト |
| | pfile/ | データベース init パラメータ・ファイル |
| | db_name2/ | db_name2 データベースの管理サブツリー |
| | doc/ | オンライン・ドキュメント |
| | local/ | ローカルの Oracle ソフトウェアのサブツリー |
| | aps6/ | Oracle6 管理パッケージ |
| | aps7/ | Oracle7 管理パッケージ |
| | product/ | 配布ファイル |
| | 7.3.3/ | 7.3.3 インスタンスの ORACLE_HOME |
| | 8.0.6/ | 8.0.6 インスタンスの ORACLE_HOME |
| | 8.1.7/ | 8.1.7 インスタンスの ORACLE_HOME |
| | oraInventory | Oracle8i Inventory のサブツリー |
| | log | インストール・ログ・ファイル |
| | home | ログイン・ホーム・ディレクトリのサブツリー |
| | ltb/ | ユーザーのホーム・ディレクトリ |
| | sbm/ | ユーザーのホーム・ディレクトリ |
| | u02/ | ユーザー・データのマウント・ポイント #2 |
| | home/ | ログイン・ホーム・ディレクトリのサブツリー |
| | cvm/ | ユーザーのホーム・ディレクトリ |
| | vrn | ユーザーのホーム・ディレクトリ |
| | oradata/ | Oracle データのサブツリー |
| | db_name1/ | db_name1 データベース・ファイルのサブツリー |
| | db_name2/ | db_name2 データベース・ファイルのサブツリー |

表 A-8 OFA インストールの階層型ファイル・マッピング (続き)

| | | |
|-----------|--|------------------------------|
| u03/ | | ユーザー・データのマウント・ポイント #3 |
| oradata/ | | Oracle データのサブツリー |
| db_name1/ | | db_name1 データベース・ファイルのサブツリー |
| db_name2/ | | db_name2 データベース・ファイルのサブツリー |
| u04/ | | ユーザー・データのマウント・ポイント #4 |
| oradata/ | | Oracle データのサブツリー |
| db_name1/ | | db_name1 データベース・ファイルのサブツリー |
| db_name2/ | | db_name2 データベース・ファイルのサブツリー |
| /var | | |
| opt/ | | |
| oracle/ | | oratab および oraInst.loc の保存場所 |
| /usr | | |
| local/ | | |
| bin/ | | oraenv/coraenv/dbhome スクリプト |

ディレクトリ構造

ORACLE_BASE ディレクトリ

ORACLE_BASE は、Oracle ディレクトリ構造のルートです。表 A-9 に、ORACLE_BASE ディレクトリ構造の詳細を示します。Oracle Universal Installer を使用して OFA に準拠したデータベースをインストールすると、ORACLE_BASE はデフォルトで `/pm/app/oracle` に設定されています。

表 A-9 ORACLE_BASE ディレクトリ構造

| | |
|---------|---------------------------|
| admin | 管理ファイル |
| doc | オンライン・ドキュメント |
| local | ローカルの Oracle ソフトウェアのサブツリー |
| product | Oracle ソフトウェア |

ORACLE_HOME ディレクトリ

OFA に準拠した Oracle Server をインストールした場合、ORACLE_HOME ディレクトリは、`/pm/app/oracle/product/release_number` です。表 A-10 に、ORACLE_HOME ディレクトリ構造の詳細を示します。UNIX の場合、ORACLE_HOME ディレクトリには、次のサブディレクトリ、および選択した各 Oracle 製品用のサブディレクトリが含まれます。インストールした製品用にのみディレクトリがあります。

表 A-10 ORACLE_HOME ディレクトリ構造および内容

| | |
|-------------------------|--|
| <code>assistants</code> | 構成アシスタント |
| <code>bin</code> | すべての製品のバイナリ・ファイル |
| <code>ctx</code> | <i>interMedia Text</i> オプション |
| <code>db</code> | <code>init<code>sid</code>.ora</code> 、 <code>lksid</code> |
| <code>install</code> | インストールした関連ファイル |
| <code>lib</code> | Oracle 製品ライブラリ |
| <code>javavm</code> | Java Virtual Machine |
| <code>jdbc</code> | JDBC ドライバ |
| <code>jlib</code> | Java クラス |
| <code>md</code> | Spatial オプション |
| <code>mlx</code> | Xerox Stemmer (<i>interMedia Text</i> オプション用) |
| <code>network</code> | Net8 |
| <code>nlsrtl</code> | NLS ランタイム読み込み可能データ |
| <code>ocommon</code> | すべての製品に共通のファイル |
| <code>odg</code> | データ収集機能 |
| <code>oracore</code> | コア・ライブラリ |
| <code>orb</code> | Object Request Broker |
| <code>ord</code> | データ・オプション |
| <code>otrace</code> | Oracle TRACE |
| <code>plssql</code> | PL/SQL |
| <code>precomp</code> | プリコンパイラ |
| <code>rdbms</code> | データベースに必要なサーバー・ファイルおよびライブラリ |
| <code>slax</code> | SLAX パーサー |
| <code>sqlplus</code> | SQL*Plus |
| <code>svrmgr</code> | Server Manager |
| <code>sysman</code> | System Management |

