

Oracle8i Net8

管理者ガイド

リリース 8.1

Oracle8i Net8 管理者ガイド リリース 8.1

部品番号 : A62733-1

第 1 版 : 1999 年 5 月

原本名 : Net8 Administrator's Guide, Release 8.1.5

原本部品番号 : A67440-01

Copyright © 1999, Oracle Corporation. All rights reserved.

原本著者 : Deborah Steiner

原本協力者 : Pierre Baudin, Toby Close, Santanu Datta, Harvey Eneman, Shuvayu Kanjila, Tong Ming Lee, Ethan Malasky, Sergio Mendiola, Ed Miner, Radhika Shah, Chandar Venkataraman, Norman Woo, Jun Wu

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かねます。当プログラムを米国国防総省の米国政府機関に提供する際には、『Restricted Rights』と共に提供してください。この場合次の Legend が適用されません。

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.....	xv
第 I 部 Net8 の概要と概念	
1 Net8 の概要	
Net8 の概要.....	1-2
Oracle の接続性の概要.....	1-3
構成の概要.....	1-4
接続ルーティング機能.....	1-6
2 アーキテクチャと概念	
Net8 の操作.....	2-2
接続操作.....	2-2
データ操作.....	2-4
例外操作.....	2-4
Net8 アーキテクチャ.....	2-5
ネットワーク環境.....	2-5
Net8 と Transparent Network Substrate (TNS).....	2-6
スタック通信.....	2-6
Oracle ネットワーク環境内のスタック通信.....	2-9
サーバー間の対話.....	2-13
IIOP 環境内のスタック通信.....	2-14
JDBC 環境内のスタック通信.....	2-16
マルチスレッド・サーバーへのネットワーク・セッション.....	2-20
専用サーバーへのネットワーク・セッション.....	2-23

専用サーバーへの Bequeathed ネットワーク・セッション	2-25
Net8 とリスナー	2-27
リスナーへのデータベース・インスタンス登録	2-27
クライアント接続	2-28
Connect-Time Failover	2-29
ロード・バランシング	2-30
Oracle Names アーキテクチャ	2-31
概要	2-31
管理リージョン	2-33
ドメイン	2-33
データ・リポジトリとしての Oracle Names	2-34
Oracle Names Server に格納されるデータ	2-35
ネットワーク・コンポーネントの編成とネーミング	2-37
管理責任を分散する複数のリージョンの使用	2-40
Oracle Names のバージョンによる違い	2-42
検出の理解	2-45
Oracle Connection Manager のアーキテクチャ	2-46
Oracle Connection Manager プロセス	2-46
接続集中化	2-48
Net8 アクセス制御	2-49
複数プロトコル・サポート	2-49
接続プーリング	2-50

第 II 部 Net8 の設定と構成

3 ネットワークのプランニング

計画の概要	3-2
ネットワーク・レイアウトの定義	3-2
ネット・サービス名の変換	3-3
ネーミング・メソッド	3-3
ホスト・ネーミング	3-3
ローカル・ネーミング	3-5
Oracle Names を使用した集中化されたネーミング	3-5
外部ネーミング	3-6
ネーミング・メソッドの選択	3-8
ネットワーク・パフォーマンスの向上	3-10
接続要求の管理	3-10

接続プーリング.....	3-11
接続集中化.....	3-11
クライアント負荷のバランス.....	3-12
セッション・データ・ユニット (SDU) サイズの調整によるデータ転送の最適化.....	3-13
TCP/IP の固定バッファ・フラッシング.....	3-14
リスナー・キューサイズの構成.....	3-14
計画のサマリー	3-15

4 リリース 8.1 のインストールと Net8 関連製品について

今回のリリースに含まれる新しい製品とその機能	4-2
Net8 リリース 8.1 で変更された構成ファイル	4-8
構成ファイルで使用される構文の変更点.....	4-9
TNSNAMES.ORA の変更点.....	4-16
LISTENER.ORA の変更点.....	4-19
CMAN.ORA の変更点.....	4-20
リリース 8.1 で廃止されたパラメータ.....	4-21
Net8 のインストールとその製品	4-22
Net8 Server のインストール.....	4-23
Net8 Client のインストール.....	4-26
Net8 製品の概要.....	4-27

5 アップグレードまたは移行の検討

Net8 と SQL*Net の違い	5-2
廃止された SQL*Net パラメータ.....	5-4
クライアントとデータベースの互換性問題	5-5
Oracle8i Client リリース 8.1 から Oracle8i データベース・リリース 8.1 への接続.....	5-5
Oracle8 リリース 8.0 または Oracle7 Client から Oracle8i データベース・リリース 8.1 への接続.....	5-6
Oracle8i Client リリース 8.1 から Oracle8 データベース・リリース 8.0 または Oracle7 データベースへの接続.....	5-8
Oracle Names.....	5-10
Oracle Net8 Assistant による互換性問題の取り扱い	5-11
リリース 8.1 機能の利用と、リリース 8.1 サービスへの接続.....	5-12
リリース 8.0 または 7.x の機能の利用と、リリース 8.0 または 7.x のサービスへの接続.....	5-14
移行とアップグレードで考慮すべき点	5-16
Net8 リリース 8.1 へのアップグレードと移行	5-17
手順 1: サービス名とインスタンス名を確認する.....	5-17

手順 2: サーバー上でソフトウェアをアップグレードまたは移行する.....	5-18
手順 3: クライアント上でソフトウェアをアップグレードまたは移行する.....	5-19
手順 4: 機能のアップグレードと移行を実行する.....	5-19
Oracle Connection Manager と Oracle Names でさらに考慮すべき点.....	5-21
Oracle Connection Manager リリース 8.1 の使用.....	5-21
Oracle MultiProtocol Interchange にかわる Oracle Connection Manager の使用.....	5-22
Oracle Names バージョン 8 の使用.....	5-23

6 ネーミング・メソッドとリスナーの構成

構成作業の概要.....	6-2
サービス名とインスタンス名の設定.....	6-2
構成モデル.....	6-4
ネーミング・メソッド.....	6-5
構成ファイル.....	6-6
ホスト・ネーミング・メソッドによるネットワークの構成.....	6-7
手順 1: グローバル・データベース名を使用してリスナーを構成する.....	6-8
手順 2: 第 1 番目のネーミング・メソッドとして HOSTNAME を構成する.....	6-9
手順 3: ホスト名解決の環境を設定する.....	6-11
ローカル・ネーミング・メソッドによるネットワークの構成.....	6-12
インストール中の構成作業.....	6-13
インストール後の構成作業.....	6-14
Oracle Names ネーミング・メソッドによるネットワークの構成.....	6-21
手順 1: Oracle Names オプションを検討する.....	6-23
手順 2: インストールする.....	6-24
手順 3: Oracle Names Server を作成する.....	6-25
手順 4: Oracle Names Server を使用するためにクライアントと データベース・サーバーを構成する.....	6-36
手順 5: リスナーを構成する.....	6-40
手順 6: Oracle Names Server にデータベースを登録する.....	6-41
手順 7: (オプション) 委任管理リージョンにドメインを委任する.....	6-50
外部ネーミング・メソッドによるネットワークの構成.....	6-51
NIS の構成.....	6-51
NDS の構成.....	6-55
リスナーの構成.....	6-62
LISTENER.ORA ファイル変更後のリスナーの再起動.....	6-64

リスナー・プロトコル・アドレスの構成.....	6-64
デフォルト以外のポートを使用するリスナーの構成.....	6-67
LISTENER 以外のリスナー名の使用.....	6-69
大量の接続要求の処理.....	6-70
サービス情報によるリスナーの構成.....	6-71
プロトコル・アドレスの構成	6-76
アドレスの構成要素.....	6-76
プロトコルのパラメータ.....	6-78

7 Net8 のオプション機能

ネット・サービス名の変更	7-2
ネット・サービス名の属性の変更.....	7-2
ネット・サービス名の改名.....	7-3
ネット・サービス名へのプロトコル・アドレスの追加.....	7-3
ネット・サービス名からのプロトコル・アドレスの削除.....	7-5
ネット・サービス名の削除.....	7-6
ネット・サービス名の接続属性の詳細構成.....	7-7
複数アドレス・オプションの構成.....	7-11
プロファイルの構成	7-15
ドメインの構成.....	7-16
ネット・サービス名のネーミング・メソッド解決の優先順位.....	7-18
デフォルトの Oracle Names Server の構成.....	7-20
プロファイルの詳細情報の構成.....	7-22
接続要求のルーティング.....	7-25
Oracle Advanced Security の構成.....	7-27
接続集中化機能の使用	7-30
手順 1: Oracle Connection Manager の構成.....	7-30
手順 2: サーバーの構成.....	7-31
手順 3: クライアントの構成.....	7-32
複数プロトコル・サポートの使用	7-34
手順 1: Oracle Connection Manager の構成.....	7-34
手順 2: クライアントの構成.....	7-35
Net8 アクセス制御の使用	7-37
手順 1: Oracle Connection Manager の構成.....	7-37
手順 2: クライアントの構成.....	7-39
Oracle Connection Manager 機能を使用する場合のクライアントの構成	7-40

手順 1: Oracle Connection Manager アドレスの構成	7-41
手順 2: リスナー・アドレスの構成	7-44
接続プーリングの使用	7-47
Java オプション接続の構成	7-48
Java スタアド・プロシージャ構成	7-49
Enterprise JavaBeans と CORBA 構成	7-49
リスナーから EJB と CORBA を使用するための Oracle8i デフォルト構成の使用	7-52
デフォルト以外の構成の使用	7-54
EJB と CORBA による SSL の使用	7-62
Oracle 以外のデータベース・サービスの構成	7-63
外部プロシージャを使用する場合の Net8 の構成	7-63
Oracle 異機種間サービスを使用する場合の Net8 の構成	7-65
Oracle RDB データベースを使用する場合の Net8 の構成	7-69

8 接続の確立とネットワークのテスト

データベースへの接続	8-2
Net8 コンポーネント起動の概要	8-2
接続文字列の入力	8-8
起動と接続の一般的な問題を解決するためのチェックリスト	8-11
ネットワークのテスト	8-14
Net8 制御ユーティリティの使用	8-15
Oracle Names Server のテスト	8-16
NAMESCTL または Oracle Net8 Assistant によるネットワーク・オブジェクトのテスト	8-17
リスナーのテスト	8-19
Oracle Connection Manager のテスト	8-19
サーバーでの構成のテスト	8-19
クライアントからのネットワーク接続のテスト	8-21

9 マルチスレッド・サーバーの構成

概要	9-2
Oracle のマルチスレッド・サーバー・アーキテクチャ構成	9-4
ディスパッチャの初期数の設定	9-8
マルチスレッド・サーバー・モードと専用サーバー・モードの両方の構成	9-10
クライアントでのマルチスレッド・サーバー・モードの変更	9-11

10 プログラマのための Net8 拡張機能

Net8 OPEN.....	10-2
Net8 OPEN API 関数コール.....	10-3
Net8 OPEN アプリケーション・プログラム・インタフェースの検索.....	10-8
独自のアプリケーションの開発.....	10-8
Net8 OPEN アプリケーションを使用するためのシステムの構築.....	10-9
サンプル・プログラム.....	10-11
Net8 OPEN API エラー.....	10-12
UNIX クライアントのプログラム.....	10-13
シグナル・ハンドラとアラームのプログラミング.....	10-14
Bequeath.....	10-14

11 Net8 のトラブルシューティング

Net8 の診断.....	11-2
サーバーの診断.....	11-3
クライアントの診断.....	11-4
一般的なエラー・メッセージの解決.....	11-7
フィールドからのトラブルシューティングのヒント.....	11-13
トラブルシューティング時の質問への回答.....	11-14
ログ・ファイルとトレース・ファイルを使用したネットワーク上の 問題点のトラブルシューティング.....	11-15
エラー情報のロギング.....	11-15
エラー・スタック.....	11-15
ログ・ファイル名.....	11-18
ログ・パラメータの理解と設定.....	11-19
制御ユーティリティ実行時のログ設定.....	11-24
ログ・ファイルの使用.....	11-25
リスナーのログ監査証跡.....	11-25
Oracle Connection Manager ログの理解.....	11-27
エラー情報のトレース.....	11-31
トレース・ファイルの名前.....	11-32
トレース・パラメータの理解と設定.....	11-32
制御ユーティリティ実行時のトレース設定.....	11-39
Net8 トレースの評価.....	11-40
Trace Assistant を使用したトレース・ファイルの検査.....	11-43
オラクル社サポート・サービスへの連絡.....	11-58

第 III 部 リファレンス

A リスナー、Oracle Names Server および Oracle Connection Manager の制御ユーティリティ

操作モード.....	A-2
リスナー制御ユーティリティ (LSNRCTL).....	A-3
SET および SHOW 修飾子.....	A-3
セキュリティ.....	A-3
LSNRCTL コマンド.....	A-4
Oracle Names 制御ユーティリティ (NAMESCTL).....	A-30
パラメータ・オプション.....	A-30
SET および SHOW 修飾子.....	A-30
NAMESCTL の分配操作.....	A-31
セキュリティ.....	A-31
NAMESCTL の確認モード.....	A-32
NAMESCTL コマンド.....	A-33
Oracle Connection Manager 制御ユーティリティ (CMCTL).....	A-88
CMCTL コマンド.....	A-89

B 構成ファイルのサンプル

プロファイル (SQLNET.ORA).....	B-2
ローカル名構成ファイル (TNSNAMES.ORA).....	B-5
リスナー構成ファイル (LISTENER.ORA).....	B-6
Oracle Names Server 構成ファイル (NAMES.ORA).....	B-7
Oracle Connection Manager 構成ファイル (CMAN.ORA).....	B-9

C 構成パラメータ

構成ファイルの構文規則.....	C-2
構成ファイルの構文規則の詳細.....	C-3
ネットワーク・キャラクタ・セット.....	C-4
キャラクタ・セット.....	C-5
ADDRESS と ADDRESS_LIST.....	C-5
プロファイル・パラメータ (SQLNET.ORA).....	C-7
ローカル・ネーミング・パラメータ (TNSNAMES.ORA).....	C-31
一般構文.....	C-31

複数の記述.....	C-32
複数のアドレス・リスト.....	C-33
クライアント負荷のバランスと DESCRIPTION_LIST の変更された機能.....	C-34
例.....	C-36
DESCRIPTIONS および DESCRIPTION_LIST.....	C-38
ADDRESS と ADDRESS_LIST.....	C-39
リストのオプション・パラメータ.....	C-40
CONNECT_DATA セクション.....	C-43
リスナー・パラメータ (LISTENER.ORA)	C-50
リスナー・アドレス・セクション.....	C-51
SID_LIST_listern_name 静的サービス・セクション.....	C-57
制御パラメータ.....	C-62
Oracle Names パラメータ (NAMES.ORA)	C-65
Oracle Connection Manager パラメータ (CMAN.ORA)	C-77
プロトコル固有のパラメータ (PROTOCOL.ORA)	C-82

用語集

索引



1-1	基本的なクライアント / サーバー接続.....	1-2
1-2	Oracle Names を使用したクライアント / サーバー接続.....	1-5
1-3	Oracle Connection Manager を通した、クライアントとサーバーの接続.....	1-6
2-1	OSI 通信スタック.....	2-7
2-2	Oracle 環境内の標準的な通信スタック.....	2-10
2-3	通常のクライアントと IIOP クライアントの通信スタック.....	2-14
2-4	IIOP 環境におけるクライアントとサーバーの通信スタック.....	2-15
2-5	通常のクライアントと JDBC クライアントの通信スタック.....	2-17
2-6	JDBC 環境におけるクライアントとサーバーの通信スタック.....	2-19
2-7	ディスパッチャにリダイレクトされた接続.....	2-22
2-8	事前に生成された専用サーバー・プロセスにリダイレクトされた接続.....	2-24
2-9	専用サーバー・プロセスへの bequeath 接続.....	2-26
2-10	代表的な Net8 接続におけるリスナー.....	2-28
2-11	Oracle Names.....	2-32
2-12	シングル・ドメイン・ネーミング・モデル.....	2-37
2-13	階層ネーミング・モデル.....	2-38
2-14	デフォルト・ドメイン.....	2-39
2-15	委任管理リージョン.....	2-41
2-16	Oracle Connection Manager による接続集中化.....	2-48
2-17	Oracle Connection Manager による複数プロトコル・サポート.....	2-50
2-18	接続プーリング.....	2-51
5-1	Oracle8i Client から Oracle8i データベースへの接続.....	5-5
5-2	Oracle8 Client リリース 8.0 または Oracle7 Client から Oracle8i データベースへの接続.....	5-6
5-3	Oracle8i Client から Oracle8 リリース 8.0 または Oracle7 データベースへの接続.....	5-8
6-1	TNSNAMES.ORA でのネット・サービス名の使用例.....	6-12
6-2	複数のリスナー・アドレス.....	6-13
7-1	Oracle Connection Manager を使用する場合と使用しない場合の TNSNAMES.ORA ファイル.....	7-33
7-2	複数プロトコル・サポートを使用するように構成された CMAN パラメータ.....	7-36
8-1	正常な Trace Route.....	8-25
8-2	エラーの発生した Trace Route.....	8-25
9-1	Oracle マルチスレッド・サーバー・アーキテクチャ.....	9-3
10-1	Net8 OPEN.....	10-2
11-1	ネットワーク製品とエラー・スタック・コンポーネント.....	11-16
11-2	一般的なエラー・スタック.....	11-17
11-3	成功した reload 要求の一般的な監査証跡情報.....	11-25
11-4	成功した connection 要求の一般的な監査証跡情報.....	11-26
11-5	CMAN_PID.LOG.....	11-27
11-6	CMADM_PID.LOG.....	11-27
11-7	パケット情報.....	11-41
11-8	トレース・ファイルのエラー出力.....	11-42
11-9	Trace Assistant の一般的な変換結果.....	11-44
11-10	bequeath 接続で送られたサマリー・データ・パケット.....	11-45

11-11	bequeath 接続で送られた詳細データ・パケット	11-46
11-12	リダイレクト接続で送られたサマリー・データ・パケット	11-47
11-13	データ・パケット	11-47
11-14	Two Task Common のサマリー情報	11-48
11-15	Two Task Common のサマリー情報	11-50
11-16	サマリー Two-Task の最上部にある詳細 SQL 情報	11-51
11-17	一般的なトレース・ファイルの統計	11-52
11-18	トレース・ファイルの例	11-53
C-1	TNSNAMES.ORA の一般構文	C-31
C-2	TNSNAMES.ORA の複数の記述	C-32
C-3	TNSNAMES.ORA の複数のアドレス・リスト	C-33
C-4	リリース 8.1 とそれ以前のリリースのクライアント負荷のバランス	C-35
C-5	TNSNAMES.ORA に指定する複数の Oracle Connection Manager アドレス	C-36
C-6	TNSNAMES.ORA におけるクライアント負荷のバランス	C-37

はじめに

この Oracle8i Net8 管理者ガイドは、Net8 製品とそれに関連するアプリケーションの理解と使用に必要な情報を提供します。

注意： このガイドには、Oracle8i と Oracle8i Enterprise Edition 製品の機能と特徴について説明した情報が記載されています。Oracle8i と Oracle8i Enterprise Edition は、同じ基本機能を備えています。ただし、いくつかの拡張機能は Enterprise Edition のみで利用できます。たとえば、Oracle Connection Manager を使用するには、Enterprise Edition が必要です。

内容は次のとおりです。

- [対象読者](#)
- [このマニュアルの構成](#)
- [関連マニュアル](#)
- [このマニュアルで使用する表記規則](#)

対象読者

このガイドの情報は、Oracle*8i* データベースなどのサービスへの接続やネットワーク構成を行う、ネットワークまたはデータベース管理者 (DBA) を対象としています。さらに、Net8 の機能を理解したい人も対象としています。

このマニュアルの構成

このマニュアルには、11 の章と 3 つの付録があります。

第 I 部: Net8 の概要と概念

- 第 1 章の「Net8 の概要」 Net8 リリース 8.1 製品を紹介します。
- 第 2 章の「アーキテクチャと概念」 Net8 アーキテクチャ、マルチスレッド・サーバーまたは専用サーバー環境における Net8 の対話、Oracle Names、Oracle Connection Manager、および接続プーリングについて説明します。

第 II 部: Net8 の設定と構成

- 第 3 章の「ネットワークのプランニング」 Net8 を使用してネットワークを計画する際の考慮事項について説明します。さらに、Net8 製品の関係と、将来のネットワーク管理をさらにうまく行うためのオプションについて説明します。
- 第 4 章の「リリース 8.1 のインストールと Net8 関連製品について」 リリース 8.1 の構成ファイルと、インストールされる Net8 製品とアプリケーションについて説明します。
- 第 5 章の「アップグレードまたは移行の検討」 Net8 リリース 8.0 から Net8 リリース 8.1 へのアップグレードについて、および SQL*Net バージョン 2 から Net8 リリース 8.1 への移行について説明します。
- 第 6 章の「ネーミング・メソッドとリスナーの構成」 Net8 の多様な方法を使用してネットワークを構成する方法を説明します。
- 第 7 章の「Net8 のオプション機能」 Net8 の高度な機能およびオプション機能を構成する方法を説明します。Oracle Connection Manager の機能、接続プーリング、クライアント負荷のバランス、connect-time failover、異種間サービス、外部プロシージャ、および Oracle Rdb データベースなどについて説明します。
- 第 8 章の「接続の確立とネットワークのテスト」 Net8 コンポーネントの起動方法、接続の確立方法、および接続のテスト方法を説明します。
- 第 9 章の「マルチスレッド・サーバーの構成」 マルチスレッド・サーバー・アーキテクチャのコンポーネントを調整する方法を説明します。
- 第 10 章の「プログラマのための Net8 拡張機能」 プログラマのための Net8 の拡張機能について説明します。この中には、Net8 OPEN、UNIX、シグナル処理、bequeath アダプタの復習が含まれています。
- 第 11 章の「Net8 のトラブルシューティング」 Net8 をトラブルシューティングする手順について説明します。この中には、トレースとロギングに関する情報が含まれています。
-

第 III 部：リファレンス

付録 A の「リスナー、Oracle Names Server および Oracle Connection Manager の制御ユーティリティ」	Net8 制御ユーティリティ、すなわちリスナー制御ユーティリティ (LSNRCTL)、Oracle Names 制御ユーティリティ (NAMESCTL) および Connection Manager 制御ユーティリティ (CMCTL) のすべてのコマンドについて説明します。
付録 B の「構成ファイルのサンプル」	プロファイル (SQLNET.ORA)、ローカル名 (TNSNAMES.ORA)、リスナー (LISTENER.ORA)、Oracle Names Server (NAMES.ORA)、Oracle Connection Manager (CMAN.ORA) のそれぞれのサンプル構成ファイルを示します。
付録 C の「構成パラメータ」	プロファイル、ローカル名、リスナー、Oracle Names Server、Oracle Connection Manager およびプロトコルの構成パラメータを一覧で示し、それらについて説明します。

関連マニュアル

- 『Oracle8i 概要』
- 『Oracle8i リファレンス・マニュアル』
- 『Oracle8i 分散システム』
- 『Oracle8i チューニング』
- 『Oracle Enterprise Manager 管理者ガイド』
- 『Oracle8i Advanced Security 管理者ガイド』

このマニュアルで使用する表記規則

このガイドでは次の表記規則を使用しています。

表記規則	例	意味
大文字	LISTENER.ORA	SQL コマンド、キーワード、ファイル名および初期化パラメータを示す。 UNIX ではほとんどの Oracle ファイル名は小文字です。
ゴシック体の小文字	<code>\$ORACLE_HOME/network/admin startup;</code>	ディレクトリおよびコマンドを示す。 UNIX のディレクトリとコマンドはほとんど小文字です。
イタリック体	次のような変数を示す <i>filename</i>	指定する必要がある値を示す。たとえば、コマンドが <i>filename</i> の入力を要求する場合、実際のファイル名を入力する必要があります。
[]	<code>change_password [listener name]</code>	オプション項目を囲む。
< >	<code>names.default_domain = <valid domain name></code>	山カッコは、ユーザーが指定する名前を囲む。
「スタート」ボタン -> を選択	「スタート」ボタン -> 「プログラム」 -> 「Oracle-HOME_NAME」 -> 「Oracle Enterprise Management」 -> 「Enterprise Manager Console」 を選択する	Windows でのプログラムの起動方法を示す。
ディレクトリの前のスラッシュ (/)	<code>/network</code>	UNIX におけるサブディレクトリを示す。
ディレクトリの前の円記号 (¥)	<code>¥network</code>	Windows におけるサブディレクトリを示す。
UNIX では \$ORACLE_HOME、Windows では ORACLE_HOME	<code>\$ORACLE_HOME¥network</code> ディレクトリに移動する	この Optimal Flexible Architecture (OFA) 互換のリリースでは、すべてのサブディレクトリを最高位の ORACLE_HOME ディレクトリ以下に配置する必要はない。複数の Oracle ホーム・ディレクトリを含めた、ORACLE_BASE ディレクトリを設定できる。

表記規則	例	意味
記号	ピリオド . カンマ , ハイフン - セミコロンの ; コロン : 等号 = 円記号 ¥ 引用符 ' 二重引用符 " 小カッコ ()	大カッコおよび縦線以外の記号は、コマンドでは記載されているとおりに入力する必要があります。

第 I 部

Net8 の概要と概念

第 I 部では、Net8 の概念と用語について紹介します。次の章があります。

- [第 1 章の「Net8 の概要」](#)
- [第 2 章の「アーキテクチャと概念」](#)

1

Net8 の概要

この章では、Net8 について紹介し、その主なアプリケーション、機能および機能性の概要を示します。内容は次のとおりです。

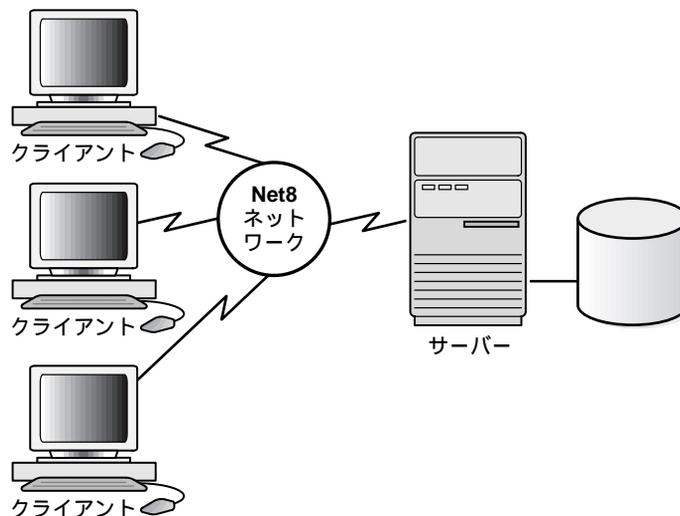
- [Net8 の概要](#)
- [Oracle の接続性の概要](#)
- [構成の概要](#)
- [接続ルーティング機能](#)

Net8 の概要

Net8 は Oracle のネットワーク製品ファミリの基礎となります。Net8 を利用することで、サービスやそのアプリケーションを異なるコンピュータに配置したり、ピア・アプリケーションとして通信させることができます。Net8 の主要な機能は、ネットワーク・セッションを確立し、クライアント・マシンとサーバー間、または 2 つのサーバー間でデータを転送することです。Net8 はネットワークの各マシンに配置されます。ネットワーク・セッションが確立されると、Net8 はクライアントやサーバーへのデータの案内役として動作します。

図 1-1 は、クライアント / サーバー接続の様子を示しています。

図 1-1 基本的なクライアント / サーバー接続



ネットワーク・セッションは、リスナーの支援を受けて確立されます。リスナーはサーバーで実行される独立したプロセスで、このサーバーはクライアントの着信接続要求をリスニングし、サーバーへのトラフィックを管理します。

リスナーはクライアントの要求を受信し、これをサーバーに渡します。クライアント（またはクライアントとして動作するサーバー）がサーバーとのネットワーク・セッションを要求するたびに、リスナーが実際の要求を受信します。クライアントの情報がリスナーの情報と一致すると、リスナーはサーバーへの接続を認めます。

同じサーバーに多数の接続が必要とされる環境では、Net8 は Oracle Connection Manager と呼ばれるルーティング・プロセスを提供します。このプロセスはクライアントまたはサーバー以外のマシンで実行されます。Oracle Connection Manager の詳細は、1-6 ページの「[接続ルーティング機能](#)」を参照してください。

Oracle の接続性の概要

Net8 を利用すると、Oracle データベース、Oracle 以外のデータベース、ゲートウェイ、外部プロシージャ（PL/SQL コードから呼び出される関数）など、さまざまなサービスに接続できます。Oracle 環境において Oracle データベースは最も一般的なサービスであり、この項ではデータベース・サービスに注目します。

エンド・ユーザーがネットワークを経由してデータベース・サーバーに接続する場合、接続文字列によってネット・サービス名からサービスが識別されます。たとえば、次のように指定します。

```
sql> CONNECT username/password@net_service_name
```

ネット・サービス名は、ネットワーク上のサービスの検索に必要なネットワーク設定情報を提供することにより、ネットワークを通してサービスにアクセスすることができます。ネット・サービス名は次のものに変換されます。

- サービスへのネットワーク・ルート。プロトコル・アドレスによるリスナーの場所が含まれます。
- サービス名。通常はデータベース・サービスの識別に使用するグローバル・データベース名で、データベース名とデータベース・ドメインで構成されます。

リスナーは、プロトコルを通してクライアントの接続を受け付けます。リスナーはクライアントの情報を、データベース・サービスから受信した情報および構成ファイル LISTENER.ORA に保存された情報と比較します。情報が一致した場合は接続が許可されます。

構成の概要

構成は、サービスに割り当てるネット・サービス名とリスナー接続先アドレスのリストを作成し、ネット・サービス名を変換するネーミング・メソッドを構成することで行います。ネーミング・メソッドは、ネット・サービス名を変換する方法です。

Net8 は複数のカテゴリのネーミング・メソッドをサポートしています。どこにサービス名を格納する場合でも、Net8 はネット・サービス名を簡単に構成できるツールを提供します。

- 既存のネーム変換サービス

ネット・サービス名は、既存の IP アドレス変換メカニズムを通して格納されます。

- クライアント構成ファイル

ネット・サービス名は、ネットワーク上の各クライアントとサーバーにあるローカル構成ファイル (TNSNAMES.ORA) に格納できます。TNSNAMES.ORA は各クライアントで構成できるので、特定のクライアントの必要性に応じて調整できます。

- 外部ネーミング・サービス

使用中の環境でサードパーティ製のネーミング・サービスがすでに構成されている場合、Net8 は追加構成を避けて、構成済みのネーミング・サービスにネット・サービス名を格納できます。

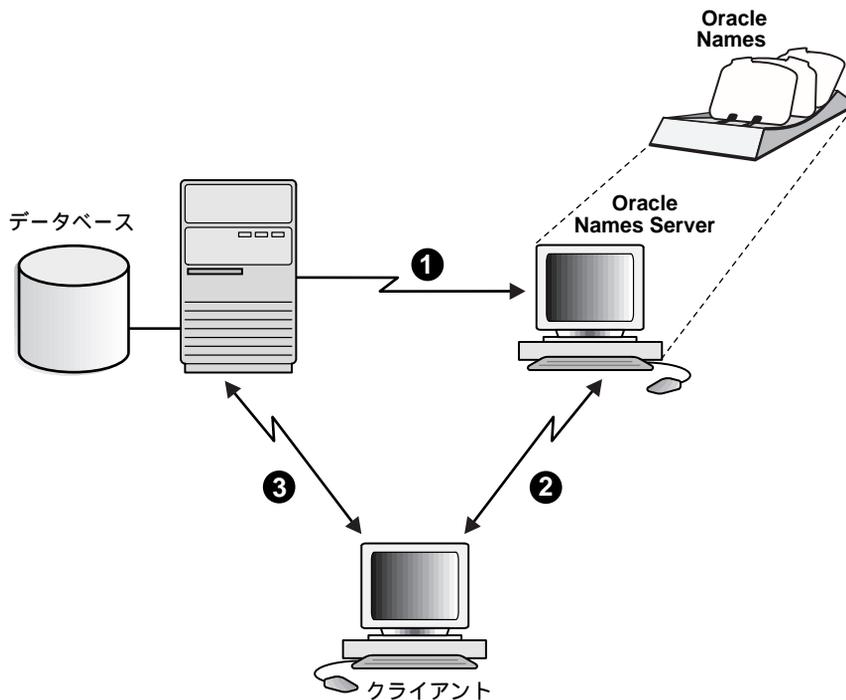
- 集中化された格納

ネット・サービス名とサービス・アドレスは、1 つのサービス・アドレスのディレクトリに集中的に格納できます。この集中化した格納は、Oracle Names と呼ばれる Oracle 独自のネーミング・サービスで、サービス・アドレスの集中的な格納を維持します。クライアント・アプリケーションが単純な名前を使用してサーバーに接続を要求する場合、Net8 クライアントは Oracle Names に接続して、サービスをネットワーク・アドレス記述子に変換します。

図 1-2 は、次の動作を示したものです。

1. データベースが Oracle Names Server にサービスを登録します。
2. クライアントが Oracle Names Server に接続してネットワーク・アドレスを取り出し、ネットワーク上のサービスを検索します。
3. 続いて、クライアントが透過的にサービスに接続します。

図 1-2 Oracle Names を使用したクライアント / サーバー接続



ネット・サービス名の格納場所の詳細は、3-3 ページの「ネット・サービス名の変換」を参照してください。

接続ルーティング機能

Net8 では、次のような環境で接続ルーティングを行います。

- 多数のユーザーが単一プロトコルを通して単一のサーバーにアクセスする必要がある環境
- クライアントとサーバーに異なるなプロトコルがインストールされ、一般的な接続が失敗する環境

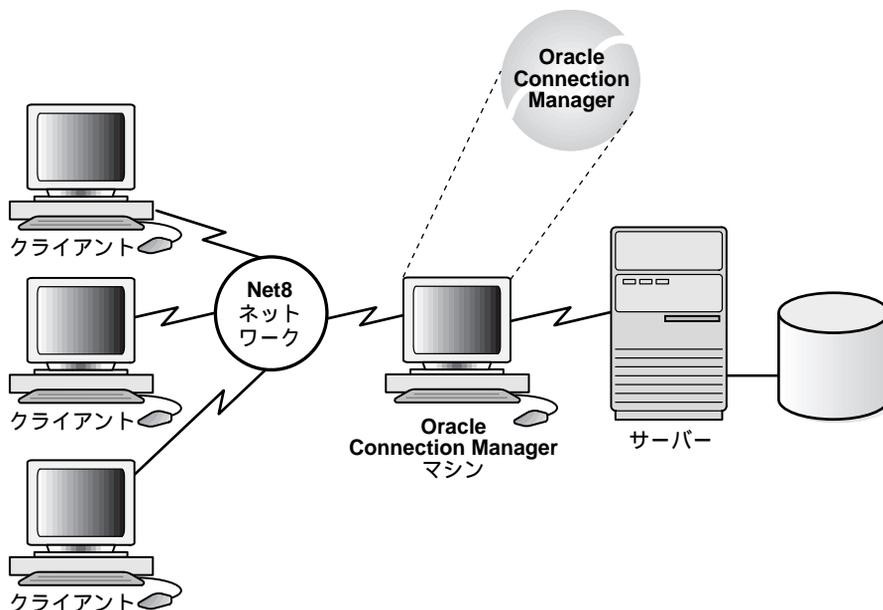
標準的な Net8 接続では、クライアントとサーバーに同じプロトコルがインストールされている必要があります。

- TCP/IP 環境で、指定したサーバーへのクライアント・アクセスの制御が必要な環境

接続ルーティングは、Oracle Connection Manager によってサポートされ、これによりクライアント接続要求がサーバーにルーティングされます。

図 1-3 は、クライアント接続が、ネットワーク中の別のマシンに存在する Oracle Connection Manager にルーティングされる様子を示したものです。

図 1-3 Oracle Connection Manager を通した、クライアントとサーバーの接続



詳細は、2-46 ページの「Oracle Connection Manager のアーキテクチャ」を参照してください。

アーキテクチャと概念

Net8 は、Transparent Network Substrate (TNS) と業界標準ネットワーク・プロトコルを使用して、サーバーにクライアントを接続し、Oracle ネットワーク・セッションを確立します。

この章では、TNS および TNS が分散システムで果たす役割について説明します。また、基本的な接続とトランスポートの各操作を行うために Net8 が TNS と対話する方法についても説明します。内容は次のとおりです。

- Net8 の操作
- Net8 アーキテクチャ
- マルチスレッド・サーバーへのネットワーク・セッション
- 専用サーバーへのネットワーク・セッション
- Net8 とリスナー
- Oracle Names アーキテクチャ
- Oracle Connection Manager のアーキテクチャ
- 接続プーリング

Net8 の操作

Net8 によって、Oracle 分散トランザクションで相互に協力するマシン間（クライアント / サーバー間でも、サーバー / サーバー間でもかまいません）の通信が可能になります。特に、Net8 には 3 つの基本的なネットワーキング操作があります。

- [接続操作](#)
- [データ操作](#)
- [例外操作](#)

接続操作

Net8 は、2 種類の接続操作をサポートします。

- [サーバーへの接続](#)
- [サーバーからの切断](#)

サーバーへの接続

ユーザーは、接続するサービスの識別子とともにユーザー名およびパスワードを渡すことによって接続要求を開始します。この識別子はネット・サービス名と呼ばれ、接続記述子に含まれているネットワーク・アドレスにマップされます。接続記述子には次のものが含まれます。

- プロトコル・アドレスによるリスナーの場所を含む、サービスへのネットワーク・ルート
- リリース 8.1 のサービスを識別するサービス名、データベースの論理表現、またはリリース 8.1 以前のデータベースを識別する Oracle システム識別子（SID）
- 接続するリリース 8.1 サービスのオプション・インスタンス名

クライアントとサーバーは、このネット・サービス名を使用してアプリケーションに接続します。

特定のネットワーク構成に従って、接続記述子は次のいずれかに格納できます。

- TNSNAMES.ORA と呼ばれるローカル名構成ファイル
- Oracle Names で使用する Oracle Names Server
- NIS または DCE CDS などの外部ネーミング・サービス

Net8 は、リスナーの支援を受けてネットワーク・セッションを調整します。

サーバーからの切断

サーバーからの切断要求は、次のようにして開始されます。

- [ユーザーによる切断](#)

- 追加接続要求
- 接続の異常終了
- タイマーによる切断またはデッド接続検出

ユーザーによる切断 ユーザーは、クライアント / サーバーのトランザクションが終了したときに、サーバーからの接続の切離しを要求できます。サーバーは、すべてのサーバー / サーバー・データ転送が終了し、リンクが必要ない場合は、第 2 サーバーからも切断できます。

追加接続要求 クライアント・アプリケーションがサーバーに接続されているときに、同じサーバーまたは他のサーバー上の別のユーザー・アカウントにアクセスを要求すると、大半の Oracle Tools は、現在接続されているサーバーからアプリケーションを切断します。接続の切離しが終了すると、対応するサーバー上の新規ユーザー・アカウントに対する接続要求が開始されます。

接続の異常終了 その他のコンポーネントでは、Net8 に予告せずに通信を切断または異常終了させる場合があります。この場合、Net8 は次のデータ操作中に障害を認識し、クライアントとサーバーの各操作をクリーン・アップして、現行の操作を効率よく切断します。

タイマーによる切断またはデッド接続検出 デッド接続検出を利用すると、Net8 はクライアントの異常終了によって中断したままになっている接続を識別できます。この機能は、無効になった接続によるリソースの無駄を最小限に抑えます。また、ブレイクされた接続のユーザーが保持しているコミットされていないトランザクションおよびロックが自動的にロールバックされます。

デッド接続検出を有効にした接続では、ユーザーが定義した間隔（通常は数分）で、サーバーからクライアントに小さなプローブ・パケットが送られます。接続が無効な場合（通常はクライアント・プロセスまたはマシンに接続できない場合）送信操作でエラーが発生すると、接続がクローズされ、サーバー・プロセスは接続を終了します。

データ操作

Net8 では、4 組のクライアント / サーバー・データ操作をサポートします。

- データの同期送信
- データの同期受信
- データの非同期送信
- データの非同期受信

クライアント側では、Net8 内の送信要求で SQL 対話要求が送られます。サーバー側では、Net8 が受信要求を処理し、データをデータベースに渡します。サーバーから戻るときは、その逆順の操作が行われます。

基本的な送信要求と受信要求は同期式です。クライアントが要求を開始する場合、クライアントはサーバーが応答してくるまで待機します。次に、クライアントはその他の要求を発行できます。

Net8 にはデータ要求を非同期で送受信する機能が追加されています。この機能は、マルチスレッド・サーバーとも呼ばれる Oracle 共有サーバーをサポートするために追加されました。

例外操作

Net8 では、3 種類の例外操作をサポートします。

- 接続を通じてブレークを発行する
- ブレーク後の同期化のために接続をリセットする
- 着信ブレークの接続状態をテストする

ユーザーは、これら 3 つの操作のうち 1 つのみ、すなわち、ブレークの発行を制御します。ユーザーが割り込みキー（マシンによっては [Ctrl] を押しながら [C]）を押すと、アプリケーションはこの機能呼び出しします。また、SQL*Loader を使用して無効なデータ行のロードを試みた場合など、異常な操作が行われると、データベースはクライアントに対してブレークを発行します。

