

# Oracle8

for AIX-Based Systems 管理者リファレンス

リリース 8.0.6

1999 年 8 月

部品番号 : J00595-01

おもなトピック :

[Oracle8 での最適フレキシブル・アーキテクチャ \(OFA\)](#)

[AIX での Oracle8 の管理](#)

[AIX での Oracle8 のチューニング](#)

[AIX での SQL\\*Plus の管理](#)

[AIX での 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 Net8 の構成](#)

[AIX での Oracle データ・カートリッジ・デモの実行](#)

部品番号 : J00595-01

原本名 : Oracle8 Administrator's Reference Release 8.0.6 for AIX-Based Systems

原本部品番号 : A74824-01

原本著者 : Kevin Adams, Donn Fisher, Nicholas Hind, Sally Norton, Nik Ormseth, Lynn Robinson

Copyright © 1999 Oracle Corporation. All rights reserved.

Printed in Japan.

制限付権利の説明

プログラムの使用、複製または開示は、オラクル社との契約に記された制約条件に従うものとします。著作権、特許権およびその他の知的財産権に関する法律により保護されています。

当ソフトウェア（プログラム）のリバース・エンジニアリングは禁止されています。

このドキュメントの情報は、予告なしに変更されることがあります。オラクル社は本ドキュメントの無謬性を保証しません。

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

危険な用途への使用について

オラクル社製品は、原子力、航空産業、大量輸送、医療あるいはその他の危険が伴うアプリケーション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兼ねます。当プログラムを米国国防総省の米国政府機関に提供する際には、『Restricted Rights』と共に提供してください。この場合次の Legend が適用されます。

#### Restricted Rights Legend

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

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

---

# 目次

|   |      |
|---|------|
| はじめに .....                                    | ix   |
| <b>1 Oracle8 での最適フレキシブル・アーキテクチャ (OFA)</b>     |      |
| <b>最適フレキシブル・アーキテクチャ (OFA)</b> .....           | 1-2  |
| OFA に準拠したデータベースの特長 .....                      | 1-2  |
| <b>Oracle8 for UNIX で適用される OFA</b> .....      | 1-5  |
| マウント・ポイントの名前 .....                            | 1-5  |
| ディレクトリの名前 .....                               | 1-5  |
| ファイルの名前 .....                                 | 1-6  |
| 表領域の名前 .....                                  | 1-8  |
| OFA 構造に基づいた Oracle ファイル .....                 | 1-9  |
| OFA ファイルのマッピング .....                          | 1-9  |
| RAW デバイスのサイズ .....                            | 1-11 |
| 複数インスタンスにおける OFA に準拠したデータベースのファイル・マッピング ..... | 1-11 |
| ディレクトリ構造 .....                                | 1-13 |
| デフォルトの OFA データベース .....                       | 1-17 |
| <b>2 AIX での Oracle8 の管理</b>                   |      |
| <b>initsid.ora ファイルのカスタマイズ</b> .....          | 2-2  |
| サンプル initsid.ora ファイル .....                   | 2-2  |
| <b>環境設定</b> .....                             | 2-4  |
| 環境変数の表示および設定 .....                            | 2-4  |
| 共通の環境設定 .....                                 | 2-5  |
| データベースの例 .....                                | 2-6  |

|  |      |
|--|------|
| <b>Oracle8 の環境変数</b> .....               | 2-7  |
| AIX での Oracle 環境変数 .....                 | 2-7  |
| Oracle8 で使用する UNIX 環境変数 .....            | 2-10 |
| システム時刻の設定 .....                          | 2-12 |
| <b>Oracle8 Server のメモリー使用量の見積り</b> ..... | 2-12 |
| <b>クラスタ・サイズおよび索引サイズの計算</b> .....         | 2-13 |
| クラスタ・サイズの計算 .....                        | 2-13 |
| 索引サイズの計算 .....                           | 2-13 |
| サーバー・リソースの制限 .....                       | 2-13 |
| <b>初期化パラメータ</b> .....                    | 2-14 |
| デフォルトの初期化パラメータの値 .....                   | 2-14 |
| <b>システム・グローバル領域の制御</b> .....             | 2-15 |
| SGA のサイズ制限 .....                         | 2-16 |
| SGA のサイズの計算 .....                        | 2-16 |
| <b>特殊アカウントおよび特殊グループの管理</b> .....         | 2-16 |
| 特殊アカウント .....                            | 2-17 |
| 特殊グループ .....                             | 2-17 |
| <b>セキュリティの管理</b> .....                   | 2-18 |
| グループとセキュリティ .....                        | 2-18 |
| Oracle Server ユーティリティのセキュリティ .....       | 2-19 |
| Server Manager コマンドのセキュリティ .....         | 2-19 |
| データベース・ファイルのセキュリティ .....                 | 2-20 |
| ネットワーク・セキュリティ .....                      | 2-20 |
| Oracle Net8 での自動ログイン .....               | 2-21 |
| 確認の順番 .....                              | 2-22 |
| セキュリティとリモート・パスワード .....                  | 2-23 |
| <b>ログイン・ホーム・ディレクトリの管理</b> .....          | 2-25 |
| <b>デモンストレーションの作成と実行</b> .....            | 2-27 |
| PL/SQL デモンストレーションのロード .....              | 2-27 |
| PL/SQL デモンストレーションの実行 .....               | 2-28 |
| SQL*Loader のデモンストレーション .....             | 2-29 |
| SQL*Loader の管理 .....                     | 2-30 |
| Oracle Security Server .....             | 2-31 |
| <b>Oracle8 Server の SQL リファレンス</b> ..... | 2-32 |
| CREATE CONTROLFILE パラメータ .....           | 2-32 |

### 3 AIX での Oracle8 のチューニング

|   |      |
|---|------|
| <b>AIX パフォーマンス・ツール</b> .....                            | 3-2  |
| AIX Base Operation System ツール .....                     | 3-2  |
| AIX Performance Toolbox .....                           | 3-2  |
| AIX System Management Interface Tool .....              | 3-3  |
| <b>Oracle パフォーマンス・ツール</b> .....                         | 3-4  |
| Utlbstat / Utilestat による統計の収集 .....                     | 3-4  |
| 動的パフォーマンス表 .....  | 3-4  |
| SQL トレース .....  | 3-4  |
| Oracle Enterprise Manager Performance Tuning Pack ..... | 3-5  |
| <b>Oracle8 Server のチューニング</b> .....                     | 3-6  |
| Oracle8 Server のチューニング順序 .....                          | 3-6  |
| <b>メモリーおよびページング</b> .....                               | 3-7  |
| バッファ・キャッシュ間のページング・アクティビティの制御 .....                      | 3-7  |
| AIX ファイル・バッファ・キャッシュのチューニング .....                        | 3-8  |
| 十分なページング領域の割当て .....                                    | 3-9  |
| ページングの制御 .....  | 3-9  |
| データベース・ブロック・サイズの設定 .....                                | 3-10 |
| アーカイバ・バッファのチューニング .....                                 | 3-10 |
| SGA サイズのチューニング .....                                    | 3-11 |
| I/O バッファと SQL*Loader .....                              | 3-11 |
| <b>ディスク I/O の問題</b> .....                               | 3-12 |
| 索引と表の分離 .....   | 3-12 |
| REDO ログの専用ディスク・デバイスへの配置 .....                           | 3-12 |
| ホット・ファイルの他のディスクへの移動 .....                               | 3-12 |
| ホット・ファイルの I/O の削減 .....                                 | 3-12 |
| データベースの過剰な断片化のチェック .....                                | 3-13 |
| AIX 論理ボリューム・マネージャ .....                                 | 3-14 |
| 異なるデータベース・ファイルの同一ディスクへの配置の回避 .....                      | 3-15 |
| ジャーナル・ファイル・システム (JFS) と RAW パーティションの使用 .....            | 3-16 |
| 非同期 I/O の使用方法 .....                                     | 3-18 |
| DB_FILE_MULTIBLOCK_READ_COUNT の使用 .....                 | 3-18 |
| RAID 機能の使用 .....  | 3-19 |
| 後書きの使用 .....  | 3-19 |
| 順次ファイルの先読みのチューニング .....                                 | 3-19 |

|                                       |      |
|---------------------------------------|------|
| ディスク I/O の歩調合せ .....                  | 3-20 |
| ディスク配置の考慮事項 .....                     | 3-20 |
| リモート I/O の最小化 .....                   | 3-20 |
| VSD キャッシュ・バッファを使用しない .....            | 3-21 |
| <b>CPU のスケジューリングとプロセスの優先順位</b> .....  | 3-22 |
| プロセスのランタイム・スライスの変更 .....              | 3-22 |
| SMP マシンでのプロセッサ・バインディング .....          | 3-22 |
| マルチ・プロセッサ・システムでのスピン・カウントのチューニング ..... | 3-24 |
| <b>その他のチューニングのヒント</b> .....           | 3-25 |
| RAW デバイスのバックアップ .....                 | 3-25 |
| パラレル回復の使用方法 .....                     | 3-26 |
| Pro*C プログラムのコンパイル .....               | 3-26 |
| <b>関連ドキュメント</b> .....                 | 3-27 |
| Oracle ドキュメント .....                   | 3-27 |
| IBM ドキュメント .....                      | 3-27 |

## 4 AIX での SQL\*Plus の管理

|   |     |
|---|-----|
| <b>SQL*Plus の管理</b> .....               | 4-2 |
| 設定ファイル .....                            | 4-2 |
| Site Profile .....                      | 4-2 |
| User Profile .....                      | 4-2 |
| PRODUCT_USER_PROFILE 表 .....            | 4-3 |
| デモンストレーション表 .....                       | 4-3 |
| ヘルプ機能 .....                             | 4-4 |
| <b>SQL*Plus の使用</b> .....               | 4-6 |
| SQL*Plus からのシステム・エディタの使用 .....          | 4-6 |
| エディタの順序設定 .....                         | 4-6 |
| _editor オプションの設定 .....                  | 4-6 |
| 環境変数の設定 .....                           | 4-7 |
| デフォルト設定 .....                           | 4-7 |
| SQL*Plus からのオペレーティング・システム・コマンドの実行 ..... | 4-7 |
| SQL*Plus への割込み .....                    | 4-8 |
| SPOOL コマンドの使用 .....                     | 4-8 |
| <b>制限事項</b> .....                       | 4-8 |
| COPY コマンド .....                         | 4-8 |

|                   |     |
|-------------------|-----|
| ウィンドウのサイズ変更 ..... | 4-9 |
| リターン・コード .....    | 4-9 |

## 5 AIX での 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

|  |      |
|--|------|
| <b>Oracle プリコンパイラの概要</b> .....                                   | 5-2  |
| プリコンパイラ実行ファイルの再リンク .....   | 5-2  |
| プリコンパイラ構成ファイル .....  | 5-3  |
| すべてのプリコンパイラに共通の問題 .....  | 5-3  |
| 参照ドキュメント .....   | 5-4  |
| <b>Pro*C/C++</b> .....   | 5-4  |
| Pro*C/C++ の管理 .....  | 5-4  |
| Pro*C/C++ の使用 .....  | 5-4  |
| <b>Pro*COBOL</b> .....   | 5-6  |
| Pro*COBOL の管理 .....  | 5-7  |
| 環境変数 .....   | 5-7  |
| Pro*COBOL の使用 .....  | 5-8  |
| <b>Pro*FORTRAN</b> .....   | 5-11 |
| Pro*FORTRAN の管理 .....  | 5-11 |
| Pro*FORTRAN の使用 .....  | 5-11 |
| <b>SQL*Module for Ada</b> .....                                  | 5-13 |
| SQL*Module for Ada の管理 .....                                     | 5-13 |
| SQL*Module for Ada の使用 .....                                     | 5-13 |
| <b>Oracle コール・インタフェース</b> .....                                  | 5-15 |
| Oracle コール・インタフェースの使用 .....                                      | 5-15 |
| <b>Oracle プリコンパイラと Oracle コール・インタフェースのリンクおよび Make ファイル</b> ..... | 5-17 |
| カスタム Make ファイル .....   | 5-17 |
| 未定義シンボル .....  | 5-17 |
| <b>スレッドのサポート</b> .....   | 5-18 |
| <b>Oracle ライブラリを使用した静的リンクおよび動的リンク</b> .....                      | 5-18 |
| <b>シグナル・ハンドラの使用方法</b> .....                                      | 5-19 |
| シグナル .....   | 5-19 |
| <b>XA 機能</b> .....   | 5-22 |

## 6 Oracle Net8 の構成

|                |     |
|----------------|-----|
| 参照ドキュメント ..... | 6-2 |
|----------------|-----|

|  |             |
|--|-------------|
| README ファイルへの補足情報 .....  | 6-2         |
| <b>主な Net8 製品および特徴 .....</b>                                     | <b>6-3</b>  |
| Net8 ファイルおよびユーティリティ .....  | 6-3         |
| Oracle Connection Manager .....                                  | 6-4         |
| マルチスレッド・サーバー .....   | 6-4         |
| Oracle Names .....   | 6-4         |
| Net8 Assistant .....   | 6-4         |
| <b>Oracle Net8 Protocol Adapter .....</b>                        | <b>6-4</b>  |
| <b>BEQ Protocol Adapter .....</b>                                | <b>6-6</b>  |
| BEQ Protocol Adapter の概要 .....                                   | 6-6         |
| BEQ の ADDRESS 指定 .....   | 6-6         |
| <b>IPC Protocol Adapter .....</b>                                | <b>6-8</b>  |
| IPC Protocol Adapter の概要 .....                                   | 6-8         |
| IPC の ADDRESS 指定 .....   | 6-8         |
| <b>RAW Protocol Adapter .....</b>                                | <b>6-9</b>  |
| <b>TCP/IP Protocol Adapter .....</b>                             | <b>6-9</b>  |
| TCP/IP Protocol Adapter の概要 .....                                | 6-9         |
| TCP/IP の ADDRESS 指定 .....  | 6-10        |
| <b>SPX/IPX Protocol Adapter .....</b>                            | <b>6-11</b> |
| ntisbsdmd 同報通信デーモン .....   | 6-11        |
| ntspcxctl ユーティリティ .....  | 6-11        |
| SPX/IPX Protocol Adapter のコマンドの要約 .....                          | 6-13        |
| getname コマンド .....   | 6-13        |
| SPX/IPX の ADDRESS 指定 .....                                       | 6-15        |
| <b>APPC/LU6.2 Protocol Adapter .....</b>                         | <b>6-16</b> |
| AIX 固有のリスナー .....  | 6-16        |
| APPC/LU6.2 の ADDRESS 指定 .....                                    | 6-17        |
| <b>Net8 Naming Adapter .....</b>                                 | <b>6-18</b> |
| <b>Oracle Enterprise Manager ( OEM ) Intelligent Agent .....</b> | <b>6-18</b> |
| <b>Oracle Advanced Networking Option .....</b>                   | <b>6-18</b> |

## 7 AIX での Oracle データ・カートリッジ・デモの実行

|   |            |
|---|------------|
| データ・カートリッジ共通の要件 .....                                       | 7-2        |
| <b>Oracle8 Time Series Cartridge .....</b>                  | <b>7-2</b> |
| Time Series Cartridge デモのインストール .....                       | 7-2        |
| <b>Oracle8 Visual Information Retrieval Cartridge .....</b> | <b>7-3</b> |



デモの作成 ..... 7-3

**Oracle8 Image Cartridge** ..... 7-3

デモの作成 ..... 7-3

**索引**



---

# はじめに

## 用途

このマニュアルでは、Oracle8 Server の管理およびチューニングに必要な UNIX および AIX 固有の情報を説明します。このマニュアルの内容は、Oracle8 マニュアル・セットを補足しています。

## 対象読者

このマニュアルは、AIX システムでの Oracle8 Server の管理者を対象としています。

## Oracle8 および Oracle8 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、Oracle8 および Oracle8 Enterprise Edition に共通です。

## このマニュアルで使用する表記規則

|         |  |
|---------|--|
| クーリエ体   | クーリエ体は、UNIX コマンド、ディレクトリ名、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、コマンド構文の中で使用される場合、大カッコは別の意味になります。 |
| イタリック体  | イタリック体は、変数（ファイル名の中で変化する部分を含む）を表します。  |
| 大文字     | 大文字は、SQL コマンド、初期化パラメータまたは環境変数を表します。  |

UNIX では大文字と小文字が別のものとして認識されるため、このマニュアルで使用する表記規則は、他の Oracle 製品の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。

## コマンドの構文

コマンド構文は、クーリエ体で表します。コマンドで使用する構文は、次のとおりです。

**バックスラッシュ \** バックスラッシュは、1 行にコマンドが入りきらない場合に使用します。このマニュアルで記載され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。

```
dd if=/dev/rdisk/c0t1d0s6 of=/dev/rst0 bs=10b \
count=10000
```

**中カッコ {}** 中カッコは、必ず選択する項目を表します。  
.DEFINE {macro1}

**大カッコ []** 大カッコは、任意に選択する項目を表します。  
cvtcrt termname [outfile]

ただし、本文中で使用される場合、大カッコは別の意味になります。

**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。  
CHKVAL fieldname value1 value2 ... valueN

**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。  
*library\_name*

**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。  
*SIZE filesize [K|M]*

## カスタマ・サポート・センターへの連絡方法

オラクル社カスタマ・サポート・センターへは、次の電話番号におかけください  
(営業時間は、各地域のカスタマ・サポート・センターにお問い合わせください)。

- アメリカ合衆国の場合 : +1.650.506.1500
- ヨーロッパの場合 : +44.1344.860160
- アジアの場合 : +81.3.5717.1850

詳細は、<http://www.oracle.com/support> にアクセスしてください。

## 参照ドキュメント

本番データベース・システム用の拡張構成およびチューニングの詳細は、次のドキュメントを参照してください。

- 『Oracle8 Server 管理者ガイド』  
Oracle8 Server に関するさまざまなタスク（データベースの作成、データベース・オブジェクトの管理、ユーザーの作成など）を行う際に、まずお読みください。
- 『Oracle Net8 管理者ガイド』
- 『Oracle8 Server チューニング』

Oracle リレーショナル・データベース管理システムに慣れていない場合は、最初に『Oracle8 Server 概要』の第1章「Oracle8 Server の基礎知識」をお読みください。



---

# Oracle8 での最適フレキシブル・アーキテクチャ (OFA)

- Oracle8 での最適フレキシブル・アーキテクチャ (OFA)
- Oracle8 for UNIX で適用される OFA

## 最適フレキシブル・アーキテクチャ（OFA）

オラクル社では、Oracle8 に対して 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起きないように、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（ソフトウェアや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、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、次の利点があります。

### 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、かつ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 負荷を十分な数のディスク・ドライブに分散させるため、パフォーマンスのボトルネックが生じません。



## ハードウェア・サポート

操作上の考慮事項に矛盾しない範囲で、ハードウェア・コストを最小限に抑えられます。

## 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、ドライブに障害が起きた場合でも、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修正しなくても、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。
- ど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、次の利点があります。

- 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。

## Parallel Server における OFA のその他の利点

Oracle Parallel Server のインストールでは、他にも次の利点があります。

- すべての管理データを一元管理できます。
- あるインスタンスの管理データは、ファイル名によってそのインスタンスと対応付けることができます。

# Oracle8 for 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、データ管理のさまざま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: Oracle8 for Open Systems』(Cary V. Millsap 著)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・ポイントの名前

### 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の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 1-1 を参照)。

表 1-1 マウント・ポイントの名前の構文

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

たとえば、`/u01/oradata/test01` および `/u01/oradata/test02` 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 1-2 を参照)。

表 1-2 ホーム・ディレクトリの名前の構文

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

たとえば、`/u01/app/oracle` は、Oracle Server ソフトウェアの所有者のホーム・ディレクトリ（`ORACLE_BASE` の値で、Oracle Installer 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/appmgr` は、Oracle アプリケーション・ソフトウェアの所有者のホーム・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（たとえば、`/*/app/*`）で参照できます。

パス名の参照

明示的なパス名は、そのパス名を保存するためのファイル（`/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 Oracle8 Server ソフトウェアを `/pm/h/product/v` という構文のディレクトリにそれぞれ保存します（表 1-3 を参照）。

表 1-3 Oracle8 Server ソフトウェアを保存するディレクトリ名の構文

|           |              |
|-----------|--------------|
| <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>v</i>  | ソフトウェアのバージョン |

たとえば、`/u01/app/oracle/product/8.0.6` は、Oracle8 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` には、このディレクトリを設定します。

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する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・ディレクトリを、*d* はデータベース名（`DB_NAME`）を、*a* は次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示します（表 1-4 を参照）。

**表 1-4 データベース管理ファイルのサブディレクトリ**

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

たとえば、サブディレクトリ `adhoc` のパス名は、  
`/u01/app/oracle/sab/admin/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、そ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 1-5 に、この構文の詳細を示します。

**表 1-5 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 (前述)   |
| <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<br>(通常、ORACLE または <code>oradata</code> ) |
| <i>d</i>  | データベースの <code>DB_NAME</code>  |
| <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` に関連する制御ファイル、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 1-6 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 1-6 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      |
| TOOLS  | 汎用ツール             |
| USERS  | その他のユーザー・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。

Oracle8 では表領域に 30 文字の名前を付けられますが、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 データおよび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ので、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 1-7 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 1-7 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/8.0.6    | Oracle Server 8.0.6 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 1-8 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 1-8 OFA インストレーションの階層型ファイル・マッピング

|           |                             |
|-----------|-----------------------------|
| /         | ルート・マウント・ポイント               |
| u01/      | 「ユーザー・データ」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     |
| 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   |
| 7.3.4/    | 7.3.4 インスタンスの ORACLE_HOME   |
| 8.0.5/    | 8.0.5 インスタンスの ORACLE_HOME   |
| 8.0.6/    | 8.0.6 インスタンスの ORACLE_HOME   |
| 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             |
| 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  |
| 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             |
| vrm/      | 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  |
| u03/      | 「ユーザー・データ」のマウント・ポイント #3     |
| home/     | ログイン・ホーム・ディレクトリのサブツリー       |
| 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  |



## RAW デバイスのサイズ

Oracle データベース・ファイルを保存するための RAW デバイスはすべて、標準的なサイズに設定します。

できるだけ 1 つのサイズに標準化してください。これによって RAW ファイルをあるパーティションから別のパーティションに確実に移動できます。必要な数のデータベース・ファイルを保存できる程度の大きさで、かつ、操作しにくくならない程度の大きさに設定してください。

たとえば、2GB のドライブのパーティションを切る場合、サイズと数のバランスを取るために、大きさが 200MB のパーティションが 10 個できるように区切るとよいでしょう。RAW デバイスを使用するすべての表領域は、複数のドライブにストライプ化してください。できれば、論理ボリューム・マネージャを使用してストライプ化してください。

## 複数インスタンスにおける OFA に準拠したデータベースのファイル・マッピング

複数インスタンス・データベース（Oracle Parallel Server インストレーション）の場合、ファイル・マッピングについて、次のガイドラインがあります。

### Oracle Parallel Server の管理用ホーム・ディレクトリ

Oracle Parallel Server を使用する場合、そのクラスタに対して Oracle 管理用ホーム・ディレクトリとして機能するノードを 1 つ選択します。管理用ホーム・ディレクトリには、管理サブツリーが含まれます。~/admin/d/ にある bdump、cdump、logbook、pfile および udump ディレクトリのデータベースにアクセスする各インスタンスにサブディレクトリを作成します。管理ホームの admin ディレクトリは、すべてのインスタンスの admin ディレクトリにマウントする必要があります。[表 1-9](#) に例を示します。

表 1-9 二重インスタンスを持つ Oracle Parallel Server の管理ディレクトリ構造

|                           |  |                                 |
|---------------------------|--|---------------------------------|
| u01/app/oracle/admin/sab/ |  | sab データベース用の管理ディレクトリ            |
| adhoc/                    |  | スクリプト用のディレクトリ                   |
| arch/                     |  | すべてのインスタンス用のログ・アーカイブ先           |
| redo001.arc               |  | アーカイブ REDO ログ・ファイル              |
| bdump/                    |  | バックグラウンド・ダンプ・ファイルのディレクトリ        |
| inst1/                    |  | inst1 インスタンスのバックグラウンド・ダンプ先      |
| inst2/                    |  | inst2 インスタンスのバックグラウンド・ダンプ先      |
| cdump/                    |  | コア・ダンプ・ファイル用のディレクトリ             |
| inst1/                    |  | inst1 インスタンスのコア・ダンプ先            |
| inst2/                    |  | inst2 インスタンスのコア・ダンプ先            |
| create/                   |  | 作成スクリプト用のディレクトリ                 |
| 1-rdbms.sql               |  | inst データベース作成用の SQL スクリプト       |
| exp/                      |  | エクスポート用のディレクトリ                  |
| 990920full.dmp            |  | 9 月 20 日のフル・エクスポート・ダンプ・ファイル     |
| export/                   |  | エクスポート・ファイル用のディレクトリ             |
| import/                   |  | インポート・ファイル用のディレクトリ              |
| logbook/                  |  | inst ログブック項目用のディレクトリ            |
| inst1/                    |  | inst1 インスタンス・レポート用のディレクトリ       |
| params.lst                |  | inst1 インスタンスの v\$parameter レポート |
| inst2/                    |  | inst2 インスタンス・レポート用のディレクトリ       |
| params.lst                |  | inst2 インスタンスの v\$parameter レポート |
| user.lst                  |  | dba_users レポート                  |
| pfile/                    |  | インスタンス・パラメータ・ファイル用のディレクトリ       |
| inst1/                    |  | inst1 インスタンス・パラメータ用のディレクトリ      |
| init                      |  | inst1 インスタンスのインスタンス・パラメータ       |
| inst2/                    |  | inst2 インスタンス・パラメータ用のディレクトリ      |
| init                      |  | inst2 インスタンスのインスタンス・パラメータ       |
| udump/                    |  | ユーザー・ダンプ・ファイル用のディレクトリ           |
| inst1/                    |  | inst1 インスタンスのユーザー・ダンプ先          |
| inst2/                    |  | inst2 インスタンスのユーザー・ダンプ先          |

## 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 1-10 に、ORACLE\_BASE ディレクトリ構造の詳細を示します。Oracle 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 1-10 ORACLE\_BASE ディレクトリ構造および内容**

|         |                           |
|---------|---------------------------|
| 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、/mount\_point/app/oracle/product/release\_number です。表 1-11 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、および各 Oracle 製品用のサブディレクトリが含まれます。

**表 1-11 ORACLE\_HOME ディレクトリ構造および内容**

|         |                                    |
|---------|------------------------------------|
| bin     | 全製品のバイナリ・ファイル                      |
| ctx     | ConText カートリッジ                     |
| db      | init $sid$ .ora、lk $sid$           |
| jdbc    | JDBC ドライバ                          |
| lib     | Oracle 製品ライブラリ                     |
| md      | Spatial カートリッジ                     |
| mlx     | Xerox Stemmer ( ConText カートリッジ用 )  |
| network | Net8                               |
| nlsrtl  | NLS ランタイム読み込み可能データ                 |
| ocommon | 全製品に共通のファイル                        |
| odg     | データ収集機能                            |
| opsm    | Parallel Server Manager Components |

表 1-11 ORACLE\_HOME ディレクトリ構造および内容

|         |                             |
|---------|-----------------------------|
| oracore | コア・ライブラリ                    |
| orainst | マスター・インストール・ファイルおよびプログラム    |
| ord     | データ・カートリッジ                  |
| otrace  | Oracle TRACE                |
| plssql  | PL/SQL                      |
| precomp | プリコンパイラ                     |
| rdbms   | 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 |
| slax    | SLAX 解析機能                   |
| sqlplus | SQL*Plus                    |
| svrmgr  | Server Manager              |