製品のサブディレクトリの内容

各製品のサブディレクトリには、表 A-11 に示すサブディレクトリが含まれます。

表 A-11 製品のサブディレクトリの内容

| | |
|----------------|---|
| admin | 管理 SQL およびシェル・スクリプト (catalog.sql、catexp.sql、demo.sql など) |
| admin/* | 他の製品専用ディレクトリ |
| admin/resource | リソース・ファイル |
| admin/terminal | ランタイム端末ファイル |
| demo | デモンストレーション・スクリプトおよびデータ・ファイル |
| doc | README ファイル (readmeunix.doc など) |
| install | 製品のインストール・スクリプト |
| jlib | 製品の Java クラス |
| lib | 製品ライブラリおよび配布された Make ファイル |
| log | トレース・ファイルおよびログ・ファイル (orasrv.log、*.trc ファイルなど) |
| mesg | アメリカ英語メッセージ・ファイル、多言語オプション (旧各国語サポート) のメッセージ・テキストおよびバイナリ・ファイル (oraus.msg、oraus.msb など) |

製品のサブディレクトリの例

表 A-12 に、製品のサブディレクトリの例を示します。

表 A-12 製品のサブディレクトリの例

| | |
|---------|---------------------------------|
| rdbms | install、lib、admin、doc、mesg、log |
| sqlplus | install、demo、lib、admin、doc、mesg |

admin ディレクトリのファイルのネーミング規則

表 A-13 に、rdbms/admin ディレクトリに含まれる SQL スクリプトを示します。

表 A-13 admin ディレクトリのファイルのネーミング規則

| | |
|-----------|---|
| cat*.sql | カタログおよびデータ・ディクショナリ表およびビューを作成します。 インストール時に次のファイルが自動的に実行されます。 catalog.sql (すべてのインストール用) catproc.sql (すべてのインストール用) catrep.sql (すべてのインストール用) catproc.sql は、標準の PL/SQL パッケージ (DBMS_SQL、DBMS_OUTPUT など) を作成するためにスクリプトを順に実行します。 |
| d*.sql | ダウングレード・スクリプト |
| dbms*.sql | その他のデータベース・パッケージ |
| u*.sql | アップグレード・スクリプト |
| utl*.sql | データベース・ユーティリティ用の表およびビューを作成するスクリプト |

ファイル名の拡張子

表 A-14 に、ファイル名の拡張子を示します。

表 A-14 ファイル名の拡張子

| | |
|------|--|
| .a | オブジェクト・ファイル・ライブラリ (Ada ランタイム・ライブラリ) |
| .aud | Oracle 監査ファイル |
| .bdf | X11 フォント記述ファイル |
| .bmp | X11 ビットマップ・ファイル |
| .c | C ソース・ファイル |
| .ctl | SQL*Loader 制御ファイル、Oracle Server 制御ファイル |
| .dat | SQL*Loader データ・ファイル |
| .dbf | Oracle Server 表領域ファイル |
| .dmp | エクスポート・ファイル |
| .doc | ASCII テキスト・ファイル |
| .env | 環境設定用のシェル・スクリプト・ファイル |
| .h | C ヘッダー・ファイル (sr.h は SQL*Report Writer のヘルプ・ファイル) |
| .jar | Java クラス・アーカイブ |