他の 2 つの例外操作は、Net8 を使用する製品の内部で実行され、ネットワーク・タイミングに関する問題を解決します。たとえば、Net8 は新しいデータを受信したかどうかを確認するために通信チャンネルのテストを開始します。リセット機能は、ブレーク操作の後で接続を同期状態に戻す場合など、異常な状態を解決する場合に使用されます。

Net8 アーキテクチャ

この項では、次のアーキテクチャ概念を説明します。

- ネットワーク環境
- Net8 と Transparent Network Substrate (TNS)
- スタック通信
- Oracle ネットワーク環境内のスタック通信
- IIOP 環境内のスタック通信
- サーバー間の対話
- JDBC 環境内のスタック通信

ネットワーク環境

Oracle ネットワーク環境は、次の 2 つの概念に基づいています。

- 分散処理
- 分散データベース

分散処理 Oracle データベースとクライアント・アプリケーションは、分散処理環境として知られている環境で動作します。分散処理または協同処理は、1 回のデータ・トランザクションを完了するために、複数のコンピュータが関与します。Oracle Tool などのアプリケーションは、特定の操作を行うためのデータを必要とするクライアントの役割を果たします。データベース・サーバーは、データを格納や提供を行います。

代表的なネットワーク構成では、クライアントとサーバーが別の物理マシン上の論理エンティティとして存在する場合があります。この構成では、労力の分割が可能で、クライアント・ワークステーションとサーバー・マシンとの間で、リソースが効果的に割り当てられます。通常、クライアントはユーザーが使い勝手のよいアプリケーションを実行するために適したメモリーを搭載したデスクトップ・コンピュータ上に存在するのに対し、サーバーにはデータベースを実行、管理するために、多くのメモリー、ディスク記憶領域、処理能力があります。

分散データベース このタイプのクライアント/サーバー・アーキテクチャでは、ネットワーク間にデータベースを分散できます。分散データベースは、ユーザーに対しては 1 つの論理データベースとして表示される、複数のコンピュータ上に格納されたデータベースのネットワークです。分散データベース・サーバーは、データベース・リンク、またはあるデータベースから別のデータベースへのパスで接続されます。1 つのサーバーがデータベース・リンクを使用して、第 2 サーバーにある情報を必要に応じて問合せしたり、変更を行い、第 2 サーバーに対するクライアントの役割を果たします。

Net8 と Transparent Network Substrate (TNS)

Net8 は、Transparent Network Substrate (TNS) と業界標準ネットワーク・プロトコルを使用して、基本的な機能を実現しています。TNS は、すべての業界標準プロトコルに対して 1 つの共通インタフェースを提供するために、Net8 に構築されている基礎技術です。

TNS を利用すると、peer-to-peer 接続が可能になります。peer-to-peer アーキテクチャでは、複数のコンピュータ (ネットワーク化された環境で使用される場合はノードと呼ばれる) が相互に直接交信でき、中間のデバイスは必要ありません。peer-to-peer システムでは、ノードはクライアントとサーバーの両方になり得ます。

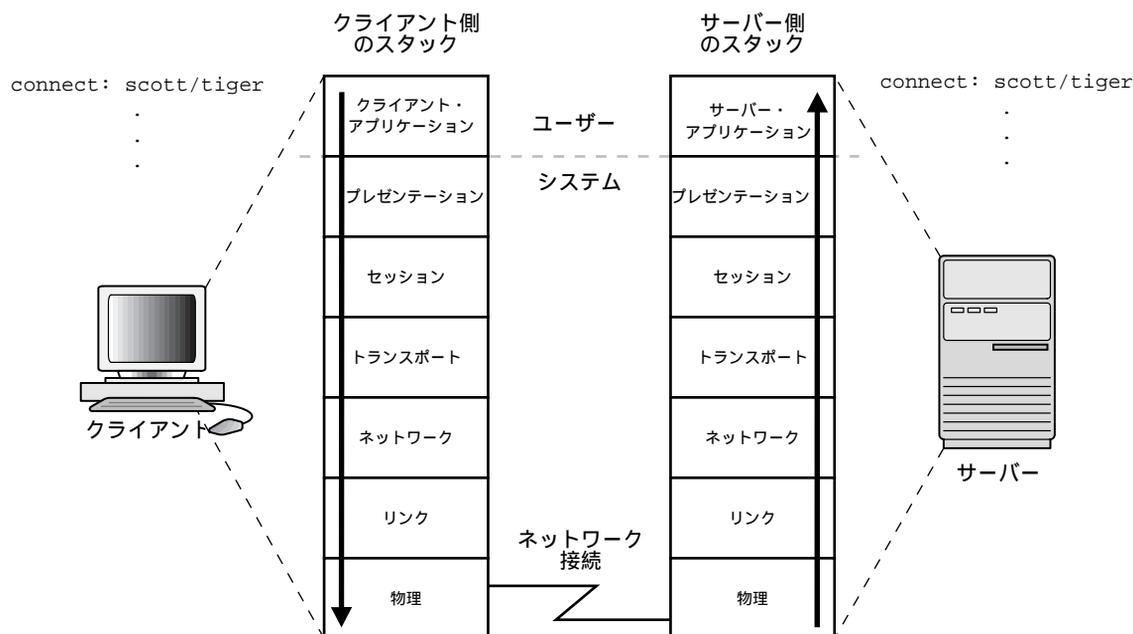
Oracle のクライアントとサーバーがどのように動作し、相互に交信するかを検討すると、TNS が何であるか、また、ネットワーク・セッションを確立するために Net8 がどのように動作するかがよく理解できます。

スタック通信

分散処理の概念は、コンピュータ相互の通信と対話を行うための設計と物理的な位置の両方で区別されているコンピュータの能力に依存しています。これは、スタック通信と呼ばれるプロセスで行われます。

スタック通信は、Open System Interconnection (OSI) 参照モデルで定義します。OSI モデルでは、いくつかのコード・レイヤーを通じてあるノードから別のノードに情報が渡されるスタックのような形式で、コンピュータ間の通信が行われます。図 2-1 は、一般的な OSI プロトコル通信スタックを示したものです。

図 2-1 OSI 通信スタック



情報はクライアント側のレイヤーを通過して降りて行き、その情報がサーバー側の対応するレイヤーで変換、解釈できるようにネットワーク・メディアを通じてトランスポート用にパッケージされます。

代表的な OSI プロトコル通信スタックには、こうしたレイヤーが7つあります。

クライアント・アプリケーション ユーザーに最も近い OSI レイヤーであるため、ユーザーが要求した機能により異なります。たとえば、データベース環境では、フォーム・アプリケーションがサーバーからのデータにアクセスするために通信を開始しようとする場合があります。

プレゼンテーション あるシステムのアプリケーション・レイヤーから送られた情報を、別のシステムのアプリケーションで読み取れるようにします。このレイヤーでは、クライアントとサーバー間で転送されたデータの構文と意味を追跡します。必要であれば、プレゼンテーション・レイヤーが共通のデータ形式を使用して、複数のデータ表現形式間で変換を行います。

セッション クライアントとサーバー間でネットワーク・セッションを確立、管理および終了します。これは、データの要求と応答を伝送する仮想パイプです。セッション・レイヤーは、データ通信を同時に両方向に送るか（非同期伝送）、一度に1つの方向のみに送るか（同期伝送）を制御します。

トランスポート データが確実に転送されるようにするデータ・トランスポートをインプリメントします。

ネットワーク 一連の相互接続サブネットワークによって、最適なパスでデータ・トランスポートが送られることを保証します。

リンク 物理リンクを通じて、信頼性の高いデータ転送を行います。

物理 クライアントとサーバーの間の物理リンクをアクティブにし、メンテナンスし、非アクティブにするための、電氣的、機械的、手順的な仕様を定義します。

Oracle ネットワーク環境内のスタック通信

スタック通信により、Oracle のクライアントとサーバーは、それらの間でデータを共有、変更、処理できます。代表的な Oracle 通信スタック内のレイヤーは、標準 OSI 通信スタックのレイヤーに類似しています。この項では、次のトピックを説明します。

- [Net8 クライアント側の対話](#)
- [サーバー側の対話](#)

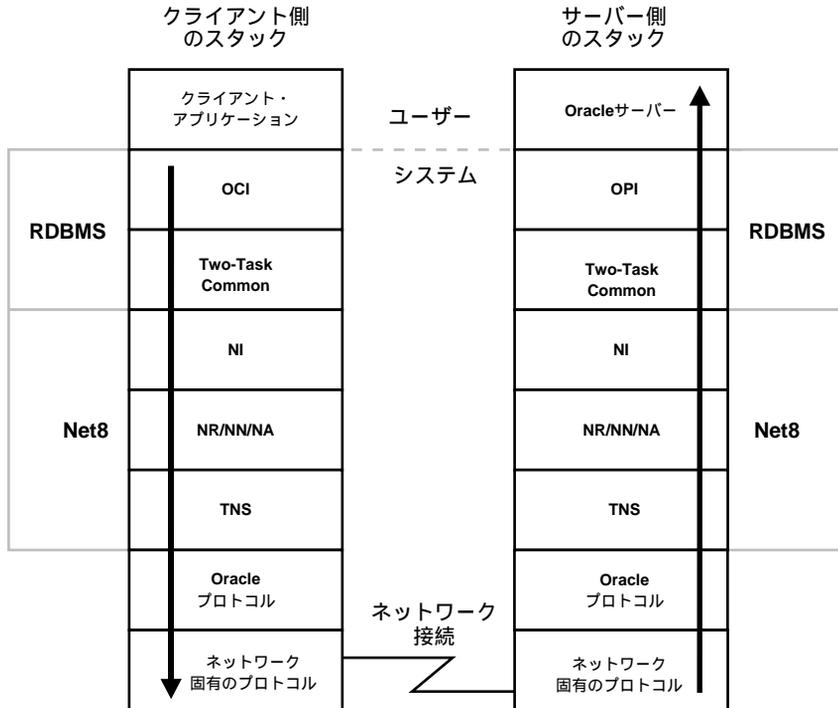
Net8 クライアント側の対話

Oracle クライアント / サーバーのトランザクションでは、情報が次のレイヤーを通過します。

- [クライアント・アプリケーション](#)
- [Oracle コール・インタフェース \(OCI\)](#)
- [Two-Task Common](#)
- [Net8](#)
- [Oracle プロトコル](#)
- [ネットワーク固有のプロトコル](#)

図 2-2 は、Oracle ネットワーク環境内の標準的な通信スタックを示したものです。

図 2-2 Oracle 環境内の標準的な通信スタック



クライアント・アプリケーション Oracle Client Applications は、文字またはグラフィック表示、スクリーン制御、データ・プレゼンテーション、アプリケーション・フロー、その他のアプリケーションの詳細など、すべてのユーザーに向けてアクティビティを提供します。アプリケーションは、サーバーに送るデータベース操作を識別し、Oracle コール・インタフェース (OCI) を通じてその操作を渡します。

Oracle コール・インタフェース (OCI) OCI コードには、クライアントとサーバーの間で SQL ダイアログを開始するために必要な情報がすべて含まれています。OCI コードは、次のことを行うサーバーに対してコールを定義します。

- SQL 文を構文解析する
- SQL 文用にカーソルをオープンする
- クライアント・アプリケーション変数をサーバー共有メモリーにバインドする
- サーバーのデータ・ディクショナリ内の値に基づいて戻されるフィールドの内容を記述する
- カーソル・メモリー領域内で SQL 文を実行する
- 1 行以上のデータをクライアント・アプリケーションに取り出す
- カーソルをクローズする

クライアント・アプリケーションは、これらのコールの組合わせを使用して、サーバー内のアクティビティを要求します。クライアント・アプリケーションの性質に応じて、OCI コールはサーバーに対する 1 つのメッセージに組み合わせたり、複数のメッセージで一度に 1 つずつ処理することができます。Oracle の製品は、多数の OCI コールを 1 つのメッセージに組み合わせ、サーバーに送信されるメッセージ数を最小限に抑えようとします。コールが実行されると、制御が Net8 に渡され、接続を確立して要求をサーバーに伝送します。

OCI の詳細は、『Oracle8i コール・インタフェース・プログラマーズ・ガイド』を参照してください。

Two-Task Common Two-Task Common は、Oracle における **プレゼンテーション**・レイヤーの実装です。Two-Task Common は、クライアントとサーバー間のキャラクタ・セットの相違または形式の相違に対して、キャラクタ・セットとデータ型の変換を行います。このレイヤーは、接続ごとに要求されたときのみ変更を行うように最適化されています。

初期の接続時に、Two Task Common は内部データとキャラクタ・セットの表現の違いを評価したり、2 つのコンピュータが通信するために変換が必要かどうかを判断します。

Net8 Net8 には、Oracle 通信スタック内のすべての**セッション**・レイヤーの機能があります。クライアント・アプリケーションとサーバーとの間のメッセージの交換をはじめ、接続の確立およびメンテナンスを行います。Net8 自体には、セッション・レイヤー機能を支援する3つのコンポーネント・レイヤーがあります。

コンポーネント	説明
ネットワーク・インタフェース (NI)	このレイヤーには、Oracle のクライアントまたはサーバー、外部プロセスが Net8 の機能にアクセスするための汎用インタフェースがある。接続のブレイク要求とリセット要求を処理する。
ネットワーク・ルーティング (NR) / ネットワーク・ネーミング (NN) / ネットワークの認証 (NA)	NR は、接続先へのネットワーク・セッションのルーティングを行う。このルーティングには、サーバー接続先までのルート上に中間の接続先またはホップを入れることができます。NN は、Net8 接続先アドレスに対するネット・サービス名を変換します。NA は、認証要件について接続先と折衝します。
Transparent Network Substrate (TNS)	業界標準の Protokol に対する共通インタフェースを提供する Net8 の基礎を形成するレイヤー。TNS は Net8 からの要求を受信し、サーバーまたは接続先の位置 (オープン、クローズの各機能) 1 つ以上の Protokol が接続に関与するかどうか (オープン、クローズの各機能)、それぞれの機能に基づいてクライアントとサーバー間の割込みを処理する方法 (送信、受信の各機能) など、一般的なマシン・レベルの接続に関する問題をすべて解決します。一般的な TNS 機能 (オープン、クローズ、送信、受信) は、Protokol 固有のコールを行うために Oracle Protokol に制御を渡します。さらに、TNS は伝送中のデータを保護するために暗号化と順次暗号メッセージ・ダイジェストをサポートします。TNS は、TNS ネットワーク・ セッション (NS) レイヤーの基本コンポーネントです。

Oracle Protokol Oracle Protokol は、Oracle における**トランスポート**・レイヤーの実装です。Oracle Protokol は、TNS の機能をクライアント / サーバー接続で使用する業界標準 Protokol にマップします。各 Protokol は、TNS と特定の Protokol との間で等価な機能をマップします。Oracle Protokol には次の Protokol が含まれます。

- LU6.2
- Named Pipes
- SPX
- TCP/IP
- SSL 付き TCP/IP

ネットワーク固有のプロトコル クライアント / サーバー接続プロセス内のすべての Oracle ソフトウェアは、2 台のマシン間の **トランスポート・レイヤー** でマシン・レベルの接続を確立するために、既存のネットワーク・プロトコル・スタックを必要とします。ネットワーク・プロトコルは、クライアント・マシンからサーバー・マシンまで、データを送るのみです。その時点で、データがサーバー側の Oracle プロトコルに渡されます。

サーバー側の対話

ネットワーク・プロトコルを通じてクライアント・アプリケーションから渡された情報は、サーバー側にある同様の通信スタックで受信されます。サーバー側のプロセス・スタックは、クライアント側で発生した内容と逆のものであり、情報は通信レイヤー内で昇順になります。サーバー側に固有な操作の 1 つは、ネットワーク・リスナーを通じて初期の接続を受信する行為です。

ネットワーク・セッション・レイヤーにある次のコンポーネントは、クライアント側のコンポーネントとは異なります。

- Oracle プログラム・インタフェース (OPI)
- Oracle サーバー

Oracle プログラム・インタフェース (OPI) OPI は、OCI の機能に対して補完的な機能を実行します。OPI は、OCI が送った各メッセージに回答します。たとえば、OCI が 25 行のデータの取出しを要求すると、OPI は取り出した 25 行のデータを OCI に戻します。

Oracle サーバー 接続の Oracle サーバー側は、クライアント OCI からのダイアログ要求を受信したり、クライアント・アプリケーションのかわりに SQL 文を解析します。要求を受信すると、要求が処理されるため、応答をフォーマットしてクライアント・アプリケーションに戻すためにデータが OPI に渡されます。

サーバー間の対話

2 台のサーバー同士が通信して分散トランザクションを処理しているときは、プロセスおよびレイヤー、ダイアログはクライアント / サーバー間の通信の場合と同じです。ただし、クライアント・アプリケーションはありません。サーバーには、ネットワーク・プログラム・インタフェース (NPI) と呼ばれるサーバー自体の OCI バージョンがあります。NPI インタフェースは、OCI がクライアントに対して実行する機能をすべて実行して、調整サーバーが追加サーバーに対する SQL 要求を構築できるようにします。

IIOP 環境内のスタック通信

このリリースの Net8 では、General Inter-Orb Protocol (GIOP) と呼ばれる新しい**プレゼンテーション**・レイヤーをサポートしています。このレイヤーは、Java オプションに接続するクライアントで使用されます。Internet Inter-Orb Protocol (IIOP) は、TCP/IP 上の GIOP または SSL 付き TCP/IP 上の GIOP の実装です。Oracle は、GIOP サービスの実装を提供します。

IIOP クライアント側の対話

図 2-3 に示すように、IIOP クライアントの通信スタックは、標準の Net8 クライアント・スタックとは異なります。IIOP クライアントは、次のように動作します。

- GIOP を **Two-Task Common** ではなく、**プレゼンテーション**・レイヤーとして使用する
- **Transparent Network Substrate (TNS)** ではなく、**非セッション**・レイヤーを使用する

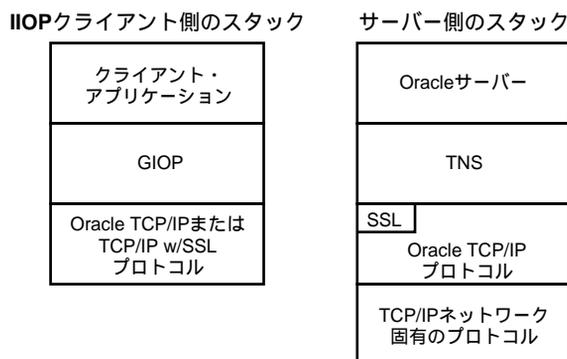
図 2-3 通常のクライアントと IIOP クライアントの通信スタック

Net8クライアント・スタック IIOPクライアント・スタック



サーバー側では、通常の Net8 接続で必要とされる多くの Net8 通信レイヤーを必要としません。かわりにサーバー側では、TCP/IP のネットワーク・プロトコルと TCP/IP の Oracle プロトコルまたは SSL 付き TCP/IP を必要とします。Net8 が必要とするコンポーネントは、Transparent Network Substrate (TNS) のみです。図 2-4 は、IIOP クライアントと Oracle データベースの Java オプション間の通信スタックを示したものです。

図 2-4 IIOP 環境におけるクライアントとサーバーの通信スタック



7-48 ページの「[Java オプション接続の構成](#)」を参照してください。

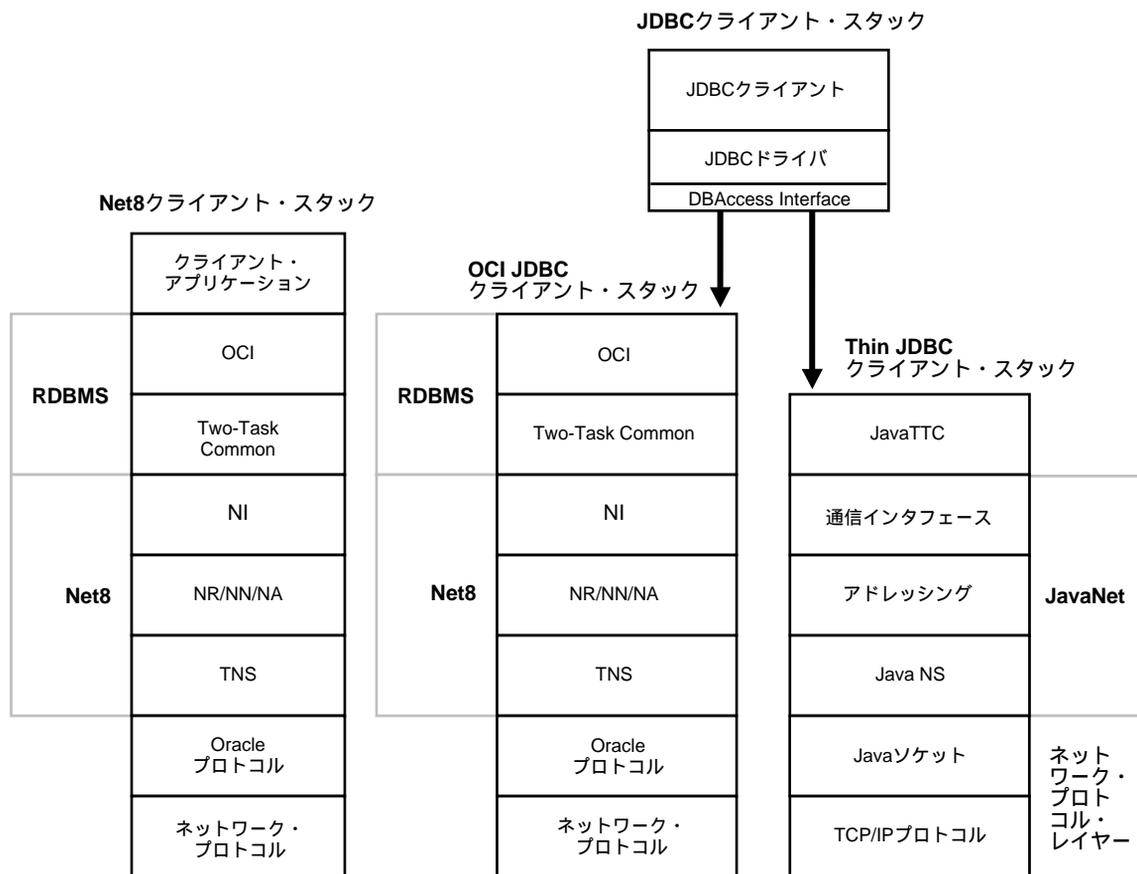
JDBC 環境内のスタック通信

Oracle の Java Database Connectivity (JDBC) Drivers を利用すると、Java プログラマは Oracle データベースにアクセスできます。Oracle では 2 つの JDBC ドライバが用意されています。1 つは、特にクライアント / サーバー型の Java アプリケーションや Java ベースの中間層 (Java Web サーバーや Java アプリケーション・サーバー) を作成している Java 開発者を対象にしたもの、もう 1 つは Java アプレットの開発者を対象としたものです。

- JDBC/OCI は、Oracle の Level 2 JDBC ドライバで、クライアント / サーバー型の Java アプリケーション・プログラマ、および Java ベースの中間層開発者を対象としています。JDBC/OCI ドライバは、JDBC の呼び出しを Oracle Call Interface (OCI) の呼び出しに変換します。変換された呼び出しは、Net8 を通して Oracle データベース・サーバーに送信されます。
- Thin JDBC ドライバは、Oracle の Type 4 ドライバで、Java アプレットおよび Java アプリケーションの開発者用に設計されています。JDBC ドライバは、Java ソケットを通して Oracle データベース・サーバーに直接接続を確立します。データベースへのアクセスは、Net8 と Two-Task Common の実装に支援されます。

JDBC クライアント通信スタック内のレイヤーは、2-17 ページの [図 2-5](#) に示すように、標準の Oracle 通信スタックのレイヤーに類似しています。

図 2-5 通常のクライアントと JDBC クライアントの通信スタック



OCI JDBC クライアント・スタックは、標準的な Net8 クライアント・スタックに類似しています。Thin JDBC クライアント・スタックには、別にインプリメントされた類似コンポーネントがあります。Java クライアント・アプレット / サーバーのトランザクションでは、情報が次のレイヤーを通過します。

- [DBAccess Interface](#)
- [JavaTTC](#)
- [JavaNet](#)
- [ネットワーク・プロトコル・レイヤー](#)

DBAccess Interface

Java DBAccess Interface レイヤーにより、標準的なクライアント通信スタックにおける **Oracle コール・インタフェース (OCI)** と同様に、クライアントは SQL セッションを開始できます。

JavaTTC

Java クライアントとデータベース間で情報を交換する際に必要な、**Two-Task Common** の実装のサブセットを提供します。JavaTTC は次の動作を行います。

- プロトコルのバージョンとデータ型を折衝する
- Java クライアントとサーバー間でキャラクタ・セットの変換が必要かどうかを決定する
- SQL 文を実行する

JavaNet

JDBC クライアントが Oracle データベースに接続できるように通信の基盤構造を提供します。Net8 の NI、NR/NN/NA および TNS の実装に相当します。

- 通信インタフェース - **JavaTTC** と他のネットワーク・レイヤー間のインタフェースを提供します。標準クライアント通信スタックにおける NI と同様に働きます。
- アドレッシング - Net8 の接続先アドレスを処理します。標準クライアント通信スタックにおける NN と同様に働きます。
- Java NS - 標準クライアント通信スタックの **Transparent Network Substrate (TNS)** と同様に、Java NS は TCP/IP に対する共通のインタフェースを提供し、サーバーまたは接続先の位置 (オープン、クローズの各機能)、1 つ以上のプロトコルが接続に関与するかどうか (オープン、クローズの各機能)、それぞれの機能に基づいてクライアントとサーバー間の割込みを処理する方法 (送信、受信の各機能) など、一般的なマシン・レベルの接続に関する問題を解決します。

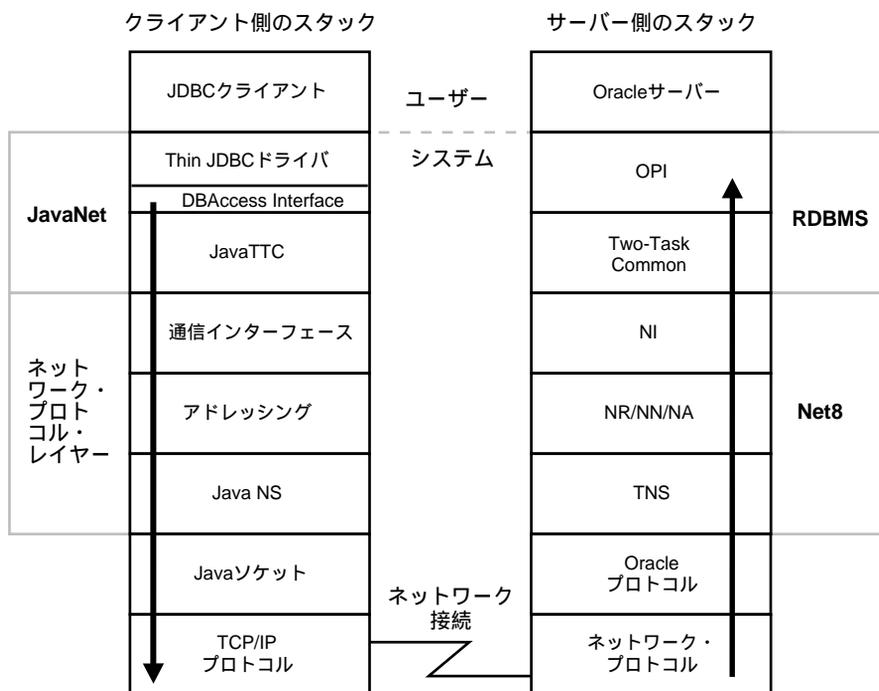
ネットワーク・プロトコル・レイヤー

Java Sockets - Java NS と TCP/IP 間の接続を提供します。

TCP/IP プロトコル - Java NS は TCP/IP を使用するため、TCP/IP 上でしか動作しません。

図 2-6 は、Java クライアント・アプレットと Oracle データベース間の通信スタックを示したものです。

図 2-6 JDBC 環境におけるクライアントとサーバーの通信スタック



マルチスレッド・サーバーへのネットワーク・セッション

注意： マルチスレッド・サーバー（MTS）を利用するには、第9章の「マルチスレッド・サーバーの構成」で説明するように構成が必要です。

マルチスレッド・サーバー（MTS）は共有サーバーとも呼ばれ、多数のユーザー・プロセスがごくわずかなサーバー・プロセスを共有することができます。

MTS の構成により、多数のユーザー・プロセスがディスパッチャに接続できます。ディスパッチャは、クライアントごとに専用サーバーを必要とせず、多数のクライアントが同じサーバーに接続できるようにします。ディスパッチャは、複数の着信ネットワーク・セッション要求を処理して、共有サーバーに送ります。MTS の利点は、システムのオーバーヘッドを減らし、使用するリソース量を削減しながら、サポートするユーザーの数を増加させられることです。

MTS を使用しない場合、各ユーザー・プロセスはそれぞれ専用のサーバー・プロセスを必要とします。接続を要求する各クライアントに対して、新しいサーバー・プロセスが作成されます。専用のサーバーは、接続が続く間、ユーザー・プロセスに関連付けられたままになります。

Oracle サーバーが MTS として構成されていると、ネットワーク・セッションが専用サーバーを特に要求している場合、またはディスパッチャが使用できない場合を除き、着信ネットワーク・セッションは必ずディスパッチャに送られます。ディスパッチャ・サーバーで発生するイベントの順序は、次のとおりです。

1. リスナーが起動され、デフォルト・アドレス、またはリスナー構成ファイルに指定したアドレスでリスニングを行います。
2. データベース・インスタンスが起動します。ディスパッチャは、初期化パラメータ・ファイル内の構成パラメータに従って起動します。次に、各ディスパッチャは割り当てられたアドレスでリスニングを行います。
3. 各ディスパッチャのアドレスがリスナーに登録されます。リスナーがデフォルトのアドレスでリスニングしない場合は、リスナーのネットワーク名をデータベース初期化ファイル（`INITSID.ORA`）内で指定できます。複数のリスナーを使用している場合は、このようなアドレスが2つ以上存在することもあります。

ディスパッチャ・アドレスを登録すると、リスナーはそれらのアドレスに着信接続要求をリダイレクトできます。

手順1の前に手順2が実行されると、サーバーは手順3でリスナーに接続できません。その場合、サーバーがリスナーに接続しようとして、遅延が発生することがあります。接続要求があるタイムフレーム内に入り、その中でディスパッチャが登録されていない場合、これらの要求は、事前に生成された専用サーバーまたは新たに生成された専用サーバーによって処理されるか、あるいは拒否されます。

これで、リスナーと Oracle ディスパッチャ・サーバーがいつでも着信ネットワーク・セッションを受信できる状態になりました。

注意： リスナー制御ユーティリティで SERVICES コマンドを発行すると、どのディスパッチャがリスナーに登録されたかをチェックできます。詳細は、A-10 ページの「SERVICES」を参照してください。

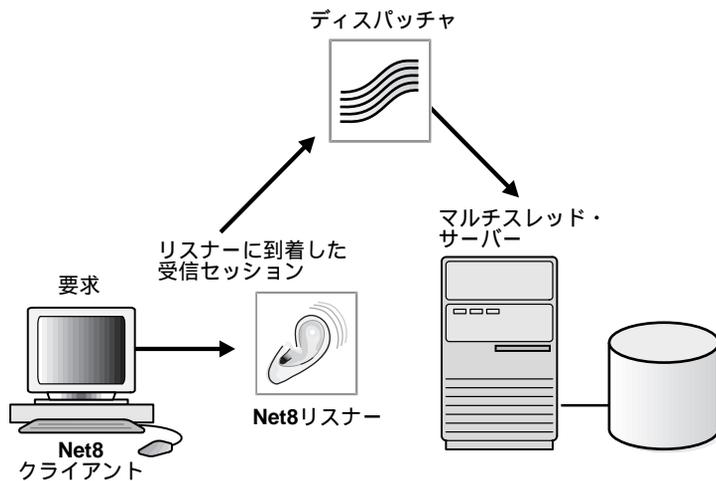
リスナーとディスパッチャがすでに起動されている場合、ネットワーク・セッション・アクティビティは次のように続行されます。

1. クライアントがネットワーク・アドレスでリスナーに接続します。
2. リスナーが接続要求を受け取り、クライアントの要求を処理できるかどうかを判別します。要求を処理できない場合、リスナーはネットワーク・セッションを拒否した後、手順 6 から再開します。
3. クライアントの要求が確認されると、リスナーは共有サーバーに対して最も使用頻度の低いディスパッチャのネットワーク・アドレスを含むクライアントに、リダイレクト・メッセージを発行します。Oracle Parallel Server では、メッセージは最もロード頻度の低いノードの、最も使用頻度の低いディスパッチャにリダイレクトされます。
4. クライアントはリスナーに要求したネットワーク・セッションを取り消し、リダイレクト・メッセージで提供されたネットワーク・アドレスを使用して、共有サーバーとのネットワーク・セッションを確立します。
5. 新しいネットワーク・セッションが確立されたため、ディスパッチャは新しい負荷値でリスナーを更新します。これによって、リスナーは同じプロトコルで動作しているディスパッチャ間で、着信ネットワーク・セッション要求を適切に振り分けることができます。
6. リスナーが、着信ネットワーク・セッションのリスニングを再開します。

クライアントが接続を切断すると、クライアントと関連付けられた共有サーバーはアクティブのまま、他の着信要求を処理します。同じクライアントからの別の要求は、他の共有サーバーで処理される場合もあります。

図 2-7 は、ディスパッチャにリダイレクトされた接続におけるリスナーの役割を示したものです。

図 2-7 ディスパッチャにリダイレクトされた接続



MTS の構成情報の詳細は、[第 9 章の「マルチスレッド・サーバーの構成」](#)を参照してください。

専用サーバーへのネットワーク・セッション

MTS のかわりに、Net8 が既存の専用サーバーに要求をリダイレクトする場合もあります。これがデフォルトの実装です。Net8 は、既存のサーバー・プロセスのアドレスをクライアントに送り返します。続いて、クライアントは指定されたサーバー・アドレスに接続要求を再送します。

Net8 には、要求を受け取る前に、専用サーバーを自動作成するオプションがあります。これらのプロセスはリスナーが稼働している間存在しており、それ以降の接続要求で再使用できます。事前に生成された専用サーバー・プロセスを使用するには、リスナー構成ファイルで指定する必要があります。

注意： 事前に生成された専用サーバーには、SQL*Net バージョン 2.1 以降、および Oracle Server リリース 7.1 以降が必要です。

事前に生成された専用サーバー・プロセスを使用してクライアント接続要求をサービスするときに発生するイベントの順序は、次のとおりです。

1. リスナーが起動され、リスナー構成ファイルで指定したアドレスで要求を待ちます。
2. 次に、リスナーは構成ファイルで定義されているプール・サイズに達するまで、一連の専用サーバー・プロセスを生成します。
3. 生成された各サーバー・プロセスは部分アドレス待機を実行して、要求を待機している部分アドレスをリスナーに知らせます。リスナーはこの時点で、事前に生成されたサーバーすべてをアイドルとしてマークします。

注意： 部分アドレス・リスニングでは、サーバー・プロセスは接続要求を待っていますが、リスニングするための特定のアドレスに関する優先権がないことを、基礎を形成するプロトコル・スタックに伝えます。このため、多くのプロトコル・スタックは空いているリスニング・アドレスを選択し、要求側のサーバー・プロセスにそのアドレスを自動的に割り当てます。

4. クライアントが接続要求をリスナーに送ります。
5. リスナーはネットワーク・セッション要求を受け取って、クライアントの要求を処理できるかどうかを判別します。要求を処理できない場合、リスナーはネットワーク・セッションを拒否した後、手順 9 から再開します。
6. リスナーは、事前に生成されたサーバーのネットワーク・アドレスのあるクライアントに、リダイレクト・メッセージを発行します。リスナーはそのサーバーをアクティブ・サーバーとして記録します。

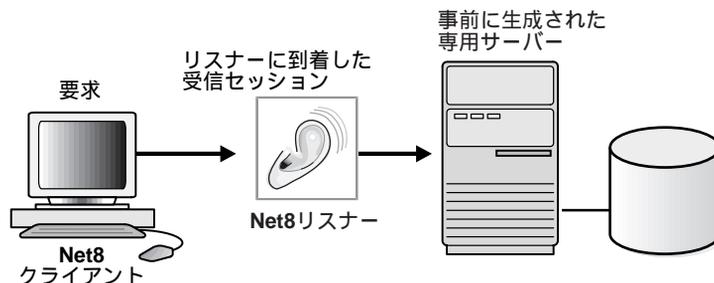
7. クライアントはリスナーに要求したネットワーク・セッションを取り消し、リダイレクト・メッセージで提供されたアドレスを使用して、事前に生成されたサーバーとのネットワーク・セッションを確立します。
8. リスナーが、アクティブな事前に生成されたサーバーと置き換えるための別のサーバー・プロセスを生成します（ただし、リスナー構成ファイルにある `PRESPAWN_MAX` の値がアクティブで、アイドル状態の事前に生成されたサーバー・プロセス数よりも大きい場合）。
9. リスナーは着信ネットワーク・セッションのリスニングを続行します。

事前に生成された数の上限に達するまで、上記のイベントが続けられます。上限に達すると、リスナーは新しい専用サーバー・プロセスを生成しません。

クライアントが接続を切断すると、クライアントと関連付けられた事前に生成された専用サーバー・プロセスがアイドル・プールに戻ります。次に、このサーバー・プロセスは、別のクライアントに割り当てられるまでの指定時間だけ待機します。事前に生成されたサーバーにクライアントが渡されずにタイムアウトが満了すると、事前に生成されたサーバーが停止します。

図 2-8 は、事前に生成された専用サーバー・プロセスにリダイレクトされた接続における、リスナーの役割を示したものです。

図 2-8 事前に生成された専用サーバー・プロセスにリダイレクトされた接続



専用サーバーへの Bequeathed ネットワーク・セッション

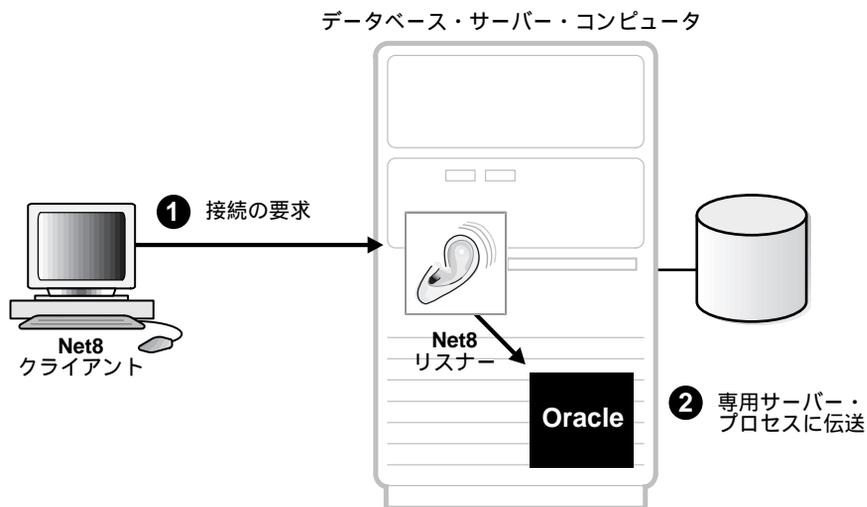
リスナーとサーバーが同じノード上に存在する場合、リスナーは、接続要求の受信時に専用サーバーを作成または起動できます。専用サーバーは1つのネットワーク・セッションのみにコミットされ、そのネットワーク・セッションを実行している間存在します。リスナーが専用サーバー・プロセスを作成し、そのプロセスにネットワーク・セッションの制御を渡す、または bequeath するときに発生するイベントの順序は、次のとおりです。

1. リスナーが起動され、リスナー構成ファイル (LISTENER.ORA) で指定したアドレスでリスニングを行います。
2. クライアントがネットワーク・アドレスでリスナーに接続します。
3. リスナーがネットワーク・セッション要求を受け取って、クライアントの要求を処理できるかどうか判別します。要求を処理できない場合、リスナーはネットワーク・セッションを拒否した後、手順 5 から再開します。
4. リスナーは、着信ネットワーク・セッションを処理するための専用サーバー・プロセスを新たに起動して、ネットワーク・セッションをそのサーバー・プロセスに渡します。ネットワーク・セッションが確立されると、クライアントと専用サーバー・プロセスの間にデータが直接流れます。
5. リスナーは着信ネットワーク・セッションのリスニングを続行します。

クライアントが接続を切断すると、そのクライアントと関連付けられた専用サーバー・プロセスがクローズします。

図 2-9 は、専用サーバー・プロセスに bequeath 接続におけるリスナーの役割を示したものです。

図 2-9 専用サーバー・プロセスへの bequeath 接続



Net8 とリスナー

リスナーは、接続要求をクライアント・アプリケーションにかわって受け取ります。

この項では、次のトピックを説明します。

- [リスナーへのデータベース・インスタンス登録](#)
- [クライアント接続](#)
- [Connect-Time Failover](#)
- [ロード・バランシング](#)

リスナーへのデータベース・インスタンス登録

データベース・インスタンスは、起動時に自身をリスナーに登録します。データベース・インスタンス登録は、2つの要素からなります。

- サービス登録。リスナーにデータベース・サービス名やインスタンス名などのインスタンス情報を提供します。
- MTS ディスパッチャの登録。リスナーにディスパッチャ情報を提供します。