Oracle 製品のサブディレクトリ

Oracle 製品には、システムで利用できる Oracle 製品および購入する製品によって、[表 1-12](#) に示すサブディレクトリがあります。

表 1-12 Oracle 製品のサブディレクトリ

|         |   |
|---------|---|
| network | Oracle Net8   |
| ocommon | ライブラリおよび SQL メッセージ。このディレクトリは自動的にインストールされ、すべての製品がこのディレクトリを使用します。 |
| plssql  | プロシージャ・オプションの PL/SQL バージョン 8.0.6                                |
| sqlplus | SQL*Plus  |
| svrmgr  | Server Manager  |

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 1-13](#) に示すサブディレクトリが含まれます。

表 1-13 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<br>(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   |

表 1-13 製品のサブディレクトリの内容

|         |   |
|---------|---|
| 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   |
| 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<br>(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<br>(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・<br>ファイル (oraus.msg、oraus.msb など) |

## 製品のサブディレクトリの例

表 1-14 に、製品のサブディレクトリの例を示します。

表 1-14 製品のサブディレクトリの例

|         |                                 |
|---------|---------------------------------|
| rdbsms  | 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

rdbsms/admin ディレクトリには、表 1-15 に示す SQL スクリプトが含まれます。

表 1-15 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 / ビューを作成します。<br>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。<br>catalog.sql (すべてのインストール)<br>catproc.sql (すべてのインストール)<br>catparr.sql (Parallel Server オプションのインストール)<br>catrep.sql (すべてのインストール) |
| dbms*.sql | その他のデータベース・パッケージ  |
| 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 1-16 に、ファイル名の拡張子を示します。

表 1-16 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)              |
| .ada | Ada ソース・ファイル                                     |
| .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                            |
| .dei | ORCA 削除スクリプト                                     |
| .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                             |
| .f   | FORTRAN ソース・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) |
| .ins | ORCA インストレーション・スクリプト                             |
| .l   | UNIX マニュアル・ページ                                   |
| .lst | SQL*Plus スクリプトの出力                                |
| .log | インストレーション・ログ・ファイル (Oracle Server REDO ログ・ファイル)   |
| .map | Installer 製品コンポーネント・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 | インストレーション・プロトタイプ・ファイル                            |
| .pad | Pro*Ada ソース・ファイル                                 |

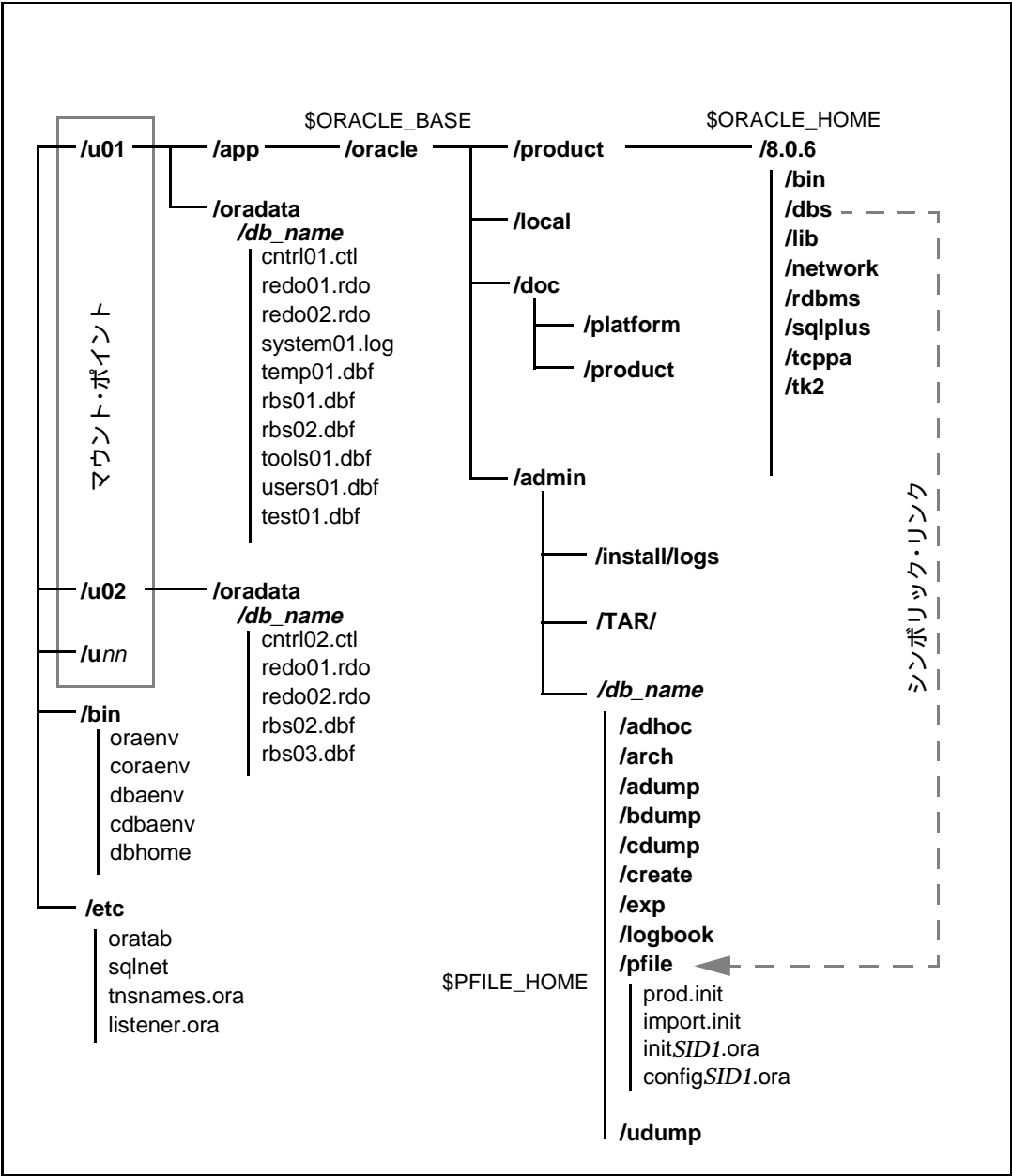
表 1-16 ファイル名の拡張子

|      |                             |
|------|-----------------------------|
| .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              |
| .pfo | Pro*FORTRAN ソース・ファイル        |
| .prd | 製品登録テンプレート・ファイル (orainst 用) |
| .res | Toolkit II リソース・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       |
| .us  | orainst メッセージ・ファイル          |
| .utd | ユニフォーム端末定義                  |
| .vrf | Installer 依存関係検証スクリプト       |

## デフォルトの OFA データベース

図 1-1 に、Oracle Installer によって作成されるデフォルトの OFA データベースを示します。

図 1-1 デフォルトの Oracle インストール





---

## AIX での Oracle8 の管理

- [initsid.ora ファイルのカスタマイズ](#)
- [環境設定](#)
- [Oracle8 の環境変数](#)
- [Oracle8 Server のメモリー使用量の見積り](#)
- [クラスタ・サイズおよび索引サイズの計算](#)
- [初期化パラメータ](#)
- [システム・グローバル領域の制御](#)
- [特殊アカウントおよび特殊グループの管理](#)
- [セキュリティの管理](#)
- [ログイン・ホーム・ディレクトリの管理](#)
- [デモンストレーションの作成と実行](#)
- [Oracle8 Server の SQL リファレンス](#)

# initSID.ora ファイルのカスタマイズ

ここでは、Oracle8 配布で提供されるデフォルトの `initSID.ora` ファイルについて説明します。このファイルは、Oracle Installer によって `$ORACLE_BASE/admin/db_name/pfile` ディレクトリに作成されます。このファイルを変更することによって、Oracle8 インストールをカスタマイズできます。

`initSID.ora` ファイルのパラメータには、どんなサイズのインストールにも共通の設定値があります。インストールのサイズに応じて値を変更する必要があるパラメータに対しては、小、中、大の3つのパターンが用意されています。サンプル `initSID.ora` ファイルでは、インストール・サイズによって設定が異なるパラメータが示されています。インストールには適用しない設定があれば、行頭に「#」を付けてコメント化できます。

表 2-1 に、`initSID.ora` ファイルに用意されている3つのパターンに対するおよそのSGAサイズを示します。

表 2-1 サンプル `initSID.ora` ファイルのブロック・サイズおよびSGAサイズ

| ブロック・サイズ | インストール / データベース・サイズ |        |         |
|----------|---------------------|--------|---------|
|          | 小                   | 中      | 大       |
| 2KB      | 4500KB              | 6800KB | 17000KB |
| 4KB      | 5500KB              | 8800KB | 21000KB |

## サンプル `initSID.ora` ファイル

サンプル `initSID.ora` ファイルは、RDBMS インストールをカスタマイズする際に使用すると便利です。サンプル `initSID.ora` ファイルのパラメータには、どんなサイズのインストールにも共通の設定値があります。インストールのサイズに応じて値を変更する必要があるパラメータに対しては、小、中、大の3つのパターンが用意されています。インストール・サイズに応じてチューニングする必要のあるパラメータには設定値が3つあり、それぞれにインストール・サイズを示すコメントが付いています。

```
# replace DEFAULT with your database name
db_name=DEFAULT

db_files = 80                                # SMALL
# db_files = 400                             # MEDIUM
# db_files = 1000                             # LARGE

db_file_multiblock_read_count = 8            # SMALL
# db_file_multiblock_read_count = 16         # MEDIUM
# db_file_multiblock_read_count = 32         # LARGE
```

```

db_block_buffers = 100                                # SMALL
# db_block_buffers = 550                                # MEDIUM
# db_block_buffers = 3200                                # LARGE

shared_pool_size = 3500000                              # SMALL
# shared_pool_size = 5000000                            # MEDIUM
# shared_pool_size = 9000000                            # LARGE

log_checkpoint_interval = 10000

processes = 50                                           # SMALL
# processes = 100                                         # MEDIUM
# processes = 200                                         # LARGE

parallel_max_servers = 5                                # SMALL
# parallel_max_servers = 4 x (number of CPUs)            # MEDIUM
# parallel_max_servers = 4 x (number of CPUs)            # LARGE

log_buffer = 8192                                        # SMALL
# log_buffer = 32768                                     # MEDIUM
# log_buffer = 163840                                    # LARGE

sequence_cache_entries = 10                             # SMALL
# sequence_cache_entries = 30                           # MEDIUM
# sequence_cache_entries = 100                          # LARGE

sequence_cache_hash_buckets = 10                       # SMALL
# sequence_cache_hash_buckets = 23                      # MEDIUM
# sequence_cache_hash_buckets = 89                      # LARGE

# audit_trail = true                                     # if you want auditing
# timed_statistics = true                               # if you want timed statistics
max_dump_file_size = 10240                              # limit trace file size to 5 Meg each

# Uncommenting the line below will cause automatic archiving if archiving has
# been enabled using ALTER DATABASE ARCHIVELOG.
# log_archive_start = true
# log_archive_dest = disk$rdms:[oracle.archive]
# log_archive_format = "T%TS%S.ARC"

# If using private rollback segments, place lines of the following
# form in each of your instance-specific init.ora files:
# rollback_segments = (name1, name2)

# If using public rollback segments, define how many
# rollback segments each instance will pick up, using the formula
# # of rollback segments = transactions / transactions_per_rollback_segment

```

```
# In this example each instance will grab 40/10 = 4:
# transactions = 40
# transactions_per_rollback_segment = 10

# Global Naming -- enforce that a dblink has same name as the db it connects to
global_names = TRUE

# Edit and uncomment the following line to provide the suffix that will be
# appended to the db_name parameter (separated with a dot) and stored as the
# global database name when a database is created. If your site uses
# Internet Domain names for e-mail, then the part of your e-mail address after
# the '@' is a good candidate for this parameter value.

# db_domain = us.acme.com # global database name is db_name.db_domain

# define parallel server (multi-instance) parameters
# ifile = ora_system:initsps.ora

# define two control files by default
control_files = (ora_control1, ora_control2)

# Uncomment the following line if you wish to enable the Oracle Trace product
# to trace server activity. This enables scheduling of server collections
# from the Oracle Enterprise Manager Console.
# Also, if the oracle_trace_collection_name parameter is non-null,
# every session will write to the named collection, as well as enabling you
# to schedule future collections from the console.

# oracle_trace_enable = TRUE
```

## 環境設定

### 環境変数の表示および設定

環境変数の値を表示するには、echo 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。

```
$ echo $ORACLE_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

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。

## 共通の環境設定

Oracle8 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 Oracle Server 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 (C シェルの場合は coraenv) 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
- Oracle Server データベース間を移動するメカニズム

たとえば、データベースを /usr/oracle から /usr1/oracle へ移動する必要があるとします。共通の環境設定のルーチンがない場合は、ユーザー起動ファイルを個々に更新する必要があります。oraenv を使用すると、各ユーザー・プロファイルから oraenv コマンド・ファイルがコールされ、そのファイルのみが更新されます。

### ローカル bin ディレクトリ

oraenv (または coraenv) 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、ローカル bin ディレクトリに入れることによって、別の ORACLE\_HOME を示すようにパスを変更した場合でも、oraenv (または coraenv) は変更せずに、そのまま使用できます。

### 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 ルーチンをコールし、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。必ず oraenv コマンド・ファイルのフルパス名を入力してください。次に例を示します。

```
$ . /usr1/bin/oraenv
ORACLE_SID= [default]? sid
```

## データベースの例

次に、ローカル bin ディレクトリが /usr/lbin で、本番データベースが PROD の場合の例を示します。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。

次の例では、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。

### 単一インスタンス

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/lbin
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/lbin
setenv ORACLE_SID PROD
set ORAENV_ASK = NO
source /usr/lbin/coraenv
unset ORAENV_ASK
```

### 複数インスタンス

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。

```
PATH=${PATH}:/usr/lbin
ORACLE_SID=PROD
export PATH ORACLE_SID
SIDLIST='awk -F: '/^[^#]/ {printf "%s ", $1}' /etc/oratab'
echo "SIDS on this machine are $SIDLIST"
ORAENV_ASK=
. oraenv
```

C シェルの場合、次のように入力します。

```
setenv PATH ${PATH}:/usr/lbin
setenv ORACLE_SID PROD
set sidlist = `awk -F: '/^[^#]/ {printf "%s ", $1}' /etc/oratab`
echo "SIDS on this machine are $sidlist"
unset ORAENV_ASK
source /usr/lbin/coraenv
```

## Oracle8 の環境変数

Oracle システムをインストールする前に、UNIX 環境の特定の変数を設定する必要があります。

**参照：** 詳細は、『Oracle8 for AIX-Based Systems インストレーション・ガイド』を参照してください。

## AIX での Oracle 環境変数

表 2-2 に、Oracle8 環境変数の機能、構文および例を示します。

表 2-2 AIX での Oracle8 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  |
|              | 例   | TRUE  |
| NLS_LANG     | 機能  | 出力に使用する言語とキャラクタ・セットを指定します。設定値の詳細は、『Oracle8 for AIX-Based Systems インストレーション・ガイド』を参照してください。 |
|              | 構文  | 言語_地域.キャラクタ・セット   |
|              | 例   | french_france.we8dec  |
| ORA_NLS33    | 機能  | 言語とキャラクタ・セットが保存されているディレクトリを指定します。   |
|              | 設定値 | \$ORACLE_HOME/ocommon/nls/admin/data  |
| ORACLE_BASE  | 機能  | OFA に準拠した Oracle データベースのディレクトリ構造のルートとなるディレクトリを指定します。                                      |
|              | 構文  | ディレクトリ・パス   |
|              | 例   | /mount_point/app/oracle   |

表 2-2 AIX での Oracle8 環境変数

| 環境変数            | 項目  | 定義  |
|-----------------|-----|---|
| ORACLE_HELP     | 機能  | ヘルプ・ファイルがあるディレクトリを指定します。  |
|                 | 構文  | ディレクトリ・パス   |
|                 | 例   | <code>\$ORACLE_HOME/help/admin/resource</code>  |
| ORACLE_HOME     | 機能  | Oracle 配布ソフトウェアがあるディレクトリを指定します。   |
|                 | 構文  | ディレクトリ・パス   |
|                 | 例   | <code>/mount_point/app/oracle/product/release_number</code>   |
| ORACLE_PATH     | 機能  | SQL*Plus 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。指定しないと、アプリケーションは現在のディレクトリに読み込みおよび書き込みを行います。   |
|                 | 構文  | ディレクトリをコロンで区切ったリスト<br>(ディレクトリ:ディレクトリ:ディレクトリ)  |
|                 | 例   | <code>/u01/oracle/adhoc/sqlplus:.</code>  |
|                 |     | <b>注意:</b> 最後にピリオドを付けることによって、検索パスに現在の作業ディレクトリが追加されます。   |
| ORACLE_SID      | 機能  | Oracle のシステム ID を指定します。   |
|                 | 構文  | 英字で始まる英数字列。詳細は、『Oracle8 for AIX-Based Systems インストレーション・ガイド』を参照してください。  |
|                 | 例   | <code>SAL1</code>   |
| ORACLE_TERM     | 機能  | 端末のタイプ識別子を指定します。Installer および Oracle 製品が正しい Toolkit II (.res) リソース・ファイルを調べる場合に使用します。設定しないと、オペレーティング・システム変数 TERM の値が使用されます。  |
|                 | 構文  | 文字列   |
|                 | 許容値 | <code>tk2c\${ORACLE_TERM}.res</code> が、Toolkit II リソース・ディレクトリにある有効なリソース・ファイルと一致するように ORACLE_TERM の値を設定します。設定値の詳細は、『Oracle8 for AIX-Based Systems インストレーション・ガイド』を参照してください。 |
|                 | 例   | <code>vt100</code>  |
| ORACLE_TERMINAL | 機能  | Toolkit II (.res) リソース・ファイルを検索するための追加ディレクトリを指定します。  |
|                 | 構文  | ディレクトリ名   |
|                 | 例   | <code>\$ORACLE_HOME/guicommon/tk21/admin/terminal</code>  |



表 2-2 AIX での Oracle8 環境変数

| 環境変数         | 項目  | 定義   |
|--------------|-----|--|
| 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を調整します。T を設定すると、多くの Oracle シェル・スクリプトは -x フラグがオンの状態で実行されます。                                 |
|              | 許容値 | T など   |
| ORAENV_ASK   | 機能  | (c)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 など  |
| TNS_ADMIN    | 機能  | Oracle Net8 構成ファイルがあるディレクトリを指定します。   |
|              | 構文  | ディレクトリ・パス  |
|              | 許容値 | 任意のディレクトリ。詳細は、『Oracle8 for AIX-Based Systems 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、Oracle Net8 接続文字列記述子の別名のデフォルトを指定します。   |
|              | 構文  | 使用可能なネットワークの別名   |
|              | 許容値 | tnsnames.ora ファイルで定義されている有効な Oracle Net8 の別名   |
|              | 例   | PRODDB_TCP   |

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

ORACLE\_HOME および ORACLE\_SID の略称

Oracle8 Server 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、Oracle8 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.ora' size 2M
```

@ マークは、\$ORACLE\_SID を表します。たとえば、ファイルが特定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@.ora'
```

## Oracle8 で使用する UNIX 環境変数

表 2-3 に、Oracle8 で使用する UNIX 環境変数の構文および例を示します。

表 2-3 Oracle8 で使用する UNIX 環境変数

| 環境変数              | 項目 | 定義   |
|-------------------|----|--|
| ADA_PATH          | 機能 | Ada コンパイラがあるディレクトリを指定します。  |
| DISPLAY           | 機能 | X ベースのツールで使います。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、AIX X-Window のドキュメントを参照してください。  |
|                   | 構文 | hostname:display<br>hostname はディスプレイ・デバイスのネットワーク ID、display は常に 0。   |
|                   | 例  | 135.287.222.12:0<br>bambi:0  |
| HOME              | 機能 | ユーザーのホーム・ディレクトリ  |
| LANG または LANGUAGE | 機能 | メッセージなどを出力するために AIX 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『Oracle8 for AIX-Based Systems インストール・ガイド』を参照してください。 |
| LD_LIBRARY_PATH   | 機能 | AIX で、Java アプリケーションの実行時に使います。  |
|                   | 構文 | ディレクトリをコロンで区切ったリスト<br>(ディレクトリ:ディレクトリ:ディレクトリ)   |
|                   | 例  | \$ORACLE_HOME/lib:\$ORACLE_HOME/network/lib  |
| LIBPATH           | 機能 | AIX で、共有ライブラリ・ローダーが実行時に共有オブジェクト・ライブラリを検索するために使います。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<br>(ディレクトリ:ディレクトリ:ディレクトリ)   |
|                   | 例  | /usr/dt/lib:\$ORACLE_HOME/lib  |
| PATH              | 機能 | シェルが実行プログラムを配置するために使います。<br>\$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<br>(ディレクトリ:ディレクトリ:ディレクトリ)   |

表 2-3 Oracle8 で使用する UNIX 環境変数

| 環境変数         | 項目  | 定義  |
|--------------|-----|---|
|              | 例   | /bin:/usr/bin:/usr/local/bin:<br>/usr/bin/X11:\$ORACLE_HOME/bin:\$HOME/bin:.<br><br><b>注意:</b> 最後にピリオドを付けることによって、検索パスに現在の作業ディレクトリが追加されます。 |
| PRINTER      | 機能  | ユーザーのデフォルト・プリンタを指定します。  |
|              | 構文  | プリンタ名   |
|              | 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  | シェルのパス名   |
|              | 許容値 | /bin/sh、/bin/csh、/bin/ksh、またはシステムに付属のその他のコマンド・インタプリタ  |
|              | 例   | /bin/sh   |
| TERM         | 機能  | Oracle Toolkit II キャラクタ・モード・ツールおよび AIX ツールが端末タイプを判断する場合に使用します。  |
|              | 例   | vt100   |
| TMPDIR       | 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  | ディレクトリ・パス   |
|              | 例   | /u02/oracle/tmp   |
| XENVIRONMENT | 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。  |

## 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。環境変数 TZ 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。

---

**警告：** できるだけ、ユーザーが個々に TZ 値を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更されてしまいます。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、表の順序を決める場合は、日付列ではなく順序番号を使用してください。

---

## Oracle8 Server のメモリー使用量の見積り

Oracle8 Server を起動する前に、次の式を使用して仮想メモリー要件を見積もることができます。

< Oracle 実行ファイルのテキストのサイズ >  
 + < SGA のサイズ >  
 +  $n \times$  ( < 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 >  
     + < Oracle 実行ファイルの未初期化データ・セクションのサイズ >  
     + < スタック用に 8192 バイト >  
     + < プロセス・ユーザー領域用に 2048 バイト > )

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 Oracle バックエンド接続に対して、次の式を使用して仮想メモリー要件を見積もります。

< Oracle 実行ファイルのデータ・セクションのサイズ >  
 + < Oracle 実行ファイルの未初期化データ・セクションのサイズ >  
 + < スタック用に 8192 バイト >  
 + < プロセス・ユーザー領域用に 2048 バイト >  
 + < アプリケーションで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 `bss`）を見積もるには、`size -f` コマンドを使用します。すべての実行ファイルのテキストは常に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 回だけです。

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 物理メモリーの使用量を計算するには、`ps` コマンドを使用します。すべてのフロント・エンド、サーバー、および Oracle バックグラウンド・プロセスの項目を検索します。項目ごとに、「プロセスの実際のサイズ（real size of process）」列を加算して、常駐メモリー使用量の小計を計算します。そして、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 回です。

**参照：** `ps` コマンドで使用できるスイッチのリストの詳細は、AIX の `man` ページを参照してください。

## クラスタ・サイズおよび索引サイズの計算

### クラスタ・サイズの計算

表 2-4 のガイドラインに従って、『Oracle8 Server 管理者ガイド』の付録 A に記載されている式を使用してクラスタ・サイズを計算します。

**表 2-4 クラスタ・サイズの値**

| タイプ              | サイズ                              |
|------------------|----------------------------------|
| 固定長ヘッダー・サイズ      | 68 バイト                           |
| 可変長トランザクション・ヘッダー | 表の <code>INITRANS</code> の値 × 24 |
| 行ディレクトリ          | クラスタ化表の行あたり 4 バイト                |

### 索引サイズの計算

表 2-5 のガイドラインに従って、『Oracle8 Server 管理者ガイド』の付録 A に記載されている式を使用して索引サイズを計算します。

**表 2-5 索引サイズの値**

| タイプ              | サイズ                               |
|------------------|-----------------------------------|
| 固定長ヘッダー・サイズ      | 113 バイト                           |
| 可変長トランザクション・ヘッダー | 索引の <code>INITRANS</code> の値 × 24 |
| 項目ヘッダー           | 5 バイト                             |

### サーバー・リソースの制限

AIX では、親プロセスからリソースの制限を継承します（`/etc/security/limits` ファイルおよび `getrlimit(2)` の `man` ページを参照）。この制限は、ユーザー・プロセスに対して実行される Oracle8 Server のシャドウ・プロセスに適用されます。AIX のデフォルトのリソース制限は、どの Oracle8 Server シャドウ・プロセスおよびバックグラウンド・プロセスに対しても十分高い値です。ただし、この制限を低くすると、Oracle8 Server システムに影響が出る場合があります。詳細は、AIX システム管理者にお問い合わせください。

Oracle dba ユーザー ID 用に設定したディスクの割当て制限によって、Oracle8 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、Oracle8 のデータベース管理者および AIX システム管理者にお問い合わせください。

## 初期化パラメータ

初期化パラメータは、Oracle8 Server インスタンスの `initSID.ora` ファイルで変更できます。

**参照：** 詳細は、『Oracle8 Server 管理者ガイド』を参照してください。

## デフォルトの初期化パラメータの値

表 2-6 に AIX 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かった場合、すべての Oracle8 Server 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・パラメータの現在の設定を表示するには、Server Manager を使用して SQL 文 `SHOW PARAMETERS` を実行します。

**参照：** 詳細は、『Oracle8 Server リファレンス・マニュアル』を参照してください。

表 2-6 デフォルトの初期化パラメータ

| パラメータ                         | デフォルト値   |
|-------------------------------|--|
| BACKGROUND_DUMP_DEST          | \$ORACLE_HOME/rdbms/log  |
| BITMAP_MERGE_AREA_SIZE        | 1048576  |
| COMMIT_POINT_STRENGTH         | 1  |
| CONTROL_FILES                 | \$ORACLE_HOME/dbs/ctrl@.dbf<br>( @ は ORACLE_SID )              |
| CREATE_BITMAP_AREA_SIZE       | 8388608  |
| DB_BLOCK_BUFFERS              | 50   |
| DB_BLOCK_SIZE                 | 4096   |
| DB_FILES                      | 30 ( 最大値は 2,000,000 )  |
| DB_FILE_DIRECT_IO_COUNT       | 64 ブロック ( 最大値は 1048576 )                                       |
| DB_FILE_MULTIBLOCK_READ_COUNT | 8 ( 許容値は 1 ~ 128。ただし、<br>DB_BLOCK_BUFFERS の 1/4 以下で設定してください。 ) |

表 2-6 デフォルトの初期化パラメータ

| パラメータ                         | デフォルト値                       |
|-------------------------------|------------------------------|
| DISTRIBUTED_TRANSACTIONS      | 8                            |
| HASH_AREA_SIZE                | 0                            |
| HASH_MULTIBLOCK_IO_COUNT      | 1                            |
| LOG_ARCHIVE_BUFFER_SIZE       | 127 ブロック                     |
| LOG_ARCHIVE_BUFFERS           | 4 ( 最大値は 128 )               |
| LOG_ARCHIVE_DEST              | \$ORACLE_HOME/dbs/arch/      |
| LOG_ARCHIVE_FORMAT            | "%t_%s.dbf"                  |
| LOG_BUFFER                    | 512KB                        |
| LOG_CHECKPOINT_INTERVAL       | 4294967294                   |
| LOG_SMALL_ENTRY_MAX_SIZE      | 80                           |
| MTS_MAX_DISPATCHERS           | 5                            |
| MTS_MAX_SERVERS               | 20                           |
| MTS_SERVERS                   | 0                            |
| MTS_LISTENER_ADDRESS          | ADDRESS=address ( 第 6 章を参照 ) |
| NLS_LANGUAGE                  | AMERICAN                     |
| NLS_TERRITORY                 | AMERICA                      |
| OBJECT_CACHE_MAX_SIZE_PERCENT | 10                           |
| OBJECT_CACHE_OPTIMAL_SIZE     | 102400                       |
| OPEN_CURSORS                  | 50                           |
| OS_AUTHENT_PREFIX             | ops\$                        |
| PROCESSES                     | 25                           |
| SHARED_POOL_SIZE              | 3500KB                       |
| SORT_AREA_SIZE                | 65536                        |
| SORT_READ_FAC                 | 5                            |
| SORT_SPACEMAP_SIZE            | 512                          |
| USER_DUMP_DEST                | \$ORACLE_HOME/rdbms/log      |