表 A-14 ファイル名の拡張子（続き）

| | |
|------|--|
| .1 | UNIX マニュアル・ページ |
| .lis | SQL*Plus スクリプトの出力 |
| .log | インストール・ログ・ファイル（Oracle Server REDO ログ・ファイル） |
| .mk | Make ファイル |
| .msb | NLS メッセージ・ファイル（バイナリ） |
| .msg | NLS メッセージ・ファイル（テキスト） |
| .o | オブジェクト・モジュール |
| .ora | Oracle 構成ファイル |
| .orc | インストール・プロトタイプ・ファイル |
| .pc | Pro*C ソース・ファイル |
| .pco | Pro*COBOL ソース・ファイル |
| .ppd | プリンタ・ドライバ・ファイル |
| .sh | Bourne シェル・スクリプト・ファイル |
| .sql | SQL* スクリプト・ファイル |
| .sys | Bourne シェル・スクリプト・ファイル |
| .tab | SQL* スクリプト・ファイル |
| .trc | トレース・ファイル |
| .tut | Bourne シェル・スクリプト・ファイル |
| .utd | ユニフォーム端末定義 |
| .zip | Zip ファイル |

索引

記号

? 略称, 1-2
@ 略称, 1-2

数字

2 タスク, 1-16
シグナル, 4-10

A

ADDRESS 指定プロトコル, 5-6

B

BEQ プロトコル, 5-6
接続パラメータの構文, 5-7
Bourne シェル・スクリプトのトレース, 1-4

C

C
Pro*C/C++, 4-4
CATPROC.SQL, A-14
CONNECT INTERNAL, 1-17
coraenv, 1-4
coraenv ファイル, 1-7
CPU の使用状況
シングルタスク・アーキテクチャ, 2-7
チューニング, 2-7
プロセスの優先順位, 2-7
CREATE CONTROL FILE パラメータ, 1-14

D

dba グループ
メンバー, 1-16
dbhome ファイル, 1-8
DBWR チューニング, 2-5
Developer/2000 (Oracle Developer) デモ, 6-8

E

expst コマンド, 2-8

G

glogin.sql ファイル, 3-2

I

impst コマンド, 2-8
init sid.ora ファイル, 1-20
Intelligent Agent
SNMP での構成, 5-11
interMedia
Java デモ, 6-6
MediaAnnotator, 6-6
サンプル Audio スクリプト, 6-2
サンプル Video スクリプト, 6-5
サンプル・プログラム, 6-3
interMedia Audio, Video, Image, 6-2
interMedia Locator, 6-6
interMedia Text, 6-2
I/O
チューニング, 2-5
I/O スレーブ, 2-6
IPC プロトコル, 5-8

ireclen, 4-3

J

Java デモ, 6-8

L

Linux ツール, 2-2
 vmstat ユーティリティ, 2-2
login.sql ファイル, 3-2

M

MAXDATAFILES パラメータ, 1-14
MAXINSTANCES パラメータ, 1-14
MAXLOGFILES パラメータ, 1-14
MAXLOGHISTORY パラメータ, 1-14
MAXLOGMEMBERS パラメータ, 1-14

N

Net8
 ADDRESS 指定, 5-5
 BEQ プロトコル, 5-6
 IPC プロトコル, 5-8
 Oracle Advanced Security, 5-13
 Oracle Connection Manager, 5-3
 Oracle Enterprise Manager Intelligent Agent, 5-10
 RAW プロトコル, 5-9
 TCP/IP プロトコル, 5-9
 アダプタ・ユーティリティ, 5-3
 製品および特徴, 5-2
 ファイルおよびユーティリティ, 5-2
 プロトコル, 5-5
 プロトコルのサポート, 5-5
 マルチスレッド・サーバー, 5-4
Net8 の構成ファイル, 5-2
NLS_LANG 環境変数, 1-3

O

OCI デモ, 6-8
OFA, 「Optimal Flexible Architecture」を参照
Optimal Flexible Architecture, A-1
 OFA の特徴, A-2
 Oracle8 for UNIX での実装, A-2