インスタンスが起動されるときに、初期化パラメータが `INITSID.ORA` ファイルから読み込まれます。この初期化パラメータの中に、サービス名が含まれます。デフォルトでは、インスタンスのバックグラウンド・プロセスは、インスタンス情報をローカル・マシンのリスナーに登録します。

リスナーがインスタンス後に起動される場合、インスタンスとディスパッチャが登録されるまでに遅延が発生する場合があります。インスタンスは、定期的にリスナーに接続を試みます。同様に、インスタンスが登録される前にリスナーが着信要求を受け取ると、リスナーは要求を拒否します。

MTS の構成情報の詳細は、[第9章の「マルチスレッド・サーバーの構成」](#)を参照してください。

利点

データベース・インスタンス登録には次の利点があります。

- `LISTENER.ORA` ファイルが、`SID_LIST_listern_name` パラメータを必要としません。このパラメータは、リスナーが提供するデータベースの情報を指定します。使用している管理ツールが必要な場合は、このパラメータが必要です。
- `connect-time failover` が有効になります。詳細は 2-29 ページの「[Connect-Time Failover](#)」を参照してください。
- 接続負荷のバランスが有効になります。詳細は 2-30 ページの「[ロード・バランシング](#)」を参照してください。

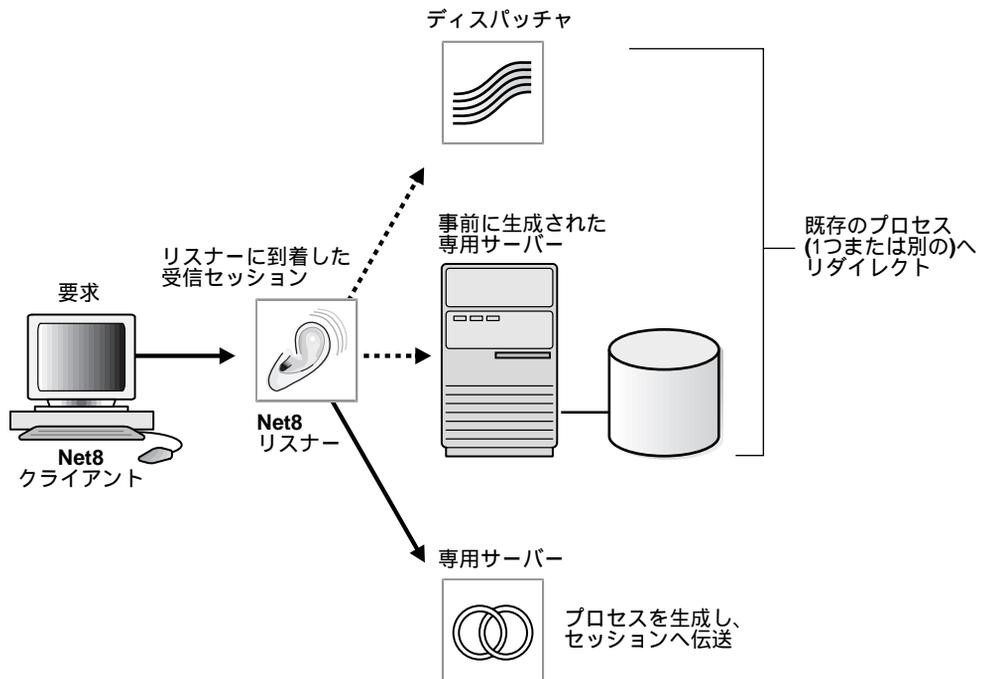
クライアント接続

クライアントは、ネット・サービス名で構成されます。ネット・サービス名には、リスナーの場所と、接続するサービスのサービス名が含まれます。

クライアントがリスナーからの接続を要求すると、サービス名がリスナーに送られます。リスナーはネットワーク・セッション要求を受け取り、渡された情報が登録されている情報と一致するか判定し、クライアントの要求を処理できるかどうかを判別します。

図 2-10 は、サーバーへの代表的な Net8 接続におけるリスナーの役割を示したものです。

図 2-10 代表的な Net8 接続におけるリスナー



Connect-Time Failover

データベース・インスタンス登録により、リスナーは、インスタンスが接続の前に準備できているか確認できます。この機能により、別のリスナーに対するクライアント接続要求のフェイルオーバーが自動化されます。

クライアントの接続動作を管理するには、各ネット・サービス名に対して複数のリスニング・アドレスを構成し、フェイルオーバー・パラメータ (FAILOVER=ON) を使用します。

connect-time failover は、クライアントが正常にリスナーに接続するまで継続されます。

注意： connect-time failover をインプリメントしても、LISTENER.ORA ファイルの静的サービス構成パラメータは使用できません。ただし、Oracle8 リリース 8.0 または Oracle7 データベース、および Oracle Enterprise Manager では、静的な構成が必要です。リスナーの静的な構成の詳細は、6-71 ページの「[サービス情報によるリスナーの構成](#)」を参照してください。

Oracle Enterprise Manager 環境で、Oracle Parallel Server の connect-time failover を設定する際の詳細な構成手順は、『Oracle8i Parallel Server セットアップおよび構成ガイド』を参照してください。

connect-time failover は、Oracle Parallel Server などの可用性の高い環境における実行時フェイルオーバーである " 透過的アプリケーション・フェイルオーバー " とは異なります。アプリケーションとサービスの、接続の失敗と再確立を参照します。透過的アプリケーション・フェイルオーバーの詳細は、『Oracle8i チューニング』を参照してください。

構成情報は、7-11 ページの「[複数アドレス・オプションの構成](#)」を参照してください。

ロード・バランシング

ロード・バランシングは、クライアント接続を複数のリスナー、ディスパッチャ、インスタンスおよびノードに均等に分散し、1つのコンポーネントに負荷がかかりすぎないようにする機能です。

ロード・バランシングは、次のレベルで行われます。

- [クライアント負荷のバランス](#)
- [接続負荷のバランス](#)

クライアント負荷のバランス

複数のリスナーが1つのデータベースをサービスする場合、クライアントは接続要求を送るリスナーをランダムに選択できます。このランダム化によって、すべてのリスナーが着信接続要求をサービスする負荷を共有できます。

クライアントがランダムにリスナーを選択できるようにするには、ネット・サービス名ごとに複数のリスニング・アドレスを構成し、ロード・バランスのパラメータ (LOAD_BALANCE=ON) を使用する必要があります。

詳細な構成情報は、7-11 ページの「[複数アドレス・オプションの構成](#)」を参照してください。

接続負荷のバランス

データベース・インスタンス登録を使用すると、リモート・リスナーにより接続負荷のバランスが有効になります。接続負荷のバランスは、同じサービスに対するさまざまなインスタンスやディスパッチャ間のアクティブな接続を均等に分散します。インスタンスとディスパッチャの負荷は、接続の数によって決定されます。

接続負荷のバランスは、MTS 環境のみで有効になります。クライアントでは構成できません。

リスナーは、特定のサービスに対する着信クライアント要求を、最も負荷の低いディスパッチャおよびインスタンスに送ります。たとえば、複数のノードに複数のインスタンスがあるサービスでは、最もロードされる頻度の低いインスタンスに基づいてディスパッチャが選択されます。インスタンスのロードは、ノードのロードに基づきます。

Oracle Names アーキテクチャ

Oracle Names は、グローバルなクライアント / サーバー・コンピューティング・ネットワークを簡単に設定、管理できるように開発された Oracle 環境用分散ネーム・サービスです。

この項では、次のトピックを説明します。

- [概要](#)
- [管理リージョン](#)
- [ドメイン](#)
- [データ・リポジトリとしての Oracle Names](#)
- [Oracle Names Server に格納されるデータ](#)
- [ネットワーク・コンポーネントの編成とネーミング](#)
- [管理責任を分散する複数のリージョンの使用](#)
- [複数の管理リージョンをもつネットワークは、1つのルート管理リージョンと1つ以上の委任管理リージョンで構成されます。](#)
- [Oracle Names のバージョンによる違い](#)
- [検出の理解](#)

概要

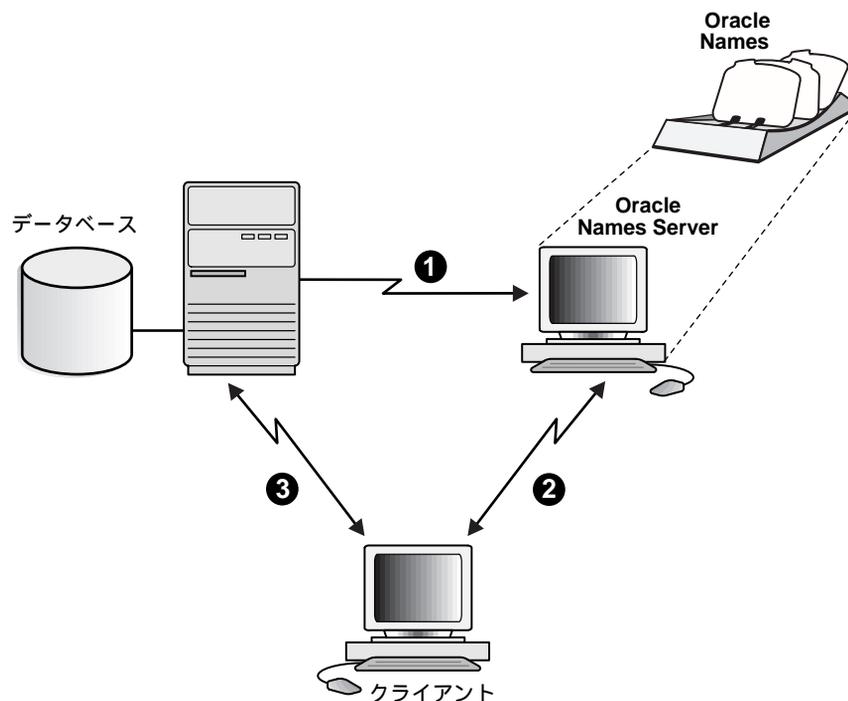
Oracle Names は、Oracle Names Server を設定し、メンテナンスします。ネーム・サーバーとは、ネットワーク上のすべてのサービスに関するアドレスを格納し、接続を行うクライアントがそのアドレスを使用できるようにするディレクトリ・サービスに似た機能を果たす統合システムです。

電話番号を検索するためにディレクトリ支援を使用する電話発信者のように、Oracle Names を使用して構成されたクライアントは、Oracle Names Server に対して接続要求の参照を行います。Oracle Names Server は、クライアントから示されたサービス名をネットワーク・アドレスに変換しようとしています。Oracle Names Server は、ネットワーク・アドレスを検出すると、その情報をクライアントに戻します。こうしてクライアントはサービスに接続するためにそのアドレスを使用できます。

図 2-11 は、クライアントとサーバー間の接続の確立を支援する Oracle Names の働きを示したものです。

1. データベースが Oracle Names Server にサービスを登録します。
2. クライアントが Oracle Names Server に接続してネットワーク・アドレスを取り出し、ネットワーク上のサービスを検索します。
3. 続いて、クライアントが透過的にサービスに接続します。

図 2-11 Oracle Names



Oracle Names には、ネット・サービス名とアドレスを各クライアントで個別に設定およびメンテナンスしなければならないファイル・ベースまたはローカル・ネーミングの変換方法にかわるものです。この情報を中央で管理してメンテナンスすることにより、Oracle Names はサービスの追加または再配置に関連する作業量を削減します。

構成情報は、6-21 ページの「Oracle Names ネーミング・メソッドによるネットワークの構成」を参照してください。

管理リージョン

Oracle Names は、1 つ以上の管理リージョンをサポートします。

大半のネットワークには、1 つの管理中心点、すなわち、1 つの管理リージョンがあります。管理リージョンは、ネットワークのサービスを管理する Oracle Names Server の集まりです。接続情報はすべて、単一のデータ・リポジトリに格納されます。データ・リポジトリには、サービス名を解釈する認証機能があります。管理リージョン内のすべての Oracle Names Server は、このデータ・リポジトリから情報を問い合わせます。管理リージョンが格納にデータベースを使用する場合、1 つの管理リージョンに対して 1 つのデータベースが存在します。Oracle Names Server の数に制限はありません。

複数のデータ・センターと多数の Oracle インスタンスが存在する大規模な企業環境では、複数管理リージョンの利用が選択されています。これにより、各データ・センターは、そのセンター自体の環境でサービスを独自に定義、管理できます。同時に、すべてのサービス・アドレスは、環境全体のすべてのクライアントに対して継続的に使用可能となります。Oracle Names Server は、外部の管理リージョンにあるクライアントから適切な Oracle Names Server に対して、ネーム変換要求を透過的に送ります。

ドメイン

管理リージョンには 1 つ以上のドメインがあり、これにより管理責任が分割されます。

ドメインは、マシンとネットワーク・サービスの論理グループです。ドメインは、ネットワーク内のサービスを識別するための方法です。ファイル・システムのディレクトリ構造と同様に、ネットワーク・サービスに固有の論理名を割り当てます。各ドメインの中では、すべての名前が一意である必要があります。ただし、異なるドメイン間では名前が重複してもかまいません。

ネットワーク・ドメインは、階層的なものであるという点で、多くのオペレーティング・システムで使用されているファイル・ディレクトリに似ています。ただし、ファイル・システムと違い、ネットワーク・ドメインはデータベースの物理的配置またはネットワーク内の他のオブジェクトに対応している場合としていない場合があります。ネットワーク・ドメインは、ネーム・スペースの矛盾を防ぐために開発されたネーム・スペースにすぎません。

注意： Oracle のネットワークのドメインと似ているように見えますが、Domain Name Service (DNS) のネーム・スペースとは完全に独立しています。便宜的に、Oracle ネットワーク内に DNS ディレクトリ構造を反映するように選択できます。

データ・リポジトリとしての Oracle Names

Oracle Names Server のデータは、リージョン内の全 Oracle Names Server 間にある継続レプリケーションを通じて、または共通の Oracle データベースで書込み / 読込みを行うことによって更新されます。

全サービスが動的に登録されている比較的小規模なワークグループ環境では、管理者が個々の Oracle Names Server すべてに対しデータを継続的にレプリケートする Oracle Names Server を構成できます。リスナーが新しいサービスを登録すると、そのサービスに関する情報が管理リージョン内のその他の Oracle Names Server に即座に渡されます。

一方、大規模な環境を管理する場合は、リージョン・データベースと呼ばれる単一の Oracle データベース内に登録データを格納することが望まれます。リージョン・データベースは、Oracle Names 情報を格納するテーブルで構成されます。Oracle データベースをリポジトリとして使用するように Oracle Names Server が構成されている場合、すべてのサービス登録がそのデータベースに書き込まれます。管理リージョン内にある所定の Oracle Names Server は、更新された登録に関して定期的にそのデータベースをポーリングします。この方法によって、所定の管理リージョン内にある全 Oracle Names Server に、新しい登録が適切な時期に伝達されます。同時に、Oracle Name サーバー同士が直接交信する必要がなくなり、高い信頼性が確保されます。

Oracle Names Server に格納されるデータ

Oracle Names Server に格納されるデータの型を次に示します。

Data	説明
グローバル・データベースの名前とアドレス	Oracle Names Server は、グローバル・データベース名（データベースの名前とドメイン）とアドレスを含むデータベースに関する情報を、リスナーから取り出す。アドレスは LISTENER.ORA ファイルで構成されます。グローバル・データベース名はデータベースの起動時に登録されるか、LISTENER.ORA ファイルで静的に構成されます。この情報を登録する必要はありません。
その他の Oracle Names Server の名前とアドレス	Oracle Names Server は、同じ管理リージョンにある他の Oracle Names Server のすべての名前とアドレスを格納する。ネットワーク内に複数の管理リージョンが存在する場合、Oracle Names Server には、ルート管理リージョンとその直属の各サブリージョンに存在する 1 つ以上の Oracle Names Server の名前とアドレスが格納される。この情報を登録する必要はありません。
ネット・サービス名	NAMESCTL 制御ユーティリティや Oracle Net8 Assistant を使用してネット・サービス名を登録すると、Oracle Names Server はこれらの名前を格納する。また、非 Oracle データベースと Oracle RDB データベースへのゲートウェイも格納される。

Data	説明
グローバル・データベース・リンク	<p>データベース・リンクにより、データベース間の通信が可能になる。データベース・リンクの名前は、リンクが指すデータベースのグローバル・データベース名と同じです。一般的には、1つのデータベースに対して1つのデータベース・リンクが存在します。</p> <p>次のタイプのデータベース・リンクが作成されます。</p> <ul style="list-style-type: none"> ■ 特定のデータベースのスキーマに、プライベート・データベース・リンクを作成できます。使用できるのは、プライベート・データベース・リンクの所有者のみです。 ■ データベースに対して、パブリック・データベース・リンクを作成できます。データベースのすべてのユーザーが使用できます。 ■ Oracle Names Server にグローバル・データベース・リンクを作成できます。ネットワーク上の誰もが使用できます。 <p>Oracle Names はグローバル・データベース名をリスナーから取り出すので、グローバル・データベース名であるグローバル・データベース・リンクは、自動的に Oracle Names Server に登録されます。したがって、この情報を登録する必要はありません。</p> <p>グローバル・データベース・リンクのユーザー名とパスワードの資格証明は、Oracle Net8 Assistant を使用して Oracle Names Server に登録されます。これらのグローバル・データベース・リンクは、Oracle Net8 Assistant で定義されたリンク修飾子で補足されます。</p> <p>個々のユーザーが作成したプライベートおよびパブリック・データベース・リンクで、グローバル・データベース・リンクを差し替えることができます。プライベート・データベース・リンクおよびパブリック・データベース・リンクの詳細は、『Oracle8i 分散システム』を参照してください。</p>
別名	<p>Oracle Names Server には、定義済みのネット・サービス名、データベース・サービスまたはグローバル・データベース・リンクに対して、別名または代替サービス名が格納される。別名は、NAMESCTL 制御ユーティリティか Oracle Net8 Assistant を使用して Oracle Names Server に登録されます。</p>
Oracle Connection Manager の情報	<p>Oracle Names Server には、ネットワーク上の全 Oracle Connection Manager の名前とリスニング・アドレスが格納される。この情報を登録する必要はありません。</p>

ネットワーク・コンポーネントの編成とネーミング

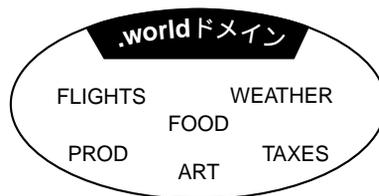
Oracle Names を使用する場合、ネットワーク化された環境にあるデータベースなどのオブジェクトには、ネットワーク内で名前が重複しないような方法でネーミングする必要があります。ネットワーク内のオブジェクトをネーミングする基本モデルは2つあります。

- シングル・ドメイン・モデル
- 階層ネーミング・モデル

シングル・ドメイン・モデル

ネットワークが小規模で名前の重複がない場合は、シングル・ドメイン・ネーミング・モデルを使用すると有効です。図 2-12 は、単一のドメイン名 .WORLD を使用した、一般的なフラット・ネーミング構造を示したものです。

図 2-12 シングル・ドメイン・ネーミング・モデル



この環境では、データベース・サービス名に拡張子 .WORLD が自動的に付けられます（たとえば、PROD.WORLD、FLIGHTS.WORLD など）。

階層ネーミング・モデル

階層ネーミング・モデルは、名前を階層構造に分割して、ネットワークの規模拡張に備えたり、名前の独立性を高めることができます。このタイプのネーミング・モデルでは、異なるドメインに同じ名前の複数のデータベースをもつネットワークを実現できます。

図 2-13 は、(ROOT)ドメイン、ACME ドメイン、US.ACME ドメイン、EUROPE.ACME ドメイン、ROW.ACME (Rest of World) ドメインで構成されるドメインの階層構造を示したものです。

図 2-13 階層ネーミング・モデル

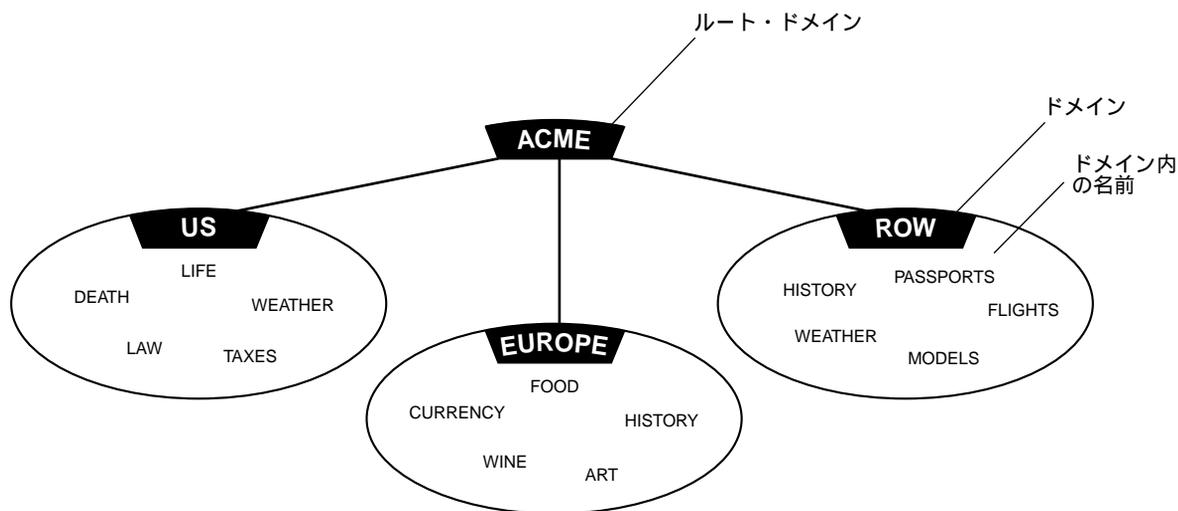
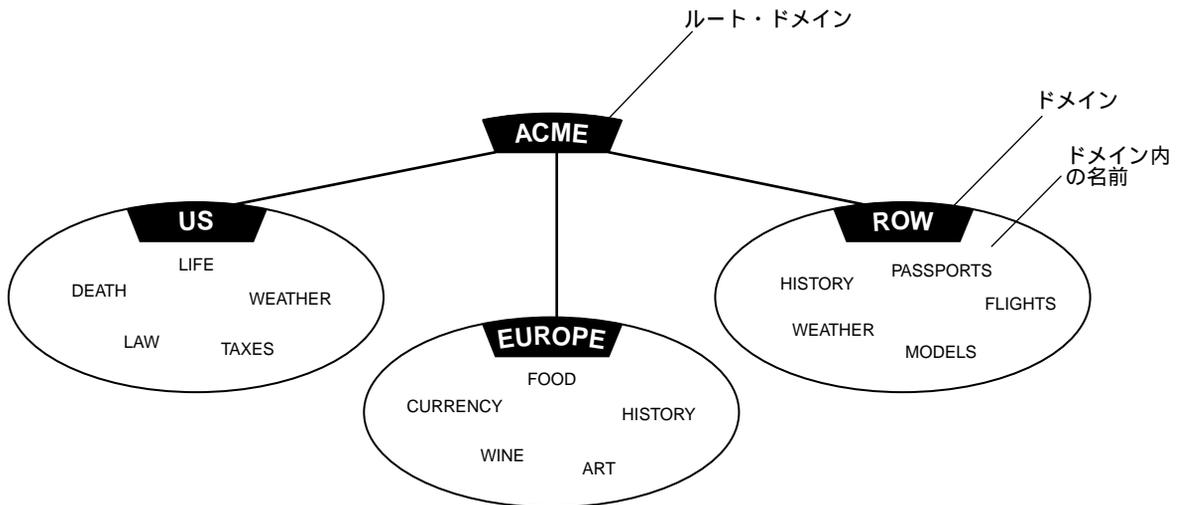


図 2-13 で、WEATHER と HISTORY はどちらも 2 つありますが、名前は一意です（すなわち、HISTORY.ROW.ACME と HISTORY.EUROPE.ACME）。

デフォルト・ドメイン デフォルト・ドメインは、クライアントの名前要求の大半を処理するドメインです。通常、これはクライアントが存在するドメインですが、クライアントが最も頻繁にサービスを要求する別のドメインを指定することもできます。クライアントがデフォルト・ドメイン内のネットワーク・サービスを要求するときは、サービスの単純な未修飾名を使用して、つまりドメイン名を指定しないでサービスを要求できます。文字 "." を含まない名前をユーザーが要求すると、要求したデータベース・サービス名またはデータベース・リンク名に、デフォルト・ドメイン名が自動的に追加されます。

図 2-14 は、クライアントとデフォルト・ドメインの EUROPE.ACME.COM を示したものです。クライアントがサービス名 "WINE" を要求すると、要求した名前にデフォルト・ドメインが付加されて、WINE.EUROPE.ACME.COM となります。

図 2-14 デフォルト・ドメイン



ドメイン名の詳細は、『Oracle8i 概要』を参照してください。

複数のドメイン 複数のドメインは、一連の親子関係で、ルート・ドメイン（階層内の最高レベルのドメイン）に階層的に関連付けられています。たとえば、ルートの下にはいくつかのドメインがあり、その1つをCOMと呼びます。COMドメインの下にはいくつかのドメインがあり、その1つをACMEと呼びます。ACMEドメインの下には、いくつかのドメイン（USやEUROPEなど）があります。

注意： SQL*Net と Oracle Names の旧リリースでは、ドメインを1つしか持たないネットワークは、デフォルトで .world と呼ばれていました。これは Net8 と Oracle Names バージョン 8 の要件ではありませんが、旧リリースからのデータベースをすべて改名することを避け、後続互換性を維持するために同じ規約を保つことができます。

管理責任を分散する複数のリージョンの使用

管理リージョンには、複数のドメインを含めることができます。完全修飾ドメイン名は、1つの管理リージョンにしか存在できませんが、ドメイン名は2つの異なるリージョンに含めることができます。ネーミング規則を柔軟にするには、リージョンの管理責任を分散させる必要があります。

Oracle Names を使用し、ネットワークが大規模または地理的に広く分散している場合は、分割した複数のリージョンをさらに分割することもできます。たとえば、企業全体のネットワークにアメリカとヨーロッパの両方が含まれている場合、ローカルに構築されているネットワークに関する管理上の決定を行う場合があります。リージョンを再分割するには、親から子またはサブリージョンに、リージョンとドメインを委任する必要があります。

管理リージョンを委任するには、それぞれの管理リージョンが1つ以上の異なるドメインを制御する必要があるため、階層ネーミング・モデルの適用が必要です。

複数の管理リージョンをもつネットワークは、1つのルート管理リージョンと1つ以上の委任管理リージョンで構成されます。

ルート管理リージョン

ルート管理リージョンにはルート・ドメインが含まれます。ルート管理リージョンには次の情報が含まれます。

- ルート・リージョン内の Oracle Names Server。
- このリージョン内で要件が管理されるときドメイン。ルート・ドメインが必ず含まれ、他のドメインが含まれる場合もあります。
- 委任管理リージョンの Oracle Names Server - ルートの直属の子リージョンの役割を果たす代替リージョン内のドメインと Oracle Names Server のアドレス。
- ルート・リージョンのデータ定義 - ルート管理リージョンに関連するすべてのデータベース・サービス名およびデータベース・リンク、別名。

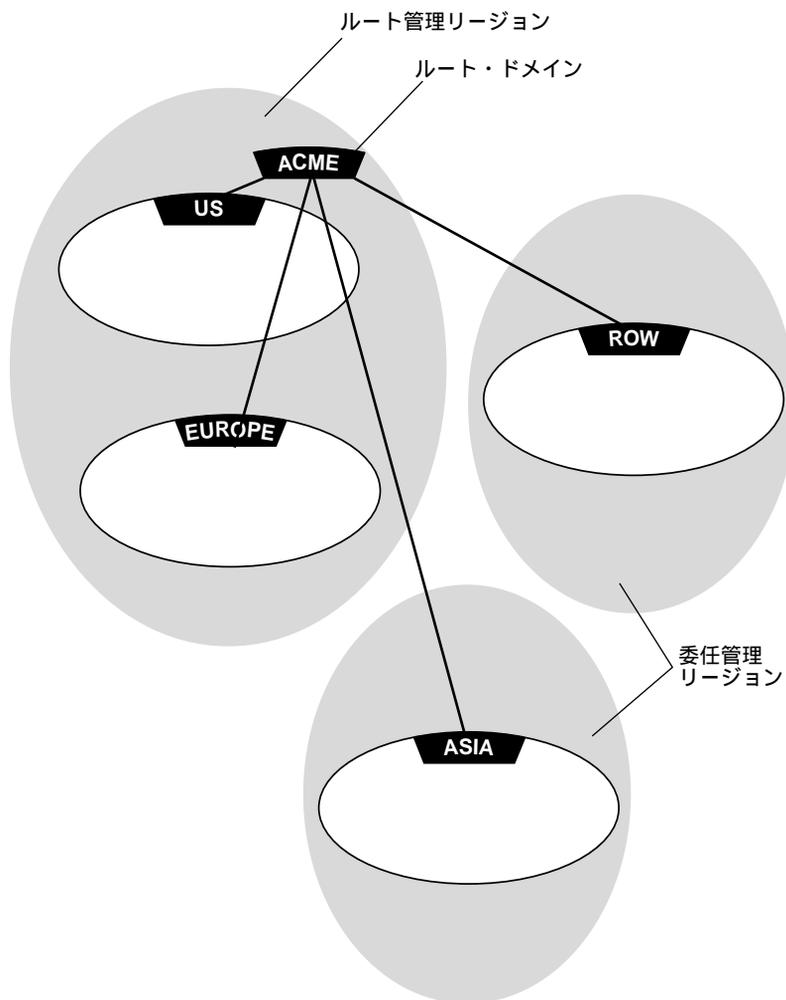
委任管理リージョン

管理リージョンは、ネーミング・モデルの階層の最高位からネーミング・モデルのその他の下位ドメインに至るまで委任できます。たとえば、ドメイン数が10のネットワークでは、1～10の管理リージョンで構成できます。

ルート以外の管理リージョンはすべて、階層構造内でルートと直接または間接的に連結している委任リージョンです。

図 2-15 は、5つのドメインと3つの管理リージョンのあるネットワークを示したものです。管理リージョンは、ROOTと2つの委任リージョン（DR1、DR2）からなります。

図 2-15 委任管理リージョン



ルートの下の委任管理リージョン

ルートの下の管理リージョンはすべて委任管理リージョンです。委任管理リージョンは、ルート管理リージョンなどの他のリージョンから、ドメインの管理責任を受け取ります。委任管理リージョンには次の情報が含まれています。

- リージョン内のすべての Oracle Names Server とドメイン。
- この管理リージョンの子リージョン内のドメインと Oracle Names Server のアドレス。
- ルート・リージョン内の Oracle Names Server のアドレス。このデータによって、委任リージョン内の Oracle Names Server が（ルート・リージョンを介して）他のリージョンと交信することができます。
- データ定義 - このローカル（委任）管理リージョン内のすべてのドメインに対するすべてのデータベース・サービス名およびデータベース・リンク、別名。

Oracle Names のバージョンによる違い

現在の Oracle Names は、以前のバージョンと大きく異なります。

- [Oracle Names バージョン 1](#)
- [Oracle Names バージョン 2](#)
- [Oracle Names バージョン 8](#)

Oracle Names バージョン 1

Oracle Names バージョン 1 では、管理者は Oracle Network Manager を使用して Oracle Names を構成し、すべてのトポロジ・データをデータベースに格納していました。リージョン内の Oracle Names Server はすべて同じデータベースにアクセスするため、サーバー間で同じ情報が共有されます。

クライアントには、SQLNET.ORA ファイルで指定される優先 Oracle Names Server のリストがあります。このリストはユーザーによって作成され、接続する Oracle Names の優先順位を表します。リストの先頭にある Oracle Names Server が、最初にクライアントから接続されます。

優先 Oracle Names Server は、7-20 ページの「[デフォルトの Oracle Names Server の構成](#)」で説明されているように、現在でも構成する場合があります。

Oracle Names バージョン 2

Oracle Names バージョン 2 では、管理者は Oracle Names Server の構成をバージョン 1 から引き継ぐか、または動的検出オプションを使用するかを選択できました。動的検出オプションは、単一リージョンおよび単一 DNS ドメインのネットワークの場合にのみ選択することをお薦めしていました。動的検出オプションは、予約済み Oracle Names Server を使用します。予約済み Oracle Names Server は、DNS または Oracle Names Server とクライアントの両方の HOSTS ファイルに組み込まれた正確な名前です。

TCP 接続用の予約済みホスト名	Named Pipes 接続用の予約済みコンピュータ名 ¹	SPX 接続用の予約済みサービス名
oranamesrvr0	ORANAMESRVR0	oranamesrvr
oranamesrvr1	ORANAMESRVR1	
oranamesrvr2	ORANAMESRVR2	
oranamesrvr3	ORANAMESRVR3	
oranamesrvr4	ORANAMESRVR4	

¹ Named Pipes 用の予約済み Names サーバー名は、すべて大文字でなければなりません。

予約済みアドレスをもつ Oracle Names Server にはそのアドレスでアクセスできるので、クライアントに優先 Oracle Names Server のリストで Oracle Names Server を探すように指示する必要はありません。

動的検出オプションが選択されると、各 Oracle Names Server は自動的に、リージョン内のすべての予約済み Oracle Names Server にデータをレプリケートします。リスナーは予約済み Oracle Names Server で登録するように構成されます。

構成の詳細は、『Oracle Names 管理者ガイド』のリリース 2.0 を参照してください。

Oracle Names バージョン 8

Oracle Names バージョン 8 では、管理者は Oracle Names Server の構成をバージョン 1 またはバージョン 2 から引き継ぐか、または新しい機能を使用するか選択できます。Oracle Names バージョン 8 には、バージョン 2 の動的検出オプション機能が組み込まれ、さらに単一バージョンおよび単一ドメインの制約がなくなっています。Oracle Names バージョン 8 の主要な機能は次のとおりです。

- サービスは、発見した Oracle Names Server に自身を登録し、名前とアドレスをリージョン内のすべての Oracle Names Server で利用可能にします。同様に、管理者がサービスを Oracle Names Server に手動で登録する場合も、サービス情報がすべての Oracle Names Server で利用可能になります。アドレス情報は、次のいずれかの方法で共有されます。
 - サービス・レプリケーション - サービス・レプリケーションでは、サービス情報が Oracle Names Server のキャッシュに格納され、すぐに他の Oracle Names Server のキャッシュにレプリケートされます。
 - リージョン・データベース - Oracle データベースが登録リポジトリとして使用される場合、登録情報はデータベースに格納され、このデータベースにすべての Oracle Names Server がアクセスします。
- 接続する Oracle Names Server のリストが、2-45 ページの「[検出の理解](#)」で説明するように、手動で構成せずに、検出プロセスを通してクライアントに作成されます。
- クライアント側のプロセスが、Oracle Names Server に格納されたデータのクライアント・キャッシュ・バージョンを作成するために実行されます。ここでは、データベース・アドレス、Oracle Names Server のアドレス、その他のサービス情報が含まれます。結果として、クライアントは Oracle Names Server に接続してアドレス情報を取得する必要がなく、アドレスの検索時間を短縮できます。
- Oracle Names バージョン 8 では、トポロジー情報を維持するためにデータベースを必要としません。ただし、管理者の設定で使用することもでき、こちらをお勧めします。

構成の情報は、6-21 ページの「[Oracle Names ネーミング・メソッドによるネットワークの構成](#)」を参照してください。

検出の理解

Oracle Names Server のリストを作成することにより、クライアントや他の Oracle Names Server が特定の Oracle Names Server に接続できるようになります。このリストを作成するプロセスを検出と呼びます。

クライアントや Oracle Names Server が Oracle Names Server を検出する場合、まず 1 つの Oracle Names Server を発見しようとします。クライアントは 1 つの Oracle Names Server を発見すると、リージョン内の他の Oracle Names Server に ping します。Oracle Names Server のリストがクライアントに作成され、UNIX の場合は .SDNS.ORA、Windows の場合は SDNS.ORA に保存されます。このリストは、応答時間の順に格納されます。

検出では、次の順で最初の Oracle Names Server が検索されます。

1. SQLNET.ORA ファイルの優先 Oracle Names Server (preferred Oracle Names Server)
2. 予約済み Oracle Names Server (well-known Oracle Names Server)
3. ポート 1575 に TCP/IP で構成されたローカル Oracle Names Server

クライアントが上の方法で最初の Oracle Names を見つけられなかった場合は、次の処理が行われます。

- Oracle Net8 Assistant を使用している場合は、Oracle Names Server のアドレスを指定するためのプロンプトが表示されます。
- NAMESCTL 制御ユーティリティが失敗します。指定したアドレスでコマンドを再実行する必要があります。

Oracle Connection Manager のアーキテクチャ

Oracle Connection Manager は、クライアント接続要求をその次のホップに送るか、または直接サーバーに送るかという処理を行うルーターの役割を果たします。Connection Manager を通じて接続要求を送るクライアントは、Connection Manager で構成される接続集中化または Net8 アクセス制御、複数プロトコル・サポートの機能を利用できます。

Oracle Connection Manager プロセス

Oracle Connection Manager に関連するプロセスは主に 3 つあります。

- [CMGW](#)
- [CMADMIN](#)
- [CMCTL](#)

CMGW

CMGW は、Oracle Connection Manager のハブとして機能するゲートウェイ・プロセスです。このプロセスは、次の役割を担います。

- [CMADMIN](#) プロセスに登録する。
- 着信接続要求をリスニングする。デフォルトでは、ポート 1600 でリスニングします。
- クライアントのリスナーに接続要求を開始する。
- クライアントとサーバー間のデータを中継する。
- CMCTL により開始された要求に応答する。

CMADMIN

CMADMIN は、Oracle Connection Manager の管理上の問題点すべてに対して責任を果たすマルチスレッド・プロセスです。このプロセスは、次の役割を担います。

- CMGW 登録を処理する
- 1 つ以上のデータベース・インスタンスで機能する全リスナーを識別する
- CMGW とリスナーに関するソース・ルート・アドレス情報を登録する
- CMCTL により開始された要求に応答する
- Oracle Names Server に登録されたリスナーを監視し、次にソース・ルート・アドレスを監視する
- ローカル Oracle Names Server を検索する
- SQL*Net 2.x と NET 8.x クライアントの Oracle Names Server でアドレス情報をメンテナンスする

Oracle Connection Manager は、使用可能なサービスの Oracle Names Server のキャッシュを更新するために、定期的に Oracle Names Server に交信します。

CMCTL

CMCTL は、Oracle Connection Manager 制御ユーティリティの実行可能名です。CMCTL には、CMADMIN と CMGW に対する管理上のアクセス権があります。Oracle Connection Manager 制御ユーティリティの詳細は、A-88 ページの「[Oracle Connection Manager 制御ユーティリティ \(CMCTL\)](#)」を参照してください。

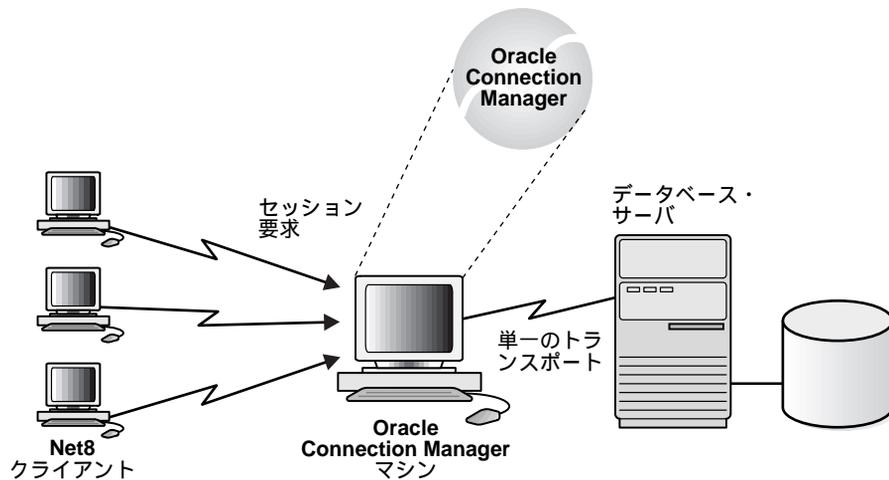
接続集中化

Oracle Connection Manager では、マルチスレッド・サーバー接続先への単一トランスポート・プロトコル接続を通じて、複数のクライアント・ネットワーク・セッションを多重化または集中化する Net8 の機能を利用できます。

集中化では、サーバーが着信要求に使用する接続とポイントの数を少なくすることにより、2つのプロセス間で複数の接続をメンテナンスするために必要なリソースを削減します。このため、サーバーが処理できるネットワーク・セッションの総数を増加できます。複数の Connection Manager を使用して、数千のユーザーを同時にサーバーに接続できます。

図 2-16 は、集中化の機能を示したものです。

図 2-16 Oracle Connection Manager による接続集中化



構成情報は、7-30 ページの「[接続集中化機能の使用](#)」を参照してください。

Net8 アクセス制御

Oracle Connection Manager には、TCP/IP 環境で指定されたサーバーへのクライアント・アクセスを制御するときに使用できる機能も含まれています。次の基準に基づいて、サーバーへの特定のクライアント・アクセスを可能にするか、または制限するフィルタリング・ルールを設定できます。