## システム・グローバル領域の制御

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。

## SGA のサイズ制限

AIX での単一共有メモリー・セグメントの最大サイズは 256MB です。プロセスでは、最大 10 個の連続共有メモリー・セグメントを割り当てられるので、AIX での最大 SGA サイズは 2.5GB です。

---

**警告：** Oracle を実行するときに、環境変数 EXTSHM を設定しないでください。EXTSHM が設定されていると、Oracle を実行できません。

---

SGA のサイズは、`init.ora` ファイルの次の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`

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

インスタンスの SGA の適切なサイズは、次の式で計算できます。

$$\begin{aligned} & ( \text{DB\_BLOCK\_BUFFERS} \times \text{DB\_BLOCK\_SIZE} ) \\ & + \text{SORT\_AREA\_SIZE} \\ & + \text{SHARED\_POOL\_SIZE} \\ & + 1\text{MB} \end{aligned}$$

実行中のデータベースの SGA サイズを表示するには、Server Manager の `show sga` コマンドを使用します。このコマンドによって、SGA サイズがバイト単位で表示されます。

## 特殊アカウントおよび特殊グループの管理

データベース管理者 (DBA) には、Oracle Server が必要とする特殊アカウントに関する知識と経験が必要で、それらのアカウントが適切なグループに属するように設定する必要があります。ここでは、特殊ユーザー・アカウントについて説明します。[表 2-7](#) に UNIX アカウントの詳細、[表 2-8](#) に Oracle Server アカウントの詳細を示します。



## 特殊アカウント

表 2-7 UNIX アカウント

|        |  |
|--------|--|
| oracle | oracle ソフトウェア所有者は、Oracle8 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 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、スーパーユーザーにはなりません。 |
| 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 2-8 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 Oracle8 アカウントです。SYS アカウントは、データ・ディクショナリの実表すべてを所有します。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 Oracle8 アカウントです。SYSTEM ユーザーでその他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。 |

## 特殊グループ

表 2-9 に、特殊グループ・アカウントの詳細を示します。

表 2-9 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、root ユーザーなどの UNIX ユーザーを追加できます。このグループのメンバーは、Server Manager 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 接続するためのパスワードを入力するか、または、Server Manager 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。デフォルトのグループ ID は dba です。 |
| oper グループ | ユーザーが作成することのできるオプションの 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。   |

## セキュリティの管理

Oracle8 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。

Oracle8 の 2 タスク・アーキテクチャによって、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。

## グループとセキュリティ

Oracle8 データベースのセキュリティを高めるため、オペレーティング・システム・レベルでユーザー・グループを作成します。グループは、UNIX ファイルの `/etc/group` が制御します。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 Server をインストールする前に、データベース管理者のグループ `dba` を作成し、このグループに `root` および `oracle` ソフトウェア所有者 ID を割り当てます。`dba` で実行できるプログラムにだけ、権限 710 が与えられています。システム権限が与えられた Server Manager コマンドは、インストール時に自動的に `dba` グループに割り当てられます。
- 権限ユーザーで構成される `oracle` グループを追加して、UNIX ユーザーのサブセットが Oracle8 へのアクセスを制限されるようにします。Oracle ユーティリティに `oracle` グループ ID を付けます。SQL\*Plus など、一般的に実行できるプログラムは、このグループから実行できる必要があります。ユーティリティの権限を 710 に設定して、このグループに実行権限を与えます。*other* に実行権限は与えないでください。
- *other* が実行できるプログラムに、権限 711 を与えます。この権限は、データベース・セキュリティに影響しないプログラムにのみ与えてください。

データベース管理者のグループには任意の名前を割り当てられますが、デフォルトのグループ名は `dba` で、このマニュアルでもその名前を使用しています。このグループ名を変更すると、Oracle Installer はインストール時にカーネルを自動的に再リンクします。同じ `ORACLE_HOME` でデータベースが複数ある場合（このような構成はお勧めしません）は、それらのデータベース管理者のグループは同じである必要があります。通常のユーザーのグループ名（`oracle` グループ）に、この制限はありません。

---

---

**警告：** oracle ソフトウェア所有者および root ユーザーは dba グループに属しますが、oracle ソフトウェア所有者は root グループのメンバーにはなりません。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。

---

---

## Oracle Server ユーティリティのセキュリティ

権限のないユーザーが Oracle8 実行プログラムを使用しないように保護する必要があります。Oracle8 実行プログラムを保護する方法は、環境やシングルタスク・ユーティリティを使用するかどうかによって異なります。次の方法で Oracle8 実行プログラムを保護します。

- すべてのプログラムを \$ORACLE\_HOME/bin ディレクトリに置き、oracle ソフトウェア所有者に所有権を与えます。
- マシン上のすべてのユーザーが Oracle Server にアクセスできるように、すべてのユーザー・ユーティリティ (sqlplus、exp、imp) に 711 の保護を与えます。
- すべての DBA ユーティリティ (Server Manager など) に 700 の保護を与えることによって、通常、それらのユーティリティを使用できるのは oracle ソフトウェア所有者である DBA ユーザー名のみになります。

## Server Manager コマンドのセキュリティ

SQL\*Plus がない場合は、Server Manager を使用して SQL 問合せを実行できます。ただし、Server Manager へのアクセスを割り当てるには注意が必要です。システム権限が与えられた次の文にアクセスできるのは、特別なオペレーティング・システム権限を持つ oracle ソフトウェア所有者および dba グループのユーザーのみとしてください。

- STARTUP
- SHUTDOWN
- CONNECT INTERNAL

---

---

**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、必要な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

Oracle8 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権限 0600 に設定します。これによって、所有者だけに読みおよび書き込み (rw) 権限が与えられ、グループや他のユーザーに書き込み権限は与えられません。

`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の所有者である必要があります。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み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。このビットを設定するには、次のように入力します。

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

これによって、Oracle プログラムの認可が次のように設定されます。

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

### ユーザー ID の設定

Oracle Installer は自動的にユーザー ID を設定します。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、事実上のユーザー ID が `oracle` であることを示しています。

## ネットワーク・セキュリティ

### ネットワーク上でのパスワードの使用

ネットワーク上のリモート・ユーザーは、クリア・テキストまたは暗号化テキストを使用してパスワードを入力できます。クリア・テキストを使用すると、許可されていないユーザーにパスワードが知られてしまい、セキュリティが失われます。Oracle Net8 では、暗号化パスワードが使用できます。

### ネットワークでの DBA 権限

ネットワークで DBA 権限を制御するには、次のいずれかの設定を行います。

- `/etc/listener.ora` ファイルで、リモート DBA アクセスを `denied` (拒否) に設定します。
- DBA 権限の `orapwd` で特別なパスワードを設定します。

## 自動 (ops\$) ログイン

Oracle8 では、ネットワークでの自動ログイン (オペレーティング・システムで許可されたログイン) をサポートしています。

UNIX では、ドル記号 (\$) が環境変数の始まりを示します。そのため、コマンド行またはスクリプトの中で、オペレーティング・システムによって許可された (ops\$) ログインを指定する場合、まず、バックスラッシュ (\) で \$ をエスケープします。たとえば、リモートにログインする場合、ユーザー ID scott には、ops\scott を指定する必要があります。

root ユーザー ID では、自動ログインできません。

---

**注意：** PC、Apple Macintosh、および OS/2 ユーザーが自動ログインすることはお薦めしません。だれでも Oracle 構成ファイルを編集したり、そのユーザー ID を変更したりできます。それらのシステムのユーザーがネットワークでログインする場合は、セキュリティ上の理由から listener.ora ファイルの ops\$ ログインを使用禁止にしてください。

---

## Oracle Net8 での自動ログイン

Oracle Net8 は、自動ログインおよびリモート DBA ログインを制御しません。自動ログインおよびリモート DBA ログインは Oracle8 Server によって制御され、initnsid.ora ファイルのパラメータを使用して構成されます。自動ログインはサポートされていますが、デフォルトでは使用禁止になっています。使用可能にするには、初期化パラメータ REMOTE\_OS\_AUTHENT を TRUE に設定し、データベースを起動します。

Oracle がそれらのログインを制御するため、root への setuid として Oracle Net8 リスナーを実行する必要はありません。

**参照：** Oracle Net8 の構成の詳細は、[第 6 章](#)を参照してください。

Oracle Net8 での自動ログインを実行するには、/etc/passwd ファイルに daemon というユーザーを作成します。この daemon ユーザーは、どのローカル・データベースにも ops\$ アカウントを持ってはいけません。また、どの DBA グループにも属してはいけません。つまり、外部ユーザーがローカル・データベースに侵入できる ops\$daemon アカウントを持ってはいけません。

## DBA グループ ID のキーワード

[表 2-10](#) に、リモート・ログインを使用可能にして制御するために、/etc/listener.ora ファイルで使用するキーワードを示します。

表 2-10 リモート・ログインを制御するキーワード

|   |  |
|---|--|
| DBA_GROUP   | 名前が、リスナーのサービス対象となるすべてのインスタンスの定数の場合に使用します。                                |
| DBA_GROUP_sid                                       | 2 つ以上の \$ORACLE_HOME がリスナーのサービス対象で、そのグループ ID が異なる場合、ORACLE_SID ごとに使用します。 |
| OPS_DOLLAR_LOGIN_ALLOWED<br>OPS_DOLLAR_LOGIN_DENIED | リモート・ログインを制御します。デフォルトは OPS_DOLLAR_LOGIN_DENIED です。                       |
| REMOTE_DBA_OPS_ALLOWED<br>REMOTE_DBA_OPS_DENIED     | リモート DBA アクセスを制御します。デフォルトは REMOTE_DBA_OPS_DENIED です。                     |

アクセスするデータベースの DBA グループ ID がデフォルト名 (dba) でない場合、デフォルト以外の名前を指定できます。

リモート・ログインおよびリモート DBA のパラメータには、ネットワーク上のデータベースの個々の ORACLE\_SID に設定するか、またはすべての sid を一度に設定します。たとえば、次の文はいずれも有効です。

```
PARAMETER=ALL_SIDS
PARAMETER=sid1[, sidn...]
```

sid にどちらの権限が割り当てられているかを調べるには、次のように入力します。

```
$ lsnrctl status
```

確認の順番

システムは、リモート・ログインのパラメータを次の順に確認します。

- 1. アクセスを拒否するパラメータ
- 2. アクセスを許可するパラメータ
- 3. デフォルト値 (拒否)

これらの権限は、Oracle Net8 リスナーの対象となるシャドウ・プロセスのユーザー ID とグループ ID を操作することによって実行されます。次に例を示します。

- 指定したインスタンスの OPS\_DOLLAR\_LOGIN\_DENIED が TRUE に設定されている場合、または、クライアント側のオペレーティング・システムによって通知されたユーザー ID のアカウントがデータベース・ホスト・マシンにない場合、そのユーザー ID およびグループ ID は daemon 項目の下の /etc/passwd ファイルにあります。

- 指定した ORACLE\_SID の OPS\_DOLLAR\_LOGIN\_ALLOWED および REMOTE\_DBA\_OPS\_ALLOWED が TRUE で、クライアントのオペレーティング・システムによって通知されたユーザー ID のアカウントがこのシステムにない場合、そのユーザー ID およびグループ ID はこのユーザー ID の /etc/passwd にあります。
- 指定した ORACLE\_SID の OPS\_DOLLAR\_LOGIN\_ALLOWED が TRUE で、REMOTE\_DBA\_OPS\_ALLOWED が FALSE の場合、そのユーザー ID に DBA 権限が与えられていれば、そのプロセスのユーザー ID およびグループ ID は daemon になります。それ以外の場合、プロセスのユーザー ID およびグループ ID は、このユーザーのユーザー ID およびグループ ID になります。

---

**注意：** デフォルトでは、REMOTE\_DBA\_OPS\_ALLOWED は FALSE です。この値は変更しないでください。このパラメータを FALSE に設定すると、DBA 権限を持つユーザーは、オペレーティング・システムで認可されたログインをネットワーク上で実行できなくなります。ただし、通常のネットワーク・ログイン（パスワードの入力が必要なネットワーク・ログイン）は実行できます。

---

## セキュリティとリモート・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・コンピュータなどのリモート・マシンからデータベースにアクセスしたり、データベースを管理したりできます。ユーザー認証は、orapwd ユーティリティで作成および管理される Oracle8 パスワード・ファイルを使用して行われます。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・ファイル検証も行えます。

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。

### 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 を実行するには、次のように入力します。

```
$ orapwd file=$ORACLE_HOME/dbs/orapwsid password=password entries=max_users
```

表 2-11 に、orapwd の構文を示します。

表 2-11 orapwd の実行構文

|                 |   |
|-----------------|---|
| <i>file</i>     | パスワード情報が書き込まれているファイル名。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>entries</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。   |

**注意：** ユーザーの最大数を増やすには、新しいパスワード・ファイルを作成する必要があります。そのため、*max\_users* は、必要な数より多く設定してください。

orapwd の例

```
$ orapwd file=/u01/app/oracle/product/8.0.6/dbs/orapwV806 \  
password=manager entries=30
```

**参照：** 詳細は、『Oracle8 Server 管理者ガイド』を参照してください。

リモート PC からデータベースへのアクセス

Oracle8 パスワード・ファイルがあれば、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。権限がないユーザーの場合は、このデータベースを使用する Oracle アプリケーションを起動することによって、データベースにアクセスできます。権限のある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、PC から、適切な Server Manager 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。

```
SVRMGR > connect internal/dba_password
```

OPERATOR として接続するには、同じ Server Manager コマンドに OPERATOR パスワードを追加して使用します。



## リモート認証

表 2-12 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 `initssid.ora` ファイルのパラメータを示します。

表 2-12 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。 |

**注意：** REMOTE\_OS\_AUTHENT を TRUE に設定すると、リモート・マシンの dba グループのメンバーであるユーザーは、パスワードなしで INTERNAL として接続できます。

## 停止時のエラー・メッセージ

Oracle インスタンスの停止時、インスタンスに接続しているクライアントでは、SQL 処理に関して次のいずれかのエラー・メッセージが表示されます。

ORA-03113: 通信チャネルでファイルの終わりが検出されました。  
ORA-12571: TNS: パケットの書き込み機に障害が発生しました。

## ログイン・ホーム・ディレクトリの管理

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せずにそのディレクトリを追加または移動するには、次のようにします。

- 明示的なパス名は、そのパス名を保存するファイル ( `/etc/passwd`、`/etc/oratab` 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。

次のいずれかの方法でユーザーのホーム・ディレクトリを参照できるため、主な参照ファイル以外にパス名を記録しておく必要はありません。

- C シェルおよび Korn シェルのユーザーは、`~login` を使用してユーザーのホーム・ディレクトリを参照できます。
- Bourne シェルのユーザーは、ユーザーのホーム・ディレクトリを参照するための簡単なプログラムを作成できます。この項で示すサンプル `lhd` スクリプトを参照してください。

同じように、グループのメンバーも `/etc/group` で参照できます。この項で示すサンプル `grpX` スクリプトを参照してください。

## サンプル lhd スクリプト

```
#!/bin/sh
#
# lhd - print login home directory name for a given user
#
# SYNTAX
# lhd [login]
#
prog='basename $0'
if [ $# -eq 0 ] ; then
    login='whoami'

elif [ $# -eq 1 ] ; then
    login=$1
else
    echo "Usage: $prog login" >&2

    exit 2
fi
awk -F: '$1==login {print $6}' login=$login /etc/passwd
```

## サンプル grp\_x スクリプト

```
#!/bin/sh
# grp_x - print the list of users belonging to a given group
#
prog='basename $0'
if [ $# -ne 1 ] ; then
    echo "Usage: $prog group" >&2
    exit 2
fi
g=$1
# calculate group id of g
gid='awk -F: '$1==g {print $3}' g=$g /etc/group'
# list users whose default group id is gid
u1='awk -F: '$4==gid {print $1}' gid=$gid /etc/passwd'
# list users who are recorded members of g
u2='awk -F: '$1==g {gsub(/,/," "); print $4}' g=$g /etc/group'
# remove duplicates from the union of the two lists
echo $u1 $u2 | tr " " "\012" | sort | uniq | tr "\012" " "
echo
```

### 例 2-1 lhd スクリプトおよび grpx スクリプトの例

次に、管理者が .profile ファイルの基本的な構成をグループの各メンバーのホーム・ディレクトリに反映させる例を示します。clerk グループのメンバー・リストを変更しても、コードを修正する必要はありません。

```
$ for u in 'grpx clerk' ; do
> cp /etc/skel/.profile 'lhd $u'
> done
```

## デモンストレーションの作成と実行

### PL/SQL デモンストレーションのロード

PL/SQL には、ロードできるサンプル・プログラムがたくさんあります。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。Oracle8 Server を起動してマウントした状態で、次の操作を行ってください。

1. 次のように入力して、Server Manager を起動し、scott/tiger (ユーザー / パスワード) で接続します。

```
$ cd $ORACLE_HOME/plsql/demo
$ svrmgrl
SVRMGR > connect scott/tiger
```

2. 次のように入力して、デモンストレーションをロードするために、Server Manager から exampbld.sql を起動します。

```
SVRMGR > @exampbld
```

---

**注意：** デモンストレーションのビルドは、権限がある 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。

---

## PL/SQL デモンストレーションの実行

表 2-13 に、カーネルのデモンストレーションを示します。

表 2-13 カーネルのデモンストレーション

| 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     |            |             |             |

表 2-14 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 2-14 プリコンパイラのデモンストレーション

| 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 |

カーネルの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、カーネルに接続します。デモンストレーション名の前に @ 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
```

プリコンパイラの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
```

1 つのデモンストレーションをビルドする場合は、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
```

現在のシェルから 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、tnsnames.ora ファイル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=/vobs/oracle)(PROGRAM=extproc)
```

次に、SQL\*Plus セッションで次のように入力します。

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

最後に、テストを実行します。

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

## SQL\*Loader のデモンストレーション

SQL\*Loader のデモンストレーションには、次の要件があります。

- 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

デモンストレーションを作成して実行するには、次の手順に従います。

1. Server Manager を起動し、scott/tiger (ユーザー ID/ パスワード) でデータベースに接続します (ライン・モード)。
2. 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。
3. 次のように入力して、scott/tiger でコマンド行からデモンストレーションを起動します。

```
$ sqlldr scott/tiger ulcasen
```

scott/tiger で、次の順に SQL\*Loader のデモンストレーションを実行します。

- ulcasel: 前述の手順 1 ~ 3 を実行します。

- ulcase3: 前述の手順 1 ~ 3 を実行します。
- ulcase4: 前述の手順 1 ~ 3 を実行します。
- ulcase5: scott/tiger で ulcase\*.sql スクリプトを実行し、コマンド行に次のように入力します。  

```
$ sqlldr scott/tiger ulcase*
```
- ulcase2: デモンストレーションを起動します (ulcase2.sql スクリプトを実行する必要はありません)。
- ulcase6: scott/tiger で ulcase6.sql スクリプトを実行し、コマンド行に次のように入力します。  

```
$ sqlldr scott/tiger ulcase1 DIRECT=true
```
- ulcase7: scott/tiger で ulcase6.sql スクリプトを実行し、コマンド行に次のように入力します。  

```
$ sqlldr scott/tiger ulcase7
```

## SQL\*Loader の管理

Oracle8 Server 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。

### ファイル処理オプション

SQL\*Loader の制御ファイルには、次のファイル処理オプション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" ]
```

`str` デフォルトです。改行文字で終了するレコードを、1 度に 1 レコードずつ読み込むストリームを指定します。

`fix` ファイルが `n` 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。`n` は整数です。

`var` ファイルが `n` 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。`n` 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走査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。

---

**警告：** `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

改行文字をロードしないで、固定長レコードから改行文字を削除するには、制御ファイルの `position(x:y)` 関数を使用します。制御ファイルに次のように入力します。

```
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 桁目にあるために、改行文字は削除されます。

## Oracle Security Server

Oracle Security Server の詳細は、『Oracle Security Server ガイド』を参照してください。

# Oracle8 Server の SQL リファレンス

## CREATE CONTROLFILE パラメータ

表 2-15 を参照して、データベースの制御ファイルのサイズを判断します。

表 2-15 制御ファイルのサイズの判断

| パラメータ         | デフォルト値 | 最大値   |
|---------------|--------|-------|
| MAXDATAFILES  | 30     | 65534 |
| MAXINSTANCES  | 1      | 1024  |
| MAXLOGFILES   | 16     | 16512 |
| MAXLOGMEMBERS | 2      | 5     |
| MAXLOGHISTORY | 100    | 65534 |



---

## AIX での Oracle8 のチューニング

この章では、AIX システムにおける Oracle8 データベース・サーバーのパフォーマンスおよびスケーラビリティを改善するツールおよびテクニックについて説明します。この章の情報と一緒に、汎用的なチューニング方法が記載された『Oracle8 Server チューニング』および IBM の『AIX Versions 4.x Performance Tuning Guide』を参照してください。また、Oracle チューニングを扱っているその他の一般的な Oracle ドキュメントも参照してください。

この章では、次のトピックについて説明します。

- [AIX パフォーマンス・ツール](#)
- [Oracle パフォーマンス・ツール](#)
- [Oracle8 Server のチューニング](#)
- [メモリーおよびページング](#)
- [ディスク I/O の問題](#)
- [CPU のスケジューリングとプロセスの優先順位](#)
- [その他のチューニングのヒント](#)
- [関連ドキュメント](#)

これらのトピックは読みやすいように並べられているだけで、特定の順序を示すものではありません。

使用している環境に伴う前述のチューニングに必要な作業量は、アプリケーションおよび作業負荷に応じて異なります。

## AIX パフォーマンス・ツール

ここでは、AIX オペレーティング・システムの標準またはオプション機能として使用可能なツールについて説明します。

### AIX Base Operation System ツール

AIX Base Operation System (BOS) には、履歴的に UNIX システムの一部であったツールやインプリメンテーション固有の AIX 機能を管理するのに必要なパフォーマンス・ツールがあります。主要な BOS 機能は、次のとおりです。

|         |   |
|---------|---|
| iostat  | CPU および I/O 統計をレポートします。   |
| lsattr  | デバイスの属性を表示します。  |
| lslv    | 論理ボリュームまたは物理ボリュームの論理ボリューム割当ての情報を表示します。                                      |
| netstat | ネットワーク関連のデータ構造の内容を表示します。  |
| nfsstat | Network File System (NFS) および Remote Procedure Call (RPC) アクティビティの統計を表示します。 |
| nice    | プロセスの最初の優先順位を変更します。   |
| no      | ネットワーク・オプションを表示または設定します。  |
| ps      | プロセスの状態を表示します。  |
| reorgvg | ボリューム・グループ内の物理的なパーティションの割当てを再編成します。   |
| sar     | システム・アクティビティ情報を収集し、レポートまたは記録します。  |
| time    | 実行が終了するまでの経過時間、ユーザーおよびシステムの CPU 処理時間を出力します。                                 |
| trace   | 選択されたシステム・イベントを記録およびレポートします。  |
| vmstat  | 仮想記憶アクティビティおよびその他のシステム統計をレポートします。   |

### AIX Performance Toolbox

AIX Performance Toolbox (PTX) には、システム・アクティビティをローカルおよびリモートで監視およびチューニングするツールがあります。PTX は、PTX Manager および PTX Agent という 2 つの主なコンポーネントで構成されています。PTX Manager は、xmperf というプログラムを使用して、構成内のさまざまなシステムからデータを収集し、表示します。PTX Agent は、xmserd というプログラムを使用して、データを収集して PTX Manager に送信します。PTX Agent は、Performance Aide for AIX という独立した製品としても提供されています。

PTX および Performance Aide には、監視用とチューニング用のツールが個別に含まれています。

|           |   |
|-----------|---|
| fdpr      | 特定の作業負荷について実行プログラムを最適化します。                            |
| filemon   | トレース機能を使用して AIX ファイル・システムのアクティビティを監視し、レポートします。        |
| fileplace | 論理ボリュームまたは物理ボリューム内でのファイルのブロックの配置を表示します。               |
| lockstat  | カーネル・ロックの競合について統計を表示します。                              |
| lvedit    | ボリューム・グループ内の論理ボリュームの対話形式の配置を簡易化します。                   |
| netpmon   | トレース機能を使用してネットワーク I/O およびネットワークに関連した CPU 使用率をレポートします。 |
| rmss      | さまざまなサイズのメモリーでシステムをシミュレートして、パフォーマンスをテストします。           |
| svmon     | 仮想メモリーの使用方法について情報を獲得し、分析します。                          |
| syscalls  | システム・コールを記録およびカウントします。                                |
| tprof     | トレース機能を使用してモジュールおよびソース・コード文レベルでの CPU 使用率をレポートします。     |
| BigFoot   | プロセスのメモリー・アクセス・パターンをレポートします。                          |
| stem      | 既存の実行ファイルをサブルーチン・レベルでの入口または出口として使用できます。               |

PTX の詳細は、『AIX Performance Toolbox/6000 User's Guide 1.2 and 2.1』を参照してください。一部のツール・コマンドの構文は、『AIX Version 4.x Commands Reference』にも記載されています。

## AIX System Management Interface Tool

AIX System Management Interface Tool ( `smit` ) は、さまざまなシステム管理ツールおよびパフォーマンス・ツールに対するメニュー方式のインタフェースを提供します。`smit` を使用すると、多数のツールを容易に操作でき、実行するジョブのみに集中できます。

## Oracle パフォーマンス・ツール

ここでは、オラクル社のパフォーマンス監視ツールについて説明します。

### Utlbstat / Utlestat による統計の収集

utlbstat.sql および utlestat.sql スクリプトは、\$ORACLE\_HOME/rdbms/admin ディレクトリにあります。通常の作業負荷期間の開始時に Server Manager のライン・モードから utlbstat を実行し、期間の終わりに utlestat を実行します。これによって、utlestat スクリプトを実行したディレクトリに report.txt ファイルが作成されます。このレポートは、測定期間のパフォーマンス統計が含まれているので、データベース競合の発見とチューニングに役立ちます。初期化パラメータ TIMED\_STATISTICS を TRUE に設定しなければ、タイミング測定がすべて 0 で表示されます。

### 動的パフォーマンス表

Oracle Server には、リソースの競合およびボトルネック、パフォーマンス問題の検出に役立つ多数の動的サーバー・パフォーマンス表またはビューがあります。よく使用される表の一部を次に示します。

- V\$system\_event
- V\$resource\_limit
- V\$waitstat
- V\$session\_wait
- V\$rowcache
- V\$ping
- V\$file\_ping
- V\$lock\_activity

### SQL トレース

初期化パラメータ SQL\_TRACE=TRUE を設定すると、インスタンスのすべてのセッションで SQL トレースがオンに設定されます。または、コマンド ALTER SESSION SET SQL\_TRACE = TRUE を発行して、SQL トレースをセッション・レベルでオンに設定することもできます。該当する各セッションのトレース・ファイルは、初期化パラメータ USER\_DUMP\_DEST で指定されたディレクトリに作成されます。このパラメータのデフォルト値は、\$ORACLE\_HOME/rdbms/log です。