 ディレクトリ構造, A-11
 ファイル・マッピング, A-2, A-9
Oracle8i Spatial, 6-9
Oracle8i で使用する UNIX 環境変数, 1-5
Oracle Advanced Security, 5-13
 .bak ファイル, 5-13
 セキュリティおよびシングル・サインオン, 5-13
Oracle Connection Manager, 5-3
Oracle Developer デモ, 6-8
ORACLE_DOC 環境変数, ix
Oracle Enterprise Manager Intelligent Agent, 5-10
 tcl スクリプトのデバッグ, 5-10
 エージェント・サービスの展開および自動構成,
 5-10
Oracle Forms デモ, 6-8
ORACLE_HOME 環境変数, 1-2, 1-3
Oracle HTTP Server, 1-24
Oracle Information Navigator, x
Oracle Server アカウント, 1-15
ORACLE_SID
 環境変数, 1-4
ORACLE_SID 環境変数, 1-2
Oracle TCP/IP プロトコル, 5-9
Oracle 環境変数
 EPC_DISABLED, 1-3
 NLS_LANG, 1-3
 ORACLE_BASE, 1-3
 ORACLE_HOME, 1-3
 ORACLE_PATH, 1-4
 ORACLE_SID, 1-4
 ORACLE_TRACE, 1-4
 ORAENV_ASK, 1-4
 ORA_NLS33, 1-3
 TNS_ADMIN, 1-5
 TWO_TASK, 1-5
 UNIX 環境変数, 1-5
Oracle コール・インタフェース, 4-7
 デモ・プログラム, 4-7
Oracle コール・インタフェース (OCI)・デモ, 6-8
Oracle システム識別子, 1-4
Oracle ソフトウェア所有者, 1-15
 特殊アカウント, 1-15
Oracle データ・オプション・デモの実行, 6-1
Oracle プリコンパイラと OCI のリンクおよび Make
 ファイル
 カスタム Make ファイル, 4-9
 未定義シンボル, 4-9

ORAENV_ASK, 1-4
oraenv ファイル, 1-7
 説明, 1-7
 データベース間の移動, 1-8
ORA_NLS 環境変数, 1-3
orapwd コマンド, 1-18
oratab ファイル, 1-30
oreclen, 4-3

P

PL/SQL デモンストレーション
 ロード, 1-28
Pro*C/C++
 Make ファイル, 4-5
 シグナル, 4-11
 使用, 4-4
 デモ・プログラム, 4-4
Pro*C/C++ デモ, 6-8
PRODUCT_USER_PROFILE 表, 3-3

R

RAW デバイス
 バッファ・キャッシュ・サイズ, 2-9
RAW プロトコル, 5-9
README.txt ファイル, 6-9
root.sh スクリプト, 1-8
root ユーザー, 1-15

S

sar コマンド, 2-4
Server Manager
 コマンド, 1-17
 セキュリティ, 1-17
SGA
 計算, 1-11
 再配置, 1-11
 チューニング, 2-5
SHELL, 1-6
SHUTDOWN コマンド, 1-17
 セキュリティ, 1-17
SIGCLD 2 タスク・シグナル, 4-11
SIGINT 2 タスク・シグナル, 4-10
SIGIO 2 タスク・シグナル, 4-11
SIGPIPE 2 タスク・シグナル, 4-11

SIGTERM 2 タスク・シグナル, 4-11
SIGURG 2 タスク・シグナル, 4-11
Simple Network Management Protocol (SNMP), 5-11
Site Profile SQL*Plus, 3-2
snmpd 実行ファイル, 5-12
SNMP および Intelligent Agent, 5-11
Solaris のツール
 swap ユーティリティ, 2-3
Spatial デモンストレーション, 6-9
SPOOL コマンド
 SQL*Plus, 3-6
 使用, 3-6
SQL*Plus, 3-5
 PRODUCT_USER_PROFILE 表, 3-3
 Site Profile, 3-2
 SPOOL コマンド, 3-6
 User Profile, 3-2
 エディタ, 3-5
 オペレーティング・システム・コマンドの実行,
 3-6
 システム・エディタ, 3-5
 使用, 3-5
 制限事項, 3-7
 設定ファイル, 3-2
 デフォルト・エディタ, 3-5
 デモンストレーション表, 3-3
 ヘルプ機能, 3-4
 割込み, 3-6
SQL*Plus の使用, 3-5
SQL*Plus への割込み, 3-6
SQL スクリプト, 2-4
SQL の管理, 3-2
start_peer スクリプト, 5-12
STARTUP コマンド, 1-17
swapon ユーティリティ, 2-2
swap ユーティリティ, 2-3
SYSDATE, 1-8
SYSTEM アカウント, 1-15
 権限, 1-15
SYS アカウント, 1-15
 権限, 1-15

T

TCP/IP プロトコル, 5-9
 ADDRESS, 5-9
TNS_ADMIN 環境変数, 1-5

TNS リスナー, 5-9
TWO_TASK 環境変数, 1-5

U

UNIX カーネル, 2-8
User Profile
 SQL*Plus, 3-2

V

vmstat コマンド, 2-4

X

XA 機能, 4-13

あ

アカウント, 1-15
アダプタ・ユーティリティ, 5-3

い

インポート, 2-7

う

受取りルーチン, 4-11

え

エクスポート, 2-7

お

オペレーティング・システム・コマンドの実行, 3-6

か

カーネル・パラメータ, 2-8
各国語サポート (NLS) 環境変数, 1-3
カプセル化プログラムの構成, 5-11
環境変数, 1-2
 NLS_LANG, 1-3
 ORACLE_DOC, ix
 ORACLE_HOME, 1-3