- クライアントのソース・ホスト名または IP アドレス
- サーバーの接続先ホスト名または IP アドレス
- 接続先データベース・サービス名

構成情報は、7-37 ページの「[Net8 アクセス制御の使用](#)」を参照してください。

Net8 Firewall Proxy

Net8 Firewall Proxy を提供しているファイアウォール・ベンダーもあります。アプリケーションのプロキシを必要とするファイアウォールにインストールされます。Net8 Firewall Proxy は、Oracle Connection Manager と同様の機能を持ちます。

注意： Net8 Firewall Proxy がサポートされているかどうかは、ファイアウォールの製造元にお尋ねください。

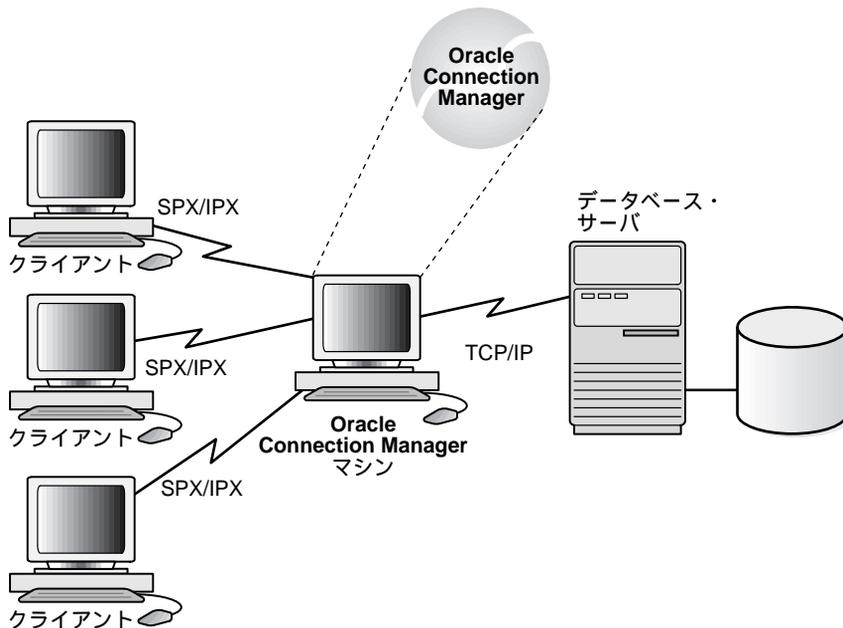
複数プロトコル・サポート

Oracle Connection Manager では、異なるネットワーク・プロトコルをもつクライアントとサーバーが相互に通信できるように、複数プロトコルがサポートされています。この機能は、SQL*Net バージョン 2 Oracle MultiProtocol Interchange ですでに実現している機能を置換したものです。

Net8 は、インストールおよびサポートが可能な大半のネットワーク・プロトコル・スタック間を行き来できます。実際に、サポートされるネットワーク・プロトコル数の制限は、特定のノードのハードウェアおよびメモリー、オペレーティング・システムに課されている制限のみです。

図 2-17 は、SPX ネットワーク内のクライアントが、Oracle Connection Manager を通じて TCP/IP トランスポート上のサーバーにネットワーク・セッションを送る方法を示したものです。

図 2-17 Oracle Connection Manager による複数プロトコル・サポート

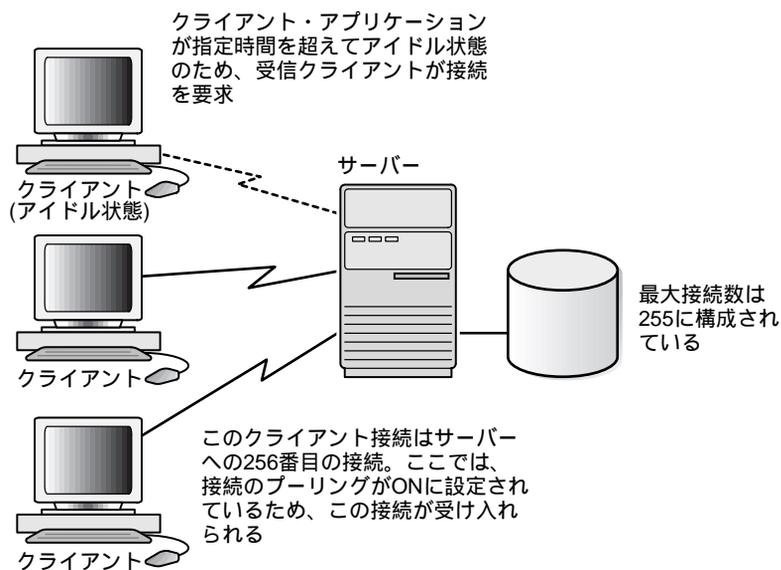


構成情報は、7-34 ページの「複数プロトコル・サポートの使用」を参照してください。

接続プーリング

接続プーリングは、マルチスレッド・サーバーに対する物理ネットワーク接続数を最大にすることができる、リソース利用およびユーザー拡張機能です。これは、複数のクライアント・プロセス間でディスパッチャの一連の接続を共有またはプールすることにより行います。図 2-18 は、接続プーリングの機能を示したものです。

図 2-18 接続プーリング



タイムアウト・メカニズムを使用して、指定された期間アイドル状態であったトランスポート接続を一時的に解放することによって、接続プーリングは、以前のアイドル接続との論理ネットワーク・セッションを維持しながら、これらの物理接続を着信クライアントに使用できるようにします。アイドル・クライアントにさらに多くのなすべき作業がある場合、物理接続はディスパッチャで再確立されます。

構成情報は、7-30 ページの「[接続集中化機能の使用](#)」を参照してください。

第 II 部

Net8 の設定と構成

第 II 部では、Net8 の設定、構成、接続およびトラブルシューティングの方法と、上級機能をインプリメントする方法について説明します。第 II 部には、次の章があります。

- 第 3 章の「ネットワークのプランニング」
- 第 4 章の「リリース 8.1 のインストールと Net8 関連製品について」
- 第 5 章の「アップグレードまたは移行の検討」
- 第 6 章の「ネーミング・メソッドとリスナーの構成」
- 第 7 章の「Net8 のオプション機能」
- 第 8 章の「接続の確立とネットワークのテスト」
- 第 9 章の「マルチスレッド・サーバーの構成」
- 第 10 章の「プログラマのための Net8 拡張機能」
- 第 11 章の「Net8 のトラブルシューティング」

ネットワークのプランニング

Net8 には、柔軟性のある使いやすいネットワークの設計と管理を支援する各種オプションがあります。Net8 では拡張性と管理容易性が向上したので、様々な環境をサポートするためのネットワークを開発できます。これらの環境が単純なワークグループであるか、大規模でミッション・クリティカルな企業であるかどうかは関係ありません。

この章では、Net8 を使用してネットワークを計画するときに考慮すべき点について説明します。また、ネットワーク製品の関係と、今後ネットワークを拡張してさらに効率的に管理するためのオプションについて説明します。内容は次のとおりです。

- [計画の概要](#)
- [ネットワーク・レイアウトの定義](#)
- [ネット・サービス名の変換](#)
- [ネットワーク・パフォーマンスの向上](#)
- [計画のサマリー](#)

計画の概要

ネットワークを構成する前に、ネットワークの検討と計画のための時間を取ってください。Oracle ネットワークを計画するときは、将来のニーズと現在の要件を念頭に置いてください。ネットワークに関して行わなければならない重要な決定は、次のとおりです。

- ネットワーク・レイアウトの定義
- ネット・サービス名の変換
- ネットワーク・パフォーマンスの向上

ネットワーク・レイアウトの定義

次のチェックリストは、ネットワークの主なコンポーネントの概要を示すためのものです。

1. ネットワークで行うことは何かを定義します。
2. クライアント・アプリケーションに必要な機能を調べ、これらの要件を満たすためのリソースを評価します。
3. どのマシンまたはノードが、クライアントまたはサーバーのアプリケーションに最適であるかを判断します。
4. 既存または将来のネットワーキング要件に最適なネットワーク・プロトコルを選択します。ネットワーク内のすべてのコンポーネントでうまく機能する1つのトランスポート・レベルのプロトコルを選択できます。多数のプラットフォーム上でほとんどの主要プロトコルが利用可能です。使用しているネットワークが、複数のプロトコルを通じて動作するクライアントまたはサーバーに関与している場合もあります。
5. ネットワークで複数のプロトコルを使用することに決定した場合は、どのノードが Oracle Connection Manager をインストールするのに最適であるかを判断します。どのノードを選択するかについては、選択したネットワーク・プロトコルと、予想される通信量を処理するマシンの能力によって決定します。

ネットワーク・コンポーネント間の物理的関係とともに論理的関係を示す図を描くと役立つことがあります。

ネット・サービス名の変換

ネットワーク・レイアウトを定義した後は、ネットワーク・インプリメンテーションを効率的に構成、管理する方法を決定する必要があります。ネーミング・メソッドを選択することが、最も重要で、最初に行う必要のある決定です。

ネーミング・メソッド

ネーミング・メソッドとは、データベース・サービスに接続しようとするときに、ネット・サービス名をネットワーク・アドレスに変換するためにクライアント・アプリケーションが使用する方法です。Net8 には、4 種類のネーミング・メソッドがあります。

- [ホスト・ネーミング](#)
- [ローカル・ネーミング](#)
- [Oracle Names を使用した集中化されたネーミング](#)
- [外部ネーミング](#)

ネットワークの規模および特性によって、各方法には、ネットワークを構成し、管理する方法に関して利点と欠点があります。

ホスト・ネーミング

ホスト・ネーミングを使用するとユーザーは、TCP/IP 環境で既存の名前解決サービスを使用してネット・サービス名を解決できます。この名前解決サービスは、DNS や NIS か、または単に中央でメンテナンスされる HOSTS ファイル・セットの場合があります。

ホスト・ネーミングでは、ユーザーはホスト名別名を使用して Oracle Server に接続できます。接続は、リスナーのデフォルトの TCP/IP ポートである 1521 を使用して確立されます。リスナー用のグローバル・データベース名とホスト名の別名を整合することにより、ノードごとの複数データベースとデータベースの位置の透過性をサポートします。

ホスト・ネーミングでは、データベースとの単純な接続性を必要とする環境で使用されるローカル名構成ファイル (TNSNAMES.ORA) が不要になります。ただし、接続プーリング、Oracle Connection Manager、異種サービスなど、高度な機能が必要とされる大規模で複雑な環境には適しません。

構成方法は、6-7 ページの「[ホスト・ネーミング・メソッドによるネットワークの構成](#)」を参照してください。

ホスト・ネーミング・オプションを使用した接続の確立

ホスト・ネーミング・オプションを使用してクライアント・セッションを確立するためのプロセスは、次のとおりです。

1. クライアントは、TCP/IP ホスト名別名を指定して接続要求を開始します。
2. ホスト・ネーミングによって、TCP/IP ホスト名とグローバル・データベース名を兼ねるネット・サービス名を使用してネットワーク・アドレスが生成され、このホスト名または別名が変換されます。TCP/IP ポートは 1521 にデフォルト設定されています。
3. Net8 は、作成されたアドレスに対して接続を要求します。
4. リスナーは、登録された TCP/IP ポート 1521 でリスニングし、クライアントの要求を受け取ってデータベースへの接続を確立します。
5. サーバーによって、接続が受け入れられます。

ローカル・ネーミング

ローカル・ネーミングとは、個々のクライアントで構成された情報を使用して、ネット・サービス名をネットワーク・アドレスに変換する方法です。アドレス帳とほぼ同じで、この情報が TNSNAMES.ORA という名前のローカル名構成ファイルに格納されます。

構成方法は、6-12 ページの「[ローカル・ネーミング・メソッドによるネットワークの構成](#)」を参照してください。

ローカル・ネーミングを使用した接続の確立

ローカル・ネーミングを使用してクライアント・セッションを確立するためのプロセスは、次のとおりです。

1. クライアントは、ネット・サービス名を指定して接続要求を開始します。
2. ネット・サービス名は、ローカル名構成ファイルに格納されているネットワーク・アドレスに変換されます。
3. Net8 は、指定されたアドレスに対して接続を要求します。
4. リスナーは、要求を受け取り、リスニング・サービスの提供先であるデータベースにその要求を送ります。
5. サーバーによって、接続が受け入れられます。

Oracle Names を使用した集中化されたネーミング

集中化されたネーミングとは、Oracle Names を使用してネット・サービス名をネットワーク・アドレスに変換する方法です。Oracle Names は、Oracle Names Server を使用してネットワーク上のすべてのデータベース・サービスの名前とアドレスを格納します。ディレクトリ支援と呼ばれるものとはほぼ同様に、直接サーバーに接続を希望するクライアントは、接続要求を Oracle Names Server に送ります。Oracle Names Server はサービス名をネットワーク・アドレスに変換し、その情報をクライアントに戻します。

構成方法は、6-12 ページの「[ローカル・ネーミング・メソッドによるネットワークの構成](#)」を参照してください。

集中化されたネーミング・オプションを使用した接続の確立

集中化されたネーミング・オプションを使用してクライアント・セッションを確立するためのプロセスは、次のとおりです。

1. クライアントは、ネット・サービス名を指定して接続要求を開始します。
2. 接続要求は Oracle Names Server に送られ、ネット・サービス名はネットワーク・アドレスに変換されます。このアドレスは、クライアントに戻されます。
3. Net8 は、指定されたアドレスに対して接続を要求します。

4. リスナーは、要求を受け取り、リスニング・サービスの提供先であるデータベースにその要求を送ります。
5. サーバーによって、接続が受け入れられます。

外部ネーミング

外部ネーミングとは、サポートされている Oracle 以外のネーミング・サービスを使用して、ネットワーク・アドレスにネット・サービス名を変換する方法です。外部ネーミングでは、顧客の外部 (Oracle 以外の) ネーミング・サービスに格納されているネット・サービス名が変換されます。これらのサービスは、次のとおりです。

- ネットワーク情報サービス (NIS)
- NetWare ディレクトリ・サービス (NDS)

注意： SQL*Net の旧リリースでは、これらの外部ネーミング・メソッドは Oracle Advanced Networking Option の一部でした。これらは、現在 Net8 の標準として組み込まれています。

分散コンピューティング環境のセル・ディレクトリ・サービス (CDS) は、Oracle Advanced Networking Option の DCE 統合部分の一部として引き続き使用できます。

外部ネーミングを使用した接続の確立

外部ネーミングを使用してクライアント・セッションを確立するためのプロセスは、次のとおりです。

1. クライアントは、ネット・サービス名を指定して接続要求を開始します。
2. 外部ネーミング・メソッドによって要求が外部ネーミング・システムに送られ、そこでネット・サービス名はネットワーク・アドレスに変換されます。アドレスは、クライアントに戻されます。
3. Net8 は、指定されたアドレスに対して接続を要求します。
4. リスナーは、要求を受け取り、リスニング・サービスの提供先であるデータベースにその要求を送ります。
5. サーバーによって、接続が受け入れられます。

外部ネーミングの構成

Oracle Net8 Assistant を使用して外部ネーミングを構成する手順は、次のとおりです。

1. クライアント・ノードに適切な外部ネーミング・メソッドがインストールされていることを確認します。
2. プロファイルに、使用する外部ネーミング・メソッド（例：CDS、NDS または NIS）を指定します。外部ネーミング・メソッドが設定されていない場合は、Oracle Net8 Assistant を使用してクライアントのプロファイルを編集します。

構成方法は、6-51 ページの「[外部ネーミング・メソッドによるネットワークの構成](#)」を参照してください。

Oracle Names と外部ネーミング

Oracle Names を他の専用ネーミング・サービスまたは公開のネーミング・サービスと一緒に使用して、環境の違いによる名前を変換できます。たとえば、複数のネーム・サービス間でネーム変換を行うために、すでに Oracle Names が実行されている企業ネットワーク内のクライアントとサーバーすべてに、CDS/DCE、NIS、または NDS の外部ネーミングをインストールできます。

Oracle Names は、Oracle データベース専用名前とアドレスを格納し、変換する専用のネーム・サービスであるため、1つのネーム変換は、Oracle Names にすべての Oracle サービスを格納し、DNS や X.500 などのディレクトリ・サービスをグローバルなネーミング・サービスとして利用できます。

ネーミング・メソッドの選択

次の表は、各ネーミング・メソッドの相対的な利点と欠点をまとめ、ネットワーク内でそれらのネーミング・メソッドを使用するためのヒントを示したものです。

ネーミング・メソッド	利点 / 欠点	推奨環境
ホスト・ネーミング	<ul style="list-style-type: none"> ■ 最小構成で行える。ユーザーは、接続を確立するためにホスト名のみを指定できます。 ■ ローカル名構成ファイル (TNSNAMES.ORA) の作成とメンテナンスの必要がない。 ■ Oracle Names 管理手順を理解する必要はない。 <p>欠点: 次の内容がすべて当てはまるときしか使用できません。</p> <ul style="list-style-type: none"> ■ クライアントとサーバーは、TCP/IP を使用して接続している。 ■ ホスト名は、ドメイン・ネーム・サービス (DNS)、ネットワーク情報サービス (NIS)、中央でメンテナンスされる TCP/IP hosts ファイルなどの IP アドレス変換メカニズムを通じて変換されている。 ■ Oracle Connection Manager の機能は不要。 	ここに記載した条件に合致する単純な TCP/IP ネットワーク
ローカル・ネーミング	<ul style="list-style-type: none"> ■ ネット・サービス名アドレスを変換する比較的簡単な方法である。 ■ 異なるプロトコルを実行しているネットワーク間でネット・サービス名を変換する。 <p>欠点: すべてのネット・サービス名からアドレスへの変換をローカル側で構成する必要がある。</p>	ほとんど変更がなく、サービス数の少ない単純な分散ネットワーク

ネーミング・メソッド	利点 / 欠点	推奨環境
Oracle Names	<ul style="list-style-type: none"> ■ 単一の場所にネットワーク名とアドレスを集中し、名前の変更と更新の管理を容易にする。たとえば、既存のサーバーに対して変更を付加する場合、または新しいサーバーをネットワークに追加する場合、変更は1つの Oracle Names Server 上で1回だけ行われます。このため、管理者は数百、数千のクライアントに対して変更を行う必要がありません。 ■ 異なるプロトコルを実行しているネットワーク間でネット・サービス名を変換する。 <p>欠点:</p> <ul style="list-style-type: none"> ■ Oracle Names は、Oracle サービス用のネットワーク名とネットワーク・アドレスのみを格納する。 ■ Oracle Names Server の補足設定と追加管理が必要。 	<p>頻繁に変更する大規模で複雑なネットワーク (20 以上のデータベースをもつ)</p>
外部ネーミング	<p>管理者がすでに理解しているツールとユーティリティを使用して、Oracle ネット・サービス名を固有のネーム・サービスにロードできる。</p>	<p>既存のネーム・サービスをもつネットワーク</p>

ネットワーク・パフォーマンスの向上

次の1つを実施することで、大規模なネットワークのパフォーマンスを向上できます。

- 接続要求の管理
- 接続プーリング
- 接続集中化
- クライアント負荷のバランス
- セッション・データ・ユニット (SDU) サイズの調整によるデータ転送の最適化
- TCP/IP の固定バッファ・フラッシング
- リスナー・キューサイズの構成

接続要求の管理

ネットワークで過剰な接続通信量を受信すると予想される場合は、事前に開始または事前に起動された専用サーバー、または共有サーバーに、これらの要求をリダイレクトすることによって、リスナーでこれらの要求を管理できます。

次の表は、各プロセスの相対的な利点をまとめ、ネットワーク内でそれらのプロセスを使用するためのヒントを示したものです。

タイプ	利点	推奨環境
共有サーバー	<ul style="list-style-type: none"> ■ 専用サーバーより効率よくネットワーク・リソースを利用するので、セッションのスループットとパフォーマンスが向上する ■ データベースに対するセッション数が増加しても、サーバー側で必要なメモリ・リソースと処理リソースを最小限に抑えることができる 	Oracle 共有サーバーまたはマルチスレッド・サーバーがサポートされるネットワーク、または新しいサーバーの作成に時間のかかるリソース集中型のネットワーク
事前に開始または事前に起動された専用サーバー (デフォルト)	<ul style="list-style-type: none"> ■ 新しい接続要求ごとに専用サーバーを作成する必要がないので、接続時間を短縮できる ■ サーバーを停止して作成しなおさなくても、他の接続で使用するサーバーを再利用できるので、割当て済みのメモリやシステム・リソースを有効に利用できる 	Oracle 共有サーバーまたはマルチスレッド・サーバーがサポートされていないネットワーク、または新しいサーバーの作成に時間のかかるリソース集中型のネットワーク

リスナーを構成して接続要求を事前に開始または事前に起動された専用サーバーにリダイレクトする方法の詳細は、[第9章の「マルチスレッド・サーバーの構成」](#)を参照してください。

接続プーリング

接続プーリングは、マルチスレッド・サーバーに対する物理ネットワーク接続数を最大にすることができるリソース利用機能です。これは、複数のクライアント・プロセス間でディスパッチャの一連の接続を共有またはプールすることにより行います。

タイムアウト・メカニズムを使用して、指定された期間アイドル状態であったトランスポート接続を一時的に解放することによって、接続プーリングは、以前のアイドル接続との論理セッションを維持しながら、これらの物理接続を着信クライアントに使用できるようにします。アイドル・クライアントにさらに多くのなすべき作業がある場合、物理接続はディスパッチャで再確立されます。

この機能は、マルチスレッド・サーバーが構成されている場合のみ動作します。

接続プーリングの概念は 2-50 ページの「[接続プーリング](#)」に、構成情報は 7-47 ページの「[接続プーリングの使用](#)」に説明があります。

接続集中化

接続集中化は、Oracle Connection Manager を使用することにより利用できる機能です。この機能により、1 つのトランスポートを通じて、複数のクライアント・セッションをマルチスレッド・サーバーに多重送信または集中送信するため、Net8 の利点を活用できます。接続プーリングと同様に、集中化はネットワーク・リソースを最適化し、一定数の物理サーバー間で可能なクライアント / サーバー・セッション数を増加させます。接続プーリングとは異なり、集中化はトランスポート接続をメンテナンスします。

接続集中化の概念は 2-46 ページの「[Oracle Connection Manager のアーキテクチャ](#)」に、構成情報は 7-30 ページの「[接続集中化機能の使用](#)」に説明があります。

接続プーリングと接続集中化の使用

次の表は、接続プーリングと接続集中化を使用するときの相対的な利点をまとめ、ネットワーク内でそれらを使用するためのヒントを示したものです。

機能	利点	推奨環境
接続プーリング	<ul style="list-style-type: none"> 1つのプロセスに対して使用されるネットワーク・リソース数を制限する 一定数に制限された物理接続数に対するクライアント / サーバー・セッション数を最大化する リソースの利用を最適化する 	メッセージ送信、OLAP など、対話形式の評価 / 検索の時間頻度が高いアプリケーションを多数のクライアントで実行するようなネットワーク
集中化	<ul style="list-style-type: none"> 大規模なクライアント集団をサポートする 実ユーザーの識別と監視が可能 中間層のアプリケーションが追加サービスをサポートできる 複数のアプリケーションをもつクライアントには、トランスポートが1つのみで済む データベース・リンクには、ネットワーク接続が1つのみで済む 	連続接続が必要なネットワーク

クライアント負荷のバランス

ロード・バランシングとは、クライアントの接続を複数のリスナー、ディスパッチャ、インスタンスおよびノード間で均一に分散し、どれか1つのコンポーネントの負荷が過大になることを防ぐ機能です。負荷のバランスをとることにより、接続パフォーマンスが向上します。

クライアントの接続要求に対してリスナーをランダムに選択できるように、クライアントを構成できます。クライアント負荷のバランスが有効に設定されていると、Net8 はリスナー・アドレスのリストから1つをランダムに選択し、各リスナーの負荷のバランスを保ちます。クライアント負荷のバランスが無効に設定されていると、Net8 はリストの順序に従ってリスナーを使用します。構成方法は、7-11 ページの「[複数アドレス・オプションの構成](#)」を参照してください。

セッション・データ・ユニット (SDU) サイズの調整によるデータ転送の最適化

ネットワーク間の往復回数を減らすようにアプリケーションを調整することは、ネットワーク・パフォーマンスを向上させる最高の方法です。これを行うと、セッション・データ・ユニット (SDU) のサイズを調整してデータ転送を最適化することもできます。

SDU は、Net8 がネットワーク間でデータを転送する前にデータを配置するバッファです。Net8 がバッファ内のデータを送信するのは、データ送信が要求されたとき、またはバッファがデータで満杯になったときです。

次の表は、SDU のサイズ変更が適切な時期と適切でない時期を示したものです。

セッション・データ・ユニット・サイズを変更するとき	セッション・データ・ユニット・サイズを変更する必要がないとき
<ul style="list-style-type: none"> ■ サーバーから戻されるデータが個別のパケットに別れる場合 ■ 遅延が長い広域ネットワーク (WAN) 上にいる場合 ■ パケット・サイズが一定している場合 ■ 戻されるデータ量が多い場合 	<ul style="list-style-type: none"> ■ 遅延に対処できるようにアプリケーションを調整できる場合 ■ データ伝送の影響がほとんどない高速ネットワークを使用している場合 ■ 要求によってサーバーから戻されるデータ量が少ない場合

注意： SDU サイズは、通常の伝送フレーム・サイズの倍数に設定する必要があります。イーサネットの通常のフレーム・サイズは 1024 のため、イーサネット・プロトコルでの最も効率的な SDU サイズは 1024 の倍数です。ただし、1024 の 4 倍までに制限されます。

接続プーリングまたは接続集中化を使用しているときは、これらの機能に対して、1 つのトランスポートについてさらに 16 バイトが必要になることに注意してください。プロトコル・フレーム・サイズの詳細については、そのプロトコルに関するそれぞれのマニュアルを参照してください。

セッション・データ・ユニット・サイズを調整するには、ローカル名構成ファイル (TNSNAMES.ORA) にパラメータを追加します。詳細は、7-7 ページの「[ネット・サービス名の接続属性の詳細構成](#)」を参照してください。

TCP/IP の固定バッファ・フラッシング

TCP/IP を使用するいくつかのアプリケーションのある状況では、Net8 のパケットがネットワークにすぐに送出されません。ほとんどの場合、この動作は大量のデータが一方から他方に送られるときに発生します。TCP/IP の実装自体がフラッシュの欠如の原因で、許容範囲を超える遅延を引き起こすおそれがあります。この問題を修正するためには、バッファ・フラッシング・プロセスで遅延なしを指定できます。

詳細は、C-82 ページの「[プロトコル固有のパラメータ \(PROTOCOL.ORA\)](#)」にある TCP.NODELAY パラメータを参照してください。

リスナー・キューサイズの構成

TCP/IP を介してリスニング・プロセス（リスナー、Oracle Connection Manager、Oracle Names など）に対する多数の接続要求を受信することが予想される場合は、Net8 を使用してシステム・デフォルトより高い値になるようにリスニング・キューを設定できます。

構成方法は、6-70 ページの「[大量の接続要求の処理](#)」を参照してください。

計画のサマリー

次の表は、ネットワークを計画したときに選択した可能性のある多数のオプションをまとめたものです。

科目	オプション
ネットワーク・レイアウト	<ul style="list-style-type: none">■ 単一または複数のプロトコル
ネット・サービス名の変換	<ul style="list-style-type: none">■ ホスト・ネーミング■ 外部ネーミング■ 集中化されたネーミング■ ローカル・ネーミング
接続要求の管理	<ul style="list-style-type: none">■ 専用のサーバー■ 事前に開始および事前に起動された専用サーバー■ 共有サーバー
ネットワーク・パフォーマンス	<ul style="list-style-type: none">■ 接続プーリング■ 接続集中化■ クライアント負荷のバランス■ セッション・データ・ユニット・サイズの最適化■ 固定バッファ・フラッシング■ リスナー・キューサイズの拡大

4

リリース 8.1 のインストールと Net8 関連製品について

この章では、リリース 8.1 の構成ファイル、インストールおよび Net8 関連製品について説明します。内容は次のとおりです。

- 今回のリリースに含まれる新しい製品とその機能
- Net8 リリース 8.1 で変更された構成ファイル
- Net8 のインストールとその製品

今回のリリースに含まれる新しい製品とその機能

次の表に、Net8 リリース 8.1.5 の新しい製品と機能を説明します。

新製品 / 新機能	説明	詳細情報の参照先
データベース・インスタンス登録	<p>データベース・インスタンスは、起動時にそれ自身をリスナーに登録する。これより前のリリースでは、インスタンスに関する情報は手動で LISTENER.ORA ファイルに構成する必要がありました。</p> <p>インスタンスの登録は、次の要素から構成されます。</p> <ul style="list-style-type: none">■ サービス登録。データベース・サービス名やインスタンス名など、インスタンスの情報をリスナーに提供します。■ MTS ディスパッチャの登録。リスナーにディスパッチャの情報を提供します。 <p>また、クライアントから接続要求があったときに最初のインスタンスがダウンしていると、データベース・インスタンス登録によって、別のインスタンスに対して自動的に connect-time failover が許可されます。</p>	2-27 ページの「 リスナーへのデータベース・インスタンス登録 」
接続負荷のバランス	<p>インスタンス登録によって、接続負荷のバランスを使用できる。接続負荷のバランスによって、同じサービスのさまざまなインスタンスとディスパッチャに対するアクティブな接続の数が均一化されます。これにより、リスナーはディスパッチャごとの接続数と、インスタンスが実行されているノードの負荷に基づいてルーティングを選択できます。</p>	2-30 ページの「 接続負荷のバランス 」

新製品 / 新機能	説明	詳細情報の参照先
サービス・ネーミング	<p>サービス・ネーミングには、単一のデータベースが提供する複数のサービスや、複数のインスタンスにまたがるサービスを含めることができる。</p> <p>これより前のリリースでは、クライアントはデータベース・インスタンスの Oracle システム識別子 (SID) を使用して構成されていました。この SID はリスナーに渡されます。次にリスナーはこの SID 情報を検査し、接続を許可または拒否します。このほかにも SID は、システム・グローバル領域 (SGA) へのポインタとしてデータベース内部で使用されていました。</p> <p>SID によってデータベース・インスタンスを識別することはできませんが、データベースは識別できません。この制限によって、データベースに 2 つ以上のサービスを関連付けることができませんでした。</p> <p>1 つのデータベースは複数のサービスに対して動作できるので、SID にかわってサービス・ネーミングが使用されています。サービス・ネーミングによって、クライアントから次の対象にアクセスできます。</p> <ul style="list-style-type: none"> ■ サービス名を使用してサービス全体 ■ インスタンス名を使用してデータベースのインスタンス (SID による方法と同じ) <p>複数のインスタンスを含むサービスをサポートするには、接続記述子の中で次の新しいパラメータを使用します。</p> <ul style="list-style-type: none"> ■ SERVICE_NAME ■ INSTANCE_NAME 	<ul style="list-style-type: none"> ■ 2-2 ページの「接続操作」 ■ 4-8 ページの「Net8 リリース 8.1 で変更された構成ファイル」 ■ 6-2 ページの「サービス名とインスタンス名の設定」

新製品 / 新機能	説明	詳細情報の参照先
<i>service_name</i> から <i>net_service_name</i> への変更	<p>これより前のリリースでは、接続記述子の別名は <i>service_name</i> と呼ばれていた。例：</p> <pre>service_name= (description= (address= (protocol_address_information)) (connect_data= (sid=sid)))</pre>	第6章の「ネーミング・メソッドとリスナーの構成」
	<p>エンド・ユーザーは、このサービス名を使用してサービスに接続していました。例：</p> <pre>sql> CONNECT username/password@service_name</pre>	
	<p>現在、サービス名は接続記述子のサービスを参照するようになったため、接続記述子として <i>net_service_name</i> が <i>service_name</i> のかわりに接続文字列中で使用されます。接続記述子に <i>net_service_name</i> をマップすると、次のようになります。</p> <pre>net_service_name= (description= (address=(protocol specific address data)) (connect_data= (service_name=service_name)))</pre>	
	<p>エンド・ユーザーは、接続文字列で <i>net_service_name</i> を使用してサービスを指定します。</p> <pre>sql> connect username/password@net_service_name</pre>	

新製品 / 新機能	説明	詳細情報の参照先
DB_DOMAIN	<p>初期化ファイル (INITSID.ORA) 中で、WORLD は DB_DOMAIN のデフォルト設定として使用されない。新しいデフォルト設定は NULL です。したがって、デフォルト設定を使用するのであれば、サービス名にドメイン名を含める必要はありません。以前のリリースの Oracle では、デフォルト設定は .WORLD でした。</p> <p>DB_DOMAIN が NULL に設定されていれば、サービス名をドメインで修飾する必要はありません。</p> <p>DB_DOMAIN が NULL に設定されていない場合は、サービス名を入力するときにドメインを含める必要があります。たとえば、Oracle8i データベースに SALES というサービス名があり、COM ドメインに所属している場合、SALES.COM というサービス名を指定してそのサービスを識別します。</p>	6-2 ページの「 サービス名とインスタンス名の設定 」
Java オプションのサポート	<p>これより前のリリースで Net8 は、Two-Task Common (TTC) プレゼンテーション・レイヤーと Transparent Network Substrate (TNS) ネットワーク・セッション (NS) レイヤーを使用する接続のみをサポートしてクライアント接続を確立していた。</p> <p>Java オプションでは、ユーザーが従来のデータベース・ストアド・プロシージャや、Enterprise JavaBeans および CORBA Server を使用してプログラミングを行い、データベース・サーバーを操作できます。Enterprise JavaBeans と CORBA Server にアクセスするクライアントをサポートするために、プレゼンテーション・レイヤーとセッション・レイヤーのサポートが拡張されています。</p>	7-48 ページの「 Java オプション接続の構成 」
Net8 Configuration Assistant	<p>インストール後に基本的な構成作業を実行するためのツールとして、Net8 Configuration Assistant が用意されている。インストールの終了後に、このツールによって自動的にデフォルトの構成ファイルが設定されます。</p> <p>このほかにスタンドアロン・モードで Net8 Configuration Assistant を実行し、次に示すさまざまな要素を構成できます。</p> <ul style="list-style-type: none"> ■ リスナー ■ ネーミング・メソッド ■ ネット・サービス名 ■ LDAP ディレクトリ・サービス 	4-22 ページの「 Net8 のインストールとその製品 」

新製品 / 新機能	説明	詳細情報の参照先
RADIUS (Remote Authentication Dial-In User Service) 認証方法	<p>軽量でオープンなプロトコルである RADIUS は、ネットワーク・クライアントと認証サーバーとの間でユーザー認証、認可および課金処理を行う。RADIUS 認証方法を使用すると、トークン・カードなどの RADIUS 準拠デバイスを使用して認証できるようになります。また、RADIUS は要求 - 応答認証と課金処理もサポートします。</p>	『Oracle8i Advanced Security 管理者ガイド』
SSL プロトコル付き TCP/IP	<p>SSL 付き TCP/IP によって、TCP/IP とセキュア・ソケット・レイヤー (Secure Sockets Layer: SSL) を使用するネットワーク上でのクライアント / サーバー対話が可能になる。</p> <p>SSL は、証明書や秘密鍵などの認証データを Oracle Wallet に格納します。クライアントがサーバーと Net8 接続を開始すると、SSL は秘密鍵と証明書を使用して 2 者間でハンドシェイクを実行します。ハンドシェイクの実行中、次の処理が行われます。</p> <ul style="list-style-type: none"> ■ クライアントとサーバーは、認証データ、暗号化方法およびデータ整合性タイプのセットである Cipher Suite についてネゴシエーションを行い、交換するメッセージに適用する。 ■ 使用する構成によっては、サーバーはクライアントの公開鍵で暗号化したメッセージにサーバー自身の証明書を入れてクライアントに送信する。サーバーは、この同じメッセージを使用して、クライアントの証明書を送るように要求する場合があります。クライアントは、それ自身の秘密鍵を使用してこのメッセージの暗号化を解除し、サーバーの証明書に認証局のシグネチャがあることを確認します。 ■ 必要に応じてクライアントは、このユーザーの証明書を Oracle サーバーに送信する。証明書によって、ユーザーの情報が正しいことと、公開鍵が実際にそのユーザーのものであることが保証されます。 	『Oracle8i Advanced Security 管理者ガイド』
	<p>サーバーはこのユーザーの証明書を調べて、認証局のシグネチャがあることを確認します。</p>	

新製品 / 新機能	説明	詳細情報の参照先
セキュア・ソケット・レイヤー (Secure Sockets Layer: SSL) 認証方法	<p>SSL は、証明書や秘密鍵などの認証データを Oracle Wallet に格納する。クライアントがサーバーと Net8 接続を開始すると、SSL は秘密鍵と証明書を使用して 2 者間でハンドシェイクを実行します。ハンドシェイクの実行中、次の処理が行われます。</p> <ul style="list-style-type: none"> ■ クライアントとサーバーは、認証データ、暗号化方法およびデータ整合性タイプのセットである Cipher Suite についてネゴシエーションを行い、交換するメッセージに適用する。 ■ SSL の構成によっては、クライアントはユーザーの証明書を Oracle サーバーに送信する。証明書によって、ユーザーの情報が正しいことと、公開鍵が実際にそのユーザーのものであることが保証されます。サーバーはこのユーザーの証明書を調べて、認証局のシグネチャがあることを確認します。 ■ 使用する構成によっては、サーバーはクライアントの公開鍵で暗号化したメッセージにサーバー自身の証明書を入れてクライアントに送信する。クライアントは、それ自身の秘密鍵を使用してこのメッセージの暗号化を解除し、サーバーの証明書に認証局のシグネチャがあることを確認します。 	『Oracle8i Advanced Security 管理者ガイド』
SSL プロトコル付き TCP/IP	TCP/IP と SSL を使用するネットワーク上でクライアント / サーバー対話を可能にする。	『Oracle8i Advanced Security 管理者ガイド』

Net8 リリース 8.1 で変更された構成ファイル

この項では、次のトピックについて説明します。

- [構成ファイルで使用される構文の変更点](#)
- [TNSNAMES.ORA の変更点](#)
- [LISTENER.ORA の変更点](#)
- [リリース 8.1 で廃止されたパラメータ](#)

構成ファイルで使用される構文の変更点

次の表は、Net8 のリリース 8.0 と 8.1 の構文における相違点を説明しています。

変更点	Net8 リリース 8.0	Net8 リリース 8.1
TNSNAMES.ORA 内で、システム識別子 (SID) ではなく、サービス名とインスタンス名を指定して接続	<p>次に示すように、データベースの SID は TNSNAMES.ORA の CONNECT_DATA セクションで指定する必要があります。</p> <pre> service_name= (description= (address_list= (address=...) (address=...)) (connect_data=(sid=sales)) </pre>	<p>データベースに複数のインスタンスにまたがるサービスを含めることができるようになったため、SID のかわりにサービス名を指定するか、またはオプションでインスタンス名を指定する。</p> <p>次のエントリによって、クライアントは SALES.COM というサービスに接続できるようになります。</p> <pre> net_service_name= (description= (address=...) (address=...) (connect_data= (service_name=sales.com))) </pre> <p>次のエントリによって、クライアントは OP.COM サービスの一部である OP1 インスタンスに接続できるようになります。</p> <pre> net_service_name= (description= (address=...) (address=...) (connect_data= (service_name=op.com) (instance_name=op1))) </pre> <p>SERVICE_NAME は、一般にグローバル・データベース名です。グローバル・データベース名は、データベース名とドメイン名から構成され、インストール中またはデータベースの作成中に入力します。INSTANCE_NAME は、一般にインストール中またはデータベースの作成中に入力する SID です。</p> <p>追加情報: 次を参照してください。</p> <ul style="list-style-type: none"> 6-2 ページの「サービス名とインスタンス名の設定」 6-12 ページの「ローカル・ネーミング・メソッドによるネットワークの構成」

変更点	Net8 リリース 8.0	Net8 リリース 8.1
<p>TNSNAMES.ORA の DESCRIPTION_LIST と ADDRESS_LIST を使用したクライアント負荷のバランス</p>	<p>複数の接続記述子からネット・サービス名へのマッピングと、複数リスナーに対するクライアント負荷のバランスでは、DESCRIPTION_LIST が使用されていた。</p> <p>クライアント負荷のバランスをとるには、ユーザーが各リスナーに対して接続記述子を個別に定義する必要がありました。</p>	<p>このリリースでも、クライアント負荷のバランスは DESCRIPTION_LIST に対してデフォルトでオンに設定されているが、新たに用意された LOAD_BALANCE パラメータを1つの ADDRESS_LIST に対して使用するか、または ADDRESS セットやセットの DESCRIPTION に LOAD_BALANCE パラメータを関連付けて明示的に指定することもできる。次のエントリによって、クライアント負荷のバランスはアドレス・リストに関連付けられ、すべてのリスナーの ADDRESS に適用されます。</p> <pre data-bbox="801 574 1136 791"> net_service_name= (description= (load_balance=on) (address=...) (address=...) (connect_data= (service_name=sales.com))) </pre> <p>クライアント負荷のバランスはどのバージョンのクライアントでも機能しますが、Oracle8i リリース 8.1 サーバーが必要です。</p> <p>追加情報: 次を参照してください。</p> <ul style="list-style-type: none"> ■ 7-11 ページの「複数アドレス・オプションの構成」 ■ C-35 ページの図 C-4 の「リリース 8.1 とそれ以前のリリースのクライアント負荷のバランス」