Oracle の TKPROF コマンドを実行して、生成されたトレース・ファイルを分析できます。SQL トレース機能および TKPROF の詳細は、『Oracle8 Server チューニング』を参照してください。

## Oracle Enterprise Manager Performance Tuning Pack

Oracle Enterprise Manager リリース 1.5.5 では、システムのパフォーマンスを監視およびチューニングするために次のパッケージが提供されています。

### Oracle Diagnostics Pack

Oracle Diagnostics Pack は、次のパッケージから構成されます。

- Oracle Performance Manager
- Oracle Capacity Planner
- Oracle Trace Manager and Viewer
- Oracle TopSessions
- Oracle Lock Manager

#### Oracle Performance Manager

Oracle データベース・サーバーおよびオペレーティング・システムのパフォーマンス・データをグラフィカルおよび動的に表示する、リアルタイム・パフォーマンス・モニターです。Oracle データベース・カートリッジおよびオペレーティング・システム・カートリッジは、ホストにインストールする必要があります。

#### Oracle Capacity Planner

Oracle Capacity Planner を使用すると、システムおよびデータベース管理者は、データベースおよびオペレーティング・システムのパフォーマンス統計値を収集し、そのデータを履歴データベースに記録できます。さらに、履歴データを分析して、将来的に必要な容量を見積もることができます。

Oracle データベース・カートリッジおよびオペレーティング・システム・カートリッジのほかに、エージェント・データの採集サービスを、オペレーティング・システムのデータを収集するすべてのホストにインストールする必要があります。

#### Oracle Trace Manager and Viewer

グラフィカルな Oracle Trace Manager アプリケーションでは、Oracle Trace API を使用して計測された製品に対する Oracle Trace の収集データを作成、スケジューリングおよび管理できます。Oracle Intelligent Agent をホストにインストールして、Oracle Trace Collection Services を使用可能な状態にしておく必要があります。

#### Oracle TopSessions

上位 n 個の Oracle セッションのパフォーマンス・アクティビティをいつでも集中的に表示できます。

#### **Oracle Lock Manager**

ロックによってブロックされたセッションを表示および解決するために使用します。

#### **Oracle Tuning Pack**

Oracle Tuning Pack は、次のパッケージから構成されます。

- Oracle Expert
- Oracle Tablespace Manager
- Oracle SQL Analyze

#### **Oracle Expert**

データベースの初めての構築や、既存データベースのパフォーマンスに見られる特徴の収集および評価をサポートして、データベース環境のパフォーマンスを最適化します。

#### **Oracle Tablespace Manager**

表領域の詳細情報を提供し、表の断片化を解消したり使用されていない領域の割当てを解除したりできます。

#### **Oracle SQL Analyze**

ツールを使用して、データベース環境およびスキーマ・オブジェクト情報の収集、SQL パフォーマンスの解析、異なるオプティマイザ手法の識別や比較、および最適なパフォーマンスの文例の編集を自動的に行えます。

## **Oracle8 Server のチューニング**

Oracle8 Server は、高度に最適化できるソフトウェア製品です。Oracle for AIX アプリケーションの有効範囲が拡大し、複雑化するなかで、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、データ障害の発生を防ぎます。

## **Oracle8 Server のチューニング順序**

Oracle Server のパフォーマンスを最大にするために、多数のタスクを実行します。1 つのチューニング・アクションでの決定が後続のチューニング手順に影響を与える可能性があるため、次の順序で作業してください。

1. アプリケーション設計のチューニング
2. データ・アクセスのチューニング
3. メモリー管理のチューニング
4. ディスク I/O のチューニング

5. CPU の使用率のチューニング
6. Oracle リソース競合のチューニング
7. Oracle Parallel Server の Oracle リソースのチューニング

この章では、メモリーおよびディスク I/O、CPU の使用率に関連した作業 3、4 および 5 を説明します。作業 1、2、6 および 7 の詳細は、この章の最後にある「[関連ドキュメント](#)」のドキュメントを参照してください。

## メモリーおよびページング

メモリーの競合は、プロセスが使用できるメモリーが不足した場合に発生します。システムはメモリー不足に対処するために、メモリーとディスクとの間でプログラムおよびデータのページングを行います。

### バッファ・キャッシュ間のページング・アクティビティの制御

過度なページング・アクティビティは、パフォーマンスを大きく低下させます。特に JFS ファイルで作成されたデータベースでは、パフォーマンスが著しく低下することがあります。この場合は、JFS バッファに入っている最も頻繁に参照されるデータが、多数の SGA データ・バッファにも入ることがあります。ファイル・システムのバッファ・キャッシュ・マネージャの動作は、パフォーマンスに重大な影響を与えます。I/O ボトルネックが発生し、システム全体のスループットを低下させる可能性があります。

AIX では、バッファ・キャッシュ間のページング・アクティビティをチューニングできますが、回数を控えて慎重に行う必要があります。

次の 4 つの AIX システム・パラメータをチューニングできます。

|         |  |
|---------|--|
| minfree | 空きリスト・サイズの最小値。このサイズ以下では、ページ・スティーリングによって空きリストが補充されます。 |
| maxfree | 空きリスト・サイズの最大値。このサイズを超えると、ページ・スティーリングが終了します。          |
| minperm | ファイル I/O 用のバッファ・ページ数の最小値。                            |
| maxperm | ファイル I/O 用のバッファ・ページ数の最大値。                            |

vmtune コマンドで前述の値を変更するには、root 権限が必要です。AIX の vmtune ユーティリティは、オペレーティング・システムのバージョンに固有のユーティリティです。異なるリリースで実行可能な vmtune コマンドを実行すると、オペレーティング・システムが故障する可能性があります。

### minfree と maxfree のチューニング

迅速な応答が要求されるプログラムのワーキング・セット・サイズに基づいて minfree を設定できます。基本的には、ロードしたプログラムが空きリストを補充する必要がないように、空きリストには十分なページを用意しておく必要があります。vmstat コマンドによってプログラムのワーキング・セットのサイズを設定します。maxfree は、8 または maxpgahead の値だけ minfree よりも大きくする必要があります。たとえば、maxpgahead が 16 で、minfree を 128 に設定する場合は、次のコマンドを使用します。

```
/usr/lpp/bos/samples/vmtune -f 128 -F 144
```

これにより、minfree が 128、maxfree が 144 に設定されます。

## AIX ファイル・バッファ・キャッシュのチューニング

AIX バッファ・キャッシュは、JFS ファイルを使用するときにディスクへのアクセス回数を減らすために使用します。キャッシュが小さすぎると、ディスク使用率が増加し、1 つ以上のディスクが一杯になる可能性があります（このパラメータの増加による影響の詳細は、「[バッファ・キャッシュ間のページング・アクティビティの制御](#)」を参照してください）。ただし、キャッシュが大きすぎても、貴重なメモリーが無駄になります。

minperm および maxperm パラメータを調整して、AIX バッファ・キャッシュを構成できます。一般的に、バッファのヒット率が低い場合（90% 未満、sar -b を実行した結果を参照）minperm を増やしてください。高いバッファ・ヒット率を維持する必要がない場合は、minperm を減らすと使用可能な物理メモリーが増えます。詳細は、AIX のドキュメントを参照してください。

パフォーマンスの向上は、マルチプログラミングの度合いおよび作業負荷の I/O 特性に左右されるので、簡単に数値で示すことはできません。

### minperm と maxperm のチューニング

AIX では、minperm 値と maxperm 値を調整することで、ファイルで使用するページ枠数と計算可能（作業用またはプログラム・テキスト）セグメントで使用するページ枠数の比率を柔軟に制御する方法を管理者に提供します。minperm 値と maxperm 値は、次のガイドラインに従って調整します。

- 実メモリーでファイル・ページの占める割合が minperm 値よりも低くなった場合は、リページ率にかかわらず、ページ置換アルゴリズムはファイルおよび計算可能ページの両方を奪います。
- 実メモリーでファイル・ページの占める割合が maxperm 値よりも高くなった場合は、ページ置換アルゴリズムはファイル・ページのみを奪います。
- 実メモリーでファイル・ページの占める割合が minperm と maxperm の中間である場合、VMM は通常、ファイル・ページのみを奪いますが、ファイル・ページのリページ率が計算可能ページのリページ率よりも高いと、計算可能ページも奪います。



デフォルト値の計算方法は、次のとおりです。

$$\text{minperm ( ページ数 )} = (( \text{ ページ枠数 } ) - 1024) \times 0.2$$

$$\text{maxperm ( ページ数 )} = (( \text{ ページ枠数 } ) - 1024) \times 0.8$$

デフォルト値を変換するには、次のコマンドを入力します。

```
vmtune -p 5 -P 20
```

このコマンドは、minperm と maxperm のデフォルト値がそれぞれ 20% と 80% であったのに対して、minperm を合計ページ枠数の 5% に変更し、maxperm を合計ページ枠数の 20% に変更します。

RAW デバイスのデータベース・ファイルの場合は、minperm と maxperm を低い値（たとえば、それぞれ 5% と 20%）に設定できます。これは、AIX ファイル・バッファ・キャッシュが RAW デバイスには使用されないからです。したがって、メモリーを他の目的（Oracle システム・グローバル領域など）に有効利用できます。

## 十分なページング領域の割当て

ページング領域が不十分であると、システムがハングしたり、応答時間が非常に遅くなったりします。AIX では、ページング領域を RAW ディスク・パーティションに動的に追加できます。必要なページング領域は、現在の物理メモリー容量および各アプリケーションのページング領域要件に応じて異なります。AIX コマンド `lspcs` を使用して、ページング領域の利用率を監視し、`vmstat` コマンドを使用してシステムのページング・アクティビティを監視します。

通常は、ほとんどの作業負荷で必要とされる物理メモリー容量の 2 倍または 3 倍になるように、ページング領域を構成するのが最適です。CASE、Oracle Applications、または Oracle Office を使用する場合は、もう少し高い値に設定してください。ページング領域の使用率を監視し、必要に応じて増やします。ページング領域を増やすには、`smit pgsp` コマンドを使用します。かなり大きい物理メモリーを持つシステム（1GB 以上）の場合は、物理メモリーの 1.5 倍のページング領域で十分です。

## ページングの制御

ページングが定期的および頻繁に発生する場合は、実メモリーの負担が大きすぎることを示します。通常は、次のことを行います。

- Oracle が SGA とディスク間でデータを読み書きする速度よりも速くメモリーと拡張記憶領域間のページングを実行する、非常に高速な拡張記憶領域を装備したシステム以外では、定期的なページングを防止する必要があります。

- システム・パフォーマンスにとって一番有利な場所に、限定されたメモリー・リソースを割り当てます。メモリー・リソース要件とつり合いがとれるまで何回も均衡化処理を繰り返すことがあります。
- メモリーが不十分な場合は、メモリーが必要なシステムのプロセスまたは要素の優先順リストを作成します。パフォーマンスの増加が最大になるように、メモリーを割り当てます。次に、各要素の優先順リストの例を示します。

OS および RDBMS のカーネル

ユーザー・プロセスおよびアプリケーション・プロセス

REDO ログ・バッファ

PGA および共有プール

PCM ロックおよび DLM

DB ブロック・バッファ・キャッシュ

たとえば、Oracle Server の動的パフォーマンス表またはビューに対して問合せを実行し、共有プールおよびデータベース・バッファ・キャッシュの両方でメモリーの増加が必要であることが判明した場合は、限られた予備のメモリーをデータベース・ブロック・バッファ・キャッシュではなく、共有プールに割り当てた方が有利です。

次の AIX コマンドは、ページングの状況および統計を表示します。

```
$ vmstat -s  
$ vmstat interval [repeats] および  
$ sar -r interval [repeats]
```

## データベース・ブロック・サイズの設定

Oracle データベース・ブロック・サイズの設定によって I/O スループットを向上できます。AIX では、2KB から 16KB までの範囲で `DB_BLOCK_SIZE` を設定できます。デフォルトは 4KB です。JFS 対応の Oracle データベースの場合は、ブロック・サイズを JFS ブロック・サイズ (AIX では 4KB) の倍数で設定する必要があります。RAW パーティション上のデータベースでは、Oracle データベース・ブロック・サイズは、オペレーティング・システムの物理ブロック・サイズ (AIX では 512 バイト) の倍数になります。

OLTP 環境または複合的作業負荷の環境では小さい Oracle データベース・ブロック・サイズ (2KB または 4KB)、DSS タイプの作業負荷では大きいブロック・サイズ (8KB または 16KB) を設定することをお勧めします。

## アーカイバ・バッファのチューニング

専用の大きいバッファを割り当てることによって、データベース・アーカイブ速度を向上させることができます。ただし、過剰な設定はシステム全体のパフォーマンスを低下させる場合があります。init.ora ファイルのパラメータ `LOG_ARCHIVE_BUFFER_SIZE` を 128 まで増やせます。LOG\_ARCHIVE\_BUFFERS のデフォルトは 4 です。

通常のデータベース・アクティビティが大きく低下することがないように、慎重にチューニングします。アーカイブ・プロセスにおいて 0 ~ 20% のパフォーマンスの向上が見込まれます。これ以上の向上を得たユーザーもいます。

## SGA サイズのチューニング

SGA での複数の共有メモリー・セグメントの使用を制約する要因は、現在のハードウェア・アーキテクチャがサポートできる物理メモリー容量です。ユーザーは、メイン・メモリー不足による Oracle SGA のページ・アウトは避けたいはずです。多数のユーザーが同時にデータベースを利用する場合は、大きい SGA をお勧めします。SGA 構成の詳細は、第 2 章の「[システム・グローバル領域の制御](#)」を参照してください。

SGA サイズに最も影響する `init.ora` ファイルのパラメータは、`DB_BLOCK_BUFFERS` および `SHARED_POOL_SIZE` です。Oracle Parallel Server が実行中の場合、`init.ora` ファイルの `LM_LOCKS`、`LM_RESS` および `LM_PROCS` パラメータは、SGA サイズに大きく影響します。これらのパラメータをどれだけ増やせるかは、SGA に割当てられるメモリーに左右されます。`DB_BLOCK_BUFFERS` および `SHARED_POOL_SIZE` に大きい値を設定すると、パフォーマンスのオーバーヘッドが生じるので、これらの値を小さくすれば CPU 使用率が減少します。

## I/O バッファと SQL\*Loader

データをパラレルでロードする以外に、SQL\*Loader ダイレクト・パス・オプションなどの高速なデータ・ロード方法を使用すると、CPU が I/O 操作の完了を待っている時間が長くなります。バッファ数を増やすと、通常は CPU 使用量が増加し、全体的なスループットが向上します。

選択するバッファ数 (SQL\*Loader の `BUFFERS` パラメータ) は、使用可能なメモリー量および CPU 使用率の拡大範囲に応じて決定します。`BUFFERS` パラメータのファイル処理オプション文字列の調整方法の詳細は、『Oracle8 Server ユーティリティ』を参照してください。

パフォーマンスの向上は、CPU 使用率およびロードするときに使用される並行度によって異なります。SQL\*Loader ユーティリティの一般情報は、『Oracle8 Server ユーティリティ』の各章を参照してください。

## ディスク I/O の問題

ディスク I/O の競合は、（後続のページングおよびスワッピングを伴った）不適切なメモリー管理、ディスク間での不適切な表領域およびファイルの分布によって生じることがあります。I/O 負荷は、すべてのディスクに均等に分散する必要があります。

### 索引と表の分離

索引とその参照先の表が同じディスク・ドライブに存在すると、索引検索に関連するすべての I/O が同じディスク上に集中します。I/O 負荷が高い場合は、索引と表を別々のドライブに保存して負荷を分散させる必要があります。

### REDO ログの専用ディスク・デバイスへの配置

INSERT および UPDATE アクティビティが頻繁に発生する Oracle アプリケーションでは、その他のディスク・アクティビティをサポートしないディスク上に REDO ログを配置することで、Oracle パフォーマンスを最大化できます。また、ARCHIVELOG オプションが使用可能になっている場合は、すべての REDO ログをそれぞれのディスクに配置し、LGWR プロセス（現行 REDO ログへの書込み）と ARCH プロセス（クローズされた REDO ログからの読込み）との間で生じるディスク競合を最小化します。REDO ログを RAW デバイス上に配置すれば、さらにパフォーマンスが向上します。次の理由から、REDO ログは、最初のファイルの 1 つとして RAW デバイスに配置してください。

- REDO ファイルではデータが順次に読み書きされ、RAW デバイスの利点を最大化する。
- REDO ログのファイル・サイズは固定されているので、RAW デバイスの管理コストを最小化する。

### ホット・ファイルの他のディスクへの移動

頻繁にアクセスされるホット・ファイルを、あまり利用されていないディスク・デバイスに分散し、I/O のバランスをとります。ファイル全体を使用頻度の高いディスクから低いディスクに移動するか、または各ディスク上にホット・ファイルの一部分が配置されるように、ホット・ファイルを複数のディスクに分散させます。

### ホット・ファイルの I/O の削減

ディスク・デバイス上にホット・ファイルが 1 つだけ存在し、このファイルが担当する要求キューが大きい場合は、ファイルを別のディスクに移動しても効果はありません。問題の Oracle ファイルまたは表領域に、複数のセグメント（表および索引など）のデータが入っている場合は、頻繁にアクセスされるセグメントを別の表領域に移動し、別々のファイルに移動します。データベース・セグメントの物理デバイスは、表領域レベルで指定する必要があります。1 つのセグメントのみが関与している場合は、1 つの表領域内の複数のファイルにセグメント・データを配置するように、表のストライプ化を検討してください。

## データベースの過剰な断片化のチェック

Oracle データ構造を断片化すると、複数の物理 I/O から 1 つの論理 I/O の要素を結合する必要があります。余分なオーバーヘッドは、応答時間を長くします。

### エクステンツの断片化

データベース・セグメントに、複数の連続していないディスク領域エクステンツが含まれている場合があります。ディスクを順次に読み込めないため、または複数の I/O に分割されるため、I/O 時間が長くなります。ディスク上で物理的に連続していない複数のエクステンツにデータが分散しているために、1 つの I/O 要求を 2 つ以上の物理 I/O に分割する必要があるときに、I/O 分割が発生します。

### 表領域の断片化

Oracle 表領域は、いくつかの個別ファイルで構成されています。表領域内で、Oracle セグメント（表、索引など）は多数の個別エクステンツから構成されており、表領域の断片化が始まります。表のストライプ化のように、断片化が好ましい場合もありますが、通常は好ましくありません。データベース・セグメントを削除するたびに、表領域の断片化が起こります。表領域が断片化すると、空き領域を有効に使用できません。

表領域が断片化すると、Oracle のマルチブロック読み込み機能が利用できなくなります。また、連続した空きエクステンツよりも大きいエクステンツが存在すると、データベース領域が無駄になります。次の SQL 文で表領域の断片化を識別できます。

```
SELECT * FROM DBA_EXTENTS;
```

次の SQL 文を使用し、空き領域に問合せを実行できます。

```
SELECT * FROM DBA_FREE_SPACE;
```

空き領域の断片化には、次の 2 つの形式があります。

- 泡立ち状。アクティブなエクステンツ間に挟まっていたエクステンツを削除したときに、泡のように小さく連続していない空き領域が形成されます。
- 蜂の巣状。隣接したエクステンツが削除されるたびに、空き領域が連続的な断片に区分化されます。

utlfbstat および utlestat で高い再帰コール値が報告された場合は、表領域の断片化が考えられます（ただし、データ・ディクショナリ・キャッシュが適切にチューニングされている場合）。

**参照：** 詳細は、『Avoiding a Database Reorganization』（Craig A. Shallahamer 著）を参照してください。  
WWW（<http://www.europa.com/~orapub>）から入手できます。

## AIX 論理ボリューム・マネージャ

AIX 論理ボリューム・マネージャ (LVM) には、ディスク競合を削減するために、複数のディスク装置にわたってデータをストライプ化する機能があります。ストライプ化の主な目的は、大きい順次ファイルを読み書きするときに、高パフォーマンスを得ることです。LVM のストライプ化機能を有効に使用することで、I/O を複数のディスクに均等に分散でき、全体的にパフォーマンスが向上します。

### ストライプ化した論理ボリュームの設計

ストライプ化した論理ボリュームを定義するときに、次の項目を指定する必要があります。

|                        |  |
|------------------------|--|
| ドライブ                   | 2 つ以上の物理ドライブ。パフォーマンス・クリティカルな順次 I/O を実行するときに、これらのドライブのアクティビティを最小限に抑える必要があります。2 つ以上のアダプタ間での論理ボリュームのストライプ化が必要になる場合があります。    |
| ストライプ・<br>ユニット・<br>サイズ | 2KB から 128KB までの範囲にある、任意の 2 の累乗です。ただし、32KB および 64KB がほとんどの作業負荷に適しています。Oracle データベース・ファイルでは、データベース・ブロック・サイズの倍数である必要があります。 |
| サイズ                    | 論理ボリュームに割り当てる物理パーティション数は、使用するディスク・ドライブ数の倍数である必要があります。  |
| 属性                     | ミラー化できません。つまり、copies=1 です。   |

### ストライプ化した論理ボリュームの推奨パラメータ

次に、通常では最高の順次 I/O スループットを達成する、ストライプ化した論理ボリュームのパラメータを示します。

|                        |   |
|------------------------|---|
| ストライプ・<br>ユニット・<br>サイズ | 64KB  |
| minpgahead             | 2 (最小先読みページ数)   |
| maxpgahead             | ディスク・ドライブ数の 16 倍 (最大先読みページ数)。ストライプ・ユニット・サイズ (64KB) とディスク・ドライブ数の積を単位として先読みします。つまり、1 回の先読みで、1 ディスク・ドライブにつき 1 ストライプ化ユニットを読み込みます。 |
| maxfree                | maxpgahead の変更を保存します。   |

(注意: minpgahead、maxpgahead および maxfree は、vmtune の調整可能なパラメータです)

filemon、sar、iostat またはその他のパフォーマンス・ツールなどの AIX ユーティリティを使用して、ホット・ディスクがないかを調べることで、I/O アクティビティが複数のディスク装置に均等に分散していることを確認します。

### その他の考慮事項

LVM を有効に利用した場合にどれだけパフォーマンスが向上するかは、使用した LVM および作業負荷の特性によって大きく異なります。意思決定支援システム (DSS) の作業負荷では、パフォーマンスが大きく向上する場合があります。OLTP タイプまたは複合的な作業負荷でも、かなりのパフォーマンスの向上を期待できます。

## 異なるデータベース・ファイルの同一ディスクへの配置の回避

同時にアクセスされる可能性があるパーティションの場合、同じ物理ディスクからパーティションを使用するのは避けてください。たとえば、データ・ファイルとログ・ファイルは同時にアクセスされることがたいへん多いので、これらのファイルは別々のディスク・セットにストライプ化する必要があります。そうしないと、ディスク・ヘッドのシーク時間がかかり、パフォーマンスが低下します。このとき、ディスク上のパーティションの位置が互いに離れていると、パフォーマンスがさらに低下します。

## ジャーナル・ファイル・システム (JFS) と RAW パーティションの使用

この話題は、時折論議的になります。この話題が論議的になるいくつかの理由を示します。

- さまざまなファイル・システム・インプリメンテーションが改善されるなかで、ファイル・システムは絶えず進歩し続けています。ファイル・システムの方が RAW デバイスよりも優れたパフォーマンスを提供する場合があります。
- さまざまなベンダーが、さまざまなディスクの長所を利用して、独自の方法でファイル・システム層をインプリメントしています。これによって、各プラットフォームに対するユーザーの理解がますます複雑化します。
- パワフルな LVM インタフェースの導入によって、RAW パーティションを基盤とした論理ディスクの構成や元に戻す作業の負担が著しく軽減されます。

パフォーマンスがどの程度向上するかは、作業負荷の特性によって大きく異なります。

JFS を使用した方が、RAW デバイスを使用する場合よりも、データベース・ファイルの管理およびメンテナンスが簡単です。ただし、JFS ではバッファの読み書きのみがサポートされており、データを I/O システム (Oracle バッファ・キャッシュ以外、SGA) に転送したり受け取ったりするたびに、追加の AIX ファイル・バッファ・キャッシュが作成されます。これは、JFS ベースのデータ・ファイルの主な短所であり、I/O サブシステムの負荷が高いときほど重大な欠点となります。

また、負荷が高いシステムでは、AIX バッファの競合が主なボトルネックになります。次のような影響があります。

- データをファイル・システム・バッファ・キャッシュからディスクへフラッシュするページデーモンの負担が増加します。
- さらに多くのメモリーが、AIX バッファ・キャッシュの犠牲になります。

RS/6000 プラットフォームでのオラクル社の経験では、作業負荷に応じて異なりますが、一般的に、ファイル・システムに比べて RAW デバイスを使用したときの方が全体的なパフォーマンスおよびスケーラビリティが向上します。したがって、使用頻度が高いデータベース構成では、RAW デバイスの使用をお勧めします。

**注意：** Oracle Parallel Server の場合は、データベース・ファイルに JFS ではなく、RAW デバイスを使用してください。

### ファイル・システムから RAW デバイスへの移動

1. UNIX の dd コマンドを使用すると、データを簡単に JFS から RAW デバイスに移動できるので、すべてのデータを手動で再ロードしなくて済みます。次のように dd コマンドを入力します。

```
dd if=/home/<my_old_JFS_file> /  
of=/dev/<my_new_raw_device> seek=1 bs=4k
```



必ず、次のことを実行してください。

- RAW デバイスの最初の 4KB ブロックは、AIX 論理ボリューム・マネージャが使用するのでスキップします。

**注意：**RAW デバイスが HSD の場合、最初の 4KB ブロックはスキップする必要はありません。

- 領域が無駄にならないように RAW デバイスのサイズを決定します (RAW デバイスは、JFS ファイルよりも数ブロック大きくする必要があります)。
  - データ・ファイルを改名します。
  - RAW パーティションに権限を設定します。
2. 順次読み込みが多少遅くなる場合があります。アプリケーションで多数の全表走査を実行していた場合は、これらの表が AIX ファイル・バッファ・キャッシュによってキャッシュされないで、全表走査でサーバーのパフォーマンスが低下する場合があります。
  3. メモリが不十分な構成では、AIX バッファ・キャッシュのサイズ変更が必要です。詳細は、「メモリーとページング」の項を参照してください。
  4. RAW デバイスの名前が変更される可能性があるので、AIX 環境ではできる限りシンボリック・リンクの使用をお勧めします。たとえば、ディスクの周辺装置を再構成したり、データベースを新しいハードウェア構成に移動したりするときに、名前が変更されることがあります。詳細は、IBM のドキュメントを参照してください。

RAW デバイスの使用に関連した操作の詳細は、『Oracle8 for AIX-Based Systems インストール・ガイド』を参照してください。

## JFS と RAW デバイスの両方の長所の利用

AIX では一部のデータ・ファイル用に JFS を、その他のデータ・ファイル用に RAW パーティションを自由に選択できます。非同期 I/O は、RAW パーティションおよび JFS の両方で動作します。データベース・オブジェクトの I/O アクティビティの性質が事前にわかっている場合は、特定のオブジェクトに対応するデータ・ファイルを JFS または RAW パーティションのどちらかに配置し、可能であれば LVM を伴った体系を考えることができます。

パフォーマンスの向上は、作業負荷の性質、ディスクまたはファイル・システムの構成に応じて大きく異なります。

## 非同期 I/O の使用方法

Oracle8 では、AIX が提供する非同期 I/O (AIO) を最大限に活用し、高速なデータベース・アクセスを実現します。AIO は複数の I/O 操作を混ぜ合せて I/O サブシステムのスループットを向上します。データを各ディスク間に適切に分散しなければ、AIO の利点は利用できません (データベース、ログおよび制御ファイルの分散の詳細は、『Oracle8 Server チューニング』を参照してください)。