ORACLE_SID, 1-4
ORACLE_TRACE, 1-4
ORAENV_ASK, 1-4
ORA_NLS, 1-3
SHELL, 1-6
TNS_ADMIN, 1-5, 5-2
TWO_TASK, 1-5

き

キャッシュ・サイズ, 2-9
共通の環境
 oraenv ファイル, 1-7
 設定, 1-7
共有メモリー
 SGA のチューニング, 2-5

け

言語, 1-3

こ

構成
 カプセル化, 5-11
 マスター・エージェント, 5-11
構成ファイル
 Net8, 5-2
 プリコンパイラ, 4-2
コマンド
 expst, 2-8
 impst, 2-8
 orapwd, 1-18
 sar, 2-4
 vmstat, 2-4
コマンド・インタプリタ, 1-6

さ

再リンク, 1-9
 パラメータ, 1-9
参照マニュアル, x

し

シグナル
 2 タスク, 4-10

- シグナル・ハンドラ
 - 使用, 4-10
- シグナル・ルーチン, 4-11
 - 例, 4-11
- システム・エディタ
 - SQL*Plus, 3-5
- システム・グローバル領域または共有グローバル領域 (SGA)
 - 再配置, 1-11
- システム時刻, 1-8
- 自動ログイン
 - remote_os_roles, 1-18
- シャドウ・プロセス, 1-16
- 上級開発者デモ, 6-8
- 初期化パラメータ
 - BACKGROUND_DUMP_DEST, 1-20, 2-10
 - BITMAP_MERGE_AREA_SIZE, 1-20
 - COMMIT_POINT_STRENGTH, 1-20
 - CONTROL_FILES, 1-20
 - CREATE_BITMAP_AREA_SIZE, 1-20
 - DB_BLOCK_BUFFERS, 1-20
 - DB_BLOCK_SIZE, 1-20
 - DB_FILE_DIRECT_IO_COUNT, 1-20
 - DB_FILE_MULTIBLOCK_READ_COUNT, 1-20
 - DB_FILES, 1-20
 - DISTRIBUTED_TRANSACTIONS, 1-20
 - HASH_AREA_SIZE, 1-20
 - HASH_MULTIBLOCK_IO_COUNT, 1-20
 - JAVA_POOL_SIZE, 1-20
 - LOCK_SGA, 1-20
 - LOG_ARCHIVE_DEST, 1-21
 - LOG_ARCHIVE_FORMAT, 1-21
 - LOG_BUFFER, 1-21
 - LOG_CHECKPOINT_INTERVAL, 1-21
 - MTS_LISTENER_ADDRESS, 1-21
 - MTS_MAX_DISPATCHERS, 1-21
 - MTS_MAX_SERVERS, 1-21
 - MTS_SERVERS, 1-21
 - NLS_LANGUAGE, 1-21
 - NLS_TERRITORY, 1-21
 - OBJECT_CACHE_MAX_SIZE_PERCENT, 1-21
 - OBJECT_CACHE_OPTIMAL_SIZE, 1-21
 - OPEN_CURSORS, 1-21
 - OS_AUTHENT_PREFIX, 1-21
 - PROCESSES, 1-21
 - SHARED_POOL_SIZE, 1-21
 - SORT_AREA_SIZE, 1-21

す

- スーパーユーザー, 1-15
- スクリプト
 - start_peer, 5-12
- スレッドのサポート, 4-10
- スワップ領域, 2-4

せ

- 制限, 1-13
- 制限事項 (SQL*Plus), 3-7
 - ウィンドウのサイズ変更, 3-7
 - リターン・コード, 3-7
- セキュリティ, 1-16, 1-17
 - 2 タスク・アーキテクチャ, 1-16
 - CONNECT INTERNAL, 1-17
 - Server Manager アクセス, 1-17
 - SHUTDOWN コマンド, 1-17
 - STARTUP コマンド, 1-17
 - UNIX の機能, 1-16
 - グループ・アカウント, 1-16
 - ファイル所有権, 1-16
- 設定ファイル
 - SQL*Plus, 3-2