変更点	Net8 リリース 8.0	Net8 リリース 8.1
TNSNAMES.ORA と LISTENER.ORA で ADDRESS_LIST 内部に ネストされたアドレス	<p>アドレス・リストが1つしかない場合でも、次に示すように ADDRESS_LIST を使用する必要があった。</p> <pre>service_name= (description= (address_list= (address=...) (address=...)) (connect_data=(sid=sales))</pre>	<p>アドレス・リストが1つしかない場合、次に示すように Net8 のアドレス・リストを ADDRESS_LIST に埋め込む必要はない。</p> <pre>net_service_name= (description= (address=...) (address=...) (connect_data= (service_name=sales.com)))</pre> <p>ネストされたアドレスはどのバージョンのクライアントでも機能しますが、Oracle8i リリース 8.1 サーバーが必要です。</p> <p>追加情報: 次を参照してください。</p> <ul style="list-style-type: none"> ■ 6-12 ページの「ローカル・ネーミング・メソッドによるネットワークの構成」 ■ C-31 ページの「ローカル・ネーミング・パラメータ (TNSNAMES.ORA)」

変更点	Net8 リリース 8.0	Net8 リリース 8.1
TNSNAMES.ORA の ADDRESS_LIST を使用した connect-time failover	connect-time failover は ADDRESS_LIST に含意されており、最初のリスナーでクライアント接続が失敗すると別のリスナーにフェイルオーバーする。Net8 と SQL*Net は、接続に成功するまですべてのリスナー・アドレスを試行します。	フェイルオーバーは、ADDRESS_LIST、DESCRIPTION_LIST および DESCRIPTION セットに対してデフォルトでサポートされている。また、フェイルオーバーは、新たに用意された FAILOVER パラメータを ADDRESS セットに対して使用することにより明示的に指定することもできます。次のエントリによって、connect-time failover はアドレス・リストに関連付けられ、すべてのリスナーの ADDRESS に適用されます。 <pre data-bbox="801 548 1136 760"> net_service_name= (description= (failover=on) (address=...) (address=...) (connect_data= (service_name=sales.com))) </pre>
		connect-time failover はどのバージョンのクライアントでも機能しますが、Oracle8i リリース 8.1 サーバーが必要です。
		<p>追加情報 : 次を参照してください。</p> <ul style="list-style-type: none"> ■ 7-11 ページの「複数アドレス・オプションの構成」 ■ C-31 ページの「ローカル・ネーミング・パラメータ (TNSNAMES.ORA)」

変更点	Net8 リリース 8.0	Net8 リリース 8.1
TNSNAMES.ORA に SOURCE_ROUTE=ON を設定し、Oracle Connection Manager を媒介とするルーティング接続	<p>Oracle Connection Manager を介して宛先サービスまでのアドレスのソース・ルートを作成する SOURCE_ROUTE パラメータは、ADDRESS_LIST の外側に記述する必要があった。</p> <pre> service_name= (description= (source_route=on) (address_list= (address=...) (address=...)) (connect_data= (sid=sales)) </pre>	<p>SOURCE_ROUTE は、ADDRESS_LIST の外側に記述しなくてもよい。次に示すように ADDRESS リストに関連付けて記述します。</p> <pre> net_service_name= (description= (source_route=on) (address=...) (address=...) (connect_data= (service_name=sales.com))) </pre> <p>追加情報: 次を参照してください。</p> <ul style="list-style-type: none"> 7-40 ページの「Oracle Connection Manager 機能を使用する場合のクライアントの構成」 C-31 ページの「ローカル・ネーミング・パラメータ (TNSNAMES.ORA)」
LISTENER.ORA の SID_LIST_listener_name 情報	<p>LISTENER.ORA では、リスナーからサービスの提供を受けているデータベースの SID の定義が必要だった。データベースの SID は、LISTENER.ORA ファイルの SID_LIST セクションに指定する必要がありました。例:</p> <pre> sid_list_listener=(sid_list= (sid_desc= (global_dbname=sales.com) (sid_name=db1) (oracle_home=/usr/bin/oracle))) </pre> <p>追加情報: 6-62 ページの「リスナーの構成」を参照してください。</p>	<p>データベースのインスタンスとマルチスレッド・サーバー・ディスパッチャは自らをリスナーに登録するため、次の場合を除いてデータベース情報を明示的に定義する必要はない。</p> <ul style="list-style-type: none"> Oracle 8 リリース 8.0 または Oracle7 のデータベースに接続する場合 この情報を必要とする Oracle Enterprise Manager のような管理ツールを使用する場合 <p>追加情報: 6-62 ページの「リスナーの構成」を参照してください。</p>

変更点	Net8 リリース 8.0	Net8 リリース 8.1
LISTENER.ORA のプレゼンテーション・レイヤーとセッション・レイヤーに関する情報	<p>Net8 では、Two-Task Common (TTC) プレゼンテーション・レイヤーと Transparent Network Substrate (TNS) ネットワーク・セッション (NS) レイヤーを使用してデータベースに接続するクライアント・タイプのみをサポートしていた。この情報は、LISTENER.ORA ファイルに入れる必要はありません。</p>	<p>従来の Net8 Client のほかに、このリリースではデータベースの Java オプションに接続する IIOP クライアントもサポートしている。これらのクライアントでは、Two-Task Common (TTC) 以外に プレゼンテーション・レイヤー として General Inter-Orb Protocol (GIOP) が使用されます。また、セッション・レイヤー は必要ありません。</p> <p>TTC と NS はデフォルトなので、標準的な Net8 Client の場合、LISTENER.ORA にこれらのレイヤーを明示的に定義する必要はありません。IIOP クライアントがいる場合は、プレゼンテーション・レイヤー と セッション・レイヤー を明示的に定義する必要があります。</p> <p>例：</p> <pre>listener= (description= (protocol_stack= (presentation=giop) (session=raw)) (address= (protocol=tcp) (host=sales-pc) (port=2481)))</pre> <p>追加情報： 7-48 ページの「Java オプション接続の構成」を参照してください。</p>

変更点	Net8 リリース 8.0	Net8 リリース 8.1
CMAN.ORA の CMADMIN プロセスのアドレス	アドレスは、Oracle Connection Manager ゲートウェイ・プロセスの CMGW のみが必要とする。管理プロセスの CMADMIN は、内部 IPC アドレスを使用します。	CMADMIN と CMGW の両方にプロトコル・アドレスが必要。 この新しいアドレスは、次に示すように CMAN_ADMIN パラメータを使用して記述します。 <pre>cman=(address=(protocol=tcp)(host=anyhost)(port=1630))</pre> <pre>cman_admin=(address=(protocol=tcp)(host=cman-pc)(port=1650))</pre> デフォルトで CMADMIN は、登録済みポートの 1830 を使用して TCP/IP 上でリスニングします。 追加情報 ：次を参照してください。 <ul style="list-style-type: none">■ 概念的な情報については、2-46 ページの「Oracle Connection Manager プロセス」■ 構成に関する情報については、7-30 ページの「接続集中化機能の使用」

TNSNAMES.ORA の変更点

次の表は、Net8 リリース 8.1 で TNSNAMES.ORA ファイルに追加された新しいパラメータを説明しています。

パラメータ	説明
FAILOVER	<p data-bbox="461 404 1268 482">これを ON に設定すると、最初のリッスナーへの接続に失敗すると Net8 は別のリッスナーにフェイルオーバーする。このパラメータによって、次に示すようにアドレスの試行回数が決まります。</p> <pre data-bbox="461 499 819 713"> net_service_name= (description= (failover=on) (address=...) (address=...) (connect_data= (service_name=service_name))) </pre> <p data-bbox="461 748 1153 769">これを OFF に設定すると、Net8 は 1 つのアドレスのみを試行します。</p> <p data-bbox="461 786 1260 838">デフォルトでこのパラメータは、ADDRESS_LIST、DESCRIPTION_LIST および DESCRIPTION セットに対して ON になっています。</p> <p data-bbox="461 855 801 876">追加情報 : 次を参照してください。</p> <ul data-bbox="461 894 1253 963" style="list-style-type: none"> ■ 7-11 ページの「複数アドレス・オプションの構成」 ■ C-31 ページの「ローカル・ネーミング・パラメータ (TNSNAMES.ORA)」

パラメータ	説明
INSTANCE_NAME	<p>SERVICE_NAME パラメータのほかに、INSTANCE_NAME パラメータを使用してアクセス先のデータベース・インスタンスを識別することも可能。INSTANCE_NAME は、接続先の特定インスタンスを識別することが重要な意味を持つ Oracle Parallel Server データベースのみに必要です。たとえば、次に示すのは複数のインスタンス OP1 と OP2 にまたがるサービス、OP に関する記述です。エンド・ユーザーは、どちらのインスタンスを使用してもサービス OP に接続できます。</p> <pre data-bbox="506 477 856 980">net_service_name= (description= (address= (protocol=tcp) (host=opsnt1) (port=1521)) (connect_data= (service_name=op.com) (instance_name=op1)) net_service_name= (description= (address= (protocol=tcp) (host=opsnt1) (port=1521)) (connect_data= (service_name=op.com) (instance_name=op2))</pre> <p>INSTANCE_NAME は、一般にインストール中またはデータベースの作成中に入力する SID です。</p> <p>追加情報: インスタンス名文字列に関する情報については、7-11 ページの「複数アドレス・オプションの構成」を参照してください。</p>

パラメータ	説明
LOAD_BALANCE	<p>これを ON に設定すると、次に示すように Net8 はリスナー・アドレスのリストからランダムにアドレスを選択し、複数のリスナーに負荷を分散する。</p> <pre data-bbox="458 331 796 546"> net_service_name= (description= (load_balance=on) (address=...) (address=...) (connect_data= (service_name=sales.com))) </pre> <p>これを OFF に設定すると、Net8 はどれか 1 つのアドレスが成功するまでアドレス・リストに書かれている順番で試行します。</p> <p>このパラメータは、デフォルトで DESCRIPTION_LIST に対して ON になっています。</p> <p>追加情報: 次を参照してください。</p> <ul style="list-style-type: none"> ■ 7-11 ページの「複数アドレス・オプションの構成」 ■ C-31 ページの「ローカル・ネーミング・パラメータ (TNSNAMES.ORA)」
SERVICE_NAME	<p>次に示すように、接続先のサービスを識別する。</p> <pre data-bbox="458 869 822 1055"> net_service_name= (description= (address=...) (address=...) (connect_data= (service_name=sales.com))) </pre> <p>SERVICE_NAME は、一般にグローバル・データベース名に設定されます。グローバル・データベース名は、データベース名とドメイン名から構成され、インストール中またはデータベースの作成中に入力します。</p> <p>追加情報: サービス名文字列に関する情報については、6-2 ページの「サービス名とインスタンス名の設定」を参照してください。</p>

LISTENER.ORA の変更点

次の表は、Net8 リリース 8.1 で LISTENER.ORA ファイルに追加された新しいパラメータを説明しています。

パラメータ	説明
PROTOCOL_STACK	<p>接続に使用されるプレゼンテーション・レイヤーとセッション・レイヤーの情報を識別する。</p> <pre>listener= (description_list= (description= (protocol_stack= (presentation=giop) (session=raw)) (address_list= (address=...))))</pre> <p>デフォルトのプレゼンテーション・レイヤーである Two-Task Common (TTC) と、デフォルトのセッション・レイヤーであるネットワーク・セッション (NS) を使用している場合、PROTOCOL_STACK を使用する必要はありません。</p> <p>追加情報: サポートするプレゼンテーション・レイヤーとセッション・レイヤーに関する説明については、2-9 ページの「Oracle ネットワーク環境内のスタック通信」を参照してください。</p>
PRESENTATION	<p>プレゼンテーション・レイヤーを識別する。</p> <ul style="list-style-type: none"> 標準的な Net8 Client はプレゼンテーション・レイヤーに TTC を使用する Java クライアントは、General Inter-Orb Protocol (GIOP) というプレゼンテーション・レイヤーを必要とする
SESSION	<p>セッション・レイヤーを識別する。</p> <ul style="list-style-type: none"> 標準的な Net8 Client はセッション・レイヤーに NS を使用する Java クライアントは、セッション・レイヤーを使用しない RAW というプレゼンテーション・レイヤーを必要とする

CMAN.ORA の変更点

CMAN.ORA ファイルには、CMADMIN アドレスを定義するための新しいパラメータが追加されています。

パラメータ	説明
CMAN_ADMIN	<p>次に示すように、管理プロセスである CMADMIN プロセスのアドレスを識別する。</p> <pre>cman_admin= (address= (protocol=tcp) (host= cman-pc) (port=1650))</pre> <p>Oracle Connection Manager 制御ユーティリティの CMCTL が実行されているときに、CMADMIN プロトコル・アドレスを探してそのコマンドを実行します。</p> <p>デフォルトの TCP/IP アドレスであるポート 1830 を使用している場合、CMAN_ADMIN を明示的に指定する必要はありません。</p> <p>追加情報 : 次を参照してください。</p> <ul style="list-style-type: none">■ 7-30 ページの「接続集中化機能の使用」■ A-88 ページの「Oracle Connection Manager 制御ユーティリティ (CMCTL)」

リリース 8.1 で廃止されたパラメータ

次の表は、リリース 8.1 で廃止されたパラメータについて説明しています。

パラメータ	該当するファイル	説明
AUTOMATIC_IPC	SQLNET.ORA	セッションに強制的に IPC アドレスを使用させるために、このパラメータを使用していた。パフォーマンス上の問題から、このパラメータは破棄されました。AUTOMATIC_IPC のかわりに、IPC アドレスを構成する必要があります。
NAMES.USE_PLUG_AND_PLAY	NAMES.ORA	動的検出オプションを有効または無効にするために、このパラメータを使用していた。Oracle Names Server を検出するために別の方法が用意されています。 追加情報 : 6-21 ページの「 Oracle Names ネーミング・メソッドによるネットワークの構成 」を参照してください。
NAMES.DOMAIN_CHECKPOINT_FILE	NAMES.ORA	チェックポイント・トポロジー・データ (ローカル・リージョン内に存在するサーバーのドメイン・アドレス) に使用されていたファイルの名前。 この情報に関するチェックポイント・ファイルを作成するには、NAMES.REGION_CHECKPOINT_FILE パラメータを使用する必要があります。
NAMES.TOPOLOGY_CHECKPOINT_FILE	NAMES.ORA	チェックポイント・ドメイン・データ (リージョン内に存在するすべてのデータベース・アドレスとその他のデータ) に使用されていたファイルの名前。 この情報に関するチェックポイント・ファイルを作成するには、NAMES.REGION_CHECKPOINT_FILE パラメータを使用する必要があります。

Net8 のインストールとその製品

Oracle ユニバーサル・インストーラを実行する場合、多くのインストール・オプションから目的の製品を選択してインストールすることができます。それぞれのインストール・オプションには、インストール・タイプのサブセットがあります。「Custom」を選択すると、必要な製品を選択してインストールできます。次の表は、各インストール・タイプを選択したときにインストールされる製品を説明しています。

インストール・タイプ	説明
「Typical」	事前に構成された初期データベースのほか、ライセンスの付与が可能な Oracle オプションとカートリッジ、ネットワーク・サービス、Oracle8i ユーティリティ、およびオンライン・マニュアルがインストールされる。このインストール・タイプは、完全なデータベース・パッケージを希望するユーザーにお勧めします。
「Minimal」	事前に構成された初期データベース、ネットワーク・サービス、および Oracle8i ユーティリティをインストールするためのオプション。このインストール・タイプは、最小限度のデータベース・パッケージを希望するユーザーにお勧めします。
「Custom」	CD-ROM から任意の製品を選択してインストールすることが可能。

インストール中に、Net8 製品もインストールされます。この項では、Net8 のインストールに関連するトピックを取り上げます。

- [Net8 Server のインストール](#)
- [Net8 Client のインストール](#)
- [Net8 製品の概要](#)

Oracle ユニバーサル・インストーラ、インストール・プロセスおよびインストール・タイプに関する詳細は、使用するオペレーション・システムの『インストール・ガイド』を参照してください。

Net8 Server のインストール

Net8 Server のインストール中、次の製品によって構成ファイルにデフォルトの情報が設定されます。

- [Net8 Configuration Assistant](#)
- [Oracle Database Configuration Assistant](#)

Net8 Configuration Assistant

サーバーのインストール中、Net8 Configuration Assistant によって次の内容で構成ファイルが作成されます。

インストール・タイプ	インストールの結果、構成される情報
「Typical」	<p>Net8 Configuration Assistant はユーザーによる入力なしで実行され、次に示すデフォルトが構成される。</p> <ul style="list-style-type: none">■ 次に示す終点でリスニングする LISTENER という名前を持つリスナー用の LISTENER.ORA ファイル<ul style="list-style-type: none">- 外部プロシージャに接続するための IPC- Oracle データベースに接続する Net8 Client のための TCP/IP- Java オプションに接続する IIOP クライアントのための TCP/IP <p>外部プロシージャに関する静的な情報も、LISTENER.ORA ファイルに構成されます。</p> <ul style="list-style-type: none">■ 外部プロシージャのために構成されるネット・サービス名を含む TNSNAMES.ORA ファイル■ コンピュータのドメインとネーミング・メソッドの順序のために設定されたパラメータを含む SQLNET.ORA ファイル

インストール・タイプ インストールの結果、構成される情報

「Minimal」

Net8 Configuration Assistant はユーザーによる入力なしで実行され、次に示すデフォルトが構成される。

- 次に示す終点でリスニングする LISTENER という名前を持つリスナー用の LISTENER.ORA ファイル

- 外部プロシージャに接続するための IPC

- Oracle データベースに接続する Net8 Client のための TCP/IP

- Java オプションに接続する IIOP クライアントのための TCP/IP

外部プロシージャに関する静的な情報も、LISTENER.ORA ファイルに構成されます。

- 外部プロシージャのために構成されるネット・サービス名を含む TNSNAMES.ORA ファイル
- コンピュータのドメインとネーミング・メソッドの順序のために設定されたパラメータを含む SQLNET.ORA ファイル

「Custom」

Net8 Configuration Assistant によって、次の情報を構成するようにプロンプトが表示される。

- リスナー。リスナーの終点に使用する名前とプロトコルも含まれます。
- **ネーミング・メソッド (naming methods)**。ネット・サービス名をネットワーク・アドレスに変換します。

標準的な Net8 Client を TCP/IP 上でリスニングするようにリスナーを構成する場合、デフォルトのポート番号であるポート 1521 を選択する必要があります。そうでない場合は、INITSID.ORA ファイルの LOCAL_LISTENER パラメータを構成し、ネーミング・メソッドによってリスナー名を解決する必要があります。Net8 Client のために、LISTENER.ORA にデフォルト以外のポート番号を構成する方法の詳細については、6-67 ページの「**デフォルト以外のポートを使用するリスナーの構成**」を参照してください。

IIOP クライアントを TCP/IP 上でリスニングするようにリスナーを構成する場合、デフォルトのポート 2481 を選択する必要があります。同様に、IIOP クライアントを SSL 付き TCP/IP 上でリスニングするには、デフォルトのポート 2482 を選択する必要があります。そうでない場合は、INITSID.ORA ファイルで LOCAL_LISTENER パラメータを構成するか、または MTS_DISPATCHERS パラメータの LISTENER 属性を構成し、ネーミング・メソッドによってリスナー名を解決する必要があります。IIOP クライアントのために LISTENER.ORA にデフォルト以外のポート番号を構成する方法の詳細については、7-57 ページの「**独自のリスナーの構成**」を参照してください。

Oracle Database Configuration Assistant

「Typical」または「Minimal」でサーバーをインストールすると、Oracle Database Configuration Assistant によってデータベース関連の情報が LISTENER.ORA に追加されます。この情報は、Oracle Enterprise Manager を使用するとき Oracle Intelligent Agent が必要とします。

デフォルトでリスナーを構成する方法の詳細は、6-62 ページの「[リスナーの構成](#)」を参照してください。

Net8 Client のインストール

どのクライアント・インストール・タイプを選んでも、Net8 Configuration Assistant はユーザーによる入力なしで実行され、次のように構成ファイルが設定されます。

インストール・タイプ	インストールの結果、構成される情報
「Typical」	<p>Net8 Configuration Assistant はユーザーによる入力なしで実行され、次に示すデフォルトが構成される。</p> <ul style="list-style-type: none"> ■ 外部プロシージャのために構成されるネット・サービス名を含む TNSNAMES.ORA ファイル ■ コンピュータのドメインとネーミング・メソッドの順序のために設定されたパラメータを含む SQLNET.ORA ファイル
「Minimal」	<p>Net8 Configuration Assistant はユーザーによる入力なしで実行され、次に示すデフォルトが構成される。</p> <ul style="list-style-type: none"> ■ 外部プロシージャのために構成されるネット・サービス名を含む TNSNAMES.ORA ファイル ■ コンピュータのドメインとネーミング・メソッドの順序のために設定されたパラメータを含む SQLNET.ORA ファイル
「Custom」	<p>Net8 Configuration Assistant の実行中、ネット・サービス名をネットワーク・アドレスに解決するためのネーミング・メソッド (naming methods) を指定するように求めるプロンプトが表示される。</p> <p>ネット・サービス名の構成に関する詳細は、第 6 章の「ネーミング・メソッドとリスナーの構成」を参照してください。</p>

Net8 製品の概要

次の表は、Net8 製品、インストール先のコンピュータ、およびインストール・タイプの一覧です。

製品	説明	インストール先	選択可能なインストール・タイプ
Net8 Client	ネットワーク経由でクライアントからデータベースへの接続を実現する製品が提供される。クライアント側アプリケーションから Net8 に送られた接続要求は、ネットワークを経由してサーバーに渡されます。	サーバーとクライアント	すべて
Net8 Server	ネットワーク・リスナーが、プロトコルを介してネットワーク上のクライアント側アプリケーションの接続要求を受け入れるための製品が提供される。	サーバー	すべて
サポートするプロトコル			
■ TCP/IP	TCP/IP を使用するネットワーク上でクライアント / サーバー対話を可能にする。上記の Oracle 製品を組み合わせることで、クライアント上の Oracle アプリケーションは TCP/IP を介してリモートの Oracle データベースと通信できるようになります (Oracle データベースが、TCP/IP を使用したネットワーク通信をサポートするホスト・システム上で実行されている場合)	サーバーとクライアント	すべて
■ SSL 付き TCP/IP	TCP/IP と Secure Sockets Layer (SSL) を使用するネットワーク上でクライアント / サーバー対話を可能にする。上記の Oracle 製品を組み合わせることで、クライアント上の Oracle アプリケーションは TCP/IP と SSL を介してリモートの Oracle データベースと通信できるようになります (Oracle データベースが、TCP/IP と SSL を使用したネットワーク通信をサポートするホスト・システム上で実行されている場合)	サーバーとクライアント	すべて
■ SPX	SPX/IPX を使用するネットワーク上でクライアント / サーバー対話を可能にする。上記の Oracle 製品を組み合わせることで、クライアント上の Oracle アプリケーションは SPX/IPX を介してリモートの Oracle データベースと通信できるようになります (Oracle データベースが、SPX/IPX を使用したネットワーク通信をサポートするホスト・システム上で実行されている場合)。このプロトコルは、Novell の Netware で広く使用されています。	サーバーとクライアント	「Custom」

製品	説明	インストール先	選択可能なインストール・タイプ
<ul style="list-style-type: none"> ■ Named Pipes 	<p>Named Pipes を使用するネットワーク上でクライアント / サーバー対話を可能にする。上記の Oracle 製品を組み合わせることで、クライアント上の Oracle アプリケーションは Named Pipes を介してリモートの Oracle データベースと通信できるようになります (Oracle データベースが、Named Pipes を使用したネットワーク通信をサポートするホスト・システム上で実行されている場合)。</p> <p>Named Pipes は、クライアント / サーバー (分散アプリケーション) 間でプロセス間通信を提供する高水準のインタフェースです。一方のプロセス (サーバー側アプリケーション) でパイプを作り、他方のプロセス (クライアント側) が名前を使用してオープンします。これによって互いに、一方が書き込む情報を他方が読み取ることができます。Named Pipes は、特に PC LAN 環境で使用することを念頭に設計されています。</p>	<p>サーバーとクライアント</p>	<p>「Custom」</p>

製品	説明	インストール先	選択可能なインストール・タイプ
<ul style="list-style-type: none"> ■ Logical Unit Type 6.2 (LU6.2) 	<p>このプロトコルは、IBM Advanced Program-to-Program Communication (APPC) アーキテクチャの一部を構成する。</p> <p>APPC は、System Network Architecture (SNA) ネットワークのピアツーピア (プログラム間) で使用される IBM のプロトコルです。SNA は、国際標準化機構 (ISO) が策定した Open Systems Interconnect (OSI) モデルに類似した IBM の参照モデルです。</p> <p>APPC アーキテクチャでは、(端末 / ホスト間プロトコルのように) クライアントで端末エミュレーションを実行することなく、SNA ネットワークを介してクライアントとホストが通信できます。また、APPC アーキテクチャではピアツーピア通信が可能のため、クライアント側からサーバーに対して通信を開始できます。</p> <p>LU6.2 プロトコルと Physical Unit Type 2.1 (PU2.1) プロトコルを使用する SNA ネットワークでは、APPC が提供されます。LU6.2 プロトコルは 2 つのアプリケーション・プログラム間のセッションを定義します。LU6.2 は、製品に依存しないタイプの LU です。</p> <p>LU6.2 を使用すると、PC 上の Oracle アプリケーションは Oracle データベースと通信できるようになります。この通信は、SNA ネットワークを介して、APPC をサポートするホスト・システム上で動作する Oracle データベースとの間で発生します。</p>	サーバーとクライアント	「Custom」

製品	説明	インストール先	選択可能なインストール・タイプ
■ Bequeath	<p>クライアントが、ネットワーク・リスナーを使用せずに、データベースから情報を取り出せるようにする。bequeath プロトコルは、クライアント・アプリケーションごとにサーバー・スレッドを内部的に起動します。ある意味で bequeath プロトコルは、リモートのネットワーク・リスナーが接続を処理するのと同じ方法でローカルの接続を処理します。</p> <p>Bequeath の特徴は次のとおりです。</p> <ul style="list-style-type: none"> ■ ネットワーク・リスナーを使用しない（つまり、ネットワーク・リスナーを構成する必要もない）。 ■ SQL*Plus などの Oracle クライアント・アプリケーションが Oracle サーバーと通信する、同じマシン上のローカル接続で使用される。 ■ 専用サーバー・モードでのみ動作する。マルチスレッド・サーバー・モード（MTS）で使用することはできません。 ■ Windows NT 上でも動作し、32 ビット・アプリケーションをサポートする。また、Windows 95 上でも動作し、32 ビット・アプリケーションと Personal Oracle8i データベースの接続をサポートする。 ■ デフォルトで、Net8 Client にインストールされる。 	サーバーとクライアント	特定のインストール・タイプに依存しない（Net8 Client および Net8 Server と一緒にインストールされる）。

製品	説明	インストール先	選択可能なインストール・タイプ
構成作業用ツール	Net8 には、Oracle ネットワークの構成作業と管理作業を簡素化するための新機能が多数用意されている。		
<ul style="list-style-type: none"> ■ Net8 Configuration Assistant 	<p>インストールの終了後、Net8 Configuration Assistant によって必要なクライアント・コンポーネントとサーバー・コンポーネントが構成されます。次に示すページで説明しています。</p> <ul style="list-style-type: none"> ■ 4-23 ページの「Net8 Server のインストール」 ■ 4-26 ページの「Net8 Client のインストール」 <p>スタンドアロン・モードで実行し、ネーミング・メソッドとリスナーを構成することもできます。Net8 Configuration Assistant を起動するには、次の手順に従います。</p> <ul style="list-style-type: none"> ■ UNIX の場合、\$ORACLE_HOME/bin で netca を実行する ■ Windows NT の場合、「スタート」ボタン -> 「プログラム」 -> 「Oracle - HOME_NAME」 -> 「Network Administration」 -> 「Net8 Configuration Assistant」を選択する <p>追加情報 : 次を参照してください。</p> <ul style="list-style-type: none"> ■ 6-12 ページの「ローカル・ネーミング・メソッドによるネットワークの構成」 ■ 第 7 章の「Net8 のオプション機能」 	サーバーとクライアント	すべて
<ul style="list-style-type: none"> ■ Net8 Easy Config 	<p>Net8 Easy Config を使用して、TNSNAMES.ORA ファイルにネット・サービス名を設定できる。この製品の機能は、Net8 Configuration Assistant に追加されています。オラクル社では、Net8 Configuration Assistant の使用をお勧めしています。</p>	サーバーとクライアント	特定のインストール・タイプに依存しない (Oracle Net8 Assistant と一緒にインストールされる)

製品	説明	インストール先	選択可能なインストール・タイプ
<ul style="list-style-type: none"> ■ Oracle Net8 Assistant 	<p>Oracle Net8 Assistant を使用すると、次のネットワーク・コンポーネントを管理できる。</p> <ul style="list-style-type: none"> ■ プロファイル - ローカル・マシン上のクライアント用 SQLNET.ORA 構成ファイルを作成または変更し、認証、ドメイン、ロギング、トレース、セキュリティなどのクライアント機能を制御できる。 ■ ネット・サービス名 - ローカル名構成ファイルの TNSNAMES.ORA でネット・サービス名を作成または変更する。 ■ リスナー - LISTENER.ORA ファイルを作成または変更し、リスナーを制御する。 ■ Oracle Names Server - 集中管理された Oracle Names Server を作成または変更し、データベース・アドレスやネット・サービス名のようなネットワーク・オブジェクトを格納する。Oracle Net8 Assistant を使用して起動や停止などを実行し、ローカルまたはリモートから Oracle Names Server を管理することもできます。 <p>Oracle Net8 Assistant を起動するには、次の手順に従います。</p> <ul style="list-style-type: none"> ■ UNIX の場合、\$ORACLE_HOME/bin で netasst を実行する ■ Windows NT の場合、「スタート」ボタン -> 「プログラム」 -> 「Oracle - HOME_NAME」 -> 「Network Administration」 -> 「Net8 Assistant」を選択する <p>追加情報: 第 6 章の「ネーミング・メソッドとリスナーの構成」と第 7 章の「Net8 のオプション機能」を参照してください。</p>	<p>サーバーとクライアント</p>	<p>すべて</p>

製品	説明	インストール先	選択可能なインストール・タイプ
Oracle Names	<p>Oracle Names は、グローバルなクライアント / サーバー・コンピューティング・ネットワークを簡単に設定、管理できるように開発された Oracle 環境用分散ネーム・サービスである。Oracle Names は、Names サーバーから成る統合されたシステムを構築し、メンテナンスすることによりこれらの機能を実現します。Oracle Names Server はディレクトリ・サービスのように動作し、ネットワーク上に存在するすべてのデータベース・サービスのアドレスを格納して、接続を要求するクライアントに提供します。</p> <p>追加情報 : 6-21 ページの「Oracle Names ネーミング・メソッドによるネットワークの構成」を参照してください。</p>	Oracle Names Server マシン	「Custom」
Oracle Connection Manager	<p>Oracle Connection Manager は、Oracle8i に次の 3 種類の機能を提供する。</p> <ul style="list-style-type: none"> ■ 接続集中化 - 宛先 Oracle データベースへの単一のトランスポート接続を介して複数の論理クライアント・セッションを多重化する。 詳細は、7-30 ページの「接続集中化機能の使用」を参照してください。 ■ Net8 アクセス制御 (ファイアウォール・サポートとも呼ばれる) - データベース・サーバーへのアクセスを拒否または許可するプロキシを提供する。 詳細は、7-37 ページの「Net8 アクセス制御の使用」を参照してください。 ■ 複数プロトコル・サポート - クライアントとデータベース・サーバーが異なるプロトコル・スタック上で互いに通信できるようにする。 詳細は、7-34 ページの「複数プロトコル・サポートの使用」を参照してください。 	Oracle Connection Manager マシン	「Custom」

製品	説明	インストール先	選択可能なインストール・タイプ
外部ネーミング・サービス	Net8 では、次の外部ネーミング・メソッドを提供している。	サーバーとクライアント	「Custom」
<ul style="list-style-type: none"> ■ NetWare ディレクトリ・サービス (NDS) 	<p>NDS 外部ネーミングは、外部の NDS ネーミング表記法を使用して、Novell NDS を利用できるネットワーク上で Oracle データベースに接続できるようにします。</p> <p>追加情報 : 6-51 ページの「外部ネーミング・メソッドによるネットワークの構成」を参照してください。</p>	サーバーとクライアント	「Custom」
<ul style="list-style-type: none"> ■ ネットワーク情報サービス (NIS) 	<p>システム・インフラの一部としてすでにネットワーク情報サービス (NIS) を使用している組織や会社のために、NIS 外部ネーミングを使用して NIS 内に Oracle サービス別名とアドレスを格納するオプションが用意されている。</p> <p>追加情報 : 6-51 ページの「外部ネーミング・メソッドによるネットワークの構成」を参照してください。</p>	サーバーとクライアント	「Custom」
Oracle Advanced Security	Oracle Advanced Security は、次のコンポーネントから構成される。	サーバーとクライアント	「Custom」
<ul style="list-style-type: none"> ■ Network Security 	<p>Oracle のネットワーク・データの暗号化とチェックサム・サービスによって、ネットワーク上での安全なデータ送信が保証されます。Network Security では、RSA Data Security 社が提供する認証と暗号化のエンジンを使用します。</p> <p>次のアルゴリズムをサポートします。</p> <p>暗号化</p> <ul style="list-style-type: none"> ■ RC4_40 (米国内、輸出用およびアップグレード用) ■ RC4_56 (米国内および輸出用) ■ RC4_128 (米国内のみ) ■ DES_56 (米国内およびアップグレード用) ■ DES_40 (米国内、輸出用およびアップグレード用) <p>チェックサム</p> <ul style="list-style-type: none"> ■ MD5 (米国、輸出用およびアップグレード用) 	サーバーとクライアント	「Custom」

製品	説明	インストール先	選択可能なインストール・タイプ
■ Single Sign-On	<p>1 つのパスワードのみで複数のアカウントとアプリケーションにユーザーがアクセスできるようにする。この機能によって、ユーザーは複数のパスワードを維持する必要がなくなり、システム管理者のユーザー・アカウントとパスワードの管理作業を簡素化します。</p> <p>集中化された、安全な認証サービスによって、分散環境でのユーザー、クライアントおよびサーバーの識別情報に高度な信頼性が与えられます。また、ネットワーク認証サービスによって、単一サイン・オンという便宜がユーザーに与えられます。</p> <p>次の認証方法をサポートします。</p> <ul style="list-style-type: none"> ■ CyberSAFE ■ Identix (Biometric) ■ Kerberos ■ RADIUS ■ SecurID ■ SSL 	サーバーとクライアント	「Custom」
■ DCE Integration	<p>分散コンピューティング環境 (DCE) Integration によって、ユーザーは透過的に各種の Oracle ツールとアプリケーションを使用できるようになり、アプリケーションは DCE 環境において Oracle8i データベースにアクセスできるようになる。Oracle DCE Integration 製品は、2 つの主要なコンポーネントから構成されます。</p> <ul style="list-style-type: none"> ■ DCE Communications/Security method ■ DCE CDS (Cell Directory Service) Naming method 	サーバーとクライアント	「Custom」

アップグレードまたは移行の検討

この章では、SQL*Net と Net8 を共存させる場合の問題や、移行とアップグレードについて説明します。内容は次のとおりです。

- Net8 と SQL*Net の違い
- クライアントとデータベースの互換性問題
- Oracle Net8 Assistant による互換性問題の取り扱い
- 移行とアップグレードで考慮すべき点
- Net8 リリース 8.1 へのアップグレードと移行
- Oracle Connection Manager と Oracle Names でさらに考慮すべき点

Net8 と SQL*Net の違い

SQL*Net から Net8 への移行では、次の機能について留意してください。

機能	SQL*Net	Net8
複数プロトコル・サポート	MultiProtocol Interchange によって、アプリケーションは TNS ネットワークで異なるプロトコルを介して通信できる。	複数プロトコル・サポート機能を備える Oracle Connection Manager によって、異なるプロトコル・スタック上で動作するクライアントとデータベース・サーバーが互いに通信できる。 追加情報 : 7-34 ページの「 複数プロトコル・サポートの使用 」を参照してください。
構成作業用ツール	SQL*Net には、次のツールが用意されています。 <ul style="list-style-type: none"> ■ SQL*Net Easy Configuration。サービス名を作成します。 ■ Oracle Network Manager。詳細な構成作業に使用します。 	Net8 には、次のツールが用意されています。 <ul style="list-style-type: none"> ■ Net8 Configuration Assistant。リスナー、ネット・サービス名およびネーミング・メソッドの基本的な構成作業に使用します。 ■ Net8 Easy Config。ネット・サービス名を作成、変更、削除またはテストするために使用します。 ■ Oracle Net8 Assistant。詳細な構成作業やネットワーク管理に使用します。
外部プロシージャ	該当なし。	PL/SQL コードからコールできる、第三世代言語 (3GL) で書かれた外部プロシージャ、ファンクションまたはプロシージャのサポート。

Net8 にネットワークを移行することは必要不可欠ではありませんが、次の理由でお薦めします。

- 最小の構成。

Net8 は、ネットワーク・コンポーネントの設定プロセスを単純化します。Net8 を使用すると、クライアント、ネットワーク・リスナー、Oracle Names Server および Oracle Connection Manager をデフォルトの設定で起動できます。これにより構成ファイルを作成し、メンテナンスする必要性が最小限に抑えられます。

インストール・プロセスの一部として、Net8 Configuration Assistant は、単純な環境で必要となる Net8 構成ファイルの基本的パラメータを構成します。

Net8 では、Oracle Network Manager ですでに提供されている機能の大半が Oracle Net8 Assistant で置き換えられます。Oracle Net8 Assistant を使用して、既存の TNSNAMES.ORA ローカル名構成ファイル、SQLNET.ORA プロファイル、LISTENER.ORA リスナー・ファイル、および NAMES.ORA Oracle Names 構成ファイルを作成または編集します。

- 拡張ネットワークの機能。Net8 は、SQL*Net バージョン 2 ですでに提供されている機能の大半を同等の機能または拡張された機能で置き換えました。

次の表は、各 Oracle リリースでサポートされているネットワーク機能を示したものです。

データベースのリリース	7.1.4	7.1.5	7.1.6	7.2.2	7.2.3	7.3.2	7.3.3	7.3.4	8.x
SQL*Net	2.1.4	2.1.5	2.1.6	2.2.2	2.2.3	2.3.2	2.3.3	2.3.4	該当なし
Oracle Names Server	1.0	1.0	1.0	1.1	2.0	2.0.2	2.0.3	2.0.4	8.x
Oracle Advanced Networking Option	該当なし	該当なし	該当なし	該当なし	該当なし	2.3.2	2.3.3	2.3.4	8.0.x
Oracle Advanced Security	該当なし	8.1.x							
Secure Network Services ¹	1.0.1	1.0.2	1.0.3	1.1	2.0	該当なし	該当なし	該当なし	該当なし
Net8	該当なし	8.x							

¹ Secure Network Services と SQL*Net/DCE の機能は、Oracle Advanced Networking Option と Oracle Advanced Security に含まれています。

廃止された SQL*Net パラメータ

次の SQL*Net パラメータは、現在の Net8 で機能しません。

パラメータ	該当するファイル	説明
COMMUNITY	TNSNAMES.ORA	COMMUNITY パラメータは、すべてのネットワーク・サービス・アドレスに必要であった。そのため、アドレスを検索すると現れます（たとえば、ローカル・ネーミングやリスナー構成ファイル）。
NAMES.DEFAULT_ZONE	SQLNET.ORA	NAMES.DEFAULT_ZONE は、NAMES.DEFAULTDOMAIN パラメータと NAMES.PREFERRED_SERVERS パラメータのわずかに異なるものとしてプロファイルに含まれていた。

これらのパラメータは削除しても、残しておいても構いません。これらのパラメータは不要なものです。ネットワーク操作に干渉するものではありません。

クライアントとデータベースの互換性問題

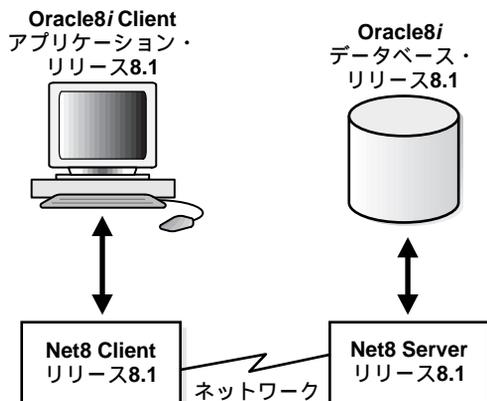
自分の環境でアップグレードまたは移行することが適切な場合は、次に示すクライアントとデータベースの接続上の問題について考慮した上で決定してください。

- Oracle8i Client リリース 8.1 から Oracle8i データベース・リリース 8.1 への接続
- Oracle8 リリース 8.0 または Oracle7 Client から Oracle8i データベース・リリース 8.1 への接続
- Oracle8i Client リリース 8.1 から Oracle8 データベース・リリース 8.0 または Oracle7 データベースへの接続
- Oracle Names

Oracle8i Client リリース 8.1 から Oracle8i データベース・リリース 8.1 への接続

図 5-1 に示すように、Oracle8i Client リリース 8.1 には Net8 Client リリース 8.1 が、Oracle8i データベース・リリース 8.1 には Net8 Server リリース 8.1 がそれぞれ必要です。