論理ボリューム・マネージャ (LVM) を使用してストライプ化を行うと、AIO の効果が強化されます。LVM は、データを複数のディスク装置にストライプ化することで、ディスク競合を削減します (詳細は、『AIX 論理ボリューム・マネージャ』の項を参照してください)。AIO を LVM と一緒に使用すると、RDBMS パフォーマンスが著しく向上します。

AIX 4.x では、ファイル・システム・パーティションおよび RAW デバイスの両方で作成されたデータベース・ファイルに対して AIO をサポートします。AIO 要求を発行した後に、アプリケーション・プロセスの実行を継続できます。ファイル・システム、VSD または HSD での AIO 要求の場合、要求をキューから取り出した時点から要求が完了するまで、サーバーと呼ばれるカーネル・プロセス (*kproc*) がそれぞれの要求を管理します。システムで同時に実行できる AIO 要求数は、サーバーの台数によって制限されます。AIX 4.x では、VSD および HSD 以外の RAW デバイスで AIO をカーネル化します。つまり、AIX サーバーは AIO 要求を処理する必要がありません。(AIX 4.3 では、APAR IX 79690 がインストールされていてファスト・パス AIO が使用可能な場合、VSD および HSD の AIO はカーネル化されています。)

サーバーの台数は、`smit (smit -> Devices -> Asynchronous I/O -> Change/Show Characteristics of Asynchronous I/O -> {MINIMUM|MAXIMUM})` または `smit aio`) または `chdev` で設定します。最小値は、システムの起動時に起動されるサーバー数です。最大値は、同時に発生した多数の要求を処理するために起動可能なサーバー数を制限します。

デフォルト値は、`minservers=1` および `maxservers=10` です。通常、これらの値は、ファイル・システム (または、ファスト・パス AIO のない VSD または HSD) で、AIO を使用して Oracle RDBMS を実行するには小さすぎます。オラクル社では、次の値を推奨します。

- `maxservers`      $10 \times (\text{AIO が同時にアクセスするディスク数})$
- `minservers` = `maxservers/2`

## DB\_FILE\_MULTIBLOCK\_READ\_COUNT の使用

一般的に、`init.ora` ファイルのパラメータ `DB_FILE_MULTIBLOCK_READ_COUNT` の値が大きい方が、I/O スループットが優れています。AIX では、このパラメータの範囲は 1 から 512 までですが、通常は 16 より大きい値に変更してもパフォーマンスは向上しません。

このパラメータの設定は、`DB_BLOCK_SIZE`  $\times$  `DB_FILE_MULTIBLOCK_READ_COUNT` の方が LVM ストライプ・サイズよりも大きくなるようにする必要があります。つまり、I/O 要求を満たすために、ディスク装置を多く起用します。

## RAID 機能の使用

RAID 5 は、順次読み込みのパフォーマンスを向上させますが、全体的な書込みパフォーマンスを低下させます。一般的に、書込みが少ない場合にのみ RAID 5 を使用することをお勧めします。RAID5 での大量の書込みは、RAID 以外の環境に比べてパフォーマンスが低下する場合があります。

RAID 0/1 では、通常、ストライプ化 / ミラー化をハードウェア・レベルで導入するとパフォーマンスが向上します。これは、AIX または Oracle のレベルで行うより効率的です。

RAID 7 は、非同期 I/O (デバイス、パス、バスなど) および複数のホスト・インタフェースを得るために、埋込み OS を採用します。これによって、大小の読み書きにおいて RAID 0 ~ 6 よりも優れたパフォーマンスを提供できます。

## 後書きの使用

ファイル・システムでは、書込みパフォーマンスを増加し、メモリー内の使用済みページ数を制限し、ディスクの断片化を最小化するために、各ファイルを 16KB パーティションに分割します。プログラムが次の 16KB パーティションの先頭バイトに書き込むまで、パーティションのページはディスクに書き込まれません。vmtune コマンドを使用して、後書きパーティションのサイズを変更できます。

```
vmtune -c 8
```

このコマンドは、後書き用のバッファ・サイズを 8 個の 16KB パーティションに設定します。

```
vmtune -c 0
```

このコマンドは、後書きを使用禁止にします。後書きを使用禁止にすると、パフォーマンスがいくらか向上することがあります。

## 順次ファイルの先読みのチューニング

Virtual Memory Manager (VMM) は、順次ファイルのページの必要性を予知します。VMM はプロセスがファイルにアクセスするパターンを監視します。プロセスがファイルにアクセスし、連続した 2 ページを読み込むと、VMM はプログラムが引き続きファイルに順次アクセスすると仮定して、追加の順次読み込みをスケジュールします。これらの読み込みは、プログラムの処理とオーバーラップし、通常よりも速くデータを使用可能にします。先読みするページ数は、次の 2 つの VMM しきい値によって決定します。

### minpgahead

VMM が最初に順次アクセス・パターンを検出したときに先読みするページ数。

### maxpgahead

順次ファイルで VMM が先読みする最大ページ数。

minpgahead および maxpgahead は、アプリケーションに応じて適切な値に設定します。デフォルト値は、それぞれ 2 および 8 です。vmtune コマンドを使用して、これらの値を変更できます。ストライプ化した論理ボリュームの順次パフォーマンスが最も重要であるシステムでは、maxpgahead の大きい値を使用できます。vmtune を使用して minpgahead および maxpgahead を設定できます。次のようにコマンドを入力します。

```
vmtune -r 512 -R 1024
```

このコマンドは、minpgahead を 512 ページに、maxpgahead を 1024 ページに設定します。minpgahead および maxpgahead は、2 の累乗に設定します。

## ディスク I/O の歩調合せ

AIX のメカニズムであるディスク I/O の歩調合せによって、システム管理者はファイルに対して保留される I/O 要求の数を制限できます。これによって、ディスク I/O が激しいプロセスによって CPU が占領されるのを防止できます。したがって、対話形式のプロセスおよび CPU 使用量の激しいプロセスの応答が遅くなることはありません。

高水位標および低水位標の 2 つのシステム・パラメータを調整して、ディスク I/O の歩調合せを達成します。保留中の I/O 要求が高水位標に達しているファイルに対してプロセスが書き込みを実行すると、そのプロセスはスリープ状態に設定されます。待機中の I/O 要求が低水位標以下に下がったとき、このプロセスはスリープ状態から解放されます。

smit を使用して、高水位標および低水位標を変更できます。smit -> System Environment -> Change/Show Characteristics of Operating System のように選択し、新しい高水位標および低水位標を入力します。試行錯誤しながら、水位標を判断します。適切に設定しないと、パフォーマンスに悪影響を及ぼすことがあるので、慎重に設定してください。ディスク I/O が 4KB を超える場合は、高水位標および低水位標をチューニングしても、あまり効果はありません。

## ディスク配置の考慮事項

AIX では、ディスク上の論理ボリュームの配置をある程度制御できます。ホット論理ボリュームを相互に近接して配置すれば、ディスクのシーク時間が短くなり、全体的なパフォーマンスが向上します。

## リモート I/O の最小化

SP アーキテクチャで実行中の Oracle Parallel Server では、Virtual Shared Disks (VSD) または Hashed Shared Disks (HSD) を、異なるノード上で実行しているすべての OPS インスタンスからアクセス可能な共通記憶装置として使用します。論理ボリュームがノードから見てローカルな VSD に対して I/O 要求が出された場合は、ローカル I/O が実行されます。ローカルでない VSD への I/O 通信量は、ネットワーク通信レイヤーを経由します。

優れたパフォーマンスを得るには、リモート I/O をできる限り少なくすることが重要です。各インスタンスの REDO ログは、ローカル論理ボリュームの VSD に配置する必要があります。更新および挿入が激しい場合、ローカル論理ボリュームにマップされる VSD 上のプライベート・ロールバック・セグメントを各インスタンスが持っている必要があります。

各セッションで、各ユーザーが使用できるテンポラリ表領域は 1 つのみです。テンポラリ表領域では、OPS を実行中の各ノードに対してローカルなデータ・ファイルが 1 つ以上必要です。

アプリケーションおよびデータベースを慎重に設計し（たとえば、インスタンスに対してアプリケーションおよびデータベースをパーティション化する）、リモート I/O を少なくしてください。

## VSD キャッシュ・バッファを使用しない

IBM では、通常の状態では VSD キャッシュ・バッファを使用しないこと（nocache）をお勧めします。これは、VSD LRU キャッシュ・バッファが、確保されたカーネル・メモリーをより有効な方法で使用するためです。また、キャッシュ・バッファが使用可能な状態に設定されていると、物理読み込みのたびにオーバーラップしたページを検索するためにキャッシュ・ブロックが検索され、キャッシュ・バッファの内外へのデータのコピーによってオーバーヘッドが生じます。

VSD のパフォーマンスを確認するには、statvsd コマンドを使用します。要求がキューに入ったままバッファを待機している状態が持続している場合でも、バッファを追加しないでください。かわりに、切替え送信プールのサイズを、次のようにして増やします。

```
# /usr/lpp/ssp/css/chgcss -l css0 -a spoolsize=new_size_in_byte
```

送信プール・サイズを増やした場合は、mbuf の上限値も増やす必要があります。

```
# /etc/no -o thewall=new_size_in_kbyte (Maximum 64MB)
```

現行の送受信プールのサイズを確認するには、次のように入力します。

```
$ /usr/sbin/lsattr -El css0
```

---

---

**参照：** SP および VSD のチューニングの詳細は、  
<http://www.rs6000.ibm.com/support/sp/perf/> および  
<http://www.rs6000.ibm.com/resource/technology/#sp>  
を参照してください。

---

---

## CPU のスケジューリングとプロセスの優先順位

CPU は、プロセス間で競合が生じる可能性のあるもう 1 つのシステム・コンポーネントです。通常は AIX カーネルが効率的に CPU を割り当てますが、多くのプロセスが CPU サイクルのために競合します。CPU を 2 つ以上搭載したシステム (SMP) では、それぞれの CPU で異なるレベルの競合が発生する可能性があります。

### プロセスのランタイム・スライスの変更

AIX RR ディスパッチャのランタイム・スライスのデフォルト値は、10 ミリ秒です。schedtune コマンドを使用して、タイム・スライスを変更できます。ただし、十分に注意して変更してください。アプリケーションの自発的な切替え率の平均が低い場合、タイム・スライスを長くすると、コンテキスト切替え率が低くなります。結果的に、プロセスでコンテキスト切替えに消費される CPU サイクルが減少し、システム・スループットが向上します。

しかし、特に単一プロセッサ・システムでは、ランタイム・スライスが長くなると、応答時間が長くなる場合があります。デフォルトのランタイム・スライス (10 ミリ秒) は通常、標準アプリケーションにとって適切な値です。実行キューが高く、多くのアプリケーションおよび Oracle シャドウ・プロセスで長時間の実行が可能となるときは、次のように入力してタイム・スライスを増やします。

```
# /usr/samples/kernel/schedtune -t n
```

この場合、*n* の値は 0 = 10 ms、1 = 20 ms、2 = 30 ms、3 = 40 ms などです。

### SMP マシンでのプロセッサ・バインディング

SMP マシンで、特定のプロセスをプロセッサにバインドすると、パフォーマンスが大きく向上することがあります。プロセッサのバインディングは、AIX 4.x で使用可能であり、全機能がサポートされています。

プロセッサ・バインディングの利点は、次のとおりです。

- 優先順位の高いアプリケーションに対して CPU 時間が相対的に多く割り当てられます。
- プロセス・コンテキストの維持期間が長くなります。

AIX でのプロセッサ・バインディングは、自動ではありません。bindprocessor コマンド (man ページ参照) を使用して、マルチプロセッサで明示的にプロセスをプロセッサにバインドする必要があります。Oracle プロセスをプロセッサにバインドできるのは、スーパーユーザーか oracle 所有者のみです。子プロセスは、プロセッサ・バインディングを継承します。

さまざまな Oracle バックグラウンド・プロセス (dbwr 以外) を別々のプロセッサにバインドし、1 つの空きプロセッサを dbwr 用に残しておくことをお勧めします。これによって、dbwr を実行するプロセッサが保証されると同時に、CPU バウンドになった場合、dbwr が自由に他のプロセッサへ移行できます。

**注意：** プロセッサ・バインディングは複雑であるため、慎重に扱う必要があります。プロセッサにバインドされたプロセスは、プロセッサが空いている場合でも、別のプロセッサに移行できません。これにより、アプリケーションのパフォーマンスが低下する場合があります。プロセッサ・バインディングに適しているのは、負荷が均等に分散した同一アプリケーション環境です。

プロセスとプロセッサのバインドは、排他的ではありません。つまり、プロセッサは自由に他のプロセスを実行できます。

## ネットワーク化したクライアント・サーバー環境でのプロセッサ・バインディング

Oracle クライアント・プロセスが Net8 リスナーを通じて Oracle サーバー・プロセスに接続されているとき（つまり、ネットワーク化した Oracle クライアント・サーバーを実行しているとき）Net8 リスナー・プロセスをバインドすることで、サーバー・プロセスを簡単にプロセッサとバインドできます。リスナーから発生する後続のすべての Oracle サーバー・プロセスは、同一プロセッサにバインドされます。

これを達成する 1 つの方法として、すべてのリスナーにそれぞれのポートを割り当て、複数のリスナーを起動します。これには、各リスナーが 1 連になるように、ファイル \$ORACLE\_HOME/network/admin/listener.ora をカスタマイズする必要があります。サーバー側で、複数のリスナーを起動します。クライアント側では、tnsnames.ora をカスタマイズし、クライアントだけでなくアプリケーションも、別々のリスナーが割り当てた異なるポートに接続します。たとえば、listener.ora を変更し、リスナー L1 および L2 にそれぞれポート 1521 および 1522 を割り当てるには、次のように入力します。

```
L1 =
  (ADDRESS_LIST =
    (ADDRESS= (PROTOCOL= TCP)(Host= nowhere)(Port= 1521))
  )
SID_LIST_L1 =
  (SID_LIST =
    (SID_DESC =
      (ORACLE_HOME= /oracle)
      (SID_NAME = ordb)
    )
  )
L2 =
  (ADDRESS_LIST =
    (ADDRESS= (PROTOCOL= TCP)(Host= nowhere)(Port= 1522))
  )
SID_LIST_L2 =
  (SID_LIST =
    (SID_DESC =
      (ORACLE_HOME= /oracle)
      (SID_NAME = ordb)
```

```
)  
)
```

次のように入力して 2 つのリスナーを起動します。

```
$ lsnrctl start L1  
$ lsnrctl start L2
```

2 つのリスナーのプロセス ID を取得します。

```
$ ps -ef|grep tnslsnr
```

これらのリスナーをプロセッサにバインドします。

```
$ bindprocessor process_id_for_L1 processor_id  
$ bindprocessor process_id_for_L2 processor_id
```

この場合、*processor\_id* は、次のとおりです。

```
processor_id = 0 ... number_of_CPUs - 1
```

特定のクライアント・アプリケーションの CPU 使用を制限する場合は、この方法でリスナーをバインドすれば、そのプロセッサが使用可能ときに常に同一プロセッサでアプリケーションを実行することができます。

## ローカル環境でのプロセッサ・バインディング

パイプ・ドライバを 2 タスクで使用し、クライアントおよび Oracle Server が同じマシン上で実行している場合は、プロセッサ・バインディングはさらに複雑です。これには、各サーバーのプロセス ID を見つけて手動でプロセッサとバインドする必要があります。管理用に過度なオーバーヘッドが生じ、サーバー・プロセスの寿命が長い場合以外は、一般的に努力する価値はありません。

Oracle プロセスをプロセッサとバインドすると、同じマシンで実行している Oracle 以外のアプリケーションのパフォーマンスが低下する場合があります。慎重なチューニングと監視が必要です。

## マルチ・プロセッサ・システムでのスピン・カウントのチューニング

リソースの取得を 1 回だけ試みた後に、プロセスをスリープさせるのでは、高価な操作になります。かわりに、プロセスが数回スピンしてもリソースを取得できなかった場合に、プロセスをスリープさせることができます。別のプロセスが保持しているラッチをまさに解放しようとしている場合には、この方針によってラッチの取得時のパフォーマンスが向上します。ただし、CPU のオーバーヘッドが大きくなる場合もあります。



Oracle8 for AIX で、`init.ora` ファイルのパラメータ `SPIN_COUNT` のデフォルト値は、シングル・プロセッサでは 0、マルチ・プロセッサでは 2000 です。これは大多数のアプリケーションにとって、適切な値です。`SPIN_COUNT` を増加すると、一般的に CPU 使用量が増加します。シングル・プロセッサ・マシンまたは CPU 使用量の多いマシンでは、`SPIN_COUNT` を 0（スピンなし）に設定するのが適切です。`SPIN_COUNT` の変更によって、パフォーマンスが向上することがありますが、値が大きすぎるとパフォーマンスが低下する可能性があります。アプリケーションをチューニングしてラッチの競合を減少させた方が、一般的により結果が得られます。

リリース 7.3 以降では、`DB_BLOCK_LRU_LATCHES` パラメータによってバッファの LRU リストで使用可能なラッチ数が決定します。このパラメータのデフォルト値はプロセッサ数の半分ですが、1 からシステムのプロセッサ数までの範囲で、値を設定できます。キャッシュ・バッファの LRU（Least Recently Used アルゴリズム）のラッチで競合が発生している場合は、このパラメータ値を大きくしてみてください。

## その他のチューニングのヒント

ここでは、『Oracle for UNIX Performance Tuning Tips』に記載されていない、その他のパフォーマンス・チューニングのヒントを説明します。

### RAW デバイスのバックアップ

RAW デバイスのバックアップには、`dd` コマンドを使用します。RAW デバイスの最初の 4KB ブロックは、AIX 論理ボリューム・マネージャが使用するので、上書きしないでください。4KB より大きいブロック・サイズでバックアップした方が早く終了します。テープへのバックアップに適したブロック・サイズは 256KB です。次に、RAW デバイスをテープにバックアップおよび復元するときに使用できるサンプル・スクリプトを示します。

1. RAW デバイスのバックアップをテープに作成します。

```
dd if=/dev/<raw_device> of=/dev/rmt0.1 bs=256k
```

2. RAW デバイスをテープから復元します。

```
dd if=/dev/rmt0.1 of=/dev/<raw_device> count=63 seek=1 skip=1 bs=4k
mt -f /dev/rmt0.1 bsf 1
dd if=/dev/rmt0.1 of=/dev/<raw_device> seek=1 skip=1 bs=256k
```

---

**注意：** HSD の場合、最初の 4KB はスキップしないでください。復元コマンドは次のように入力します。

```
dd if=/dev/remt0.1 of=/dev/<raw_device> bs=256K
```

---

## パラレル回復の使用方法

init.ora ファイルのパラメータ `RECOVERY_PARALLELISM = N` ( $N$  は 1 より大きい整数で、通常はマシンのプロセッサ数) を設定することで、さらにパフォーマンスは向上します。これによって、ログ・ファイルの変更をデータベースに適用するタスクを分担するために、複数の回復スレーブ・プロセスが生成されます。

## Pro\*C プログラムのコンパイル

AIX XLC コンパイラを使用すると、いくつかのコンパイラ・フラグによって、Pro\*C または Oracle OCI インタフェースで書かれた Oracle アプリケーションのパフォーマンスが向上する場合があります。たとえば、次のことを試みてください。

1. 最高の最適化レベル (-O3) を使用します。
2. アーキテクチャ固有のコンパイル・オプションを使用します。AIX XLC コンパイラがサポートするアーキテクチャ固有のコンパイル・オプションは次のとおりです。

`-qarch=com`

3 つの命令セットの共通サブセットのコンパイルです。このオプションでコンパイルしたプログラムは、3 つのアーキテクチャのすべてで正しく実行されます。

`-qarch=pwr`

元の RISC System/6000 の POWER アーキテクチャのコンパイルです。このオプションでコンパイルしたプログラムは、3 つのアーキテクチャのすべてで正しく実行されます。ただし、パフォーマンスを犠牲にして一部の命令が PowerPC でシミュレートされる場合があります。

`-qarch=pwrx`

特に、POWER2 アーキテクチャに対するコンパイルです。倍精度浮動小数点または浮動小数点平方根をよく使用するプログラムでは、パフォーマンスが向上する場合があります。この実行プログラムは、Power2 でだけ実行できます。

`-qarch=ppc`

特に PowerPC アーキテクチャに対するコンパイルです。単精度浮動小数点をよく使用するプログラムでは、パフォーマンスが向上する場合があります。この実行プログラムは、Power PC でのみ実行できます。

アプリケーションの性質に応じて、このように積極的に最適化するとパフォーマンスが向上する場合があります。これらのフラグの詳細は、コンパイラのドキュメントを参照してください。

## 関連ドキュメント

### Oracle ドキュメント

1. 『Oracle8 for AIX-Based Systems インストレーション・ガイド』
2. 『Oracle8 Server チューニング』
3. 『Oracle8 Server 管理者ガイド』
4. 『Oracle8 Server 概要 : Vol.1』
5. 『Oracle8 Server 概要 : Vol.2』
6. 『Oracle8 Server SQL リファレンス : Vol.1』
7. 『Oracle8 Server SQL リファレンス : Vol.2』
8. 『Oracle8 Server ユーティリティ』
9. 『Oracle8 Server アプリケーション開発者ガイド』

### IBM ドキュメント

1. 『AIX Versions 4.x Performance Tuning Guide』
2. 『AIX Performance Toolbox/6000 User's Guide 1.2 and 2.1』
3. 『AIX Version 4.x System Management Guide: Operating System and Devices』
4. 『AIX Version 4.x System Management Guide: Communication and Networks』
5. 『AIX Version 4.x Commands Reference』



---

# AIX での SQL\*Plus の管理

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

## SQL\*Plus の管理

### 設定ファイル

SQL\*Plus の設定ファイルには、サイトごとに定義するグローバルなファイル (glogin.sql) およびユーザーごとに定義するファイル (login.sql) があります。glogin.sql および login.sql ファイルには、SQL\*Plus セッションの最初に行われる SQL 文または SQL\*Plus コマンドが記述されています。SQL\*Plus を起動すると、最初に glogin.sql が読み込まれ、次に login.sql が読み込まれます。

### Site Profile

Site Profile ファイルは、\$ORACLE\_HOME/sqlplus/admin/glogin.sql です。SQL\*Plus は、ユーザーが SQL\*Plus を起動し、SQL\*Plus が Oracle に接続するときに、このコマンド・ファイルを実行します。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

User Profile ファイルは login.sql です。SQL\*Plus は、ユーザーが SQL\*Plus を起動し、SQL\*Plus が Oracle に接続するときに、このコマンド・ファイルを実行します。User Profile は、Site Profile の後で実行されます。SQL\*Plus は、起動されたディレクトリから User Profile を検索します。環境変数 SQLPATH にディレクトリ・パスをコロンで区切って設定すると、SQL\*Plus はその順番で User Profile を検索します。

たとえば、現在のディレクトリが /u02/oracle で、次のような SQLPATH を設定しているとします。

```
% echo $SQLPATH
/home:/home/oracle:/u01/oracle
```

SQL\*Plus は、最初に現在のディレクトリ /u02/oracle で login.sql を検索します。現在のディレクトリに login.sql がない場合、SQL\*Plus は /home、/home/oracle および /u01/oracle ディレクトリを検索します。

次に、login.sql ファイルのサンプルを示します。

```
set echo off
set feedback 4
set pause on
set pause "PLEASE PRESS RETURN TO CONTINUE"
set message on
set echo on
```

## PRODUCT\_USER\_PROFILE 表

SYSTEM ユーザーで \$ORACLE\_HOME/sqlplus/admin/pupbld.sql を実行し、Product 表および User Profile 表を作成します。

\$ORACLE\_HOME/sqlplus/admin/pupbld.sql は、シェル・スクリプト \$ORACLE\_HOME/bin/pupbld を使用して実行することもできます。このスクリプトを使用するには、環境変数 ORACLE\_HOME および SYSTEM\_PASS を設定する必要があります。SYSTEM\_PASS には、SYSTEM ユーザーのユーザー名およびパスワードを設定します。次に例を示します。

```
% setenv SYSTEM_PASS SYSTEM/manager
% pupbld

Installing product user profile tables...

Product user profile tables installed.
```

pupbld.sql は、SQL\*Plus のインストール時に「Create Database Objects」を選択した場合のみ、Installer によって実行されます。

## デモンストレーション表

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。

### デフォルト・インストール

「Default Install」によって「Create Database Objects」を選択すると、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。

### カスタム・インストール

「Custom Install」によって SQL\*Plus をインストールする場合、「Create Database Objects」を選択し、「Would you like to load the SQL\*Plus Demo Tables?」というプロンプトに対し「Yes」と答えます。Installer はユーザー SCOTT (パスワードは TIGER) およびデモンストレーション表を作成します。

## 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 `demobld.sql` ファイルを実行すると、ユーザーはスキーマにデモンストレーション表を作成することができます。次に例を示します。

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

次のようにシェル・スクリプト `$ORACLE_HOME/bin/demobld` を使用することによって、`$ORACLE_HOME/sqlplus/demo/demobld.sql` を実行できます。

```
% 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/demodrop.sql` を実行します。SQL\*Plus で `demodrop.sql` ファイルを実行すると、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。次に例を示します。

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

次のようにシェル・スクリプト `$ORACLE_HOME/bin/demodrop` を使用することによって、`$ORACLE_HOME/sqlplus/demo/demodrop.sql` を実行できます。

```
% 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。これらと同じ名前の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

### デフォルト・インストール

「Default Install」によって「Create Database Objects」を選択すると、ヘルプ機能が自動的にインストールされます。



## カスタム・インストール

SQL\*Plus のインストール時に、「Create Database Objects」を選択し、「Would you like to load the SQL\*Plus Help Facility?」というプロンプトに対し「Yes」と答えます。Installer はヘルプ機能を作成します。

## ヘルプ機能の手動インストール

手動でヘルプ機能をインストールするには、シェル・スクリプト

\$ORACLE\_HOME/bin/helpins を使用します。このスクリプトを使用するには、環境変数 ORACLE\_HOME および SYSTEM\_PASS を設定する必要があります。SYSTEM\_PASS には、SYSTEM ユーザーのユーザー名およびパスワードを設定します。次に例を示します。

```
$ setenv SYSTEM_PASS SYSTEM/manager
$ helpins
```

```
SQL*Loader: Release 8.0.6.0.0 - Production
```

```
(c) Copyright 1999 Oracle Corporation. All rights reserved.
```

```
Commit point reached - logical record count 828
```

```
SQL*Loader: Release 8.0.6.0.0 - Production
```

```
(c) Copyright 1999 Oracle Corporation. All rights reserved.
```

```
Commit point reached - logical record count 1024
```

```
Commit point reached - logical record count 1207
```

```
SQL*Loader: Release 8.0.6.0.0 - Production
```

```
(c) Copyright 1999 Oracle Corporation. All rights reserved.
```

```
Commit point reached - logical record count 1024
```

```
Commit point reached - logical record count 1304
```

```
Commit point reached - logical record count 2328
```

```
Commit point reached - logical record count 2724
```

```
Commit point reached - logical record count 2835
```

**参照：** 詳細は、『SQL\*Plus ユーザーズ・ガイドおよびリファレンス』および README ファイル \$ORACLE\_HOME/sqlplus/doc/release.doc を参照してください。

## SQL\*Plus の使用

### 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` などのデフォルト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。

グローバルなデフォルト・エディタは、SQL\*Plus の `_editor` オプションを使用して、DBA が `glogin.sql` に設定します。`login.sql` にエディタを指定して、この設定を上書きします。SQL\*Plus は、起動時に両方のファイルを読み込みますが、ローカル・ファイル (`login.sql`) が優先されます。また、SQL\*Plus セッション中に `_editor` オプションを設定し、両方のファイルの設定を上書きすることもできます。

`_editor` オプションが設定されていない場合、環境変数 `EDITOR` および `VISUAL` によって SQL\*Plus エディタが指定されます。これらの環境変数は、`glogin.sql` または `login.sql` では設定できません。ユーザー起動ファイルまたはシステム・プロンプトで設定します。環境変数 `EDITOR` および `VISUAL` が両方設定されている場合、環境変数 `EDITOR` が優先されます。