た

- 単一共有メモリー・セグメント, 2-5

ち

- チューニング, 2-4, 2-5, 2-7, 2-8
 - CPU の使用状況, 2-7
 - I/O ボトルネック, 2-5
 - Linux バッファ・キャッシュ・サイズ, 2-9
 - ディスク I/O, 2-5
 - トレース・ファイルおよびアラート・ファイル, 2-9, 2-10
 - ブロック・サイズおよびファイル・サイズ, 2-8
 - メモリー管理, 2-4
 - リソースの競合, 2-8

つ

- ツール, 2-2

て

- ディスク
 - パフォーマンスの監視, 2-7
 - 割当て制限, 1-13
- ディスク I/O
 - I/O スレーブ, 2-6
 - チューニング, 2-5
 - データベース・ライターのチューニング, 2-5
- ディスク・パフォーマンス, 2-7
- データ・オプション, 6-1
- データベースの変更, 1-8
- データベース・パフォーマンス, 2-4
- データベース・ファイル, 1-17
- データベース・ライター
 - チューニング, 2-5
- データベース・ライターのチューニング, 2-5
- デバッグ・プログラム, 4-3
- デモ (デモンストレーション・ファイル)
 - Java, 6-8
 - Oracle Developer, 6-8
 - Oracle コール・インタフェース (OCI), 6-8
 - Pro*C/C++, 6-8
 - 既存表のレトロフィット, 6-8
 - 上級開発者, 6-8
 - 使用方法, 6-8
 - 電子ユーティリティ, 6-8
- デモの使用, 6-8
- デモ・プログラム
 - Oracle コール・インタフェース, 4-7
 - Pro*C/C++, 4-4
- デモンストレーション
 - プリコンパイラ, 1-29
 - プロシージャ・オプション、PL/SQL, 1-28
- デモンストレーション表
 - SQL*Plus, 3-3
 - 削除, 3-4
 - 手動で作成, 3-3
- 電子ユーティリティ・デモ, 6-8

と

- ドキュメント
 - アクセス, ix
 - インストール先, ix
 - ナビゲーション・ファイル, ix
- 特殊アカウント, 1-15

特殊グループ

- root, 1-16
- トレース・ファイルおよびアラート・ファイル
 - アラート・ファイル, 2-10
 - 使用, 2-9, 2-10
 - トレース・ファイル名, 2-10

に

- 認可, 1-17

は

- 配布ソフトウェア, 1-3
- パスワード, 1-18
- バッファ・キャッシュ・サイズ, 2-9
 - チューニング, 2-9
- バッファ・マネージャ, 2-4

ふ

- ファイル
 - coraenv, 1-7
 - dbhome, 1-8
 - oraenv, 1-7
 - oratab, 1-30
 - root.sh, 1-8
- ファイル・サイズ, 2-8
- 複数のシグナル・ハンドラ, 4-11
- プリコンパイラ
 - ireclen および oreclen の値, 4-3
 - 値, 4-3
 - 大文字から小文字への変換, 4-3
 - 概要, 4-2
 - シグナル, 4-11
 - デモンストレーションの実行, 1-29
 - ベンダー提供のデバッグ・プログラム, 4-3
- プリコンパイラ構成ファイル, 4-2
- ブロック・サイズ, 2-8
- ブロック・サイズおよびファイル・サイズ, 2-8
- プロトコル, 5-5
 - ADDRESS 指定, 5-6

へ

- ページング領域, 2-4
- チューニング, 2-5

ヘルプ機能, 3-4

ま

マスター・エージェントの構成, 5-11
マニュアル
 参照, x
マルチスレッド・サーバー, 5-4

め

メモリー
 SGA のチューニング, 2-5
 仮想, 1-12
 使用量の見積り, 1-12
 チューニング, 2-4
メモリー管理, 2-4
 UNIX カーネル, 2-5
 スワップ領域, 2-4
 単一共有メモリー・セグメント, 2-5
 ページングの制御, 2-5
メモリーの使用量, 1-13

ゆ

ユーザー割込みハンドラ, 4-11
ユーティリティ (電子) デモ, 6-8

ら

ライターのアクティビティ, 2-5

り

リソースの競合
 カーネル・パラメータ, 2-8
 チューニング, 2-8
リソースの制限, 1-13
リモート接続パラメータ
 OS_AUTHENT_PREFIX, 1-18
 REMOTE_OS_AUTHENT, 1-18
 REMOTE_OS_ROLES, 1-18
リモート・パスワード, 1-18
リンク, 2-7

れ

レトロフィット・デモ, 6-8

ろ

ログ・ライター・プロセス (LGWR), 2-7