図 5-1 Oracle8i Client から Oracle8i データベースへの接続

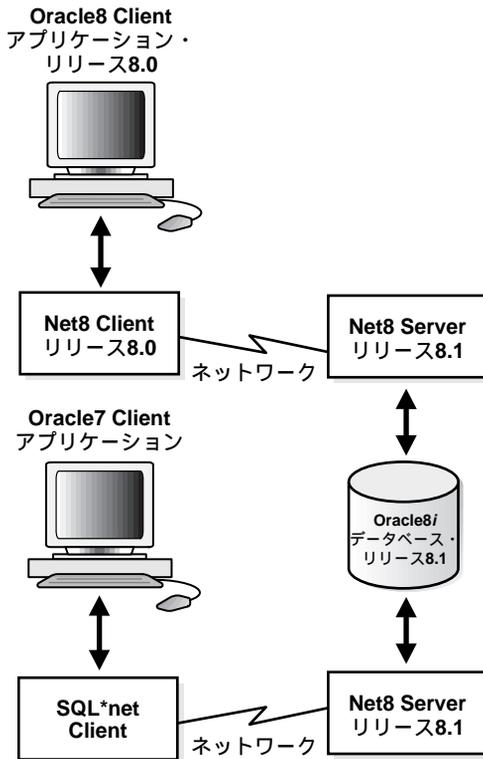


Oracle8i Client には、6-12 ページの「ローカル・ネーミング・メソッドによるネットワークの構成」の説明に従ってサービス名を構成する必要があります。

Oracle8 リリース 8.0 または Oracle7 Client から Oracle8i データベース・リリース 8.1 への接続

図 5-2 に示すように、Oracle8 Client リリース 8.0 には互換リリースの Net8 Client が、Oracle7 Client には SQL*Net Client が、そして Oracle8i データベースには Net8 Server リリース 8.1 がそれぞれ必要です。

図 5-2 Oracle8 Client リリース 8.0 または Oracle7 Client から Oracle8i データベースへの接続



データベースの Oracle システム識別子 (SID) をサービス名で置き換えることは必要不可欠ではありませんが、新しい機能を利用するためにサービス名の使用をお勧めします。たとえば、次のように指定します。

```
net_service_name=  
(description=  
  (address=...)   
  (address=...)   
  )   
(connect_data=  
  (service_name=sales.com)===  
  )
```

Oracle Net8 Assistant の互換モードを使用して、SID=*SID* を SERVICE_NAME=*SERVICE_NAME* と置き換えることにより、この結果を得ることができます。5-11 ページの「[Oracle Net8 Assistant による互換性問題の取り扱い](#)」を参照してください。

Oracle7 クライアントから Oracle8i リリース 8.1 データベースに接続する環境では、次の予期される質問について検討する必要があります。

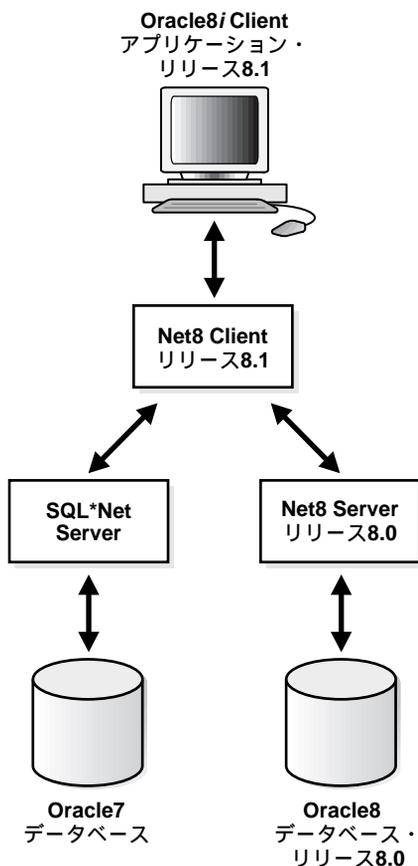
- 現在使用中のサードパーティ製 Oracle7 アプリケーションから Net8 の機能を活用できるか。
活用できません。Net8 のライブラリを利用するには、アプリケーションを再構築するかアップグレードする必要があります。
- リモートの Oracle8i データベースに接続するには、現在使用中の Oracle7 クライアントで Net8 Client が必要か。
必要ありません。Oracle7 クライアントからリモートの Oracle8i データベースへの接続が必要な場合、SQL*Net Client リリース 2.x を Oracle7 クライアントに構成するだけでかまいません。Net8 は、SQL*Net Client リリース 2.x と下位互換です。唯一の制限事項は、Net8 から新たに追加されたネットワーク機能がこの接続タイプで利用できないことです。
- ローカルの Oracle8i データベースに接続するには、現在使用中の Oracle7 クライアントに Net8 Client が必要か。
必要です。Oracle7 クライアントからローカルの Oracle8i データベースに接続する場合、SQL*Net Client リリース 2.x、Net8 Client および Net8 Server が同じシステム上に必要です。ただし、Net8 Client と Net8 Server は Oracle8i Enterprise Edition または Oracle8i をインストールしたときに一緒にインストールされています。

Oracle8i Client リリース 8.1 から Oracle8 データベース・リリース 8.0 または Oracle7 データベースへの接続

図 5-3 に、この接続を示します。

- Oracle8i Client リリース 8.1 には、Net8 Client リリース 8.1 が必要
- Oracle7 Server には、SQL*Net Server が必要
- Oracle8 データベース・リリース 8.0 には、Net8 Server リリース 8.0 が必要

図 5-3 Oracle8i Client から Oracle8 リリース 8.0 または Oracle7 データベースへの接続



リリース 8.1 クライアントは、データベースの SID を使用して構成する必要があります。たとえば、TNSNAMES.ORA は次のようになります。

```
net_service_name=  
(description=  
  (address=...)  
  (address=...)  
  )  
(connect_data=  
  (sid=sales)  
  )
```

Oracle Net8 Assistant の互換モードを使用しても、これと同じ結果を得ることができます。5-14 ページの「[リリース 8.0 または 7.x の機能の利用と、リリース 8.0 または 7.x のサービスへの接続](#)」を参照してください。

6-71 ページの「[サービス情報によるリスナーの構成](#)」に説明されているように、データベース・サーバーの LISTENER.ORA は、SID を記述して構成する必要があります。

Oracle8i クライアント・リリース 8.1 から Oracle7 データベースに接続する環境では、次の予期される質問について考慮してください。

- リモートの Oracle7 データベースに接続するには、使用中の Oracle8i クライアントに SQL*Net Client リリース 2.x が必要か。

必要ありません。Oracle8i クライアントがリモートの Oracle7 データベースに接続する必要がある場合は、Oracle8i クライアント上で Net8 クライアントを構成するだけでかまいません。SQL*Net リリース 2.x は、Net8 と上位互換です。唯一の制限事項は、Net8 から新たに追加されたネットワーク機能がこの接続タイプで利用できないことです。

- ローカルの Oracle7 データベースに接続するには、使用中の Oracle8i クライアントに SQL*Net Client リリース 2.x が必要か。

必要です。Oracle8i クライアントからローカルの Oracle7 データベースへの接続が必要な場合、SQL*Net Server リリース 2.x と Net8 Client の両方が同じシステム上に必要です。

Oracle Names

使用中のネットワークの全部または一部を Net8 に移行あるいはアップグレードする場合、リージョン内にあるすべての Oracle Names Server をバージョン 8 に移行するかアップグレードする必要があります。

- 使用中の Oracle7 クライアントは、Oracle Names バージョン 8 を使用してサービス名を解決できるか。
解決できません。
- 使用中の Oracle7 クライアントは、Oracle Names バージョン 8 から戻された接続文字列を使用して Oracle7 データベースか Oracle8i データベースに接続できるか。

Oracle Names に入力されたときに接続文字列が正しく指定されていれば、接続できます。つまり、TNSNAMES.ORA ファイルの接続文字列が動作すれば、Oracle Names でも動作します。

Oracle Net8 Assistant による互換性問題の取り扱い

一部のパラメータはリリース 8.1 でのみ使用できるので、Oracle Net8 Assistant では TNSNAMES.ORA ファイルに適切なパラメータを設定するために、Net8 リリース 8.1 クライアントと、Net8 リリース 8.0 および SQL*Net リリース 2.x クライアントのために 2 種類のオプションが用意されています。

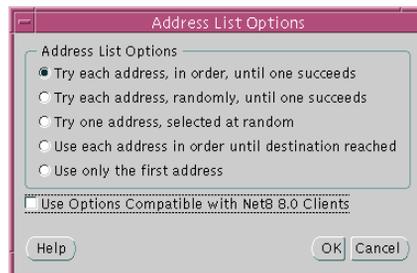
Oracle Net8 Assistant オプション	説明
「Use Options Compatible with Net8 8.0 Clients」	<p>複数のアドレスに対してリリース 8.1 のオプションを使用可能または使用不可にする。</p> <p>このオプションをオンにすると、リリース 8.1 より前のクライアントの接続で、Oracle Connection Manager パラメータの SOURCE_ROUTE のみ選択できます。</p> <p>このオプションをオフにすると、リリース 8.1 クライアントの接続で、SOURCE_ROUTE、LOAD_BALANCE および FAILOVER の各パラメータを有効または無効にできます。</p> <p>追加情報: 7-11 ページの「複数アドレス・オプションの構成」を参照してください。</p>
「Use Oracle8i Release 8.0 Compatible Identification」	<p>TNSNAMES.ORA の CONNECT_DATA セクションで、リリース 8.1 の詳細設定を使用可能または使用不可にする。</p> <p>このオプションをオンにすると、リリース 8.1 より前のデータベースの SID を入力できます。</p> <p>このオプションをオフにすると、リリース 8.1 のサービス名 (SERVICE_NAME) を入力できます。</p> <p>注意: このオプションがオンまたはオフであるかによって、「Advanced Service Options」ダイアログ・ボックスも影響を受けます。このダイアログ・ボックスは、「Service Identification」グループの「Advanced」ボタンが選択されているときに表示されます。一部の設定は、リリース 8.1 サービスに対する接続でのみ利用できます。</p> <p>追加情報: 7-7 ページの「ネット・サービス名の接続属性の詳細構成」を参照してください。</p>

リリース 8.1 機能の利用と、リリース 8.1 サービスへの接続

リリース 8.1 より前の構成ファイルを使用する場合は、TNSNAMES.ORA を変更することにより新機能を利用したり、リリース 8.1 サービスに接続したりできます。TNSNAMES.ORA に対するこれらの変更は、手動で行うか Oracle Net8 Assistant を使用して実行できます。

古い構成ファイルを構成してリリース 8.1 機能を利用できるようにするには、Oracle Net8 Assistant を使用して次の手順を実行します。

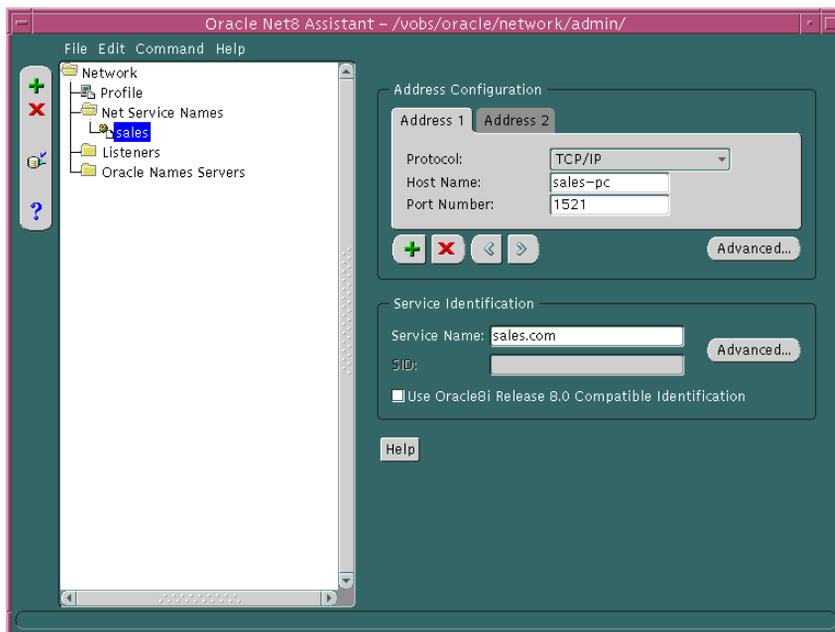
1. Oracle Net8 Assistant を起動します。
 - UNIX の場合、`$ORACLE_HOME/bin` で `netasst` を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」-> 「Oracle - HOME_NAME」-> 「Network Administration」-> 「Net8 Assistant」を選択します。
2. ディレクトリ・ツリーの「Net Service Names」フォルダをダブルクリックします。
3. 互換モードを適用するネット・サービス名を選択します。
4. 選択したネット・サービス名に複数のアドレスを構成する場合は、クライアント負荷のバランス、connect-time failover および Oracle Connection Manager の各オプションを設定できます。
 - a. 「Address Configuration」グループの「Advanced」ボタンをクリックします。
「Address List Options」ダイアログ・ボックスが表示されます。



- b. 「Use Options Compatible with Net8 8.0 Clients」がオフになっていることを確認します。
- c. 任意のアドレス・リスト・オプションを選択して「OK」をクリックします。

これらのオプションに関する詳細は、7-11 ページの「複数アドレス・オプションの構成」を参照してください。

5. リリース 8.1 サービスに接続します。



- a. 「Service Identification」グループ・ボックスの「Use Oracle8i Release 8.0 Compatible Identification」がオフになっていることを確認します。
- b. 「Service Name」フィールドに接続先のサービス名を入力します。

このサービス名は、一般にグローバル・データベース名です。グローバル・データベース名は、データベース名とドメイン名から構成され、インストール中またはデータベースの作成中に入力します。

サービス名の値に関する詳細は、次を参照してください。

- サービス名を設定するには、6-2 ページの「サービス名とインスタンス名の設定」と 5-17 ページの「手順 1: サービス名とインスタンス名を確認する」

- オプションの接続設定に関しては、7-7 ページの「ネット・サービス名の接続属性の詳細構成」

6. 「File」メニューから「Save Network Configuration」を選択します。
7. 「File」メニューから「Exit」を選択します。

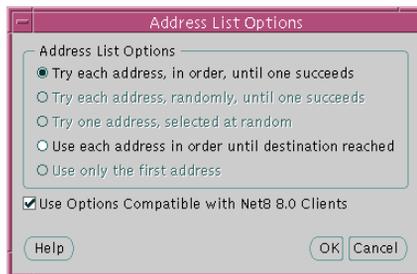
Oracle Net8 Assistant アプリケーションが終了します。

リリース 8.0 または 7.x の機能の利用と、リリース 8.0 または 7.x のサービスへの接続

リリース 8.0 または 7.x のサービスに接続するクライアントを構成する場合、TNSNAMES.ORA にデータベースの SID を設定する必要があります。

リリース 8.0 機能を構成するには、Oracle Net8 Assistant を使用するかまたは手動で TNSNAMES.ORA ファイルを編集します。

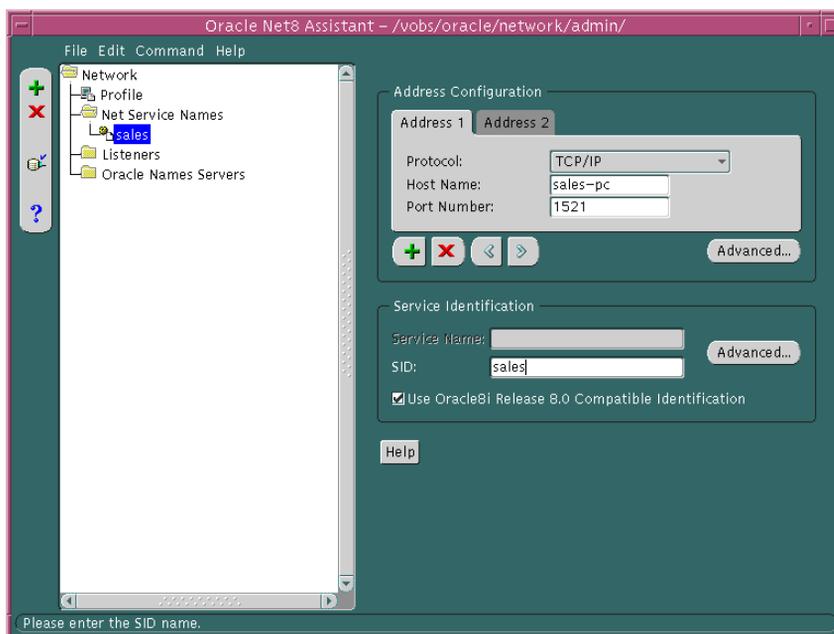
1. Oracle Net8 Assistant を起動します。
 - UNIX の場合、`$ORACLE_HOME/bin` で `netasst` を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」 -> 「Oracle - HOME_NAME」 -> 「Network Administration」 -> 「Net8 Assistant」を選択します。
2. ディレクトリ・ツリーの「Net Service Names」フォルダをダブルクリックします。
3. 互換モードを適用するネット・サービス名を選択します。
4. 選択したネット・サービス名に複数のアドレスを構成する場合は、Oracle Connection Manager を構成できます。
 - a. 「Address Configuration」グループの「Advanced」をクリックします。
「Address List Options」ダイアログ・ボックスが表示されます。



- b. 「Use Options Compatible with Net8 8.0 Clients」がオンになっていることを確認します。
- c. 「Use each address in order until destination reached」を選択します。「Try each address, in order, until one succeeds」オプションを選択すると、ソース・ルート・オプションはオフになります。このほかに、connect-time failover とクライアント負荷のバランスのオプションがありますが、8.1 より前のクライアントは利用できません。

Oracle Connection Manager の構成に関する詳細は、7-40 ページの「[Oracle Connection Manager 機能を使用する場合のクライアントの構成](#)」を参照してください。

5. リリース 8.0 またはリリース 7.x サービスに接続します。



- a. 「Use Oracle8i Release 8.0 Compatible Identification」がオンになっていることを確認します。
 - b. 「SID」フィールドに接続先 SID の名前を入力します。
6. 「File」メニューから「Save Network Configuration」を選択します。
 7. 「File」メニューから「Exit」を選択します。

Oracle Net8 Assistant アプリケーションが終了します。

移行とアップグレードで考慮すべき点

次に示すいずれかの方法を使用して Net8 をインストールします。

インストールの種類 説明	
別の Oracle ホームにインストールする	同じサーバー上の別の Oracle ホームにリリース 8.1.5 をインストールする。
移行	移行とは、SQL*Net リリース 2.x から Net8 に移るプロセスのことをいう。移行プロセスには、Net8 のインストールと SQL*Net リリース 2.x の削除が含まれます。
アップグレード	アップグレードとは、同じバージョン内でデータベースのリリースを変更するプロセスのことをいう。たとえば、Net8 リリース 8.0.5 から Oracle8i データベース・リリース 8.1.5 への変更は、データベース・システムのアップグレードです。

注意： リリース 8.1.5 製品を既存の Oracle ホームにインストールすることはできません。

Net8 リリース 8.1 へのアップグレードと移行

Net8 リリース 8.0 から Net8 リリース 8.1 にアップグレードするか、または SQL*Net リリース 2.x から Net8 リリース 8.1 に移行するには、次の手順に従います。

- 手順 1: サービス名とインスタンス名を確認する
- 手順 2: サーバー上でソフトウェアをアップグレードまたは移行する
- 手順 3: クライアント上でソフトウェアをアップグレードまたは移行する
- 手順 4: 機能のアップグレードと移行を実行する

手順 1: サービス名とインスタンス名を確認する

Net8 リリース 8.1 の機能を利用して TNSNAMES.ORA ファイルのサービスとインスタンスを識別する場合は、データベース初期化ファイル (INITSID.ORA) に SERVICE_NAMES パラメータと INSTANCE_NAMES パラメータが設定されていることを確認してください。

パラメータ	説明
SERVICE_NAMES	ネットワーク上のサービスの名前を表す 1 つ以上のカンマで区切られた文字列。グローバル・データベース名には、サービスのいずれか 1 つを設定する必要があります。グローバル・データベース名は、データベース名 (DB_NAME) とドメイン名 (DB_DOMAIN) から構成される名前です。
INSTANCE_NAME	インスタンスの名前を表す文字列値。複数インスタンスが同じサービス名を共有する場合、特定のインスタンスを識別するために使用されます。INSTANCE_NAME を SID と混同しないでください。SID は、実際に 1 台のホスト上にあるインスタンス共有メモリーを一意に識別します。 INSTANCE_NAME は、SID の値に設定する必要があります。

手順 2: サーバー上でソフトウェアをアップグレードまたは移行する

注意: リリース 8.1.5 を、前のリリースの Oracle8i データベースを使用して作成された Oracle ホームにインストールすることはできません。また、リリース 8.1.5 より前のリリースの Oracle8i データベースを、リリース 8.1.5 によって作成された Oracle ホームにインストールすることもできません。

データベース・サーバー上でソフトウェアをアップグレードまたは移行するには、次の作業を実行します。

Oracle Universal Installer から最新リリースの Net8 Server をインストールし、最新の実行ファイルを取り込みます。

Net8 Sever が Oracle8i データベースと一緒にインストールされている場合、Oracle Universal Installer によってリリース 8.0 データベースがシステム上で検出されると、Oracle Data Migration Assistant を使用してデータベースをアップグレードするようにプロンプトが表示されます。インストール・プロセスの途中でアップグレードしない場合は、このアシスタントをインストールして後で使用することができます。

Oracle Universal Installer によって、次の作業が自動的に実行されます。

- Net8 リリース 8.0 または SQL*Net リスナーを停止する
- Net8 リリース 8.1 リスナーを起動する

手順 3: クライアント上でソフトウェアをアップグレードまたは移行する

クライアント上でソフトウェアをアップグレードまたは移行するには、次の作業を実行します。

Oracle Universal Installer から最新のリリースの Net8 Client をインストールし、最新の実行ファイルを取り込みます。

手順 4: 機能のアップグレードと移行を実行する

ソフトウェアのアップグレードと移行が終了した後、リリース 8.1 の新機能を使用しないのであれば構成ファイルをアップグレードする必要はありません。新機能を利用するには、次の作業を実行します。

変更対象の構成ファイル	必要な変更作業
クライアントとサーバー上の SQLNET.ORA	AUTOMATIC_IPC を削除する。このパラメータはセッションによる IPC の使用を強制しますが、これは無視されます。
TNSNAMES.ORA	<p>SID を SERVICE_NAME パラメータに置き換えて、リリース 8.1 サービスに接続する。</p> <pre>net_service_name= (description= (address=...) (address=...)) (connect_data=(service_name=sales.com))</pre> <p>SERVICE_NAME は、一般にグローバル・データベース名です。グローバル・データベース名は、データベース名とドメイン名から構成され、インストール中またはデータベースの作成中に入力します。</p> <p>追加情報：</p> <ul style="list-style-type: none"> ■ サービス名の構成に関する情報は、5-11 ページの「Oracle Net8 Assistant による互換性問題の取り扱い」を参照してください。 ■ サービス名文字列に関する情報は、6-2 ページの「サービス名とインスタンス名の設定」を参照してください。 <p>クライアント負荷のバランスまたは connect-time failover、あるいはその両方を有効にする。</p> <p>追加情報： 次の情報を参照してください。</p> <ul style="list-style-type: none"> ■ ADDRESS_LIST に LOAD_BALANCE または FAILOVER、あるいはその両方を設定するには、5-11 ページの「Oracle Net8 Assistant による互換性問題の取り扱い」 ■ 7-11 ページの「複数アドレス・オプションの構成」

変更対象の構成ファイル **必要な変更作業**

LISTENER.ORA

インスタンス情報はリリース 8.1 のリスナーに登録されるため、LISTENER.ORA ファイルの `SID_LIST_listener_name` セクションにインスタンス情報を含める必要はない。

ただし、まだリリース 8.1.5 では Oracle Enterprise Manager をはじめとする多くの管理用ツールでは、この情報が必要です。Oracle Enterprise Manager を使用してデータベース・オブジェクトを管理する場合は、次に示すように LISTENER.ORA ファイルにデータベースに関する情報を構成する必要があります。

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

`connect-time failover` をインプリメントすると、LISTENER.ORA ファイルに静的なサービス構成パラメータは使用できなくなります。ただし、Oracle8 リリース 8.0、または Oracle7 データベースと Oracle Enterprise Manager で静的な構成が必要です。リスナーを静的に構成する方法の詳細は、6-71 ページの「サービス情報によるリスナーの構成」を参照してください。

Oracle Enterprise Manager 環境で Oracle Parallel Server に `connect-time failover` をセットアップするための構成方法の詳細は、『Oracle8i Parallel Server セットアップおよび構成ガイド』を参照してください。

追加情報 : 6-71 ページの「サービス情報によるリスナーの構成」を参照してください。

Oracle Connection Manager と Oracle Names の移行に関する詳細は、次の項の「Oracle Connection Manager と Oracle Names でさらに考慮すべき点」を参照してください。

Oracle Connection Manager と Oracle Names でさらに考慮すべき点

現在 Oracle MultiProtocol Interchange または Oracle Names を使用している場合は、次のトピックを参照してください。

- [Oracle Connection Manager リリース 8.1 の使用](#)
- [Oracle MultiProtocol Interchange にかわる Oracle Connection Manager の使用](#)
- [Oracle Names バージョン 8 の使用](#)

Oracle Connection Manager リリース 8.1 の使用

Oracle Connection Manager の管理プロセスである CMADMIN は、管理コマンドを実行する CMCTL 制御ユーティリティによって使用されます。

リリース 8.1 の新しい構成では、CMADMIN のプロトコル・アドレスが必要です。デフォルトのアドレスは、次のとおりです。

```
cman_admin=  
(address=  
  (protocol=tcp)  
  (host= host)  
  (port=1830)  
)
```

Oracle Connection Manager リリース 8.1 は、リリース 8.0 と下位互換です。これは、次のことを意味します。

- Oracle Connection Manager リリース 8.0 は、Oracle8i リリース 8.1 データベースにアクセスできません。古い CMAN.ORA ファイルが使用されている場合、CMADMIN プロセスはポート 1830 を使用して TCP/IP 上でリスニングします。ポート 1830 がすでに使用されている場合、別の CMAN_ADMIN アドレスを使用して CMAN.ORA を手動で構成してください。
- Oracle Connection Manager リリース 8.1 は、Oracle8 リリース 8.0 データベースにアクセスできます。ただし、機能的にはリリース 8.0 に戻ります。
- Oracle8 リリース 8.0 クライアントは、追加の構成作業なしに Oracle Connection Manager リリース 8.1 にアクセスできます。

Oracle MultiProtocol Interchange にかわる Oracle Connection Manager の使用

クライアントは Net8 に移行済みだが、さらに複数プロトコル・サポートが必要な場合は、Oracle Connection Manager をインストールし、セッションを Oracle Connection Manager 経由にする必要があります。Oracle Multiprotocol Interchange は、SQL*Net リリース 2.3 以降サポートされていません。

Oracle Connection Manager が正しく機能するために、次のファイルがこれらのノードから削除されていることを確認してください。

- TNSNET.ORA
- TNSNAV.ORA
- INTCHG.ORA

ユーザーのネットワーク構成に特有の移行に関するその他の考慮事項があります。

複数プロトコルのサポートに関する詳細は、7-34 ページの「[複数プロトコル・サポートの使用](#)」を参照してください。

Oracle Names バージョン 8 の使用

Oracle Names バージョン 8 は、SQL*Net バージョン 2 と下位互換です。これは、SQL*Net バージョン 2 で動作するクライアントが Oracle8i データベースに接続するために Oracle Names バージョン 8 を使用して Oracle Names Server にアクセスできることを意味します。

Oracle Names バージョン 8 で提供されている新しい機能を利用するには、既存 Oracle Names Server のノードすべてに Oracle Names バージョン 8 をインストールして、リージョン内にある既存の全 Oracle Names Server をバージョン 8 に移行する必要があります。

移行するときの注意点は、次のとおりです。

- [データベースによる Oracle Names バージョン 2 からの移行](#)
- [動的検出オプションによる Oracle Names バージョン 2 からの移行](#)
- [ROSFILES からデータベースの Oracle Names 表への移行](#)
- [Oracle Names バージョン 8 に確実に正しく移行するためのチェックリスト](#)

データベースによる Oracle Names バージョン 2 からの移行

既存の Oracle Names Server データベースからバージョン 8 データベースに移行し、データを転送するには、Oracle Network Manager によってネットワーク定義が格納されているノード上で、UNIX の場合は \$ORACLE_HOME/network/admin にある NAMESUPG.SQL スクリプトを実行し、Windows NT の場合は `ORACLE_HOME\network\admin` のものを実行します。

```
sql> CONNECT user/password
sql> @oracle_home/network/admin/namesupg.sql;
```

動的検出オプションによる Oracle Names バージョン 2 からの移行

動的検出オプションを使用して Oracle Names バージョン 2 を移行する手順は、Oracle Names バージョン 8 にリージョン・データベースの情報を格納するかどうかによって異なります。

- リージョン以外のデータベースの移行
- リージョン・データベースの移行

リージョン以外のデータベースの移行 動的検出オプションを使用して Oracle Names バージョン 2 から Oracle Names バージョン 8 に移行する場合、新規の Oracle Names Server は古いチェックポイント・ファイルから登録データを取得する必要があります。なんらかの理由でデータが登録されていない場合は、6-40 ページの「[手順 5: リスナーを構成する](#)」の手順に従って USE_PLUG_AND_PLAY パラメータでリスナーを正しく構成し、6-41 ページの「[手順 6: Oracle Names Server にデータベースを登録する](#)」の手順に従ってオブジェクトを登録できます。

リージョン・データベースの移行 動的検出オプションを使用してすでに Oracle Names バージョン 2 が実行されている状態で、Oracle Names 情報用リポジトリとしてデータベースを構成する場合は、次の手順を実行する必要があります。

1. Oracle Names バージョン 2 のローカル管理リージョンに格納されている情報を TNSNAMES.ORA ファイルに書き込みます。コマンド行から、次のように入力します。

```
namesctl  
namesctl> dump_tnsnames
```

2. データベースの格納先にする Oracle Names Server ノード上で、\$ORACLE_HOME/network/admin (UNIX の場合) または ORACLE_HOME¥network¥admin (Windows NT の場合) にある NAMESINI.SQL スクリプトを実行します。

```
sql> CONNECT user/password  
sql> @oracle_home/network/admin/namesini.sql;
```

- Oracle Net8 Assistant または NAMESCTL 制御ユーティリティを使用して、TNSNAMES.ORA ファイルをバージョン 8 の Oracle Names Server にロードします。

Oracle Net8 Assistant による操作	制御ユーティリティの NAMESCTL による操作
<ol style="list-style-type: none"> Oracle Net8 Assistant を起動します。 -UNIX の場合、\$ORACLE_HOME/bin で <code>netasst</code> を実行します。 -Windows NT の場合、「スタート」ボタン -> 「プログラム」-> 「Oracle - HOME_NAME」-> 「Network Administration」-> 「Net8 Assistant」を選択します。 Oracle Names Server フォルダをダブルクリックします。 Oracle Names Server を選択します。 ドロップダウン・リスト・ボックスから「Manage Data」を選択します。 「Load」を選択します。 Oracle Network Manager によって生成された TNSNAMES.ORA ファイルのパスとファイル名を「File」フィールドに入力します。 「Execute」を選択します。 「File」メニューから「Save Network Configuration」を選択します。 「File」メニューから「Exit」を選択し、Oracle Net8 Assistant アプリケーションを終了します。 	<p>コマンド行から、次のように入力します。</p> <pre>namesctl namesctl> load_tnsnames file_name</pre>

- Oracle Net8 Assistant を使用して、すべての Oracle Names Server 構成ファイル (NAMES.ORA) に NAMES.ADMIN_REGION パラメータを設定します。6-25 ページの「[手順 3: Oracle Names Server を作成する](#)」を参照してください。
- Oracle Names ノードの 1 つで NAMESCTL REORDER_NS コマンドを実行し、少なくとも 1 つの Oracle Names Server に Oracle Names Server 情報を再ロードします。6-36 ページの「[手順 4: Oracle Names Server を使用するためにクライアントとデータベース・サーバーを構成する](#)」を参照してください。

ROFILES からデータベースの Oracle Names 表への移行

Oracle Names バージョン 8 では、Resource Object Store (ROS) ファイル (ROFILES) を使用する古い構成をサポートしていません。ROFILES は、直接 Oracle Names データベース表に移行するか、または最初に TNSNAMES.ORA ファイルに移行してから Oracle Names に移行する必要があります。

- [ROFILES からデータベース表](#)
- [ROFILES から TNSNAMES.ORA ファイル](#)

ROFILES からデータベース表 ROFILES をデータベース表に移行するには、次の手順に従います。

1. Oracle Network Manager 用のデータベース・ユーザー・アカウントを作成します。

```
sql> CONNECT system/password
sql> CREATE USER user
      identified by password
      default tablespace users
      temporary tablespace temp;
```

2. このユーザーに SYSDBA 権限を付与します。

```
sql> GRANT sysdba to username identified by password;
```

3. 必要な表を作成するには、次のスクリプトをサーバーに対して実行する必要があります。一般に、これらのスクリプトは Oracle Network Manager ノード上で実行します。

```
sql> CONNECT user/password
sql> @oracle_home%db%rosbild.sql;
sql> @oracle_home%db%nmcbild.sql;
sql> @oracle_home%db%rosgmnt.sql;
sql> @oracle_home%db%nmcgrnt.sql;
```

スクリプト	説明
ROSBILD.SQL	ROS が使用する表を作成する。
NMCBILD.SQL	Oracle Network Manager Objects (NMO) コンポーネントが使用する表を作成する。
ROSGRNT.SQL	共通表にアクセスするユーザーにアクセスを許可する。ユーザー名の入力を求めるプロンプトが表示されます。Oracle Network Manager アカウントをセットアップするときに使用したユーザー名を入力します。
NMCGRNT.SQL	Oracle Network Manager 表にアクセスするユーザーにアクセスを許可する。

4. Oracle Network Manager から、ROFILES をデータベースに保存します。
- 「File」メニューから「Save As」を選択します。
 - 「Save Network Definition」ダイアログ・ボックスで「Database」を選択し、「OK」をクリックします。
 - 手順 1 で作成したデータベースのユーザー名とパスワードを入力し、「Connect」ダイアログ・ボックスにデータベースのネット・サービス名を入力します。「OK」をクリックします。
 - 「Save Network Definition」ダイアログ・ボックスで、保存するネットワークの名前を選択または入力します。
 - 「File」メニューから「Exit」を選択し、Oracle Network Manager を終了します。
5. サーバー上で、`$ORACLE_HOME/network/admin` (UNIX の場合) または `ORACLE_HOME%network%admin` (Windows NT の場合) にある NAMESUPG.SQL スクリプトを実行し、データベース・ファイルをアップグレードします。

```
sql> CONNECT user/password
sql> @oracle_home/network/admin/namesupg.sql;
```

ROSFILES から TNSNAMES.ORA ファイル ROSFILES を TNSNAMES.ORA ファイルに移行した後、TNSNAMES.ORA ファイルを Oracle Names にインポートするには、次の手順に従います。

1. TNSNAMES.ORA ファイルを作成します。
 - a. Oracle Network Manager から、「Special」メニューの「Preferences」を選択します。
 - b. 「Preferences」ダイアログ・ボックスで Oracle Names が選択されていないことを確認します。
 - c. 「File」メニューから「Generate」を選択し、ネットワーク定義を更新して TNSNAMES.ORA ファイルを作成します。
 - d. 「File」メニューから「Exit」を選択し、Oracle Network Manager を終了します。
2. Oracle Net8 Assistant または NAMESCTL 制御ユーティリティを使用して、TNSNAMES.ORA を Oracle Names Server にロードします。

Oracle Net8 Assistant による操作	制御ユーティリティの NAMESCTL による操作
1. Oracle Net8 Assistant を起動します。 -UNIX の場合、\$ORACLE_HOME/bin で <code>netasst</code> を実行します。 -Windows NT の場合、「スタート」ボタン -> 「プログラム」-> 「Oracle - HOME_NAME」-> 「Network Administration」-> 「Net8 Assistant」を選択します。	コマンド行から、次のように入力します。 <pre>namesctl namesctl> load_tnsnames file_name</pre>
2. Oracle Names Server フォルダをダブルクリックします。	
3. Oracle Names Server を選択します。	
4. ドロップダウン・リスト・ボックスから「Manage Data」を選択します。	
5. 「Load」を選択します。	
6. Oracle Network Manager によって生成された TNSNAMES.ORA ファイルのパスとファイル名を「File」フィールドに入力します。	
7. 「Execute」を選択します。	
8. 「File」メニューから「Save Network Configuration」を選択します。	
9. 「File」メニューから「Exit」を選択し、Oracle Net8 Assistant アプリケーションを終了します。	

次のマニュアルも参照してください。

- 『Oracle Network Manager 管理者ガイド、リリース 3.1』
- 『Oracle Names 管理者ガイド、バージョン 2』

Oracle Names バージョン 8 に確実に正しく移行するための チェックリスト

次のチェックリストは、Oracle Names バージョン 8 に確実に正しく移行するためのものです。

- 各リージョンにあるすべての Oracle Names Server を同じバージョン 8 に移行します。
- 以前に Oracle Names バージョン 2 を実行していて、自分のデータベースを Oracle Names 情報用リポジトリとしてアップデートする場合は、ネットワーク定義が格納されているノードで、`$ORACLE_HOME/network/admin` (UNIX の場合) または `ORACLE_HOME¥network¥admin¥names` (Windows NT の場合) にある NAMESUPG.SQL スクリプトを実行する必要があります。
- 動的検出オプションを使用してすでに Oracle Names バージョン 2 が実行されている状態で、Oracle Names 情報用リポジトリとしてデータベースを構成する場合は、次の手順を実行する必要があります。
 1. データベースのインストール先にするノード上で、`$ORACLE_HOME/network/admin` (UNIX の場合) または `ORACLE_HOME¥network¥admin¥names` (Windows NT の場合) にある NAMESINI.SQL スクリプトを実行します。

```
sql> CONNECT user/password
sql> @oracle_home/network/admin/names/namesini.sql;
```
 2. Oracle Net8 Assistant を使用して、すべての Oracle Names Server 構成ファイルに NAMES.ADMIN_REGION パラメータを構成します。NAMES.ADMIN_REGION パラメータについての詳細は、付録 C の「構成パラメータ」を参照してください。
- 障害耐性を実現するために各リージョン内に複数の Oracle Names Server を設定します。

- 2-42 ページの「[Oracle Names のバージョンによる違い](#)」と 2-45 ページの「[検出の理解](#)」に説明されているように、クライアントはデフォルトの Oracle Names Server リスト、予約済み Oracle Names Server、または検出によって取得された生成済みリストを使用して Oracle Names Server にアクセスします。

バージョン 8 によって生成された Oracle Names Server リストである SDNS.ORA (UNIX) または SDNS.ORA (Windows プラットフォーム) を使用する場合は、6-36 ページの「[手順 4: Oracle Names Server を使用するためにクライアントとデータベース・サーバーを構成する](#)」の手順に従ってください。

このファイルの作成後、SQLNET.ORA ファイルの NAMES.PREFERRED_SERVERS パラメータを削除します。NAMES.PREFERRED_SERVERS パラメータで指定した Oracle Names Server は、すべて検出プロセスの結果を上書きします。

ネーミング・メソッドとリスナーの構成

この章では、ネットワークの構成方法について説明します。

内容は次のとおりです。

- [構成作業の概要](#)
- [サービス名とインスタンス名の設定](#)
- [構成モデル](#)
- [ネーミング・メソッドとリスナーの構成](#)
- [ローカル・ネーミング・メソッドによるネットワークの構成](#)
- [Oracle Names ネーミング・メソッドによるネットワークの構成](#)
- [外部ネーミング・メソッドによるネットワークの構成](#)
- [リスナーの構成](#)
- [プロトコル・アドレスの構成](#)

構成作業の概要

データベースなどのサービスがクライアントからの接続を受け入れるには、ネット・サービス名を使用するようにクライアントを構成する必要があります。ネット・サービス名は、ネットワーク設定に基づいてサービス名を記述したものです。それぞれのネットワーク設定では、1つ以上のネットワーク・ルートがサービスに対して指定され、リスナーなどの Net8 コンポーネントが任意の数のみ含まれます。ネット・サービス名では、ネットワーク接続の負荷を複数のリスナー間で均衡化するかどうか、また、ネットワーク接続の確立に失敗したときに代替用のリスナーにフェイルオーバーするかどうかについても指定します。

具体的にネット・サービス名が識別するものは、次のとおりです。

- プロトコル・アドレスによるリスナーの位置情報など、サービスへのネットワーク・ルート。
- リリース 8.1 サービスを識別するための 1 つ以上のサービス名。または、リリース 8.1 より前のデータベースを識別するための Oracle システム識別子 (SID)。
- オプションで、接続先となるリリース 8.1 サービスのインスタンス名。

リスナー・アドレスは、クライアントがリスナーに接続するときに使用します。接続に成功すると、アドレスとサービス情報がクライアントからリスナーに渡されます。

インスタンスは、起動時に次の登録情報をリスナーに登録します。

- サービス登録。サービス名やインスタンス名など、インスタンスの情報をリスナーに提供します。
- プロトコル・アドレス情報。この情報に従ってリスナーはインスタンスに接続します。