### エディタの順序設定

SQL\*Plus は、次の順序でデフォルトのエディタを検索します。

1. SQL\*Plus セッション中の `_editor` 変数
2. `login.sql` の `_editor` 変数
3. `glogin.sql` の `_editor` 変数
4. 環境変数 `EDITOR`
5. 環境変数 `VISUAL`

これらの値がどれも設定されていない場合、SQL\*Plus はエディタに `ed` を使用します。

### `_editor` オプションの設定

SQL\*Plus の `_editor` オプションを設定するには、`login.sql` ファイルに次の行を追加します。

```
define _editor=editor_name
```

この場合、`editor_name` は UNIX エディタを表します。

## 環境変数の設定

Bourne シェルまたは Korn シェルの場合は、次のように入力して環境変数にデフォルト・エディタを設定します。

```
$ UNIX_VAR=editor_name; export UNIX_VAR
```

C シェルの場合は、次のように入力して環境変数にデフォルト・エディタを設定します。

```
% setenv UNIX_VAR editor_name
```

表 4-1 に、環境変数の構文の詳細を示します。

**表 4-1 UNIX 環境変数の構文**

|                          |                          |
|--------------------------|--------------------------|
| <code>UNIX_VAR</code>    | 環境変数 EDITOR または VISUAL   |
| <code>editor_name</code> | UNIX エディタ (たとえば、vi や ed) |

## デフォルト設定

システム・エディタを起動すると、現行の SQL バッファは編集バッファに入れられるため、エディタで利用できる文はすべて SQL 文を変更できます。SQL\*Plus はテンポラリ・ファイル `afiedt.buf` を使用します。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 からのオペレーティング・システム・コマンドの実行

SQL\*Plus プロンプトの後の最初の文字を感嘆符 (!) にすると、後続の文字列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。

次の SQL\*Plus コマンドを使用すると、それぞれに特定のタスクを実行できます。

- `[!]+command` (1 つのオペレーティング・システム・コマンドを実行する場合)  
コマンドの実行後、制御は SQL\*Plus に戻ります。
- `[!]+[Return]` (2 つ以上のオペレーティング・システム・コマンドを実行する場合)  
終了後に `[Ctrl]+[d]` を押すと、制御は SQL\*Plus に戻ります。

## SQL\*Plus への割込み

SQL\*Plus の実行中に、次のような割込みができます。

- BSD マシンの場合は [Ctrl]+[c] を、また、System V マシンの場合は [Delete] 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。
- SQL\*Plus プロンプトで [Interrupt] を押すと、別の SQL\*Plus プロンプトが表示されません。

## SPOOL コマンドの使用

SPOOL コマンドで生成されるファイルの拡張子のデフォルトは、.lst 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。

次に例を示します。

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

## 制限事項

### COPY コマンド

SQL\*Plus の COPY コマンドは、同じバージョンのオペレーティング・システムが実行されているマシン上で使用できます。制限事項はありません。

COPY コマンドは、異なるバージョンのオペレーティング・システムが実行されているマシン間でも動作します。COPY コマンドが失敗した場合は、rcp または ftp を使用して接続をテストします。ベンダーが提供するネットワーク・ソフトウェアによっては、rcp、ftp または COPY コマンドがシステム間で正常に動作しない場合があります。

---

**注意：** rlogin コマンドは、大きなデータ・パケットを送信または受信しないので、接続のテストに適していません。

---

COPY コマンドがシステム間で動作しない場合、システムおよびユーザー ID へのデータベース・リンクを作成して、コピーする表を指定します。そのためには、次のように入力します。

```
SQL> create table newtable as (SELECT * FROM table@database_link_name)
```

これによって、リモート・システムにあるコピー元の表で選択された行および列が、ローカル・システム上のコピー先の表に挿入されます。

## ウィンドウのサイズ変更

SQL\*Plus の LINESIZE のデフォルト値は 80、PAGESIZE のデフォルト値は 25 です。これらの環境変数によって、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。



---

# AIX での 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
- Pro\*C/C++
- Pro\*COBOL
- Pro\*FORTRAN
- SQL\*Module for Ada
- Oracle コール・インタフェース
- Oracle プリコンパイラと Oracle コール・インタフェースのリンクおよび Make ファイル
- スレッドのサポート
- Oracle ライブラリを使用した静的リンクおよび動的リンク
- シグナル・ハンドラの使用法
- 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、Oracle8 やその他の ANSI SQL DBMS 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。

## プリコンパイラ実行ファイルの再リンク

すべてのプリコンパイラ実行ファイルは、Make ファイル `$ORACLE_HOME/precomp/lib/ins_precomp.mk` を使用して再リンクされます。make コマンドの形式は次のとおりです。

```
$ make -f ins_precomp.mk relink EXENAME=executable
```

このコマンドを実行すると、新しい実行ファイルが `$ORACLE_HOME/precomp/lib` ディレクトリに作成され、その後、その実行ファイルは `$ORACLE_HOME/bin` ディレクトリに移されます。`$ORACLE_HOME/bin` に移さずに新しい実行ファイルを作成するには、次のコマンドを使用します。

```
$ make -f ins_precomp.mk executable
```

表 5-1 に、使用する製品に対する実行ファイル名を示します。

表 5-1 製品と実行ファイル名

| 製品                        | 実行ファイル              |
|---------------------------|---------------------|
| Pro*C/C++                 | proc                |
| Pro*COBOL 1.8.28          | procob18 または rtsora |
| Pro*COBOL 8.0.6           | procob または rtsora   |
| Pro*FORTRAN               | profor              |
| Oracle SQL*Module for Ada | modada              |
| Object Type Translator    | ott                 |

たとえば、Pro\*C/C++ の実行ファイルを再リンクするには、次のように入力します。

```
$ cd $ORACLE_HOME/precomp/lib
$ make -f ins_precomp.mk relink EXENAME=proc
```



## プリコンパイラ構成ファイル

`$ORACLE_HOME/precomp/admin` には、6 つの `.cfg` システム構成ファイルがあります。  
表 5-2 に、製品とその構成ファイルを示します。

表 5-2 システム構成ファイル

| 製品                              | 構成ファイル                  |
|---------------------------------|-------------------------|
| Pro*C/C++ 8.0.6                 | <code>pcscfg.cfg</code> |
| Pro*COBOL 8.0.6                 | <code>pcbcfg.cfg</code> |
| Pro*COBOL 1.8.28                | <code>pcccob.cfg</code> |
| Pro*FORTRAN 1.8.28              | <code>pccfor.cfg</code> |
| Oracle SQL*Module for Ada 8.0.6 | <code>pmscfg.cfg</code> |
| Object Type Translator 8.0.6    | <code>ottcfg.cfg</code> |

## 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」エラーが発生する場合があります。この場合、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グ・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。

## 参照ドキュメント

次のドキュメントでは、プリコンパイラおよびインタフェース機能についての追加情報を説明しています。

- 『Oracle Pro\*C/C++ プリコンパイラ・プログラマーズ・ガイド』
- 『Oracle Pro\*COBOL プリコンパイラ・プログラマーズ・ガイド』
- 『Oracle プリコンパイラ・ガイド Pro\*FORTRAN サプリメント』
- 『Oracle8 コール・インタフェース・プログラマーズ・ガイド』
- 『Oracle8 Server アプリケーション開発者ガイド』
- 『Programmer's Guide to SQL\*Module for Ada』

## Pro\*C/C++

Pro\*C/C++ リリース 8.0.6 の追加情報は、README ファイル  
\$ORACLE\_HOME/precomp/doc/proc2/readme.doc を参照してください。

## Pro\*C/C++ の管理

### システム構成ファイル

Pro\*C/C++ のシステム構成ファイルは、\$ORACLE\_HOME/precomp/admin/pcscfg.cfg  
です。

## Pro\*C/C++ の使用

Pro\*C/C++ を使用する前に、適切なバージョンの AIX コンパイラがインストールされていることを確認してください。必要なバージョンについては、『Oracle8 for AIX-Based Systems インストレーション・ガイド』の第 1 章を参照してください。

### デモンストレーション・プログラム

Pro\*C/C++ プリコンパイラのさまざまな機能を参照するために、デモンストレーション・プログラムが提供されています。C、C++ および Object プログラムの 3 種類のデモンストレーション・プログラムがあります。Object プログラムのデモンストレーション・プログラムでは、Oracle8 Object の新機能が表示されます。すべてのデモンストレーション・プログラムは \$ORACLE\_HOME/precomp/demo/proc にあります。また、すべてのデモンストレーション・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。

SQL\*Plus を使用してデモンストレーション・プログラムを作成する場合の詳細は、このマニュアルの [4-3 ページの「デモンストレーション表」](#) を参照してください。また、デモンストレーション・プログラムの詳細は、『Oracle 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。

Make ファイル \$ORACLE\_HOME/precomp/demo/proc/demo\_proc.mk は、デモンストレーション・プログラムの作成に必要です。たとえば、sample1 というデモンストレーション・プログラムをプリコンパイル、コンパイルまたはリンクする場合は、次のように入力します。

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

また、次の例は構文が明示的になっているだけで、同様の結果になります。

```
$ make -f demo_proc.mk build OBJS=sample1.o EXE=sample1
```

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.a に動的にリンクされます。Make ファイルでは、環境変数 ORA\_CLIENT\_LIB が使用されなくなりました。

すべての Pro\*C/C++ の C デモンストレーション・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
```

すべての Pro\*C/C++ の C++ デモンストレーション・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
```

すべての Pro\*C/C++ の Object デモンストレーション・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
```

デモンストレーション・プログラムによっては、実行する際に \$ORACLE\_HOME/precomp/demo/sql にある SQL スクリプトが必要な場合があります。このようなデモンストレーション・プログラムを作成し、SQL スクリプトを実行するには、コマンド行に 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 デモンストレーション・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
```

また、Object デモンストレーション・プログラムを作成し、必要な SQL スクリプトを実行するには、次のように入力します。

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

必要に応じて SQL スクリプトを手動で実行することもできます。

ユーザー・プログラム

Make ファイル \$ORACLE\_HOME/precomp/demo/proc/demo\_proc.mk は、ユーザー・プログラムを作成する場合に使用します。ユーザー・プログラムを demo\_proc.mk にリンクする場合の構文は、次のとおりです。

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

たとえば、Pro\*C/C++ のソース myprog.pc からプログラム myprog を作成する場合、ソースおよび実行ファイルの形式によって、次のいずれかを入力します。

C ソースの場合で、クライアント共有ライブラリに動的にリンクする場合

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

C ソースの場合で、静的にリンクする場合

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

C++ ソースの場合で、クライアント共有ライブラリに動的にリンクする場合

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

C++ ソースの場合で、静的にリンクする場合

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

Pro\*COBOL

今回のリリースの Pro\*COBOL には、次の 2 つのバージョンがあります。

- Pro\*COBOL 8.0.6
- Pro\*COBOL 1.8.28

表 5-3 に、この 2 つバージョンのネーミング方法の違いを示します。

表 5-3 Pro\*COBOL のネーミング方法の違い

|                                   | Pro*COBOL 8.0.6 | Pro*COBOL 1.8.28 |
|-----------------------------------|-----------------|------------------|
| 実行ファイル                            | procob          | procob18         |
| デモ・ディレクトリ                         | procob2         | procob           |
| Make ファイル<br>( MicroFocus COBOL ) | demo_procob.mk  | demo_procob18.mk |

Pro\*COBOL では、静的リンク、動的リンク、または動的読み込みプログラムをサポートしています。動的リンク・プログラムでは、Oracle クライアント共有ライブラリ `$ORACLE_HOME/lib/libclntsh.a` が使用されます。動的読み込みプログラムでは、`rtsora` 実行ファイルが使用されます。

Pro\*COBOL 8.0.6 の追加情報は、README ファイル  
`$ORACLE_HOME/precomp/doc/procob2/readme.doc` を参照してください。  
Pro\*COBOL 1.8.28 の追加情報は、README ファイル  
`$ORACLE_HOME/precomp/doc/prolx/readme.txt` を参照してください。

## Pro\*COBOL の管理

### システム構成ファイル

Pro\*COBOL 8.0.6 のシステム構成ファイルは、  
`$ORACLE_HOME/precomp/admin/pcbcfg.cfg` です。

Pro\*COBOL 1.8.28 のシステム構成ファイルは、  
`$ORACLE_HOME/precomp/admin/pcccob.cfg` です。

## 環境変数

### MicroFocus COBOL コンパイラ

MicroFocus COBOL コンパイラには、環境変数 `COBDIR` および `LIBPATH` を設定する必要があります。`COBDIR` には、コンパイラがインストールされているディレクトリを設定します。次に例を示します。

```
$ setenv COBDIR /usr/lpp/mfcobol
```

`LIBPATH` には、`$COBDIR/coblib` ディレクトリを設定する必要があります。たとえば、次のように入力して `$COBDIR/coblib` を `LIBPATH` に追加します。

```
$ setenv LIBPATH ${LIBPATH}:%COBDIR/coblib
```

`LIBPATH` に `$COBDIR/coblib` が設定されていないと、プログラムをコンパイルしたときに次のエラーが発生します。

```
System error - error data is: rts32
Cannot load library libfhutil.2.0.a[shr.o]
System error: A file or directory in the pathname does not exist
```

## Pro\*COBOL の使用

Pro\*COBOL を使用する前に、適切なバージョンの COBOL コンパイラがインストールされていることを確認してください。必要なバージョンについては、『Oracle8 for AIX-Based Systems インストレーション・ガイド』の第 1 章を参照してください。

### Pro\*COBOL 実行ファイルのリンク

Pro\*COBOL 実行ファイルをリンクする場合は、未定義のシンボルを解決するために、\$COBDIR/coblib/liblist ファイルに指定されたオブジェクト・ファイルの 1 つを変更する必要があります。\$COBDIR/coblib/liblist ファイルで、次の行を置き換えます。

```
i/lib/crt0.o  
s-lc
```

これを次のように置き換えます。

```
i/lib/crt0_r.o  
s-lc_r
```

### Oracle ランタイム・システム

Oracle では、動的読込み Pro\*COBOL プログラムを実行するために、rtsora という専用のランタイム・システムが提供されています。rtsora ランタイム・システムは、動的読込み Pro\*COBOL プログラムを実行する際に、MicroFocus の cobrun ランタイム・システムのかわりに使用します。cobrun で Pro\*COBOL プログラムを実行すると、次のエラーが発生します。

```
$ cobrun sample1.gnt  
Load error : file 'SQLADR'  
error code: 173, pc=0, call=1, seg=0  
173      Called program file not found in drive/directory
```

### デモンストレーション・プログラム

デモンストレーション・プログラムは、Pro\*COBOL プリコンパイラのさまざまな機能を参照するために提供されています。すべてのプログラムは、Pro\*COBOL のバージョンによって、\$ORACLE\_HOME/precomp/demo/procob または \$ORACLE\_HOME/precomp/demo/procob2 に保存されています。すべてのプログラムでは、\$ORACLE\_HOME/sqlplus/demo/demobld.sql で作成したデモンストレーション表は 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。

SQL\*Plus を使用してデモンストレーション・プログラムを作成する場合の詳細は、このマニュアルの [4.3 ページの「デモンストレーション表」](#)を参照してください。また、デモンストレーション・プログラムの詳細は、『Oracle Pro\*COBOL プリコンパイラ・プログラマーズ・ガイド』を参照してください。

デモンストレーション Make ファイルは、サンプル・プログラムを作成する場合に必要です。Pro\*COBOL 8.0.6 のデモンストレーション Make ファイルは、`$ORACLE_HOME/precomp/demo/procob2/demo_procob.mk` です。Pro\*COBOL 1.8.28 のデモンストレーション Make ファイルは、`$ORACLE_HOME/precomp/demo/procob/demo_procob18.mk` です。たとえば、Pro\*COBOL 8.0.6 のデモンストレーション・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には、次のコマンドを使用します。

```
$ cd $ORACLE_HOME/precomp/demo/procob2
$ make -f demo_procob.mk sample1
```

また、次の例は構文が明示的になっているだけで、同様の結果になります。

```
$ make -f demo_procob.mk build COBS=sample1.cob EXE=sample1
```

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.a` に動的にリンクされます。

すべての Pro\*COBOL デモンストレーション・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_procob.mk samples
```

`rtsora` で使用する動的読み込みプログラム `sample1.gnt` を作成するには、次のように入力します。

```
$ make -f demo_procob.mk sample1.gnt
```

その後、次のように入力して、`rtsora` でプログラムを実行します。

```
$ rtsora sample1.gnt
```

デモンストレーション・プログラムによっては、実行する際に `$ORACLE_HOME/precomp/demo/sql` にある SQL スクリプトが必要な場合があります。このようなデモンストレーション・プログラムを作成し、SQL スクリプトを実行するには、コマンド行に `make` マクロ引数 `RUNSQL=run` を追加する必要があります。

たとえば、`sample9` デモンストレーション・プログラムを作成し、必要な `$ORACLE_HOME/precomp/demo/sql/sample9.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_procob.mk sample9 RUNSQL=run
```

必要に応じて SQL スクリプトを手動で実行することもできます。

## ユーザー・プログラム

デモンストレーション Make ファイルは、ユーザー・プログラムを作成する場合に使用します。ご使用の Pro\*COBOL のバージョンおよび COBOL コンパイラに応じて、適切な Make ファイルを使用してください。ユーザー・プログラムをデモンストレーション Make ファイルにリンクする構文は、次のとおりです。

```
$ make -f demo_procob.mk target COBS="cobfile1 cobfile2 ..." \  
EXE=exename
```

たとえば、Pro\*COBOL のソース `myprog.pco` からプログラム `myprog` を作成する場合、実行ファイルの形式および共有ライブラリ使用方法によって、次のいずれかのコマンドを実行します。

クライアント共有ライブラリに動的にリンクする実行ファイルの場合

```
$ make -f demo_procob.mk build COBS=myprog.cob EXE=myprog
```

クライアント共有ライブラリを使用せずに静的にリンクする実行ファイルの場合

```
$ make -f demo_procob.mk build_static COBS=myprog.cob EXE=myprog
```

`rtsora` で使用できる動的読み込みモジュールの場合

```
$ make -f demo_procob.mk myprog.gnt
```

## FORMAT プリコンパイラ

FORMAT プリコンパイラ・オプションは、COBOL の入力行の形式を指定します。

FORMAT=ANSI（デフォルト）を指定した場合、カラム 1 ~ 6 はオプションの順序番号、カラム 7 はコメントまたは継続行を示す標識です。さらに、カラム 8 ~ 11 は段落の名前で、カラム 12 ~ 72 が文となります。

FORMAT=TERMINAL を指定した場合、カラム 1 ~ 6 は削除され、カラム 7 が左端のカラムになります。



## Pro\*FORTRAN

Pro\*FORTRAN 1.8.28 の追加情報は、README ファイル  
`$ORACLE_HOME/precomp/doc/pro1x/readme.txt` を参照してください。

## Pro\*FORTRAN の管理

### システム構成ファイル

Pro\*FORTRAN のシステム構成ファイルは、  
`$ORACLE_HOME/precomp/admin/pccfor.cfg` です。

## Pro\*FORTRAN の使用

Pro\*FORTRAN を使用する前に、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。必要なバージョンについては、『Oracle8 for AIX-Based Systems インストレーション・ガイド』の第 1 章を参照してください。

### デモンストレーション・プログラム

Pro\*FORTRAN プリコンパイラのさまざまな機能を参照するために、デモンストレーション・プログラムが提供されています。すべてのプログラムは、  
`$ORACLE_HOME/precomp/demo/profor` にあります。また、  
`$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。

SQL\*Plus を使用してデモンストレーション・プログラムを作成する場合の詳細は、このマニュアルの [4-3 ページの「デモンストレーション表」](#) を参照してください。また、デモンストレーション・プログラムの詳細は、『Oracle プリコンパイラ・ガイド Pro\*FORTRAN サプリメント』を参照してください。

Make ファイル `$ORACLE_HOME/precomp/demo/profor/demo_profor.mk` は、デモンストレーション・プログラムの作成に必要です。たとえば、`sample1` というデモンストレーション・プログラムをプリコンパイル、コンパイルまたはリンクする場合は、次のように入力します。

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

また、次の例は構文が明示的になっているだけで、同様の結果になります。

```
$ make -f demo_profor.mk build FORS=sample1.pfo EXE=sample1
```

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.a` に動的にリンクされます。

すべての Pro\*FORTRAN デモンストレーション・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_profor.mk samples
```

デモンストレーション・プログラムによっては、実行する際に

\$ORACLE\_HOME/precomp/demo/sql にある SQL スクリプトが必要な場合があります。このようなデモンストレーション・プログラムを作成し、SQL スクリプトを実行するには、コマンド行に make マクロ引数 RUNSQL=run を追加する必要があります。たとえば、sample11 デモンストレーション・プログラムを作成し、必要な \$ORACLE\_HOME/precomp/demo/sql/sample11.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_profor.mk sample11 RUNSQL=run
```

必要に応じて SQL スクリプトを手動で実行することもできます。

## ユーザー・プログラム

Make ファイル \$ORACLE\_HOME/precomp/demo/profor/demo\_profor.mk は、ユーザー・プログラムを作成する場合に使用します。ユーザー・プログラムを demo\_profor.mk にリンクする場合の構文は、次のとおりです。

```
$ make -f demo_profor.mk target FORS="forfile1 forfile2 ..." \
EXE=exename
```

たとえば、Pro\*FORTRAN のソース myprog.pfo からプログラム myprog を作成する場合、実行ファイルの形式によって、次のいずれかを実行します。

クライアント共有ライブラリに動的にリンクする実行ファイルの場合

```
$ make -f demo_profor.mk build FORS=myprog.f EXE=myprog
```

静的にリンクする実行ファイルの場合

```
$ make -f demo_profor.mk build_static FORS=myprog.f EXE=myprog
```

# SQL\*Module for Ada

## SQL\*Module for Ada の管理

### システム構成ファイル

Oracle SQL\*Module のシステム構成ファイルは、  
\$ORACLE\_HOME/precomp/admin/pmscfg.cfg です。

## SQL\*Module for Ada の使用

SQL\*Module for Ada を使用する前に、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。必要なバージョンについては、『Oracle8 for AIX-Based Systems インストレーション・ガイド』の第 1 章を参照してください。

### デモンストレーション・プログラム

デモンストレーション・プログラムは、SQL\*Module for Ada のさまざまな機能を参照するために提供されています。すべてのプログラムは、  
\$ORACLE\_HOME/precomp/demo/modada に格納されています。

デモンストレーション・プログラム chl\_drv 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。

デモンストレーション・プログラム demcalsp および demohost では、  
\$ORACLE\_HOME/precomp/demo/sql/mktables.sql で作成した大学のサンプル・データベースが MODTEST スキーマにあると仮定しています。

すべてのプログラムでは、Net8 の接続文字列またはインスタンスの別名 INST1\_ALIAS が定義されていて、適切な表が存在するデータベースに接続できると仮定しています。

SQL\*Plus を使用してデモンストレーション・プログラムを作成する場合の詳細は、このマニュアルの [4-3 ページの「デモンストレーション表」](#) を参照してください。また、デモンストレーション・プログラムの詳細は、『Programmer's Guide to SQL\*Module for Ada』を参照してください。

Make ファイル \$ORACLE\_HOME/precomp/demo/modada/demo\_modada.mk は、デモンストレーション・プログラムの作成に必要です。たとえば、chl\_drv というデモンストレーション・プログラム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_modada.mk chl_drv
```

また、次の例は構文が明示的になっているだけで、同様の結果になります。

```
$ make -f demo_modada.mk ada OBJS="ch1_mod.ada ch1_drv.ada" \
    EXE=ch1_drv MODARAGS=user=modtest/yes
```

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.a に動的にリンクされます。

すべての SQL\*Module for Ada デモンストレーション・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_modada.mk samples
```

サンプル・プログラム demcalsp および demohost では、MODTEST スキーマの  
\$ORACLE\_HOME/precomp/demo/sql/mktables.sql で作成した大学のサンプル・デー  
タベースが必要です。MODTEST ユーザーは、SQL スクリプト  
\$ORACLE\_HOME/precomp/demo/sql/grant.sql を実行して作成することができます。  
MODTEST ユーザーおよび大学のサンプル・データベースを作成し、デモンストレーショ  
ン・プログラムを作成する場合、作成ターゲットとして makeuser および loaddb を使用  
します。たとえば、必要な SQL スクリプトを実行して demohost プログラムを作成するに  
は、次の構文のコマンドを実行します。

```
$ make -f demo_modada.mk makeuser loaddb demohost
```

必要に応じて SQL スクリプトを手動で実行することもできます。

すべての SQL\*Module for Ada デモンストレーション・プログラムを作成し、必要な SQL  
スクリプトを実行して MODTEST ユーザーを作成し、大学のサンプル・データベースを作  
成する場合、次のように入力します。

```
$ make -f demo_modada.mk all
```

## ユーザー・プログラム

Make ファイル \$ORACLE\_HOME/precomp/demo/modada/demo\_modada.mk は、ユー  
ザー・プログラムを作成する場合に使用します。ユーザー・プログラムを  
demo\_modada.mk にリンクする構文は、次のとおりです。

```
$ make -f demo_modada.mk ada OBJS="module1 module2 ..." \
    EXE=exename MODARAGS=SQL*Module_arguments
```

# Oracle コール・インタフェース

## Oracle コール・インタフェースの使用

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。必要なバージョンについては、『Oracle8 for AIX-Based Systems インストレーション・ガイド』の第 1 章を参照してください。

### デモンストレーション・プログラム

デモンストレーション・プログラムは、OCI のさまざまな機能を参照するために提供されています。デモンストレーション・プログラムには、C および C++ の 2 種類があります。すべてのデモンストレーション・プログラムは、`$ORACLE_HOME/rdbms/demo` にあります。ほとんどのデモンストレーション・プログラムでは、`$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。

SQL\*Plus を使用してデモンストレーション・プログラムを作成する場合の詳細は、このマニュアルの [4-3 ページの「デモンストレーション表」](#) を参照してください。また、デモンストレーション・プログラムの詳細は『Oracle コール・インタフェース・プログラマーズ・ガイド』を、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。

Make ファイル `$ORACLE_HOME/rdbms/demo/demo_rdbms.mk` は、デモンストレーション・プログラムを作成する場合に必要です。たとえば、`cdemo1` というデモンストレーション・プログラム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
```

また、次の例は構文が明示的になっているだけで、同様の結果になります。

```
$ make -f demo_rdbms.mk build OBJS=cdemo1.o EXE=cdemo1
```

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.a` に動的にリンクされます。

すべての OCI の C デモンストレーション・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
```

すべての OCI の C++ デモンストレーション・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
```

デモンストレーション・プログラムによっては、プログラムを実行する前に、`$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。

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。たとえば、プログラム `oci02` の SQL スクリプトは `oci02.sql` です。

プログラムの先頭にコメントがある場合は、そのコメントから必要な SQL スクリプトを判断します。

## ユーザー・プログラム

Make ファイル `$ORACLE_HOME/rdbms/demo/demo_rdbms.mk` は、ユーザー・プログラムを作成する場合に使用します。ユーザー・プログラムを `demo_rdbms.mk` にリンクする構文は、次のとおりです。

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

たとえば、C のソース `myprog.c` からプログラム `myprog` を作成する場合、実行ファイルの形式によって、次のいずれかを実行します。

C ソースの場合で、クライアント共有ライブラリに動的にリンクする場合

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

C ソースの場合で、静的にリンクする場合

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

C++ のソース `myprog.cc` からプログラム `myprog` を作成する場合は、次のいずれかのコマンドを実行します。

C++ ソースの場合で、クライアント共有ライブラリに動的にリンクする場合

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

C++ ソースの場合で、静的にリンクする場合

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

# Oracle プリコンパイラと Oracle コール・インタフェースのリンクおよび Make ファイル

## カスタム Make ファイル

この章で説明している各製品には、オラクル社から提供されている 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は、次のことに注意してください。

- Oracle ライブラリの順番は変更しないでください。
- リンク行に独自のライブラリを追加する場合は、リンク行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 ライブラリの間に置かないでください。
- `nm` または `GNU make` 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、AIX で提供されている `make` ユーティリティ (`/usr/ccs/bin/make`) との違いに注意してください。Oracle の Make ファイルは、すでにテスト済みで、AIX 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、現在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

プログラムのリンク時に共通するエラーに、次のような未定義シンボルがあります。

```
$ make -f demo_proc.mk sample1
ld: 0711-317 ERROR: Undefined symbol: .sqlcex
ld: 0711-317 ERROR: Undefined symbol: .sqlglm
ld: 0711-345 Use the -bloadmap or -bnoquiet option to obtain more information.
```

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。通常、このような問題を解決するには、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・ファイルがリンク行にあるかどうか、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。

オラクル社が提供している `symfind` という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のに役立ちます。次に、シンボル `sqlcex` が定義されている場所を調べる `symfind` の出力例を示します。

```
$ symfind sqlcex
```

```
SymFind - Find Symbol <sqlcex> in <*>.a, .o, .so
```

```
-----
```

```
Command:           /u01/app/oracle/product/8.0.6/bin/symfind sqlcex  
Local Directory:   /u01/app/oracle/product/8.0.6  
Output File:       (none)  
Note:              I do not traverse symbolic links  
                   Use '-v' option to show any symbolic links
```

```
Locating Archive and Object files ...  
[11645] |      467572|      44|FUNC |GLOB |0      |8      |sqlcex  
^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^  
                                              ./lib/libclntsh.a  
[35]    |          0|      44|FUNC |GLOB |0      |5      |sqlcex  
^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^  
                                              ./lib/libsql.a
```

## スレッドのサポート

今回のリリースで提供されている Oracle ライブラリはスレッド・セーフで、マルチスレッド・アプリケーションをサポートできます。AIX 4.3 に関連した制限および詳細は、『Oracle8 for AIX-Based Systems インストレーション・ガイド』の第 1 章の「スレッド・セーフ・ライブラリ」を参照してください。

## Oracle ライブラリを使用した静的リンクおよび動的リンク

プリコンパイラおよび OCI アプリケーションは、静的および動的に 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、ある部分は実行プログラムに存在し、また別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も存在します。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呼びます。動的リンクには、主に次の2つ利点があります。

1. 必要なディスク領域が少なくて済む  
異なるアプリケーション、または同一アプリケーションからの異なる呼出しであっても、同一の共有ライブラリまたは動的ライブラリを使用できます。その結果、必要なディスク領域を削減できます。
2. 必要なメイン・メモリーが少なくて済む  
異なるアプリケーションで、同一の共有ライブラリ・イメージまたは動的ライブラリ・イメージ（メモリー内コピー）を共有できます。つまり、1つのライブラリをメイン・メモリーに1回だけ読み込めば、複数のアプリケーションでそのライブラリを使用できます。その結果、必要なメイン・メモリーを削減できます。



## Oracle 共有ライブラリ

Oracle 共有ライブラリは、`$ORACLE_HOME/lib/libclntsh.a` 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。

環境変数 `LIBPATH` は、プロセスの起動時にランタイム・ローダーが Oracle 共有ライブラリを検索できるように設定する必要があります。実行ファイルの起動時に次のエラーが発生する場合は、`LIBPATH` に Oracle 共有ライブラリがあるディレクトリを設定してください。

```
% sample1
exec(): 0509-036 Cannot load program ./sample1 because of the following
errors:
0509-022 Cannot load library libclntsh.a[shr.o].
0509-026 System error: A file or directory in the path name does not exist.
```

`LIBPATH` を次のように設定します。

```
% setenv LIBPATH $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、`oracle` ユーザーでログインして、次のように入力します。

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

## シグナル・ハンドラの使用方法

ここでは、2 タスク通信で Oracle8 が使用するシグナル、および独自のシグナル・ハンドラを設定する方法について説明します。

### シグナル

シグナルは、データベースに接続するとユーザー・プロセスに作成され、切断すると削除されます。

表 5-4 に、Oracle8 の 2 タスク通信で使用するシグナルを示します。

表 5-4 2 タスク通信のためのシグナル

| シグナル    | 目的  |
|---------|---|
| SIGCONT | アウトオブバンド・ブレークをユーザー・プロセスから Oracle プロセスに送信する場合に、パイプ 2 タスク・ドライバが使用します。   |
| SIGINT  | ユーザーの割り込み要求を検出する場合に、すべての 2 タスク・ドライバが使用します。SIGINT は、Oracle ではなく、ユーザー・プロセスが受け取ります。  |
| SIGPIPE | 通信チャネルでファイルの終わりを検出するために、パイプ・ドライバが使用します。パイプへの書き込み時に、読み込み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  | パイプ・ドライバが使用します。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。Oracle プロセスが異常終了すると、UNIX カーネルはユーザー・プロセスに SIGCLD を送ります（サーバー・プロセスが異常終了しているかどうかを調べるには、シグナル・ハンドラで wait() を使用します）。SIGCLD は、ユーザー・プロセスが受け取り、Oracle は受け取りません。 |
| SIGTERM | パイプ・ドライバが、ユーザー側から Oracle プロセスに割り込みシグナルを送る場合に使用します。ユーザーが割り込みキー（[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。  |
| SIGIO   | ネットワーク・イベントの着信を示すために、Oracle Net8 のプロトコル・アダプタが使用します。   |
| SIGURG  | ユーザー・プロセスから Oracle プロセスへアウトオブバンド・ブレークを送る場合に、Oracle Net8 TCP/IP ドライバが使用します。  |

ここに示されているシグナルは、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() ルーチンをコールして設定すると、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。その他のシグナルには、必要なだけのシグナル・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。

シグナル・ルーチンの例

次に、独自のシグナル・ルーチンおよび受取りルーチを設定する方法を示します。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。

```

/* 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。

```

/*
** My own user interrupt handler.
*/
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
Do something appropriate here. */
...
}
...
/* clear my interrupt handler */
if (err = osncui(handle))
{
/* if the return value is non-zero, an error has occurred
Do something appropriate here. */
...
}
...
}
```

## XA 機能

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、リンク行で Oracle 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`）を使用する場合のみ必要です。

Oracle8 Server 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（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、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、Oracle8 の XA ライブラリにリンクする必要があります。Oracle8 の XA コールは、共有ライブラリ `$ORACLE_HOME/lib/libclntsh.a` および静的ライブラリ `$ORACLE_HOME/lib/libclient.a` の両方で定義されています。

---

## Oracle Net8 の構成

- 参照ドキュメント
- 主な Net8 製品および特徴
- Oracle Net8 Protocol Adapter
- BEQ Protocol Adapter
- IPC Protocol Adapter
- RAW Protocol Adapter
- TCP/IP Protocol Adapter
- SPX/IPX Protocol Adapter
- APPC/LU6.2 Protocol Adapter
- Net8 Naming Adapter
- Oracle Enterprise Manager ( OEM ) Intelligent Agent
- Oracle Advanced Networking Option

# 参照ドキュメント

**参照：** Oracle Net8 の特徴の詳細は、次のドキュメントを参照してください。

- 『Oracle Net8 管理者ガイド』
- 『Oracle Networking Quick Reference Card for Net8』
- 『Oracle Advanced Networking Option 管理者ガイド』
- 『Oracle Security Server ガイド』
- 『Oracle Cryptographic Toolkit Programmer's Guide』

## README ファイルへの補足情報

[表 6-1](#) 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 6-1 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 Networking Option         | \$ORACLE_HOME/network/doc/README.ANO                |
| Oracle Intelligent Agent           | \$ORACLE_HOME/network/doc/README.oemagent           |
| Oracle Security Server             | \$ORACLE_HOME/network/doc/README.OSS                |
| Oracle APPC/LU6.2 Protocol Adapter | \$ORACLE_HOME/network/install/lu62pa/doc/README.doc |
| Oracle Names Server                | \$ORACLE_HOME/network/install/names/doc/README.doc  |
| Oracle SPX/IPX Protocol Adapter    | \$ORACLE_HOME/network/install/spxpa/doc/README.doc  |
| Oracle TCP/IP Protocol Adapter     | \$ORACLE_HOME/network/install/tcppa/doc/README.doc  |

## 主な Net8 製品および特徴

参照: サンプル・ファイルの詳細は、『Oracle Net8 管理者ガイド』を参照してください。

## 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

AIX では、Oracle Net8 および Connection Manager のグローバル・ファイルは、デフォルトで `/etc` ディレクトリに保存されています。

Oracle Net8 および Connection Manager 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` に設定したディレクトリ（設定した場合）
2. `/etc/` ディレクトリ
3. `$ORACLE_HOME/network/admin`

ファイルがデフォルトのディレクトリにない場合、すべてのネットワーク・ユーザーの起動ファイルの環境変数 `TNS_ADMIN` で、異なるディレクトリを設定します。

C シェルの場合、次のように入力します。

```
% setenv TNS_ADMIN new_default
```

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`、`listner.ora`、`names.ora`、`sqlnet.ora` および `tnsnames.ora` のサンプル構成ファイルは、`$ORACLE_HOME/network/admin/samples` に保存されています。

### アダプタ・ユーティリティ

インストールされている Oracle Net8 アダプタを表示するには、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アダプタが表示されます。

```
% adapters oracle
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
Advanced Networking Option/Network Security products linked with oracle are:
    Oracle Security Server Authentication Adapter
```

## Oracle Connection Manager

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

## マルチスレッド・サーバー

マルチスレッド・サーバーの詳細は、『Oracle8 Server 概要』および『Oracle8 Server 管理者ガイド』を参照してください。

## Oracle Names

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

## Net8 Assistant

`$ORACLE_HOME/bin/net8asst.sh` を実行して、Net8 Assistant を起動します。Net8 Assistant を実行する前に、`LD_LIBRARY_PATH` に `$ORACLE_HOME/network/lib` ディレクトリを追加します。

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

## Oracle Net8 Protocol Adapter

AIX では、Net8 リリース 8.0.6 用に次のプロトコル・アダプタがサポートされています。

- BEQ Protocol Adapter



- IPC Protocol Adapter
- RAW Protocol Adapter
- TCP/IP Protocol Adapter
- SPX/IPX Protocol Adapter
- APPC/LU6.2 Protocol Adapter

TCP/IP、APPC/LU6.2 または SPX/IPX Protocol Adapter 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。要件の詳細は、『Oracle8 for AIX-Based Systems インストレーション・ガイド』を参照してください。BEQ および IPC Protocol Adapter に固有のオペレーティング・システム要件はありません。

IPC、TCP/IP、APPC/LU6.2 または SPX/IPX Protocol Adapter 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 `init.ora` ファイルの中のデータベース初期化パラメータ `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・アダプタの項を参照してください。

表 6-2 に、各プロトコル・アダプタの ADDRESS 指定を示します。

表 6-2 ADDRESS 指定

| プロトコル・ ADDRESS の指定<br>アダプタ |  |
|----------------------------|--|
| BEQ                        | (ADDRESS =<br>(PROTOCOL = BEQ)<br>(PROGRAM = ORACLE_HOME/bin/oracle)<br>(ARGV0 = oracleORACLE_SID)<br>(ARGS = '(DESCRIPTION=(LOCAL=YES) (ADDRESS=(PROTOCOL=BEQ)))')<br>(ENVS = 'ORACLE_HOME=ORACLE_HOME,ORACLE_SID=ORACLE_SID')<br>) |
| IPC                        | (ADDRESS =<br>(PROTOCOL=IPC)<br>(KEY=key)<br>)   |
| RAW                        | N/A  |
| TCP/IP                     | (ADDRESS =<br>(PROTOCOL=TCP)<br>(HOST=hostname)<br>(PORT=port_id)<br>)   |

表 6-2 ADDRESS 指定

| プロトコル・アダプタ | ADDRESS の指定  |
|------------|--|
| SPX/IPX    | (ADDRESS =<br>(PROTOCOL=SPX)<br>(SERVICE=service_name)<br>)  |
| APPC/LU6.2 | (ADDRESS =<br>(PROTOCOL=LU62)<br>(TP_NAME=transaction_program_name)<br>(LU_NAME=logically_unit_name)<br>(MODE=mode_name)<br>(PLU=partner_lu_name)<br>) |

## BEQ Protocol Adapter

### BEQ Protocol Adapter の概要

BEQ Protocol Adapter 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。サービス名を指定（コマンド行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 Protocol Adapter が使用されます。その場合、専用の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 Protocol Adapter 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 Protocol Adapter 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 Protocol Adapter が動作するためにネットワーク・リスナーは不要です。これは、アダプタがクライアント・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 Protocol Adapter は、クライアント・プログラムおよび Oracle8 Server が同一のマシンに常駐する場合にのみ使用できます。BEQ Protocol Adapter 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 Oracle8 Server にリンクされています。

### BEQ の ADDRESS 指定

ADDRESS のキーワードとその値の組合せが BEQ Protocol Adapter の接続パラメータです。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 6-3 に、BEQ Protocol Adapter の接続パラメータの構文を示します。

表 6-3 BEQ Protocol Adapter の接続パラメータの構文

|          |   |
|----------|---|
| PROTOCOL | 使用するアダプタを指定します。<br>値は beq で、大文字または小文字のいずれかで指定します。   |
| 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。<br>推奨値は oracleORACLE_SID です。  |
| ARGS     | '(DESCRIPTION=(LOCAL=YES)(ADDRESS=(PROTOCOL=BEQ)))'   |
| ENVS     | ORACLE_HOME には、接続先データベースの ORACLE_HOME ディレク<br>トリをフルパス名で指定します。ORACLE_SID には、接続先データベ<br>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。

例 6-1 クライアントを指定するための BEQ ADDRESS

```
(ADDRESS =  
  (PROTOCOL = BEQ)  
  (PROGRAM = /u01/app/oracle/product/8.0.6/bin/oracle)  
  (ARGV0 = oracleV806)  
  (ARGS = '(DESCRIPTION=(LOCAL=YES)(ADDRESS=(PROTOCOL=BEQ)))')  
  (ENVS = 'ORACLE_HOME=/u01/app/oracle/product/8.0.6,ORACLE_SID=V806')  
)
```

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。

# IPC Protocol Adapter

## IPC Protocol Adapter の概要

IPC Protocol Adapter は、クライアント・プログラムおよび Oracle8 Server 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 Protocol Adapter と似ています。IPC Protocol Adapter と BEQ Protocol Adapter の違いは、IPC Protocol Adapter 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 Protocol Adapter が動作するには、ネットワーク・リスナーが必要です。IPC Protocol Adapter 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 Oracle8 Server にリンクされています。

IPC Protocol Adapter では、Oracle7 リリース 7.1 以降、UNIX システムでの UNIX Domain Socket (IPC) 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 Oracle8 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。

## IPC の ADDRESS 指定

ADDRESS のキーワードとその値の組合せが IPC Protocol Adapter の接続パラメータです。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 6-4 に、IPC Protocol Adapter の接続パラメータの構文を示します。

表 6-4 IPC Protocol Adapter の接続パラメータの構文

|          |   |
|----------|---|
| PROTOCOL | 使用するアダプタを指定します。<br>値は ipc で、大文字または小文字のいずれかで指定します。 |
| 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 (SID) を指定<br>します。       |

次に、IPC ADDRESS の例を示します。

**例 6-2 クライアントを指定するための IPC ADDRESS**

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

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。

## RAW Protocol Adapter

データがクライアントとサーバー間で転送されると、Net8 は、そのヘッダー情報を各パケット（ネットワークを介して送信される情報のひとまとまり）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の送信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、接続後は、データを送受信するすべてのデータ・パケットや、接続を再設定または切断するパケットから Net8 のヘッダー情報は削除され、Net8 の NT および Oracle Protocol Adapter 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。たとえば、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、この場合は、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 Protocol Adapter

### TCP/IP Protocol Adapter の概要

Oracle Net8 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、Oracle Net8 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してください。

## TCP/IP の ADDRESS 指定

ADDRESS のキーワードとその値の組合せが TCP/IP Protocol Adapter の接続パラメータです。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 6-5 に、TCP/IP Protocol Adapter の接続パラメータの構文を示します。

表 6-5 TCP/IP Protocol Adapter の接続パラメータの構文

|          |   |
|----------|---|
| PROTOCOL | 使用するアダプタを指定します。<br>値は大文字でも小文字でもかまいません。デフォルトは <code>tcp</code> です。                     |
| HOST     | ホスト名またはホストの IP アドレスを指定します。  |
| PORT     | TCP/IP のポートを指定します。 <code>/etc/services</code> 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。

### 例 6-3 クライアントを指定するための TCP/IP の ADDRESS

```
(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。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。

## SPX/IPX Protocol Adapter

SPX/IPX Protocol Adapter の `help` コマンドについては、[表 6-6](#) を参照してください。

### ntisbsdmd 同報通信デーモン

クライアントは名前を使用し、その名前を SPX アドレスに変換してサーバーを識別し、サーバーと通信します。Netware バインダリは、変換メカニズムを提供するディレクトリ・サービスです。サーバーがバインダリに登録されると、そのアドレスのバインダリを定期的に通知します。これは、サーバー通知プロトコル (SAP) を使用して行われます。

サーバーは、IPX データグラムで 60 秒ごとに SAP パケットを同報通信します。この SAP パケットには、関連するすべてのアドレス情報が含まれています。その後、どのクライアントからも、必要なサーバーのアドレスに最も近いサーバーに対して問合せできます。

Oracle SPX/IPX Protocol Adapter は、`$ORACLE_HOME/bin` の `ntisbsdmd` 同報通信デーモンを使用して同報通信を行います。`ntisbsdmd` は、`ntspcxctl` ユーティリティによって起動および終了されます。

### ntspcxctl ユーティリティ

`ntspcxctl` ユーティリティには、名前の登録と取消し、およびバインダリの問合せを実行する機能が含まれています。また、同報通信デーモンを起動または停止する場合に使用します。リスナーは、このデーモンを使用して、使用中のサービス名を自動登録します。

[例 6-4](#) に、`ntspcxctl` ユーティリティのユーザーが複数の場合の例を示します。

#### 例 6-4 ntspcxctl ユーティリティの使用

`ntspcxctl` ユーティリティは、コマンド行からコマンドを読み取ります。パラメータを指定していない場合、それらのパラメータを指定するためのプロンプトが表示されます。

`ntspcxctl` を起動するには、次のように入力します。

```
$ ntspcxctl
```

次のように表示されます。

```
ntspcxctl: Version 2.0.12.1 - on Fri Jul 3 11:43:50 1999
```

同報通信デーモンを起動するには、次のように入力します。

```
ntspcxctl> startup
```

次のように表示されます。

```
ntisbsdsm started at Fri Jul 3 11:43:47 1999
```

デーモンがすでに起動されている場合は、システム・メッセージが表示されます。

同報通信デーモンは、マシンの起動時に常に自動的に起動されるように設定する必要があります。`/etc/inittab` ファイルにエントリを追加して、デーモンの起動を自動化します。たとえば、システムの起動時に `ntisbsdsm` を起動するには、`/etc/inittab` に次の行を追加します。

```
ntspcxctl:2:once:/u/oracle/bin/ntisbsdsm &
```

この場合、`/u/oracle` は `$ORACLE_HOME` へのフルパス名を示します。

テスト用の名前を登録するには、`register` および名前を入力します。

```
ntspcxctl> register stet
```

これによって `ntisbsdsm` が所有するソケットが作成され、登録されます。

次のメッセージが表示されます。

```
Name YYY successfully registered
YYY address 00eee045:000000000001:4454
```

`ntisbsdsm` の状態を表示するには、次のように入力します。

```
ntspcxctl> status
```

次のメッセージが表示されます。

```
ntisbsdsm started at Fri Jul 3 11:43:47 1999
Tracing is off
Pid: 14784 YYY
```

次のように入力すると、状態をより簡潔に表示できます。

```
ntspcxctl> summary
```

次のメッセージが表示されます。

```
ntisbsdsm started at Fri Jul 3 11:43:47 1999
Tracing is off
1 names are registered
```



## SPX/IPX Protocol Adapter のコマンドの要約

表 6-6 に、SPX/IPX Protocol Adapter の help コマンドを示します。

表 6-6 help コマンド

|                           |                     |
|---------------------------|---------------------|
| register name             | エントリを登録します。         |
| remove name               | エントリを取り消します。        |
| shutdown [force]          | ntisbsdsm を停止します。   |
| startup                   | 状態の要約を表示します。        |
| traceon                   | トレースをアクティブにします。     |
| traceoff                  | トレースを非アクティブにします。    |
| status                    | 状態の詳細情報を表示します。      |
| getname name / hex_number | ネーム・サービスの問合せを実行します。 |
| exit                      | プログラムを終了します。        |
| help [command]            | コマンド情報を印刷します。       |
| !                         | シェルを取り消します。         |

## getname コマンド

getname コマンドは、Novell システムに名前を要求する場合に使用します。同報通信デモンは必要ありません。

次のように入力します。

```
getname name servicetype
```

次のメッセージが表示されます。

```
getname name servicetype (address number_of_hops)
```

表 6-7 に、getname コマンドの構文を示します。

表 6-7 getname コマンドの構文

|                |  |
|----------------|--|
| name           | 入力した名前です。  |
| servicetype    | Novell で割り当てられた番号です。Oracle の番号は 103 です。  |
| address        | 入力した名前のアドレスです。   |
| number_of_hops | 宛先までのホップ数（16 進数）です。値が 10 の場合は、名前が登録解除されたことを表します。SAP 問合せがサポートされていない場合、値は 0000 です。 |

付けることができるすべての名前を表示するには、次のように入力します。

```
getname * *
```

例 6-5 に、getname コマンドで名前を付ける例を示します。

#### 例 6-5 getname コマンドの使用

```
ntspxctl> getname YYY *
YYY  servertype x0103 address 00eee045:0000000000001:
      4465 hops 0000
ntspxctl> getname * 103
LSNR  servertype x0103 address 00eee053:0000000000001:
      502c hops 0000
IBM6  servertype x0103 address 00eee058:0000000000001:
      507f hops 0000
DESK  servertype x0004 address 00eee055:0000000000001:
      5451 hops 0000
DESK  servertype x0107 address 00eee055:0000000000001:
      5104 hops 0000
CXY4  servertype x009e address 00eee055:0000000000001:
      5063 hops 0000
IBM2  servertype x0004 address 00eee057:0000000000001:
      5451 hops 0000
```

ntisbsdsm を停止するには、次のように入力します。

```
ntspxctl> shutdown
```

このデーモンは、名前がまだ登録されている場合は停止しません。次のメッセージが表示されます。

```
1 names are registered
ntisbsdsm not stopped
```

名前を取り消すには、*remove* および名前を入力します。名前が YYY の場合の例を次に示します。

```
ntspxctl> remove YYY
```

次のメッセージが表示されます。

```
Name xxxremoved.
ntspxctl> shutdown
ntisbsdsm stopped
```

強制終了するには、次のように入力します。

```
ntspxctl> shutdown force
```

次のメッセージが表示されます。

```
ntisbsdsm stopped
```

## SPX/IPX の ADDRESS 指定

SPX/IPX Protocol および Oracle SPX/IPX がシステムにインストール済みの場合、TNS 接続記述子を指定した SPX/IPX パラメータを使用して、SPX/IPX コミュニティ・ノードを識別できます。

ADDRESS のキーワードとその値の組合せが SPX/IPX Protocol Adapter パラメータです。

```
(ADDRESS=
  (PROTOCOL=SPX)
  (SERVICE=service_name)
)
```

表 6-8 に、SPX/IPX Protocol Adapter 接続の構文を示します。

**表 6-8 SPX/IPX Protocol Adapter 接続の構文**

|          |  |
|----------|--|
| PROTOCOL | アダプタ名を指定します。SPX/IPX の場合、値は <code>spx</code> です。  |
| SERVICE  | ネットワーク上のアプリケーションを示す一意の名前を 30 文字以下で指定します。サービスには起動中に名前が付けられ、ネットワーク全体で使用できます。クライアントは、ネットワーク・ディレクトリのバインダリの検索を使用してこのサービスを参照します。 |

例 6-6 に、リモート・サーバー上にサービス MAILDB1 を指定する SPX/IPX ADDRESS を示します。

例 6-6 SPX/IPX Protocol Adapter 接続

```
(ADDRESS=
  (PROTOCOL=SPX)
  (SERVICE=MAILDB1)
)
```

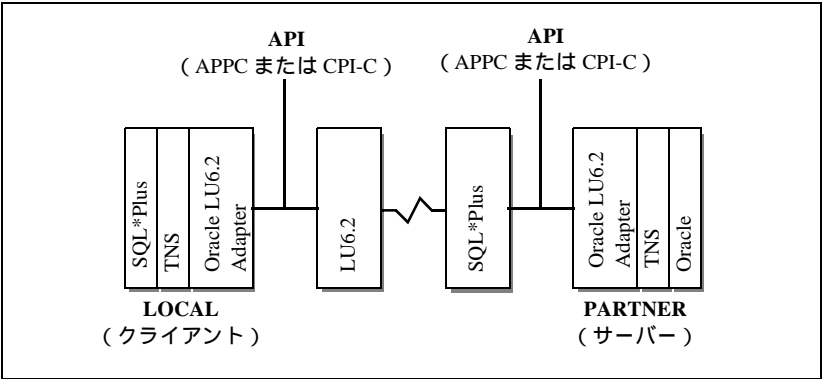
通常、ADDRESS は、接続記述子、構成ファイルなどを構成している内容の一部です。

APPC/LU6.2 Protocol Adapter

Oracle APPC/LU6.2 Protocol Adapter は、Oracle プログラム間の通信に LU6.2 サービスを使用するネットワークで使用できます。たとえば、APPC/LU6.2 を使用することで TNS アプリケーションは、標準インタフェースとして API を使用できるようになります。

図 6-1 に、LU6.2 通信サービスおよび Oracle APPC/LU6.2 Protocol Adapter を使用した Oracle プログラム間での通信レイヤーを示します。

図 6-1 Oracle プログラムおよび LU6.2 間での通信レイヤー



AIX 固有のリスナー

AIX では、汎用のリスナーをサポートしていません。かわりに AIX では、ドライバ nlobt がトランザクション・プログラムとして働き、個々の LU で接続要求の着信時に SNA サービスによって起動されます。このドライバがかわりに、Oracle シャドウ・プロセスを起動します。

## APPC/LU6.2 の ADDRESS 指定

APPC/LU6.2 Protocol Adapter パラメータは、各ノードの接続記述子で指定します。各接続記述子には、[ キーワード ]=[ 値 ] という形の組合せがいくつか含まれています。APPC/LU6.2 固有のキーワードは、接続記述子内で任意の順に入力できます。

```
(ADDRESS=
  (PROTOCOL=LU62)
  (TP_NAME=transaction_program_name)
  (LU_NAME=logical_unit_name)
  (MODE=mode_name)
  (PLU=partner_lu_name)
)
```

表 6-9 に、APPC/LU6.2 Protocol Adapter 接続の構文を示します。

表 6-9 APPC/LU6.2 Protocol Adapter の構文

|          |   |
|----------|---|
| PROTOCOL | 使用するプロトコル・アダプタを示します。値は大文字でも小文字でもかまいません。APPC/LU6.2 の場合、値は lu62 です。   |
| TP_NAME  | ターゲットで実行するトランザクション・プログラム名、または着信接続要求をリスニングするときに使用するトランザクション・プログラム名を指定します。この値は必須です。   |
| LU_NAME  | tnsnames.ora を参照する場合、リモート・パートナ LU の名前を指定します。このキーワードが listener.ora に指定されている場合は、ローカル LU の名前を指定します。LU_NAME は、無視されたり、他のパラメータで上書きされたりすることがあります。一部の APPC/LU6.2 実装要件のために、LU_NAME には、常に完全修飾された LU 名 (netid.lu_name) を指定する必要があります。 |
| MODE     | 論理デバイス間のセッションの特性を定義します。モードは、パートナ LU およびトランザクション・プログラム名とともに、ALLOCATE セグメントで指定します。モード名は、ローカル LU およびパートナ LU の両方に共通である必要があります。この値は必須です。   |
| PLU      | パートナ LU の名前を指定します。AIX では、この値は必須で、TP_NAME/transaction_program_name に設定できます。  |