リスナーは、クライアントからの接続要求を受け取ると、インスタンスから受け取った情報と、LISTENER.ORA ファイルに格納されている情報に基づいてそのクライアントの情報を検証します。これらの情報が一致すると、接続が許可されます。

クライアントは、リスナーの接続先サービスとアドレスにマップされるネット・サービス名のリストを作成して構成します。これらのサービスとアドレスは、TNSNAMES.ORA クライアント構成ファイル、Oracle Names Server または別のネーミング・メソッドに格納されています。エンド・ユーザーがサービスに接続する場合、ネット・サービス名によってそのサービスを識別します。たとえば、次のように指定します。

```
sql> CONNECT username/password@net_service_name
```

サービス名とインスタンス名の設定

サービス名は、データベースなどのサービスを論理的に表現したものです。サービスは、1つ以上のインスタンスから構成されます。ネット・サービス名を作成する場合、サービス名を使用してリリース 8.1 サービスを識別する必要があります。オプションで、インスタンス名を使用してリリース 8.1 インスタンスを識別することもできます。

次は、TNSNAMES.ORA ファイルの例です。ネット・サービスによってどのようにサービスのネットワーク設定を記述するかを示しています。

```
net_service_name=  
(description=  
  (address= (protocol_address_information))  
  (connect_data=  
    (service_name=service_name)  
    (instance_name=instance_name)))
```

この例では、SERVICE_NAME パラメータによってサービス名を識別する方法と、INSTANCE_NAME パラメータによってインスタンス名を識別する方法に注目してください。

サービス名は、一般にグローバル・データベース名を表す文字列です。グローバル・データベース名は、データベース名とドメイン名から構成され、インストール中またはデータベースの作成中に入力します。グローバル・データベース名が不明な場合は、データベース初期化ファイルの INITSID.ORA にある SERVICE_NAMES パラメータの結合値から取得できます。たとえば、SALES.COM というサービス名では、SALES がデータベース名で COM がドメインです。

インスタンス名は、インスタンスの名前を含みます。一般に、インスタンス名はインストール中に入力する SID のことです。SID が不明な場合は、INSTANCE_NAME パラメータから取得できます。

次の表は、SERVICE_NAMES パラメータと INSTANCE_NAME パラメータを説明しています。

パラメータ	説明
SERVICE_NAMES	ネットワーク上のサービスの名前を表す 1 つ以上のカンマで区切られた文字列。使用されている異なるインスタンスを個別に識別するために、(個々の SERVICE_NAMES エントリによって) 複数のサービス名を記述することができます。レプリケーションの使用によって複数インスタンスから利用できるサービスを識別するために、サービス名を使用することもできます。
INSTANCE_NAME	インスタンスの名前を表す文字列値。複数インスタンスが同じサービス名を共有する場合、特定のインスタンスを識別するために使用されます。INSTANCE_NAME を SID と混同しないでください。SID は、実際に 1 台のホスト上でインスタンスが共有しているメモリーを一意に識別します。

構成モデル

Net8 の構成は、2 つのモデルのいずれかを基盤としています。

ネットワークの構成モデル	適したネットワーク	ネーミング・メソッド
ローカル管理	ネットワーク・アドレスは、各ノード上の TNSNAMES.ORA ファイル内でマップされる (Oracle Names Server は存在しない)。	ホスト・ネーミング ローカル・ネーミング 外部ネーミング
集中管理	Oracle Names Server によってネットワーク・アドレスが解決されるネットワーク。Oracle Names Server は、クライアント構成プロファイルを 1 箇所に格納します。	Oracle Names

ネーミング・メソッド

Net8 では、次の方法を使用してネット・サービス名を解決できます。

ネーミング・メソッド	説明
ホスト・ネーミング	<p>接続時にクライアントは、サービス名のかわりにデータベースのグローバル・データベース名にマップされている別名を使用する。既存の名前解決サービスを介してデータベースのグローバル・データベース名を識別するためにホスト名が使用されます。この名前解決サービスには、ドメイン名サービス (DNS)、ネットワーク・インフォメーション・サービス (NIS) または単に集中管理された <code>/etc/hosts</code> ファイルのセットなどがあります。ホスト・ネーミングでは、サーバーのホスト名またはホスト名の別名を使用するのみで、ユーザーは Oracle サーバーに接続できます。この機能を利用するために、とくにクライアントを構成する必要はありません。この方法は、簡単な TCP/IP 環境にお勧めします。</p> <p>追加情報: 6-7 ページの「ホスト・ネーミング・メソッドによるネットワークの構成」を参照してください。</p>
ローカル・ネーミング	<p>個々のクライアントで構成され、格納されている情報を使用して、ネット・サービス名をネットワーク・アドレスに解決する。あまり変更されることのない少数のサービスからなる簡単な分散ネットワークでは、ローカル・ネーミングが最適な方法です。</p> <p>追加情報: 6-12 ページの「ローカル・ネーミング・メソッドによるネットワークの構成」を参照してください。</p>
Oracle Names	<p>ネットワーク上に存在するすべてのサービスの名前とアドレスが Oracle Names Server に格納される。接続要求は Oracle Names Server に送られ、そこでネット・サービス名からネットワーク・アドレスに変換されます。この情報は、クライアントに戻されます。</p> <p>追加情報: 6-21 ページの「Oracle Names ネーミング・メソッドによるネットワークの構成」を参照してください。</p>
外部ネーミング	<p>サポートされているサードパーティ製ネーム・サービスを利用してネット・サービス名を解決する。</p> <p>追加情報: 6-51 ページの「外部ネーミング・メソッドによるネットワークの構成」を参照してください。</p>

ネーミング・メソッドの正しい選択に関する詳細は、3-3 ページの「[ネット・サービス名の変換](#)」を参照してください。

構成ファイル

使用するネーミング・メソッドに合わせて、`$ORACLE_HOME/network/admin` (UNIX の場合) または `ORACLE_HOME¥network¥admin` (Windows プラットフォームの場合) で、次の構成ファイルを作成または変更することができます。

構成ファイル	説明
LISTENER.ORA	サーバー上に置かれ、サーバー上のすべてのリスナーのアドレスと、リスナーが使用するさまざまな制御パラメータを含む。
NAMES.ORA	Oracle Names Server 上に置かれ、各 Oracle Names Server の位置情報、ドメイン情報およびオプションの構成パラメータを含む。
SQLNET.ORA	名前解決方法が書かれている。このファイルは、クライアント上に置かれます。
TNSNAMES.ORA	クライアント上に置かれ、ネット・サービス名および対応するリスナー・アドレスのリストと、サービスに関する記述を含む。

次の章も参照してください。

- [付録 B の「構成ファイルのサンプル」](#) (前述のファイル例)
- [付録 C の「構成パラメータ」](#) (パラメータ情報)

これらのファイルの構成方法は、この章の各ネーミング・メソッドで説明します。

ホスト・ネーミング・メソッドによるネットワークの構成

注意: この項では、リリース 8.1 を対象に構成方法について説明していません。既存のリリース 8.0 またはリリース 7.x の構成がある場合は、[第 4 章の「リリース 8.1 のインストールと Net8 関連製品について」](#)を参照してください。

簡単な接続方法が求められる環境では、ホスト・ネーミングを使用することにより TNSNAMES.ORA 構成ファイルでサービス名を参照する必要がなくなります。ただしホスト・ネーミングは、接続プーリング、外部プロシージャ、異機種間サービスなど、追加の接続情報が必要な、高度の機能が要求される大規模で複雑な環境には不向きです。このような場合は、別のネーミング・メソッドの使用をお勧めします。

ホスト・ネーミング・メソッドは、TCP/IP ネットワーク環境でのみ利用可能です。インスタンスがリスナーに登録されると、データベース名とドメイン名から構成される、サーバーのグローバル・データベース名が登録されます。グローバル・データベース名は、既存の名前解決サービスのホスト名と別名にマップされます。クライアントは、接続文字列の中で別名を使用し、データベースに接続します。

次の場合にクライアントは、別名を使用してサーバーに接続できます。

- インストール済みの Net8 Server と Net8 Client ソフトウェアを使用して、Oracle8i データベース・サービスに接続する場合
- クライアントとサーバーは、TCP/IP プロトコルを介して接続する場合
- すべての名前は、ドメイン名サービス (DNS) や集中的に管理される TCP/IP hosts ファイルのような IP アドレス変換メカニズムを通して解決される場合
- Oracle Connection Manager やセキュリティのような高度な機能が不要な場合

ホスト・ネーミング・メソッドを構成するには、次の作業を実行します。

手順 1: グローバル・データベース名を使用してリスナーを構成する

手順 2: 第 1 番目のネーミング・メソッドとして HOSTNAME を構成する

手順 3: ホスト名解決の環境を設定する

手順 1: グローバル・データベース名を使用してリスナーを構成する

グローバル・データベース名は、リスナーに登録する必要があります。登録がどのように行われるかは、データベースのリリースによって異なります。

Oracle8i リリース 8.1

データベースがリスナーを検出できる場合は、グローバル・データベース名を含むデータベースに関する情報は自動的にリスナーに登録されます。リスナーは、次の場合に検出されます。

- LISTENER という名前のデフォルトのリスナーが、ポート 1521 の TCP/IP 上で実行されている場合
- LOCAL_LISTENER パラメータが、INITSID.ORA ファイルに設定されている場合

データベースがリスナーを検出できない場合、次の「[Oracle8 リリース 8.0 またはそれより前のリリース](#)」で説明するように GLOBAL_DBNAME パラメータを使用して LISTENER.ORA ファイルを構成する必要があります。

Oracle8 リリース 8.0 またはそれより前のリリース

Oracle8 リリース 8.0 データベースまたは Oracle7 データベースの場合、データベース情報は LISTENER.ORA ファイルを介してリスナーに登録されます。これらのデータベースでは、次に示すように SID_LIST_listener_name セクションを静的に構成し、そこに GLOBAL_DBNAME パラメータを含めます。

```
sid_list_listener=(sid_list=  
(sid_desc=  
(global_dbname=sales.com)  
(sid_name=db1)  
(oracle_home=/u01/app/oracle/8.0.5)  
)
```

詳細は、6-71 ページの「[サービス情報によるリスナーの構成](#)」を参照してください。

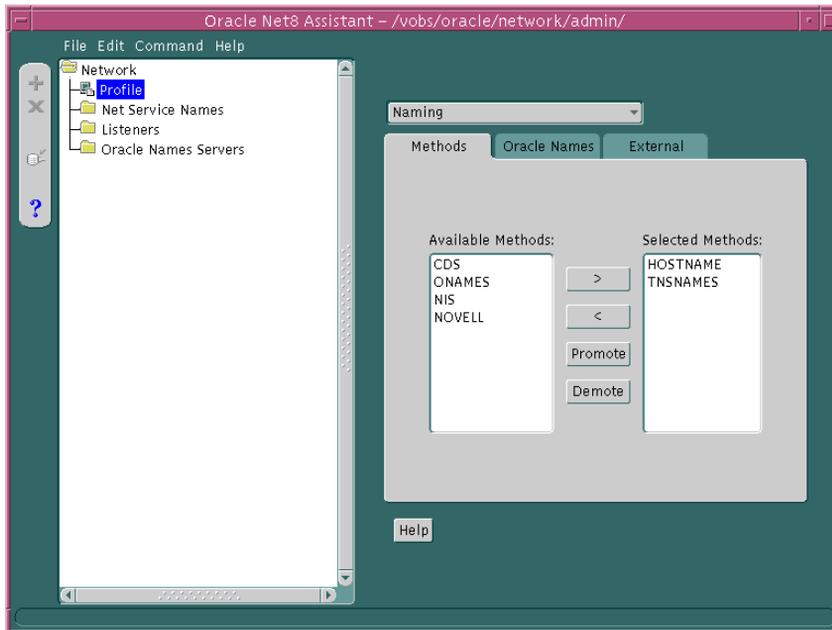
手順 2: 第 1 番目のネーミング・メソッドとして HOSTNAME を構成する

SQLNET.ORA ファイルの NAMES.DIRECTORY_PATH パラメータに指定する第 1 番目のネーミング・メソッドとしてホスト・ネーミングを構成します。このパラメータによって、Net8 が名前解決に使用するネーミング・メソッドの順序を指定します。

ローカル・ネーミングを第 1 番目のネーミング・メソッドに指定するには、次の手順に従います。

1. Oracle Net8 Assistant を起動します。
 - UNIX の場合、\$ORACLE_HOME/bin で netasst を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」 -> 「Oracle - HOME_NAME」 -> 「Network Administration」 -> 「Net8 Assistant」を選択します。
2. 「Profile」アイコンをクリックします。
3. ドロップダウン・リスト・ボックスから「Naming」を選択します。
4. 「Methods」タブをクリックします。
5. 「Available Methods」リストから「TNSNAMES」を選択し、「>」をクリックします。

- 「Selected Methods」リストで「TNSNAMES」を選択し、「Promote」ボタンを使用して選択した項目をリストの最上位に移動します。



- 「File」メニューから「Save Network Configuration」を選択します。
- 「File」メニューから「Exit」を選択し、Oracle Net8 Assistant アプリケーションを終了します。

これにより SQLNET.ORA ファイルに、NAMES.DIRECTORY_PATH パラメータの先頭に HOSTNAME を含むエントリが入ります。

```
names.directory_path=(hostname, tnsnames)
```

手順 3: ホスト名解決の環境を設定する

グローバル・データベース名は、DNS、NIS または集中管理される TCP/IP ホスト・ファイルである `/etc/hosts` のような IP アドレス変換メカニズムを通して解決する必要があります。グローバル・データベース名のために作成する別名は、グローバル・データベース名と同じ名前にする必要があります。

グローバル・データベース名のドメイン部分は、ネットワーク・ドメインと一致しなければならないことに注意してください。

たとえば、あるデータベースの SALES.COM というグローバル・データベース名が SALES-PC という名前のマシン上に存在する場合、`/etc/hosts` ファイルのエントリは次のようになります。

```
#IP address of server      host name  alias
144.25.219.146            sales-pc   sales.com
```

8-2 ページの「[データベースへの接続](#)」に説明されているように、クライアントは接続文字列中で SALES.COM を使用します。クライアントとサーバーが同じ COM ドメインに所属していれば、クライアントは接続文字列に SALES を入力するだけでかまいません。

ローカル・ネーミング・メソッドによるネットワークの構成

注意： この項では、リリース 8.1 を対象に構成方法について説明していません。既存のリリース 8.0 またはリリース 7.x の構成がある場合は、[第 4 章の「リリース 8.1 のインストールと Net8 関連製品について」](#)を参照してください。

ローカル・ネーミング・メソッドを使用する場合、ネット・サービス名は TNSNAMES.ORA ファイルに追加します。ネット・サービス名は、接続記述子に含まれるネットワーク・アドレスにマップされます。接続記述子には、プロトコル・アドレスで表されたリスナーの位置情報と、接続先サービスの名前があります。クライアントがアプリケーションから接続するときは、このネット・サービス名を使用します。

接続記述子にマップされたネット・サービス名を含む TNSNAMES.ORA ファイルの基本的な構文例を、[図 6-1](#) に示します。DESCRIPTION は接続記述子を含み、ADDRESS はリスナー・アドレスを表し、CONNECT_DATA はサービス名を含んでいます。

図 6-1 TNSNAMES.ORA でのネット・サービス名の使用例

```
net_service_name=
(description=
  (address= (protocol_address_information))
  (connect_data=
    (service_name=service_name)
  )
)
```

図 6-2 に示すように、各接続記述子には 1 つ以上のリスナー・アドレスを含めることができます。

図 6-2 複数のリスナー・アドレス

```
net_service_name=  
(description=  
  (address= (protocol_address_information))  
  (address= (protocol_address_information))  
  (address= (protocol_address_information))  
  (connect_data=  
    (service_name=service_name)  
  )  
)  
)
```

注意: 前のリリースでは、アドレス・リストが 1 つしかない場合、ADDRESS_LIST を使用する必要がありました。

connect-time failover、クライアント負荷のバランスおよび Oracle Connection Manager 機能のようなオプションのパラメータは、1 つの接続記述子内部のアドレス・リストに設定することも、複数の接続記述子に含まれるすべてのアドレスに設定することもできます。

これらの機能の詳細は、第 7 章の「Net8 のオプション機能」と C-31 ページの「ローカル・ネーミング・パラメータ (TNSNAMES.ORA)」を参照してください。

インストール中の構成作業

クライアントまたはサーバーで「Custom」インストールを終了すると、Net8 Configuration Assistant によってネーミング・メソッドの構成を求めるプロンプトが表示されます。ローカル・ネーミング・メソッドを選択すると、Net8 Configuration Assistant によって TNSNAMES.ORA ファイルにネット・サービス名を構成するように求めるプロンプトが表示されます。Net8 Configuration Assistant を使用してネット・サービス名を作成する場合は、TNSNAMES.ORA ファイルと LISTENER.ORA ファイルのリスナー・プロトコル・アドレスが一致することを確認してください。この手順は、オンライン・ヘルプを参照してください。

インストール後の構成作業

ネット・サービス名は、任意の時点で TNSNAMES.ORA ファイルに追加できます。ローカル・ネーミング・メソッドを構成するには、次の作業を実行します。

手順 1: ネット・サービス名を構成する

手順 2: 第 1 番目のネーミング・メソッドとして TNSNAMES を構成する

手順 3: 構成情報を配布する

手順 4: リスナーを構成する

手順 1: ネット・サービス名を構成する

ローカル・ネーミング・メソッドによって構成するには、Oracle Net8 Assistant またはサーバー上の Net8 Configuration Assistant を使用します。

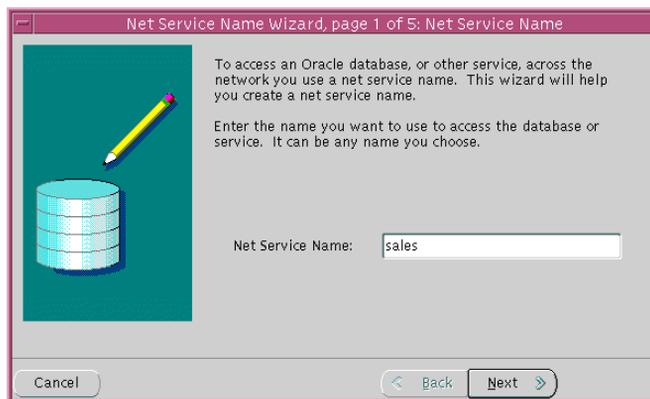
注意: ネット・サービス名は、Net8 Easy Config を実行して作成することもできます。ただし、このツールは将来提供しない可能性があります。このため、オラクル社では Net8 Configuration Assistant が Oracle Net8 Assistant の使用をお勧めしています。Net8 Easy Config を実行するには、次の手順に従います。

- UNIX の場合、`$ORACLE_HOME/bin` で `netec` を実行します。
 - Windows プラットフォームの場合、「スタート」ボタン -> 「プログラム」 -> 「Oracle - HOME_NAME」 -> 「Network Administration」 -> 「Net8 Easy Config」を選択します。
-
-

Oracle Net8 Assistant Oracle Net8 Assistant を使用するには、次の手順に従います。

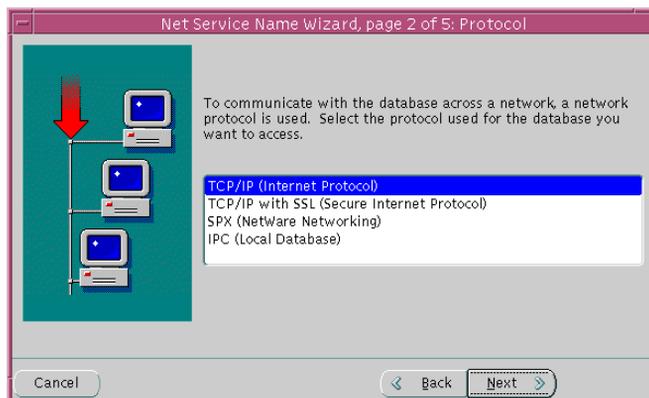
1. Oracle Net8 Assistant を起動します。
 - UNIX の場合、`$ORACLE_HOME/bin` で `netasst` を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」 -> 「Oracle - HOME_NAME」 -> 「Network Administration」 -> 「Net8 Assistant」を選択します。
2. ディレクトリ・ツリーの「Net Service Names」フォルダをクリックします。ネット・サービス名が存在する場合は、ツリー構造に表示されます。
3. 「Edit」メニューから「Create」を選択するか、または「+」ボタンをクリックして、新規のネット・サービス名を作成します。「Net Service Name Wizard」が起動します。

4. 「Net Service Name」フィールドに任意の名前を入力し、「Next」をクリックします。

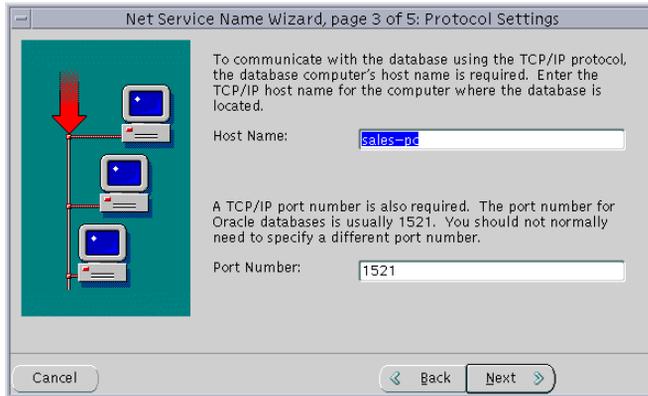


明示的に指定する場合は、ネット・サービス名をクライアントのドメインで修飾できます。SQLNET.ORA に NAMES.DEFAULT_DOMAIN パラメータを指定すると、ネット・サービス名は自動的にドメインによって修飾されます。詳細は、7-16 ページの「[ドメインの構成](#)」を参照してください。

5. リスナーに構成するリスニング用のプロトコルを選択し、「Next」をクリックします（このプロトコルはクライアントにもインストールする必要があります）。



6. フィールドで選択したプロトコルのために、リスナーの適切なプロトコル・アドレス情報を入力し、「Next」をクリックします。



プロトコル・アドレスに関する詳細は、6-76 ページの「[プロトコル・アドレスの構成](#)」を参照してください。

TCP/IP 上で標準的な Net8 クライアントをリスニングするリスナーを構成する場合、デフォルトのポート 1521 を選択する必要があります。このポートを選択しない場合は、`INITSID.ORA` ファイルに `LOCAL_LISTENER` パラメータを設定し、なんらかのネーミング・メソッドによってリスナー名を解決する必要があります。`LISTENER.ORA` にデフォルト以外のポート番号を構成する方法の詳細は、6-67 ページの「[デフォルト以外のポートを使用するリスナーの構成](#)」を参照してください。

- 宛先サービスがリリース 8.1 の場合は、「Oracle8i release 8.1」をクリックして「Service Name」フィールドにサービス名を入力し、「Next」をクリックします。

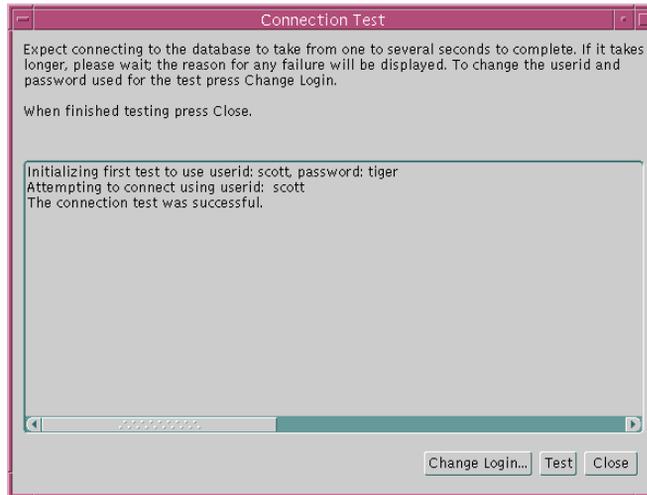


サービス名は、一般にグローバル・データベース名です。使用するサービス名文字列についての詳細は、6-2 ページの「サービス名とインスタンス名の設定」を参照してください。

宛先サービスがリリース 8.1 より前のリリースの場合は、「Oracle8 Version 8.0」または「Previous」をクリックして「Database SID」フィールドに「SID」と入力し、「Next」をクリックします。

- ネット・サービス名の動作を検証するには「Test」をクリックし、「Next」をクリックします。テストでは、データベースとリスナーが実行されている必要があります。実行されていない場合は、8-2 ページの「Net8 コンポーネント起動の概要」の手順に従ってコンポーネントを起動するか、または「Next」をクリックしてテストを省略し、手順 10 に進んでください。

テストに成功すると、「Connect Test」ダイアログ・ボックスに「The connect test was successful.」というメッセージが表示されます。



テストに成功しなかった場合は、次に従って再試行してください。

- データベースとリスナーが実行されていることを確認し、「Test」をクリックします。
 - 「Change Login」をクリックして接続に使用するユーザー名とパスワードを変更し、「Test」をクリックします。
9. 「Close」をクリックし、「Connect Test」ダイアログ・ボックスを閉じます。
 10. 「Finish」をクリックして構成を保存し、「Net Service Names Wizard」を終了します。
 11. 「File」メニューから「Save Network Configuration」を選択します。
 12. 「File」メニューから「Exit」を選択し、Oracle Net8 Assistant を終了します。

次の項目も参照してください。

- オプションのアドレスと複数アドレスの特性を設定する方法は、7-7 ページの「[ネット・サービス名の接続属性の詳細構成](#)」と 7-11 ページの「[複数アドレス・オプションの構成](#)」を参照してください。
- ドメインを指定してネット・サービス名を要求する方法は、7-16 ページの「[ドメインの構成](#)」を参照してください。

Net8 Configuration Assistant

Net8 Configuration Assistant を使用する場合は、次の手順に従います。

1. Net8 Configuration Assistant を起動します。

- UNIX の場合、`$ORACLE_HOME/bin` で `netca` を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」 -> 「Oracle - *HOME_NAME*」 -> 「Network Administration」 -> 「Net8 Configuration Assistant」を選択します。
2. 「Local Net8 Service Name configuration」を選択し、「Next」をクリックします。
 3. 「Create」をクリックします。
 4. ウィザードのプロンプトとオンライン・ヘルプに従って、ネット・サービス名を作成します。

注意：「Oracle8i release 8.1 database or service」オプションを選択した場合は通常、グローバル・サービス名をサービス名として入力します。使用するサービス名文字列についての詳細は、6-2 ページの「サービス名とインスタンス名の設定」を参照してください。

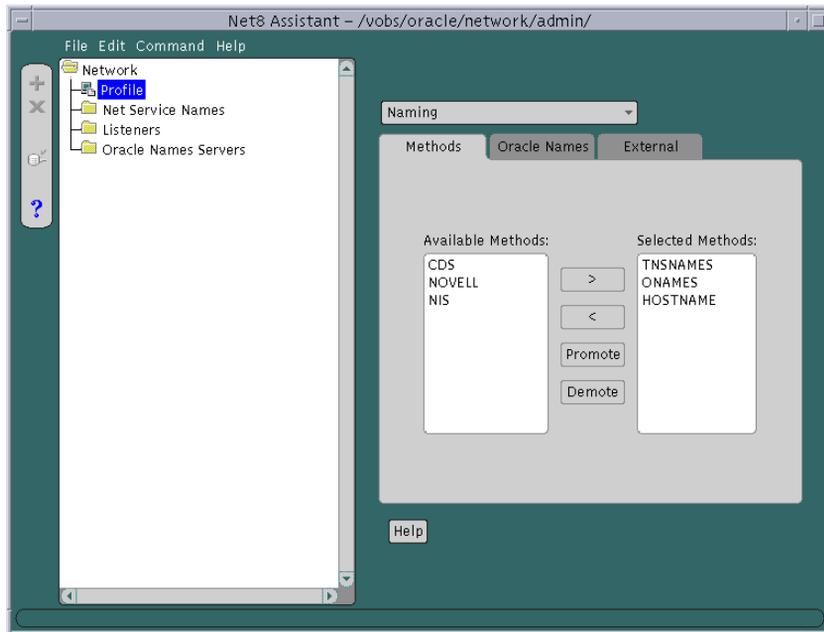
手順 2: 第 1 番目のネーミング・メソッドとして TNSNAMES を構成する

SQLNET.ORA ファイルの NAMES.DIRECTORY_PATH パラメータに指定する第 1 番目のネーミング・メソッドとしてローカル・ネーミングを構成します。このパラメータによって、Net8 が名前解決に使用するネーミング・メソッドの順序を指定します。

ローカル・ネーミングを第 1 番目のネーミング・メソッドに指定するには、次の手順に従います。

1. Oracle Net8 Assistant を起動します。
 - UNIX の場合、`$ORACLE_HOME/bin` で `netasst` を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」 -> 「Oracle - *HOME_NAME*」 -> 「Network Administration」 -> 「Net8 Assistant」を選択します。
2. 「Profile」アイコンをクリックします。
3. ドロップダウン・リスト・ボックスから「Naming」を選択します。
4. 「Methods」タブをクリックします。
5. 「Available Methods」リストから「TNSNAMES」を選択し、「>」をクリックします。

- 「Selected Methods」リストで「TNSNAMES」を選択し、「Promote」ボタンを使用して選択した項目をリストの最上位に移動します。



- 「File」メニューから「Save Network Configuration」を選択します。
- 「File」メニューから「Exit」を選択し、Oracle Net8 Assistant アプリケーションを終了します。

これにより SQLNET.ORA ファイルに、NAMES.DIRECTORY_PATH パラメータの先頭に TNSNAMES を含むエントリが入ります。

```
names.directory_path=(tnsnames, onames, hostname)
```

手順 3: 構成情報を配布する

クライアントを 1 台構成したら、単に TNSNAMES.ORA と SQLNET.ORA の各構成ファイルを他のクライアントの同じ場所にコピーする方法が最も確実です。これによって、ファイルの一貫性が保たれます。このようにしないと、Oracle Net8 Assistant か Net8 Configuration Assistant をすべてのクライアントで使用しなければならず、エラーの原因になります。

手順 4: リスナーを構成する

サーバー上のリスナーは、ネット・サービス名用に構成したものと同一プロトコルとプロトコル・アドレスを使用してリスニングするように構成する必要があります。デフォルトでリスナーは、オペレーティング・システムの第 1 番目のプロトコル（通常は TCP/IP）を使用するようにすでに構成されています。リスナーの構成方法の詳細は、6-76 ページの「[プロトコル・アドレスの構成](#)」を参照してください。

Oracle Names ネーミング・メソッドによるネットワークの構成

注意： この項では、リリース 8.1 を対象に構成方法について説明しています。既存のリリース 8.0 またはリリース 7.x の構成がある場合は、[第 4 章の「リリース 8.1 のインストールと Net8 関連製品について」](#)を参照してください。

Oracle Names を使用すると、クライアント / サーバーによるグローバルなコンピュータ・ネットワークの設定と管理が容易になります。Oracle Names によって、ネットワーク・アドレスとデータベース・リンクの情報がネットワーク全体のすべてのノードから利用できるようになります。各データベース・サーバーのネットワーク・アドレスは、単純なサービス名によって識別されます。これによりクライアント・アプリケーションは、長いアドレスを使用せずに、この単純な名前を使用してデータベース接続を要求できます。Oracle Names では、ネットワーク・インフラストラクチャが変更されても、ユーザーとアプリケーションはその影響を受けずに済みます。Oracle Names は、ネットワーク・サービス名の集中的な管理を提供します。

Oracle Names Server には、次の情報が格納されます。

データ	説明
データベースのグローバル・データベース名とアドレス	Oracle Names Server は、グローバル・データベース名（データベース名とドメイン）とアドレスなどのデータベースに関する情報をリスナーから取得する。このアドレスは、LISTENER.ORA ファイルに構成されています。また、グローバル・データベース名はデータベースの起動時に登録されるか、または静的に LISTENER.ORA ファイルに構成されます。管理者は、この情報を登録しません。

データ	説明
その他の Oracle Names Server 名とアドレス	Oracle Names Server には、同じ管理リージョン内にあるすべての Oracle Names Server の名前とアドレスが格納される。ネットワーク内に複数の管理リージョンが存在する場合、Oracle Names Server には、ルート管理リージョンとその直属の各サブリージョンに存在する 1 つ以上の Oracle Names Server の名前とアドレスが格納されます。管理者がこの情報を登録する必要はありません。
ネット・サービス名	NAMESCTL 制御ユーティリティまたは Oracle Net8 Assistant に ネット・サービス名を登録すると、これらが Oracle Names Server に格納される。このほか、Oracle Names Server には非 Oracle データベースと Oracle RDB データベースへのゲートウェイも格納されます。
グローバル・データベース・リンク	<p>データベース・リンクによって、データベース同士の通信が可能になる。データベース・リンクの名前は、リンクのポイント先となっているデータベースのグローバル・データベース名と同じです。通常、1 個のデータベースに存在するリンクは 1 つのみです。</p> <p>次に示すタイプのデータベース・リンクを作成できます。</p> <ul style="list-style-type: none"> ■ データベースの特定のスキーマ内でのプライベート・データベース・リンク。プライベート・データベース・リンクの所有者のみが、このデータベースを使用できます。 ■ データベース用のパブリック・データベース・リンク。すべてのユーザーが、このデータベースを使用できます。 ■ Oracle Names Server 内のグローバル・データベース・リンク。ネットワーク内のだれもがこのデータベースを使用できるようにします。 <p>Oracle Names はリスナーからグローバル・データベース名を取得するため、グローバル・データベース名であるグローバル・データベース・リンクは Oracle Names Server に自動的に登録されます。したがって、管理者がこの情報を登録する必要はありません。</p> <p>Net8 Assistant を使用して、グローバル・データベース・リンクのためにユーザー名とパスワードからなる資格証明を Oracle Names Server に登録できます。Oracle Net8 Assistant を使用して、グローバル・データベース・リンクにリンク修飾子を追加できます。</p> <p>グローバル・データベース・リンクは、個々のユーザーが作成したプライベート・データベース・リンクやパブリック・データベース・リンクによって置き換えることができます。プライベート・データベース・リンクおよびパブリック・データベース・リンクについての詳細は、『Oracle8i 分散システム』を参照してください。</p>

データ	説明
別名	Oracle Names Server には、定義済みの任意のネット・サービス名、データベース・サービス、またはグローバル・データベース・リンクに対する別名（サービス名の代用となるもの）が格納される。別名は、NAMECTL 制御ユーティリティまたは Oracle Net8 Assistant を使用して Oracle Names Server に登録できます。
Oracle Connection Manager	Oracle Names Server には、ネットワーク上に存在するすべての Oracle Connection Manager の名前とリスニング・アドレスが格納される。管理者がこの情報を登録する必要はありません。

Oracle Names の構成に必要な作業は、次のとおりです。

手順 1: Oracle Names オプションを検討する

手順 2: インストールする

手順 3: Oracle Names Server を作成する

手順 4: Oracle Names Server を使用するためにクライアントとデータベース・サーバーを構成する

手順 5: リスナーを構成する

手順 6: Oracle Names Server にデータベースを登録する

手順 7: (オプション) 委任管理リージョンにドメインを委任する

手順 1: Oracle Names オプションを検討する

Oracle Names Server を作成する前に、次の項目について決定する必要があります。

- Oracle Names Server 間でサービス登録データを継続的にレプリケートするか、または同一リージョン内のすべての Oracle Names Server がそれぞれの登録データを Oracle データベースに格納する必要があるかどうか。
- 1つ以上の管理リージョンに対するサポートが必要であるかどうか。管理リージョンは、1つ以上のドメインに存在する Oracle Names Server の集合であり、データベースなどのネットワーク・オブジェクトをグループ化したものです。複数の管理リージョンを持つネットワークには、1つのルート管理リージョンと1つ以上の委任管理リージョンが必要です。委任管理リージョンには、ルートの直属の子リージョンの役割を果たす代替リージョン内のドメインと Oracle Names Server のアドレスが格納されます。

これらの決定事項に関してさらに詳しい情報が必要な場合は、2-31 ページの「Oracle Names アーキテクチャ」を参照してください。

手順 2: インストールする

次のコンポーネントがインストールされていることを確認します。

- Oracle Names Server として指定するマシン上の Oracle Names
- クライアント上の Net8 Client または SQL*Net Client
- サーバー上の Net8 Server または SQL*Net Server

重要： UNIX プラットフォームでは、Oracle Universal Installer によって \$ORACLE_HOME/network/names ディレクトリは作成されません。このディレクトリは、正しく構成作業を実行するために必要です。このディレクトリが存在しない場合は、作成してください。

手順 3: Oracle Names Server を作成する

Oracle Names Server を作成する方法は、リージョン・データの格納場所によって異なります。

- Oracle Names Server にアクセス可能な Oracle データベース内の、リージョン・データベースと呼ばれる表に格納する場合

Oracle Names Server に追加した情報は、データベースに格納されるほか、Oracle Names Server のキャッシュ・ファイルにも格納されます。データベースが利用できない場合は、キャッシュ・ファイル内の情報が使用されます。

- Oracle Names Server のキャッシュ・ファイルに格納し、ただちに他のすべての Oracle Names Server のキャッシュ・ファイルにレプリケートする場合

キャッシュ・ファイルは、UNIX プラットフォームでは `$ORACLE_HOME/network/names` に、Windows NT では `ORACLE_HOME¥network¥names` に格納されます。

該当するファイル	説明
CKCFG.ORA	Oracle Names Server の構成に関する情報を格納する。NAMES.ORA ファイルと似ています。
CKPCCH.ORA	接続文字列の情報など、クライアントのキャッシュ情報を格納する。
CKPREG.ORA	Oracle Names Server に格納されているデータに関する情報を格納する。TNSNAMES.ORA ファイルと似ています。

ONAMES_*host* という名前を持つデフォルトの Oracle Names Server が、リージョン以外のデータベースの構成用にインストール中に作成されます。Oracle Names は、ポート 1575 を使用して TCP/IP 上でリスニングするように構成されます。これは基本的な構成であり、複数ドメインや複数リージョンは含まれません。この Oracle Names Server を使用する場合は、この手順を省略して 6-35 ページの「[手順 2: Oracle Names Server を検出する](#)」に進んでください。

次の項では、これら両方のモードについてそれぞれ説明します。

- [リージョン・データベースへのデータの格納](#)
- [ファイルへのデータの格納](#)

どちらの方法を選択するにしても、オラクル社では Oracle Names Server がダウンした場合に備えて複数の Oracle Names Server を作成するようお勧めしています。

リージョン・データベースへのデータの格納

Oracle データベースにサービス・データを格納するには、次の作業を実行します。

- 手順 1: データベース・サーバーを構成する
- 手順 2: Oracle Names Server を作成する
- 手順 3: Oracle Names Server を起動する

手順 1: データベース・サーバーを構成する データベース・サーバーを構成するには、次の手順に従います。

1. まだデータベースが実行されていない場合は、データベースを起動します。データベースがすでに起動されている場合は、手順 2 に進みます。

- a. INTERNAL としてデータベースに接続します。

```
sql> CONNECT internal/password
```

INTERNAL ユーザー・アカウントの *password* は、デフォルトで ORACLE です。

- b. データベースを起動します。

```
sql> STARTUP;
```

2. SYSTEM ユーザーとしてデータベースに接続します。

```
sql> CONNECT system/password
```

SYSTEM ユーザー・アカウントの *password* は、デフォルトで MANAGER です。

3. ユーザー・アカウントとパスワードを作成します。

```
sql> CREATE user user
      identified by password
      default tablespace users
      temporary tablespace temp;
```

4. NAMESINI.SQL スクリプトを実行します。このファイルは、UNIX では \$ORACLE_HOME/network/admin に、Windows NT プラットフォームでは ORACLE_HOME¥network¥admin¥names にあります。このスクリプトによって、Oracle Names が情報を格納するために必要な表が作成されます。オプションで NAMESUPG.SQL を実行し、古い表を定義します。

```
sql> CONNECT user/password
sql> @oracle_home/network/admin/namesini.sql;
```

手順 2: Oracle Names Server を作成する Oracle Names がインストールされている各コンピュータに Oracle Names Server が必要な場合、Oracle Net8 Assistant を使用して Oracle Names Server を作成します。

1. Oracle Net8 Assistant を起動します。
 - UNIX の場合、\$ORACLE_HOME/bin で netasst を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」 -> 「Oracle - HOME_NAME」 -> 「Network Administration」 -> 「Net8 Assistant」を選択します。
2. Oracle Net8 Assistant で、「Oracle Names Servers」フォルダをクリックします。
3. 「Edit」メニューから「Create」を選択するか、または「+」ボタンをクリックして、新規の Oracle Names Server を作成します。

Names ウィザードが起動します。

このウィザードでは、Oracle Names Server を作成し、指示に従って構成作業を進めていく過程で、次の項目の入力を求めるプロンプトが表示されます。

- Oracle Name サーバーの一意的な名前
- Oracle Names Server のプロトコル・アドレス
TCP/IP を選択する場合、オラクル社では正規に登録されたデフォルトのポート番号である 1575 を Oracle Names Server に使用することをお薦めしています。
- データベースに情報を格納するか、または Oracle Names Server 間で情報をレプリケートするかを選択
(「Use a region database.」をクリックします)
- データベースのリスナーのプロトコル・アドレス
TCP/IP を選択する場合、オラクル社では正規に登録されたデフォルトのポート番号である 1521 をリスナーに使用することをお薦めしています。
- データベース・ユーザー ID、パスワード、およびサービス名または SID
- この Oracle Names Server がルート管理リージョンにあるかどうかの識別