## Net8 Naming Adapter

### NIS Naming Adapter

NIS Naming Adapter の構成の詳細は、『Oracle Net8 管理者ガイド』を参照してください。

## Oracle Enterprise Manager ( OEM ) Intelligent Agent

### エージェント・サービスの展開および自動構成

**参照：** 詳細は、『Oracle Enterprise Manager 構成ガイド』を参照してください。

### tcl スクリプトのデバッグ

実行ファイル `oratclsh` は、tcl スクリプトをデバッグ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 アプリケーション開発者ガイド』を参照してください。

## Oracle Advanced Networking Option

### .bak ファイル

Oracle Advanced Networking Option のインストール中に、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` に保存されています。これらのファイルは、Oracle Advanced Networking Option の削除（デインストール）中の再リンクに必要なので、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 Networking Option 管理者ガイド』を参照してください。

### DCE 統合

DCE 統合の詳細は、『Oracle Advanced Networking Option 管理者ガイド』を参照してください。

---

## AIX での Oracle データ・カートリッジ・デモの実行

- データ・カートリッジ共通の要件
- Oracle8 Time Series Cartridge
- Oracle8 Visual Information Retrieval Cartridge
- Oracle8 Image Cartridge

## データ・カートリッジ共通の要件

データ・カートリッジでは、インストール後にサンプル・アプリケーションを組み込むために C コンパイラが必要です。

**参照：** Oracle データ・カートリッジの詳細は、『Oracle Enterprise Manager 構成ガイド』および各カートリッジのドキュメントを参照してください。

## Oracle8 Time Series Cartridge

**参照：** 詳細は、『Oracle Time Series カートリッジ・ユーザズ・ガイド』を参照してください。

## Time Series Cartridge デモのインストール

Time Series Cartridge デモのファイルは、`$ORACLE_HOME/ord/ts/demo` に保存されています。このディレクトリの各サブディレクトリに、それぞれのデモが含まれています。

- `usage` デモでは、Time Series オブジェクトの組み込み方法および Time Series 機能をコールする方法を示します。最初に `usage` デモを実行して、他のデモで使用する Time Series スキーマのサンプルを作成します。
- `extend` デモでは、カスタマイズされた Time Series 機能の書込みおよび追加方法を示します。
- `proc` デモでは、Pro\*C を使用してカートリッジが保存されているデータへのアクセス方法を説明します。
- `oci` デモでは、Oracle コール・インタフェースを通してデータにアクセスする方法を説明します。
- `dev2k` デモには、カートリッジを使用してデータを取り出す Developer 2000 形式が含まれています。

各デモの詳細は、デモ・ディレクトリにある README ファイルを参照してください。



## Oracle8 Visual Information Retrieval Cartridge

**参照：** 詳細は、『Oracle Visual Information Retrieval Cartridge ユーザーズ・ガイド』を参照してください。

### デモの作成

1. 次のように入力して、VIR デモのディレクトリを作成します。

```
% svrmgrl
SVRMGR> connect internal;
SVRMGR> create or replace directory virdemodir
2> as "$ORACLE_HOME/ord/vir/demo";
```

2. 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 virdemodir to public
2> with grant option;
SVRMGR> exit;
```

3. 次のように入力して、virdemo プログラムを作成します。

```
% cd $ORACLE_HOME/ord/vir/demo
% make -f demo_ordvir.mk virdemo
```

## Oracle8 Image Cartridge

**参照：** 詳細は、『Oracle Image Cartridge ユーザーズ・ガイド』を参照してください。

### デモの作成

1. 次のように入力して、Image Cartridge のデモ・ディレクトリを作成します。

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

2. 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
2> with grant option;
SVRMGR> exit;
```

3. 次のように入力して、imgdemo プログラムを作成します。

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

## 記号

- ! (感嘆符)
  - SQL\*Plus プロンプト, 4-7
- \$ (ドル記号)
  - 使用, 2-21
- ? (疑問符)
  - 使用, 2-9
- @
  - SORACLE\_SID, 2-10
- \_editor
  - 設定, 4-6

## 数字

- 2 タスク
  - アーキテクチャ
  - セキュリティ, 2-18

## A

- ADA\_PATH, 2-10
- Ada コンパイラ, 2-10
- ADDRESS の指定
  - プロトコル・アダプタ, 6-5
- Advanced Networking Option, 6-18
  - .bak ファイル, 6-18
  - DCE 統合, 6-18
  - セキュリティおよびシングル・サインオン, 6-18
- afiedt.buf, 4-7
- AIX
  - システム・パラメータ, 3-7
- APPC/LU6.2 Protocol Adapter, 6-16
  - ADDRESS, 6-17
  - AIX リスナー, 6-16

## B

- BEQ Protocol Adapter, 6-6
  - ADDRESS, 6-6
  - 概要, 6-6
  - 接続パラメータの構文, 6-7
- Bourne シェル・スクリプトのトレース, 2-9
- BSD, 2-11

## C

- C
  - Pro\*C/C++, 5-4
  - SQL\*Module for, 5-13
- CATPROC.SQL, 1-15
- CDE ツール, 2-8
- chmod コマンド
  - 例, 2-20
- COBOL
  - Pro\*COBOL, 5-6
- CONNECT INTERNAL
  - セキュリティ, 2-19
- COPY コマンド
  - SQL\*Plus, 4-8
- coraenv, 2-9

## D

- daemon ユーザー
  - セキュリティ, 2-21
- DB\_BLOCK\_SIZE, 3-10
- DBA\_GROUP
  - /etc/listener.ora ファイル, 2-22
- DBA\_GROUP\_sid
  - /etc/listener.ora ファイル, 2-22

dba グループ  
    再リンク, 2-18  
    メンバー, 2-17  
DBA グループ ID  
    キーワード, 2-21  
    デフォルト以外の名前, 2-22  
DISPLAY  
    環境変数, 2-10

---

## E

echo コマンド, 2-4  
EXTSHM  
    警告, 2-16

---

## F

FORMAT プリコンパイラ  
    Pro\*COBOL, 5-10

---

## G

getname コマンド, 6-13  
glogin.sql, 4-2

---

## H

HOME, 2-10

---

## I

Image Cartridge, 7-3  
    デモの作成, 7-3  
IPC Protocol Adapter, 6-8  
    ADDRESS, 6-8  
    概要, 6-8  
ireclen, 5-3

---

## L

LANGUAGE  
    環境変数, 2-10  
LD\_LIBRARY\_PATH  
    環境変数, 2-10  
login.sql, 4-2

---

## M

Make, 5-6  
MAXDATAFILES パラメータ, 2-32  
maxfree, 3-7  
MAXINSTANCES パラメータ, 2-32  
MAXLOGFILES パラメータ, 2-32  
MAXLOGHISTORY パラメータ, 2-32  
MAXLOGMEMBERS パラメータ, 2-32  
maxper, 3-8  
maxperm, 3-7  
MicroFocus COBOL コンパイラ, 5-7  
minfree, 3-7  
minperm, 3-7, 3-8

---

## N

Naming Adapter, 6-18  
Net8  
    Advanced Networking Option, 6-18  
    APPC/LU6.2 Protocol Adapter, 6-16  
    BEQ Protocol Adapter, 6-6  
    IPC Protocol Adapter, 6-8  
    Net8 Assistant, 6-4  
    Oracle Connection Manager, 6-4  
    Oracle Enterprise Manager Intelligent Agent, 6-18  
    Oracle Names, 6-4  
    RAW Protocol Adapter, 6-9  
    README ファイル, 6-2  
    SPX/IPX Protocol Adapter, 6-11  
    TCP/IP Protocol Adapter, 6-9  
    アダプタ・ユーティリティ, 6-3  
    製品および特徴, 6-3  
    ネーム・アダプタ, 6-18  
    ファイルおよびユーティリティ, 6-3  
    プロトコル・アダプタ, 6-4  
    マルチスレッド・サーバー, 6-4  
Net8 Assistant, 6-4  
Net8 の構成ファイル  
    保存場所, 6-3  
NLS\_LANG  
    環境変数, 2-7  
ntisbsdsm 同報通信デーモン, 6-11  
ntspxtl ユーティリティ, 6-11

## O

---

### ORA\_NLS

環境変数, 2-7

### Oracle, 5-8

メモリーの使用量, 2-12

### Oracle Connection Manager, 6-4

### Oracle Diagnostics Pack, 3-5

### Oracle Enterprise Manager Intelligent Agent, 6-18

ttl スクリプトのデバッグ, 6-18

エージェント・サービスの展開と自動構成, 6-18

### Oracle Names, 6-4

### Oracle Server

アカウント, 2-17

### Oracle Tuning Pack, 3-6

### ORACLE\_HELP

環境変数, 2-8

### ORACLE\_HOME

? の使用方法, 2-9

環境変数, 2-8

### ORACLE\_SID

環境変数, 2-8

プロンプトの抑制, 2-6

### ORACLE\_TERM

環境変数, 2-8

### Oracle8 で使用する UNIX 環境変数

ADA\_PATH, 2-10

DISPLAY, 2-10

HOME, 2-10

LANG, 2-10

LD\_LIBRARY\_PATH, 2-10

PATH, 2-10

PRINTER, 2-11

SHELL, 2-11

TERM, 2-11

TMPDIR, 2-11

XENVIRONMENT, 2-11

### oracle アカウント

認可の設定, 2-20

### Oracle 環境変数

EPC\_DISABLED, 2-7

NLS\_LANG, 2-7

ORA\_NLS33, 2-7

ORACLE\_BASE, 2-7

ORACLE\_HELP, 2-8

ORACLE\_HOME, 2-8

ORACLE\_PATH, 2-8

ORACLE\_SID, 2-8

ORACLE\_TERM, 2-8

ORACLE\_TERMINAL, 2-8

ORACLE\_TRACE, 2-9

ORAENV\_ASK, 2-9

TNS\_ADMIN, 2-9

TWO\_TASK, 2-9

### oracle グループ

権限と実行プログラム, 2-18

### Oracle コール・インタフェース, 5-15

使用, 5-15

デモンストレーション・プログラム, 5-15

ユーザー・プログラム, 5-16

### oracle ソフトウェア所有者, 2-17

特殊アカウント, 2-17

### Oracle のシステム ID, 2-8

### Oracle プリコンパイラ、OCI のリンクおよび Make

ファイル, 5-17

カスタム Make ファイル, 5-17

未定義シンボル, 5-17

### Oracle ライブラリ

Oracle 共有ライブラリ, 5-19

静的リンクおよび動的リンク, 5-18

### Oracle ランタイム・システム

Pro\*COBOL, 5-8

### ORAENV\_ASK, 2-9

設定, 2-6

### oraenv ファイル

説明, 2-5

データベース間の移動, 2-5

### orainst, 2-8

### orapwd コマンド, 2-23

### orecln, 5-3

## P

---

### PATH, 2-10

### PL/SQL

デモンストレーション

ロード, 2-27

### PRINTER, 2-11

### Pro\*C/C++

Make ファイル, 5-5, 5-6

管理, 5-4

シグナル, 5-20

システム構成ファイル, 5-4

使用, 5-4

- デモンストレーション・プログラム, 5-4
- ユーザー・プログラム, 5-6
- Pro\*COBOL, 5-6
  - FORMAT プリコンパイラ, 5-10
  - Oracle ランタイム・システム, 5-8
  - 環境変数, 5-7
  - 管理, 5-7
  - システム構成ファイル, 5-7
  - デモンストレーション・プログラム, 5-8
  - ネーミング方法の違い, 5-6
  - ユーザー・プログラム, 5-10
- PRODUCT\_USER\_PROFILE 表
  - SQL\*Plus, 4-3
- PTX Agent, 3-2
- PTX Manager, 3-2
- pupbld.sql, 4-3

## R

---

- RAW Protocol Adapter, 6-9
- README ファイル
  - Net8, 6-2
- root
  - ユーザー, 2-17

## S

---

- Server Manager
  - SHOW PARAMETERS, 2-14
  - コマンド, 2-19
  - セキュリティ, 2-19
- SGA, 3-11
- SHELL, 2-11
- SHUTDOWN コマンド, 2-19
  - セキュリティ, 2-19
- SIGCLD 2 タスク・シグナル, 5-20
- SIGINT 2 タスク・シグナル, 5-20
- SIGIO 2 タスク・シグナル, 5-20
- SIGPIPE 2 タスク・シグナル, 5-20
- SIGTERM 2 タスク・シグナル, 5-20
- SIGURG 2 タスク・シグナル, 5-20
- Site Profile
  - SQL\*Plus, 4-2
- smit
  - 「システム管理インタフェース・ツール」を参照
- SPOOL コマンド
  - SQL\*Plus, 4-8

- 使用, 4-8
- SPX/IPX Protocol Adapter, 6-11
  - ADDRESS, 6-15
  - getname コマンド, 6-13
  - ntisbsdsm 同報通信デーモン, 6-11
  - ntspxtcl ユーティリティ, 6-11
  - コマンドの要約, 6-13
- SQL
  - 管理, 4-2
- SQL\*DBA, 2-8
  - SHOW PARAMETERS, 2-14
- SQL\*Loader, 3-11
- SQL\*Module
  - for Ada, 5-13
  - for C, 5-13
- SQL\*Module for Ada, 5-13
  - 管理, 5-13
  - システム構成ファイル, 5-13
  - 使用, 5-13
  - デモンストレーション・プログラム, 5-13
  - ユーザー・プログラム, 5-14
- SQL\*Net バージョン 1, 2-9
- SQL\*Net バージョン 2, 2-9
  - /etc/oratab ファイル, 6-3
  - オペレーティング・システム許可のログイン, 2-21
  - デフォルト・ディレクトリ, 6-3
- SQL\*Plus
  - COPY コマンド, 4-8
  - PRODUCT\_USER\_PROFILE 表, 4-3
  - Site Profile, 4-2
  - SPOOL コマンド, 4-8
  - UNIX コマンド, 4-7
  - User Profile, 4-2
  - エディタ, 4-6
  - 環境変数, 4-7
  - システム・エディタ, 4-6
  - 使用, 4-6
  - 制限事項, 4-8
  - 設定ファイルのサンプル, 4-3
  - デフォルト・エディタ, 4-6
  - デモンストレーション表, 4-3
  - ファイルの設定, 4-2
  - ヘルプ機能, 4-4
  - 割込み, 4-8
- SQL\*Plus の使用, 4-6
- SQL\*Plus への割込み, 4-8
- SQL トレース, 3-4

STARTUP コマンド  
    セキュリティ , 2-19  
SYSDATE  
    TZ , 2-12  
SYSTEM アカウント  
    権限 , 2-17  
SYS アカウント  
    権限 , 2-17

## T

---

TCP/IP Protocol Adapter , 6-9  
    ADDRESS , 6-10  
    概要 , 6-9  
TERM  
    環境変数 , 2-11  
Time Series Cartridge , 7-2  
    dev2k デモ , 7-2  
    extend デモ , 7-2  
    oci デモ , 7-2  
    proc デモ , 7-2  
    usage デモ , 7-2  
Time Series Cartridge のデモ  
    インストール , 7-2  
TIMED\_STATISTICS , 3-4  
TKPROF , 3-5  
TMPDIR , 2-11  
TNS\_ADMIN  
    環境変数 , 2-9  
TNS リスナー  
    Oracle TCP/IP Protocol Adapter 用の構成 , 6-9  
Toolkit II リソース・ファイル , 2-8  
TWO\_TASK  
    環境変数 , 2-9  
TZ  
    および SYSDATE , 2-12  
    環境変数 , 2-12

## U

---

UNIX  
    セキュリティ , 2-18  
UNIX コマンド  
    SQL\*Plus からの実行 , 4-7  
User Profile  
    SQL\*Plus , 4-2  
utlbstat.sql , 3-4

utlestat.sql , 3-4

## V

---

Visual Information Retrieval Cartridge , 7-3  
    デモの作成 , 7-3  
vmtune , 3-7

## X

---

XA 機能 , 5-22  
XENVIRONMENT  
    環境変数 , 2-11  
X-Window , 2-10 , 2-11

## あ

---

アダプタ・ユーティリティ , 6-3

## う

---

受取り、ルーチン  
    例 , 5-20

## え

---

エディタ  
    SQL\*Plus , 4-6  
    順序の設定 , 4-6

## お

---

オペレーティング・システム許可のログイン  
    SQL\*Net バージョン 2 , 2-21  
オペレーティング・システム・コマンド  
    SQL\*Plus からの実行 , 4-7

## か

---

カートリッジのデモ  
    Image Cartridge , 7-3  
    Time Series のインストール , 7-2  
    Visual Information Retrieval Cartridge , 7-3  
各国語サポート ( NLS )  
    環境変数 , 2-7  
環境変数 , 5-7  
    ADA\_PATH , 2-10

DISPLAY , 2-10  
HOME , 2-10  
LANG , 2-10  
LANGUAGE , 2-10  
LD\_LIBRARY\_PATH , 2-10  
MicroFocus COBOL コンパイラ , 5-7  
NLS\_LANG , 2-7  
ORA\_NLS , 2-7  
ORACLE\_HELP , 2-8  
ORACLE\_HOME , 2-8  
ORACLE\_SID , 2-8  
ORACLE\_TERM , 2-8  
ORACLE\_TERMINAL , 2-8  
ORACLE\_TRACE , 2-9  
ORAENV\_ASK , 2-9  
PATH , 2-10  
PRINTER , 2-11  
Pro\*COBOL , 5-7  
SHELL , 2-11  
SQL\*Plus の設定 , 4-7  
TERM , 2-11  
TMPDIR , 2-11  
TNS\_ADMIN , 2-9 , 6-3  
TWO\_TASK , 2-9  
TZ , 2-12  
XENVIRONMENT , 2-11  
管理  
    SQL , 4-2

## き

---

キーワード  
    DBA グループ ID , 2-21  
疑問符  
    使用例 , 2-9  
共通の環境  
    oraenv ファイル , 2-5  
    設定 , 2-5  
共有  
    オブジェクト・ライブラリ , 2-10  
    ライブラリ・ローダー , 2-10  
共有メモリー  
    SGA , 2-15

## く

---

クラスタ  
    サイズの見積り , 2-13  
グループ  
    サンプル・スクリプト , 2-26

## け

---

警告  
    EXTSHM , 2-16  
権限  
    dba グループ , 2-18  
    付与 , 2-18  
言語 , 2-7

## こ

---

構成ファイル  
    Net8 , 6-3  
    ブリコンパイラ , 5-3  
コマンド  
    orapwd , 2-23  
コマンド・インタプリタ , 2-11

## さ

---

索引サイズ  
    計算 , 2-13  
サブシェル  
    SQL\*Plus での作成 , 4-7  
参照マニュアル , xi

## し

---

シグナル  
    2 タスク , 5-19  
    ハンドラの作成 , 5-19  
シグナル・ハンドラ  
    シグナル , 5-19  
    使用 , 5-19  
シグナル・ルーチン  
    例 , 5-20  
システム管理インタフェース・ツール , 3-3  
システム構成ファイル  
    Pro\*C/C++ , 5-4  
    Pro\*COBOL , 5-7



- SQL\*Module for Ada , 5-13
- システム時刻
  - 設定 , 2-12
- システム・エディタ
  - SQL\*Plus , 4-6
  - 順序の設定 , 4-6
  - デフォルト設定 , 4-7
- システム・グローバル領域 (SGA)
  - 要件 , 2-15
- システム・パラメータ
  - AIX , 3-7
- 実行プログラム , 2-10
- 自動ログイン
  - listener.ora ファイル , 2-21
  - REMOTE\_OS\_ROLES , 2-25
  - UNIX 以外のシステム , 2-21
- シャドウ・プロセス
  - セキュリティ , 2-18
- 初期化パラメータ
  - BACKGROUND\_DUMP\_DEST , 2-14
  - BITMAP\_MERGE\_AREA\_SIZE , 2-14
  - COMMIT\_POINT\_STRENGTH , 2-14
  - CONTROL\_FILES , 2-14
  - CREATE\_BITMAP\_AREA\_SIZE , 2-14
  - DB\_BLOCK\_BUFFERS , 2-14
  - DB\_BLOCK\_SIZE , 2-14
  - DB\_FILE\_DIRECT\_IO\_COUNT , 2-14
  - DB\_FILE\_MULTIBLOCK\_READ\_COUNT , 2-14
  - DB\_FILES , 2-14
  - DISTRIBUTED\_TRANSACTIONS , 2-15
  - HASH\_AREA\_SIZE , 2-15
  - HASH\_MULTIBLOCK\_IO\_COUNT , 2-15
  - LOG\_ARCHIVE\_BUFFER\_SIZE , 2-15
  - LOG\_ARCHIVE\_BUFFERS , 2-15
  - LOG\_ARCHIVE\_DEST , 2-15
  - LOG\_ARCHIVE\_FORMAT , 2-15
  - LOG\_BUFFER , 2-15
  - LOG\_CHECKPOINT\_INTERVAL , 2-15
  - LOG\_SMALL\_ENTRY\_MAX\_SIZE , 2-15
  - MTS\_LISTENER\_ADDRESS , 2-15
  - MTS\_MAX\_DISPATCHERS , 2-15
  - MTS\_MAX\_SERVERS , 2-15
  - MTS\_SERVERS , 2-15
  - NLS\_LANGUAGE , 2-15
  - NLS\_TERRITORY , 2-15
  - OBJECT\_CACHE\_MAX\_SIZE\_PERCENT , 2-15
  - OBJECT\_CACHE\_OPTIMAL\_SIZE , 2-15

- OPEN\_CURSORS , 2-15
- OS\_AUTHENT\_PREFIX , 2-15
- PROCESSES , 2-15
- REMOTE\_OS\_AUTHENT , 2-21
- SHARED\_POOL\_SIZE , 2-15
- SHOW PARAMETERS コマンド , 2-14
- SORT\_AREA\_SIZE , 2-15
- SORT\_READ\_FAC , 2-15
- SORT\_SPACEMAP\_SIZE , 2-15
- USER\_DUMP\_DEST , 2-15
- デフォルト , 2-14

## す

- スーパーユーザー , 2-17
- スレッドのサポート , 5-18

## せ

- 制御、リモート DBA アクセス
  - REMOTE\_DBA\_OPS\_ALLOWED , 2-22
  - REMOTE\_DBA\_OPS\_DENIED , 2-22
- 制御、リモート・ログイン , 2-22
  - OPS\_DOLLAR\_LOGIN\_DENIED , 2-22
- 制限
  - リソース , 2-13
- 制限事項 (SQL\*Plus) , 4-8
  - COPY コマンド , 4-8
  - ウィンドウのサイズ変更 , 4-9
  - リターン・コード , 4-9
- 静的リンクおよび動的リンク
  - Oracle ライブラリ , 5-18
- セキュリティ
  - 2 タスク・アーキテクチャ , 2-18
  - CONNECT INTERNAL , 2-19
  - Server Manager アクセス , 2-19
  - SHUTDOWN コマンド , 2-19
  - STARTUP コマンド , 2-19
  - グループ・アカウント , 2-18
  - 権限の割当て , 2-18
  - デフォルト・グループ名 , 2-18
  - ネットワーク DBA 権限 , 2-20
  - ファイル所有権 , 2-18

## た

---

タイム・ゾーン  
TZ での設定, 2-12

## ち

---

チューニング  
SGA, 3-11  
アーカイバ・バッファ, 3-10  
メモリーおよびページング, 3-7  
チューニング、パフォーマンス, 3-5

## て

---

ディスク  
割当て制限, 2-13  
データベース  
管理者  
実行プログラムへの権限, 2-18  
ファイル  
セキュリティ, 2-20  
認可, 2-20  
ブロック・サイズ, 3-10  
データベースの変更, 2-5  
データ・カートリッジ  
デモの実行, 7-1  
デバック・プログラム, 5-3  
デフォルト  
ディレクトリ, 2-11  
デフォルト設定  
システム・エディタ, 4-7  
デフォルト・プリンタ, 2-11  
デモ  
Image Cartridge の作成, 7-3  
Time Series Cartridge のインストール, 7-2  
Visual Information Retrieval Cartridge の作成, 7-3  
デモンストレーション  
ブリコンパイラ, 2-28  
プロシージャ・オプション、PL/SQL, 2-27  
デモンストレーション表  
SQL\*Plus, 4-3  
削除, 4-4  
手動で作成, 4-4  
デモンストレーション・プログラム  
Oracle コール・インタフェース, 5-15  
Pro\*C/C++, 5-4

Pro\*COBOL, 5-8  
SQL\*Module for Ada, 5-13  
テンポラリ・ディスク・ファイル, 2-11

## と

---

動的リンクおよび静的リンク  
Oracle ライブラリ, 5-18  
特殊アカウント, 2-17  
特殊グループ  
root, 2-17

## に

---

入出力, 2-10

## ね

---

ネットワーク  
DBA 権限, 2-20  
ネットワーク・セキュリティ, 2-20  
ネットワーク・パスワード, 2-20

## は

---

配布ソフトウェア, 2-8  
バッファ・キャッシュ間  
チューニング, 3-7  
パフォーマンス監視ツール  
AIX, 3-2  
Oracle, 3-4  
Performance Toolbox, 3-2

## ふ

---

ファイルの設定  
SQL\*Plus, 4-2  
ファイル名  
SQL\*Plus でのデフォルト拡張子, 4-8  
複数のシグナル・ハンドラ, 5-20  
ブリコンパイラ  
ireclen および oreclen の値, 5-3  
値, 5-3  
大文字から小文字への変換, 5-3  
概要, 5-2  
シグナル, 5-20  
実行ファイルの再リンク, 5-2

- デモンストレーションの実行, 2-28
- ベンダー提供のデバッグ・プログラム, 5-3
- プリコンパイラ構成ファイル, 5-3
- プリコンパイラ実行ファイルの再リンク, 5-2
- ブロック・サイズ, 3-10
- プロトコル・アダプタ, 6-4
  - ADDRESS の指定, 6-5

## へ

---

- ページング
  - 過度な, 3-7, 3-9
  - 不十分な, 3-9
- ヘルプ機能
  - SQL\*Plus, 4-4
- ヘルプ・ファイル, 2-8

## ほ

---

- ホーム・ディレクトリ, 2-10

## ま

---

- マルチスレッド・サーバー, 6-4

## め

---

- メモリー
  - 仮想, 2-12
  - 競合, 3-7
  - 共有, 2-15
  - 使用量の見積り, 2-12

## ゆ

---

- ユーザー
  - サンプル・スクリプト, 2-26
- ユーザー・プログラム
  - Oracle コール・インタフェース, 5-16
  - Pro\*C/C++, 5-6
  - Pro\*COBOL, 5-10
  - SQL\*Module for Ada, 5-14
- ユーザー割込みハンドラ, 5-20

## り

---

- リソース定義, 2-11

- リソースの制限, 2-13
- リモート
  - 接続
    - INTERNAL として, 2-24
    - OPERATOR として, 2-24
  - パラメータ
    - REMOTE\_OS\_AUTHENT, 2-21
  - ログイン
    - セキュリティ, 2-21
- リモート接続パラメータ
  - OS\_AUTHENT\_PREFIX, 2-25
  - REMOTE\_OS\_AUTHENT, 2-25
  - REMOTE\_OS\_ROLES, 2-25

## ろ

---

- ログイン・ホーム・ディレクトリ
  - 管理, 2-25
  - サンプル・スクリプト, 2-26