この Oracle Names Server がルート管理リージョンにあると指定した場合、構成作業は完了です。

この Oracle Names Server がルート管理リージョンにないと指定した場合は、委任管理リージョンにあるものと見なされます。その場合ウィザードは、ローカル管理リージョンのドメイン名と、ルート管理リージョンにある Oracle Names Server のアドレスを求めるプロンプトを表示してから完了します。

ウィザードが完了すると、次のメッセージが表示されます。

```
「 A Names Server, onames_server, has been created with default settings. Use the Configure
  Server section to modify the default configuration. 」
```

4. 「File」メニューから「Save Network Configuration」を選択します。
5. 「File」メニューから「Exit」を選択し、Oracle Net8 Assistant アプリケーションを終了します。
6. Oracle Names Server をさらに追加する場合は、手順 2 ~ 5 を繰り返してリージョン内に Oracle Names Server を作成します。Oracle Net8 Assistant では、1 台のコンピュータ上に複数の Oracle Names Server を作成することをサポートしていません。

Oracle Net8 Assistant では、少なくとも次の項目を設定して NAMES.ORA ファイルを作成します。

- Oracle Names Server の名前 (NAMES.SERVER_NAME)
- Oracle Names Server のリスニング用プロトコル・アドレス (NAMES.ADDRESSES)
- データベース情報 (NAMES.ADMIN_REGION)

次に、NAMES.ORA ファイルを内容に関する注釈付きで示します。

```
#Oracle Names server name. The name should include the name of the domain this
Oracle Names server is in.
names.server_name=namesvr2.com

#Oracle Names server listening protocol address
names.addresses=
  (address=(protocol=tcp)(host=namesrv2-pc)(port=1575))
  [(address=...)]

#Database repository information
names.admin_region=
  (region=
    (description=
      (address=(protocol=tcp)(host=sales-pc)(port=1521))
      (connect_data=
        (service_name=sales.com)
      )
    )
    (userid=system)
    (password=password)
    (name=local_region)
    (refresh=86400)
    (retry=60)
    (expire=600)
  )
```

```
#If an Oracle Names server is in a delegated administrative region, identify the
address of an Oracle Names server in the root administrative region.
names.domain_hints=
  (address=(protocol=tcp)(host=namesrv1-pc)(port=1575))
```

```
# Specify the domain controlled by this region and the time to live (TTL). If
this is the root administrative region, you must have NAME= (null) to identify
the root domain properly.
```

```
names.domains=
  (domain=
    (name=)
    (min_ttl=86400)
  )
```

アドレス情報に必要なプロトコル構文に関する詳細は、6-76 ページの「[プロトコル・アドレスの構成](#)」を参照してください。

複数ドメインの管理

リージョンに複数のドメインを管理させたい場合は、Oracle Net8 Assistant を使用して NAMES.DOMAINS パラメータに追加のドメインを指定します。

1. Oracle Names Server フォルダをダブルクリックします。
2. Oracle Names Server を選択します。
3. ドロップダウン・リスト・ボックスから「Domains」を選択します。
4. 「Domains」タブを選択します。
5. 「Domain Name」フィールドにドメイン名を入力します。
6. TTL 情報を入力します。
7. 「Add」をクリックします。
8. 「File」メニューから「Save Network Configuration」を選択します。
9. 別のドメインを追加する場合は、手順 1 ~ 8 を繰り返します。

次の例では、NAMES.DOMAINS にルート、COM、ORACLE.COM および HQ.ORACLE.COM ドメインのリストが含まれています。HQ.ORACLE.COM を定義するためには、HQ.ORACLE.COM より手前に位置するすべてのドメインを定義しておく必要があります。

```
names.domains=  
  (domain_list=  
    (domain=  
      (name=)  
      (min_ttl=86400)  
    )  
    (domain=  
      (name=com)  
      (min_ttl=86400)  
    )  
    (domain=  
      (name=oracle.com)  
      (min_ttl=86400)  
    )  
    (domain=  
      (name=hq.oracle.com)  
      (min_ttl=86400)  
    )  
  )  
)
```

手順 3: Oracle Names Server を起動する Oracle Net8 Assistant または NAMESCTL 制御ユーティリティを使用して、Oracle Names Server を起動します。

Oracle Net8 Assistant による操作	制御ユーティリティの NAMESCTL による操作
<ol style="list-style-type: none"> Oracle Names Server フォルダをダブルクリックします。 Oracle Names Server を選択します。 ドロップダウン・リスト・ボックスから「Manage Server」を選択します。 「Control」タブを選択します。 「Server Operations」フィールドの「Start」ラジオ・ボタンをクリックします。 「Apply」をクリックします。 次のメッセージが表示されます。 「Server started successfully.」 Windows NT では、OracleHOME_ NAMENAMESONAMES_SERVER というサービスが作成されます。 	<p>コマンド行から、次のように入力します。</p> <pre>namesctl</pre> <p>次のエラー・メッセージが表示されても、無視します。</p> <pre>NNL-00024: warning: no preferred names servers in SQLNET.ORA >NNL-00018: warning: could not contact default name server</pre> <pre>namesctl> start</pre> <p>NAMESCTL の START コマンドは Oracle Names Server をメモリにロードし、実行の開始を指示します。Oracle Names Server は、起動時に構成情報とデータをロードします。</p> <p>これによって、Windows NT では OracleORACLE_HOMENAMESonames_serverService というサービスが作成されます。</p>

ファイルへのデータの格納

ファイルにデータを格納し、更新中に Oracle Names Server 間でレプリケートするには、次の作業を実行します。

- 手順 1: Oracle Names Server を作成する
- 手順 2: Oracle Names Server を検出する

手順 1: Oracle Names Server を作成する Oracle Names がインストールされている各コンピュータに Oracle Names Server が必要な場合、Oracle Net8 Assistant を使用して Oracle Names Server を作成します。

1. Oracle Net8 Assistant を起動します。
 - UNIX の場合、`$ORACLE_HOME/bin` で `netasst` を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」 -> 「Oracle - HOME_NAME」 -> 「Network Administration」 -> 「Net8 Assistant」を選択します。
2. Oracle Net8 Assistant で、「Oracle Names Servers」フォルダをクリックします。
3. 「Edit」メニューから「Create」を選択するか、または「+」ボタンをクリックして、新規の Oracle Names Server を作成します。

Names ウィザードが起動します。

このウィザードでは、Oracle Names Server を作成し、指示に従って構成作業を進めていく過程で、次の項目の入力を求めるプロンプトが表示されます。

- Oracle Name サーバーの一意的な名前
- Oracle Names Server のリスニング用プロトコル・アドレス
TCP/IP を選択する場合、オラクル社では正規に登録されたデフォルトのポート番号である 1575 を Oracle Names Server に使用することをお薦めしています。
- データベースに情報を格納するか、または Oracle Names Server 間で情報をレプリケートするかを選択

(「Don't use a region database.」をクリックします)

- この Oracle Names Server が、リージョンで最初のネーム・サーバーであるかどうかの識別
これがリージョンで最初の Oracle Names Server でなければ、別の Oracle Names Server を検出するか、またはリージョンに存在する別の Oracle Names Server のアドレスを指定するように求めるプロンプトが表示されます。
- この Oracle Names Server がルート管理リージョンにあるかどうかの識別

この Oracle Names Server がルート管理リージョンにあると指定した場合、構成作業は完了です。

この Oracle Names Server がルート管理リージョンにないと指定した場合は、委任管理リージョンにあるものと見なされます。その場合ウィザードは、ローカル管理リージョ

ンのドメイン名と、ルート管理リージョンにある Oracle Names Server のアドレスを求め
るプロンプトを表示してから完了します。

ウィザードが完了すると、次のメッセージが表示されます。

```
「 A Names Server, onames_server, has been created with default settings. Use the Configure
Server section to modify the default configuration. 」
```

4. 「File」メニューから「Save Network Configuration」を選択します。
5. Oracle Names Server をさらに追加する場合は、手順 2 ~ 4 を繰り返してリージョンに
Oracle Names Server を作成します。Oracle Net8 Assistant では、1 台のコンピュータ上で
複数の Oracle Names Server の作成をサポートしていません。

Oracle Net8 Assistant では、少なくとも次の項目を設定して NAMES.ORA ファイルを作成し
ます。

- Oracle Names Server の名前 (NAMES.SERVER_NAME)
- Oracle Names のリスニング用プロトコル・アドレス (NAMES.ADDRESSES)

次に、NAMES.ORA ファイルを内容に関する注釈付きで示します。

```
#Oracle Names server name. The name should include the name of the domain this
Oracle Names server is in.
names.server_name=namesvr2.com
```

```
#Oracle Names server listening protocol address
(address=(protocol=tcp)(host=namesrv2-pc)(port=1575))
[(address=...)]
```

```
#If an Oracle Names server is in a delegated administrative region, identify the
address of an Oracle Names server in the root administrative region.
names.domain_hints=
(address=(protocol=tcp)(host=namesrv1-pc)(port=1575))
```

```
# Specify the domain controlled by this region and the time to live (TTL). If
this is the root administrative region, you must have NAME= (null) to identify
the root domain properly.
names.domains=
(domain=
(name=)
(min_ttl=86400)
)
```

アドレス情報に必要なプロトコル構文に関する詳細は、6-76 ページの「[プロトコル・アドレスの構成](#)」を参照してください。

複数ドメインの管理

リージョンに複数のドメインを管理させたい場合は、Oracle Net8 Assistant を使用して
NAMES.DOMAINS パラメータに追加のドメインを指定します。

1. Oracle Names Server フォルダをダブルクリックします。
2. Oracle Names Server を選択します。
3. ドロップダウン・リスト・ボックスから「Domains」を選択します。
4. 「Domains」タブを選択します。
5. 「Domain Name」フィールドにドメイン名を入力します。
6. 存続時間情報を入力します。
7. 「Add」をクリックします。
8. 「File」メニューから「Save Network Configuration」を選択します。
9. 別のドメインを追加する場合は、手順 1 ~ 8 を繰り返します。

次の例では、NAMES.DOMAINS にルート、COM、ORACLE.COM および HQ.ORACLE.COM ドメインのリストが含まれています。HQ.ORACLE.COM を定義するためには、HQ.ORACLE.COM より前に位置するすべてのドメインを定義しておく必要があります。

```
names.domains=  
  (domain_list=  
    (domain=  
      (name=  
        (min_ttl=86400)  
      )  
    (domain=  
      (name=com)  
      (min_ttl=86400)  
    )  
    (domain=  
      (name=oracle.com)  
      (min_ttl=86400)  
    )  
    (domain=  
      (name=hq.oracle.com)  
      (min_ttl=86400)  
    )  
  )  
)
```

手順 2: Oracle Names Server を検出する すべての Oracle Names Server の構成が終わったら、同じリージョン内で最初の Oracle Names Server を除いて、各 Oracle Names がリージョン内にある他のすべての Oracle Names Server を検出するようにします。

検出に関する説明は、2-45 ページの「[検出の理解](#)」を参照してください。

Oracle Names がインストールされているマシン上で、Oracle Net8 Assistant または NAMESCTL 制御ユーティリティを使用して、次の作業を実行します。

Oracle Net8 Assistant による操作	制御ユーティリティの NAMESCTL による操作
<p>1. Oracle Names Server フォルダをダブルクリックします。</p> <p>2. Oracle Names Server を選択します。</p> <p>3. ドロップダウン・リスト・ボックスから「Manage Server」を選択します。</p> <p>4. 「Control」タブをクリックします。</p> <p>5. 「Server Operations」フィールドの「Start」オプション・ボタンをクリックします。</p> <p>6. 「Apply」をクリックします。 次のメッセージが表示されます。 「Server started successfully.」</p> <p>7. 「Command」メニューから「Discover Oracle Names Servers」を選択します。 これによって、UNIX では \$ORACLE_HOME/network/names に .SDNS.ORA ファイルが、Windows プラットフォームでは ORACLE_HOME¥network¥names に SDNS.ORA ファイルが作成されます。このファイルには、Oracle Names Server の名前とアドレスが記述されています。Oracle Names Server のアドレスを検索するときに、このファイルが読み取られます。 次のメッセージが表示されます。 「Discovered Oracle Names Server in the region.Please exit the tool and start again.」 Oracle Names Server から応答がなく、特定の Oracle Names Server のネットワーク・アドレスを知っている場合は、入力を求めるダイアログが表示されるのでそのアドレスを入力します。</p>	<p>1. Oracle Names Server を起動します。 namesctl namesctl> start</p> <p>2. 他のすべての Oracle Names Server を検出します。 namesctl> reorder_ns これによって、UNIX では \$ORACLE_HOME/network/names に .SDNS.ORA ファイルが、Windows プラットフォームでは ORACLE_HOME¥network¥names に SDNS.ORA ファイルが作成されます。このファイルには、Oracle Names Server の名前とアドレスが記述されています。Oracle Names Server のアドレスを検索するときに、このファイルが読み取られます。 Oracle Names Server が見つからず、特定の Oracle Names Server のネットワーク・アドレスを知っている場合は、プロンプトに対してそのアドレスを入力します。例： namesctl> reorder_ns (address=(protocol=tcp)(host=mail-pc)(port=1575))</p> <p>3. 別の Oracle Names Server に問い合わせ、検出を確認します。 namesctl> query onames_server</p>

手順 4: Oracle Names Server を使用するためにクライアントとデータベース・サーバーを構成する

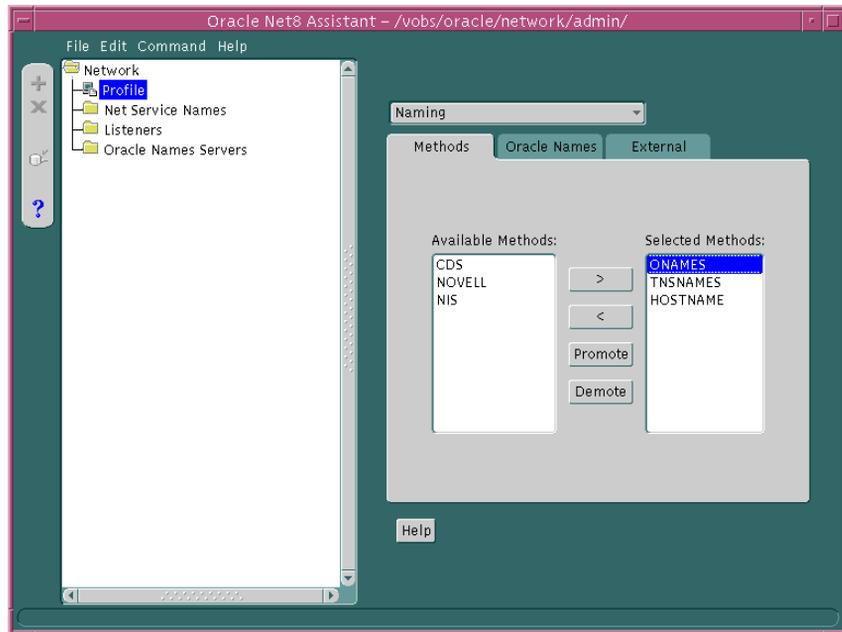
クライアントとデータベース・サーバーのコンピュータを構成するには、次の手順に従います。

1. 7-20 ページの「[デフォルトの Oracle Names Server の構成](#)」の説明に従って、オプションで優先する Oracle Names Server を作成します。
2. UNIX では \$ORACLE_HOME/network/names に .SDNS.ORA ファイルを作成し、Windows プラットフォームでは `ORACLE_HOME\network\names` に SDNS.ORA ファイルを作成して Oracle Names Server を検出します。このファイルには、Oracle Names Server の名前とアドレスが問い合わせ順に記述されています。クライアント・マシンは、このファイルを読み取って問い合わせ先の Oracle Names Server のアドレスを検索します。次の作業は、Oracle Net8 Assistant または NAMESCTL ユーティリティを使用して実行します。

検出に関する説明は、2-45 ページの「[検出の理解](#)」を参照してください。

Oracle Net8 Assistant による操作	制御ユーティリティの NAMESCTL による操作
<p>1. Oracle Net8 Assistant を起動します。</p> <p>-UNIX の場合、\$ORACLE_HOME/bin で netasst を実行します。</p> <p>-Windows NT の場合、「スタート」ボタン -> 「プログラム」-> 「Oracle - HOME_NAME」-> 「Network Administration」-> 「Net8 Assistant」を選択します。</p> <p>2. 「Command」メニューから「Discover Oracle Names Servers」を選択します。</p> <p>これによって、UNIX では \$ORACLE_HOME/network/names に .SDNS.ORA ファイルが、Windows プラットフォームでは ORACLE_HOME¥network¥names に SDNS.ORA ファイルが作成されます。このファイルには、Oracle Names Server の名前とアドレスが記述されています。Oracle Names Server のアドレスを検索するときに、このファイルが読み取られます。</p> <p>次のメッセージが表示されます。</p> <p>「Discovered Oracle Names Server in the region.Please exit the tool and start again.」</p> <p>Oracle Names Server から応答がなく、特定の Oracle Names Server のネットワーク・アドレスを知っている場合は、入力を求めるダイアログが表示されるのでそのアドレスを入力します。</p>	<p>他のすべての Oracle Names Server を検出します。</p> <pre>namesctl namesctl> reorder_ns</pre> <p>これによって、UNIX では \$ORACLE_HOME/network/names に .SDNS.ORA ファイルが、Windows プラットフォームでは ORACLE_HOME¥network¥names に SDNS.ORA ファイルが作成されます。このファイルには、Oracle Names Server の名前とアドレスが記述されています。Oracle Names Server のアドレスを検索するときに、このファイルが読み取られます。</p> <p>Oracle Names Server が見つからず、特定の Oracle Names Server のネットワーク・アドレスを知っている場合は、プロンプトに対してそのアドレスを入力します。例：</p> <pre>namesctl> reorder_ns (address=(protocol=tcp)(host=mail-pc)(port=1575))</pre>

3. Oracle Names をネット・サービス名を解決するための第 1 番目の方法とする場合は、次の手順に従って SQLNET.ORA ファイルに NAMES.DIRECTORY_PATH パラメータを設定します。
 - a. Oracle Net8 Assistant の「Profile」アイコンをクリックします。
 - b. ドロップダウン・リスト・ボックスから「Naming」を選択します。
 - c. 「Methods」タブをクリックします。
 - d. 「Available Methods」リストから「ONAMES」を選択し、「>」をクリックします。
 - e. 「Selected Methods」リストで「ONAMES」を選択し、「Promote」ボタンを使用して選択した項目をリストの最上位に移動します。



- f. 「File」メニューから「Save Network Configuration」を選択します。
- g. 「File」メニューから「Exit」を選択し、Oracle Net8 Assistant アプリケーションを終了します。

これにより SQLNET.ORA ファイルに、NAMES.DIRECTORY_PATH パラメータの先頭に ONAMES を含むエントリが入ります。

```
names.directory_path=(onames, tnsnames, hostname)
```

4. これ以外にも、クライアントが Oracle Names を使用方法として次の機能が構成可能です。

Oracle Net8 Assistant オプション	SQLNET.ORA パラメータ	説明
「Default Domain」	NAMES.DEFAULT_DOMAIN	クライアントから Oracle Names Server への要求が最も多く発生するドメイン名スペースを指定する。デフォルト・ドメインが設定されると、Oracle Names 要求ですべての未修飾の名前に対してこの名前が自動的に追加されます。Net8 では、デフォルト・ドメインがデフォルトで NULL に設定されています。
「Maximum Wait Each Attempt」	NAMES.INITIAL_RETRY_TIMEOUT	Oracle Names Server からの応答を待機する時間を指定する。この時間が経過すると、別の Oracle Names Server に要求を再送信します。デフォルトで、Net8 は 15 秒間待機します。
「Attempts Per Names Server」	NAMES.REQUEST_RETRIES	Oracle Names Server のリスト全体を試行する回数を指定する。この回数試行しても応答がないと操作は失敗します。デフォルトで Net8 は、操作が失敗になる前に Oracle Names Server のリストを 1 回反復します。
「Maximum Open Connections」	NAMES.MAX_OPEN_CONNECTIONS	Oracle Names クライアントが一度にオープンできる接続数を指定する。デフォルト設定で、Net8 では、クライアントを一度に 10 接続までオープンできます。このデフォルト値は、ほとんどの状況で十分な数です。
「Initial Preallocated Requests」	NAMES.MESSAGE_POOL_START_SIZE	クライアントのメッセージ・プール内に事前割当てできる初期メッセージ数。これらのメッセージは、Oracle Names Server に対して将来要求するときに使用できます。デフォルトの設定で、Net8 はプールに 10 メッセージを割り当てます。このデフォルト値は、ほとんどの状況で十分な数です。

これらの機能を追加または構成するには、次の手順に従います。

- a. Oracle Net8 Assistant の「Profile」アイコンをクリックします。
- b. ドロップダウン・リスト・ボックスから「Naming」を選択します。
- c. 「Oracle Names」タブをクリックします。
- d. 前の表に示されているオプションに必要なデータを入力します。
- e. 「File」メニューから「Save Network Configuration」を選択します。
- f. 「File」メニューから「Exit」を選択し、Oracle Net8 Assistant アプリケーションを終了します。

手順 5: リスナーを構成する

オペレーティング・システムの主プロトコルを使用しない場合は、データベース・プロトコルのリスニング・アドレスを構成します。リスナーの構成方法の詳細は、6-76 ページの「[プロトコル・アドレスの構成](#)」を参照してください。

さらに Oracle Net8 Assistant で、LISTENER.ORA ファイルに `USE_PLUG_AND_PLAY_listener_name=ON` を設定します。これによりリスナーは、サービス名、SID、グローバル・データベース名などのデータベース情報を Oracle Names Server に登録できるようになります。

1. Oracle Net8 Assistant を起動します。
 - UNIX の場合、`$ORACLE_HOME/bin` で `netasst` を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」 -> 「Oracle - HOME_NAME」 -> 「Network Administration」 -> 「Net8 Assistant」を選択します。
2. 「Listeners」フォルダをダブルクリックします。
3. 目的のリスナーをクリックします。
4. ドロップダウン・リスト・ボックスから「General Parameter」を選択します。
5. 「Register Service with Oracle Names」をクリックします。
6. 「File」メニューから「Save Network Configuration」を選択します。
7. 「File」メニューから「Exit」を選択し、Oracle Net8 Assistant アプリケーションを終了します。

手順 6: Oracle Names Server にデータベースを登録する

6-21 ページの「Oracle Names ネーミング・メソッドによるネットワークの構成」の表に説明されているように、Oracle Names Server はデータベースの場所、Oracle Names Server の場所、ネット・サービス名、別名、グローバル・データベース・リンクおよび Oracle Connection Manager を含む多くの種類の情報を格納します。データベースはリスナーに情報を登録するため、これらの情報の一部は Oracle Names Server に自動的に登録されます。ただし、次に示す情報を格納するには、Oracle Net8 Assistant または NAMESCTL 制御ユーティリティを使用して登録する必要があります。

- ネット・サービス名
- グローバル・データベース・リンクの資格証明とデータベース修飾子
- 別名

登録は、1つの管理リージョンにつき1台の Oracle Names Server でのみ行われる必要があります。データがリージョン・データベースに格納されている場合は、この Oracle Names Server 以外のすべての Oracle Names Server によってこの情報が参照され、リージョン・データベースを使用しない場合は、他のすべての Oracle Names に対してこの情報が伝送されません。

これらの情報の登録後、8-17 ページの「NAMESCTL または Oracle Net8 Assistant によるネットワーク・オブジェクトのテスト」の説明に従って問い合わせする必要があります。

ネット・サービス名

オラクル社では、各サービスについてネット・サービス名を作成するようにお薦めしています。ネット・サービス名を作成しないと、接続を試みるユーザーはグローバル・データベース名を渡す必要があります。エンド・ユーザーにこの情報を提供するかわりに、サービスに対してネット・サービス名を作成するようにしてください。

次の作業は、Oracle Net8 Assistant または NAMESCTL 制御ユーティリティを使用して実行します。

Oracle Net8 Assistant による操作	制御ユーティリティの NAMESCTL による操作
<p>1. Oracle Net8 Assistant を起動します。</p> <p>-UNIX の場合、\$ORACLE_HOME/bin で <code>netasst</code> を実行します。</p> <p>-Windows NT の場合、「スタート」ボタン -> 「プログラム」-> 「Oracle - HOME_NAME」-> 「Network Administration」-> 「Net8 Assistant」を選択します。</p> <p>2. Oracle Names Server フォルダをダブルクリックします。</p> <p>3. 目的の Oracle Names Server を選択します。</p> <p>4. ドロップダウン・リスト・ボックスから「Manage Data」を選択します。</p> <p>5. 「Net Service Names」タブをクリックします。</p> <p>6. 「Add」を選択します。</p> <p>7. 「Service Name」フィールドに別名を覚えやすい名前を入力します。</p> <p>8. プロトコルを選択し、リスナー・アドレスを入力します。</p> <p>追加情報 : アドレス情報に必要なプロトコル構文に関する詳細は、6-76 ページの「プロトコル・アドレスの構成」を参照してください。</p> <p>9. データベースのサービス名を入力し、セッション・タイプを選択します。</p> <p>10. 「Execute」をクリックします。</p> <p>11. 「File」メニューから「Save Network Configuration」を選択します。</p> <p>12. 「File」メニューから「Exit」を選択し、Oracle Net8 Assistant アプリケーションを終了します。</p>	<p>ネット・サービス名を作成し、Oracle Names Server に登録します。</p> <pre>namesctl namesctl> register net_service_name -D address_data</pre> <p>たとえば、デフォルトの TCP/IP リスニング・アドレスを使用して SALES というネット・サービス名を登録するには、次のように入力します。</p> <pre>register sales -D (description=(address=(protocol=tcp)(host=sales-pc)(port=1521)))</pre> <p>追加情報 : アドレス情報に必要なプロトコル構文に関する詳細は、6-76 ページの「プロトコル・アドレスの構成」を参照してください。</p>

これまで構成方法としてローカル・ネーミングを使用していて、Oracle Names 構成に移行する場合は、TNSNAMES.ORA ファイルで使用されているネット・サービス名を Oracle Names Server にアップロードすることができます。ローカル名構成ファイルの情報を Oracle Names Server にロードするには、Oracle Net8 Assistant または NAMESCTL 制御ユーティリティを使用します。

Oracle Net8 Assistant による操作	制御ユーティリティの NAMESCTL による操作
1. Oracle Names Server を選択します。	コマンド行から、次のように入力します。
2. ドロップダウン・リスト・ボックスから「Manage Data」を選択します。	<code>namesctl</code> <code>namesctl> load_tnsnames file_name</code>
3. 「Load」を選択します。	
4. 「File」フィールドに、現行のマスター・ローカル名構成ファイルへのパスを入力します。	
5. 「Execute」をクリックします。	
6. 「File」メニューから「Save Network Configuration」を選択します。	
7. 「File」メニューから「Exit」を選択し、Oracle Net8 Assistant アプリケーションを終了します。	

グローバル・データベース・リンクの資格証明とデータベース修飾子

オラクル社では、データベースごとに1個のグローバル・データベース・リンクを作成することをお薦めしています。グローバル・データベース名であるグローバル・データベース・リンクは、SALES.COMのようにOracle Names Serverに自動的に登録されます。

たとえば、グローバル・データベース・リンクを使用してリモート・データベースに接続するには次のように入力します。

```
sql> CONNECT @sales.com
```

したがって、ユーザー名とパスワードからなる資格証明を入力するの でなければ、この情報を登録する必要はありません。

グローバル・データベース・リンクを登録するには、Oracle Net8 Assistant または NAMESCTL 制御ユーティリティを使用します。NAMESCTL 制御ユーティリティはグローバル・データベース・リンクを登録する上で制限があるので、オラクル社では Oracle Net8 Assistant の使用をお薦めしています。

Oracle Net8 Assistant による操作	制御ユーティリティの NAMESCTL による操作
<ol style="list-style-type: none"> Oracle Net8 Assistant を起動します。 -UNIX の場合、\$ORACLE_HOME/bin で <code>netasst</code> を実行します。 -Windows NT の場合、「スタート」ボタン -> 「プログラム」 -> 「Oracle - HOME_NAME」 -> 「Network Administration」 -> 「Net8 Assistant」を選択します。 Oracle Names Server フォルダをダブルクリックします。 目的の Oracle Names Server を選択します。 ドロップダウン・リスト・ボックスから「Manage Data」を選択します。 「DB Links」タブをクリックします。 「Add」を選択します。 「DB Link Name」フィールドにグローバル・データベース・リンクを入力します。 注意: INITSID.ORA ファイルの GLOBAL_NAMES パラメータが TRUE に設定されている場合、グローバル・データベース名を入力する必要があります。 「User and Password」フィールドに、このデータベースの有効なユーザー名とパスワードからなる資格証明を入力します。 「Execute」をクリックします。 リンク修飾子を使用する場合は、6-46 ページの「リンク修飾子の追加」を参照してください。 「File」メニューから「Save Network Configuration」を選択します。 「File」メニューから「Exit」を選択し、Oracle Net8 Assistant アプリケーションを終了します。 	<p>グローバル・データベース・リンクを作成し、Oracle Names Server に登録します。</p> <pre>namesctl namesctl> register dblink -d address_data</pre> <p>たとえば、デフォルトの TCP/IP リスニング・アドレスを使用して SALES.COM というグローバル・データベース・リンクを登録するには、次のように入力します。</p> <pre>register sales.com -d (description=(address=(protocol-tcp)(host=sales-pc)(port=1521)))</pre> <p>追加情報: アドレス情報に必要なプロトコル構文に関する詳細は、6-76 ページの「プロトコル・アドレスの構成」を参照してください。</p> <p>NAMESCTL ユーティリティでは、ユーザー名とパスワードの資格証明を設定できません。</p>

作成するグローバル・データベース・リンクの数はデータベースごとに 1 個をお勧めしますが、追加できるリンク修飾子の数に制限はありません。リンク修飾子はグローバル・データベース・リンクに付加され、データベースのユーザー名とパスワードからなる資格証明の代替設定を提供します。たとえば、RDBMS.OMD というリンク修飾子は SALES.COM に付加できます。グローバル・データベース・リンクを使用してリモート・データベースに接続するには、次のように入力します。

```
sql> CONNECT @sales.com@emp
```

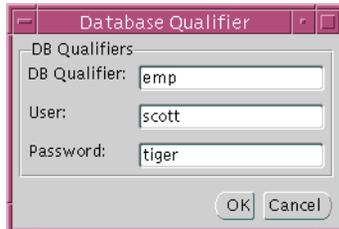
リンク修飾子の追加

リンク修飾子は、主に複数のデータベース・リンクが使用されている古い Oracle データベース環境で使用されます。複数のデータベース・リンクを使用しなくても、1 個のデータベース・リンクに対して複数のリンク修飾子を作成できます。

グローバル・データベース・リンクに対してグローバル・リンク修飾子を登録するには、Oracle Net8 Assistant を使用します。NAMESCTL ユーティリティでは、リンク修飾子を作成できません。

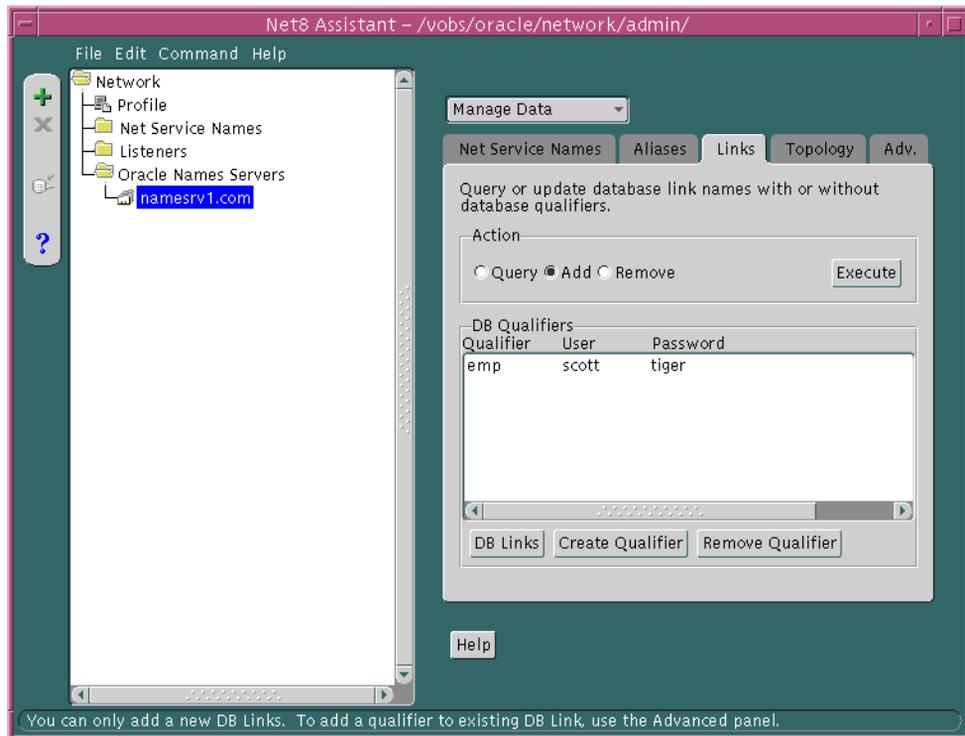
1. 前の手順の手順 1 ~ 8 を実行します。
2. 「DB Qualifiers」をクリックします。
3. 「Create Qualifier」をクリックします。

「Database Qualifier」ダイアログ・ボックスが表示されます。



4. リンク修飾子の名前と、有効なユーザー名とパスワードからなる資格証明を入力し、「OK」をクリックします。

リンク修飾子が「DB Qualifiers」リスト・ボックスに追加されます。



5. 追加するリンク修飾子ごとに手順3～5を繰り返します。
6. 「Execute」をクリックします。
7. 「File」メニューから「Save Network Configuration」を選択します。
8. 「File」メニューから「Exit」を選択し、Oracle Net8 Assistant アプリケーションを終了します。

グローバル・データベース・リンクとリンク修飾子が正しく動作していることを確認するには、次の作業を実行します。

1. リモート・データベースに接続します。

```
sql> CONNECT user_name/password@db_link;  
sql> CONNECT user_name/password@db_link@link_qualifier;
```

例：

```
sql> CONNECT scott/tiger@sales.com;  
sql> CONNECT scott/tiger@sales.com@emp;
```

2. ユーザーが認可されている表に対して SELECT 文を実行します。

```
sql> SELECT * from table@db_link;  
sql> SELECT * from table@db_link@link_qualifier;
```

例：

```
sql> SELECT * from EMP@sales.com;  
sql> SELECT * from EMP@sales.com@emp;
```

データベース・リンクの詳細は、『Oracle8i 分散システム』を参照してください。

別名

別名は、Oracle Names Server、ネット・サービス名、グローバル・データベース・リンクなど、Oracle Names Server に格納されている項目タイプに対して使用されます。

グローバル・データベース・リンクを登録するには、Oracle Net8 Assistant または NAMESCTL 制御ユーティリティを使用します。

Oracle Net8 Assistant による操作	制御ユーティリティの NAMESCTL による操作
<p>1. Oracle Net8 Assistant を起動します。</p> <p>-UNIX の場合、\$ORACLE_HOME/bin で netasst を実行します。</p> <p>-Windows NT の場合、「スタート」ボタン -> 「プログラム」 -> 「Oracle - HOME_NAME」 -> 「Network Administration」 -> 「Net8 Assistant」を選択します。</p>	<p>別名を作成し、Oracle Names Server に登録します。</p> <pre>namesctl namesctl> register alias -d address_data</pre> <p>たとえば、デフォルトの TCP/IP リスニング・アドレスを使用して SALES というネット・サービス名を登録するには、次のように入力します。</p> <pre>register sales -d (description=(address=(protocol=tcp)(host=sales-pc)(port=1521)))</pre> <p>追加情報: アドレス情報に必要なプロトコル構文に関する詳細は、6-76 ページの「プロトコル・アドレスの構成」を参照してください。</p>
<p>2. Oracle Names Server フォルダをダブルクリックします。</p>	
<p>3. 管理リージョンにある目的の Oracle Names Server を選択します。</p>	
<p>4. ドロップダウン・リスト・ボックスから「Manage Data」を選択します。</p>	
<p>5. 「Aliases」タブをクリックします。</p>	
<p>6. 「Add」を選択します。</p>	
<p>7. 「Domain Name」フィールドでサブドメインを指定します。</p>	
<p>8. 「Name Server Name」フィールドに Oracle Names Server 名を入力します。</p>	
<p>9. 「Execute」をクリックします。</p>	
<p>10. 「File」メニューから「Save Network Configuration」を選択します。</p>	
<p>11. 「File」メニューから「Exit」を選択し、Oracle Net8 Assistant アプリケーションを終了します。</p>	

手順 7: (オプション) 委任管理リージョンにドメインを委任する

委任管理リージョンが構成されている場合、上位レベルの管理リージョンはサブドメインを委任管理リージョンに委任する必要があります。このようにしないと、上位レベルの管理リージョンにある Oracle Names Server がすべてのサブドメインに対する権限を持ちます。

いったんドメインが委任されると、現行リージョンにある Oracle Names Server はそれ以降のすべての処理を、別の Oracle Names Server によって管理されるドメインのあるサブリージョンに渡します。

ドメインを委任するには、Oracle Net8 Assistant または NAMESCTL 制御ユーティリティを使用します。

Oracle Net8 Assistant による操作	制御ユーティリティの NAMESCTL による操作
<p>1. Oracle Net8 Assistant を起動します。</p> <p>-UNIX の場合、\$ORACLE_HOME/bin で <code>netasst</code> を実行します。</p> <p>-Windows NT の場合、「スタート」ボタン -> 「プログラム」-> 「Oracle - HOME_NAME」-> 「Network Administration」-> 「Net8 Assistant」を選択します。</p>	<p>別名を作成し、Oracle Names Server に登録します。</p> <pre>namesctl namesctl> delegate_domain domain_name onames_server [(description=(address_list=) (address= ...)[)]]</pre> <p>追加情報: アドレス情報に必要なプロトコル構文に関する詳細は、6-76 ページの「プロトコル・アドレスの構成」を参照してください。</p>
<p>2. Oracle Names Server フォルダをダブルクリックします。</p>	
<p>3. 上位レベル・リージョンにある目的の Oracle Names Server を選択します。</p>	<p>たとえば、NAMESRV3.ACME.COM からサービスの提供を受ける ACME.COM ドメインを委任するには、次のように入力します。</p>
<p>4. ドロップダウン・リスト・ボックスから「Manage Data」を選択します。</p>	<pre>delegate_domain acme.com namesrv3.acme.com (address=(protocol-tcp)(host=namesrv3-pc)(port=1575))</pre>
<p>5. 「Topology」タブをクリックします。</p>	
<p>6. 「Delegate Domain」を選択します。</p>	
<p>7. 「Domain Name」フィールドにサブドメインの名前を入力します。</p>	
<p>8. 「Names Server Name」フィールドにドメインにサービスを提供する Oracle Names Server と、そのリスニング・アドレスを入力します。</p> <p>追加情報: アドレス情報に必要なプロトコル構文に関する詳細は、6-76 ページの「プロトコル・アドレスの構成」を参照してください。</p>	
<p>9. 「Execute」をクリックします。</p>	
<p>10. 「File」メニューから「Save Network Configuration」を選択します。</p>	
<p>11. 「File」メニューから「Exit」を選択し、Oracle Net8 Assistant アプリケーションを終了します。</p>	

外部ネーミング・メソッドによるネットワークの構成

外部ネーミングとは、サポートされている Oracle 以外のネーミング・サービスを使用して、ネットワーク・アドレスにネット・サービス名を変換する方法をいいます。外部ネーミングでは、顧客の固有の (Oracle 以外の) ネーミング・サービスに格納されているネット・サービス名が変換されます。これらのサービスは、次のとおりです。

- ネットワーク情報サービス (NIS)
- NetWare ディレクトリ・サービス (NDS)

NIS の構成

システム・インフラの一部としてすでにネットワーク情報サービス (NIS) を使用している組織や会社のために、NIS 外部ネーミングを使用して NIS 内に Oracle ネット・サービス名とアドレスを格納するオプションが用意されています。

ユーザーが次のようなコマンドを指定した場合、

```
sqlplus scott/tiger@payroll
```

(*payroll* は Oracle ネット・サービス名) クライアント・プログラム (またはクライアント・プログラムとしての役割を果たすサーバー) を実行するノード上の NIS 外部ネーミングは、ネットワーク内にある NIS サーバーと接続し、ネット・サービス名を NIS サーバーに渡します。NIS サーバーはネット・サービス名を Net8 アドレスに変換し、このアドレスをクライアント・プログラム (またはクライアント・プログラムとしての役割を果たすサーバー) に返します。次に、クライアント・プログラムはこのアドレスを使用して Oracle データベースに接続します。

NIS サーバーとして動作するコンピュータは、名前解決要求を処理する *yppserv* というプログラムを実行します。*yppserv* は、*maps* と呼ばれる特殊なファイルに異なるタイプのデータを格納します。たとえば、パスワードは *passwd.byname* と呼ばれるマップに格納されます。Oracle データベース・ネット・サービス名は、*tnsnames* と呼ばれるマップに格納されます。

前の項で紹介したようなコマンドをユーザーが発行すると、NIS 外部ネーミングは RPC コールを使用して *yppserv* に接続し、Oracle ネット・サービス名の *payroll* とマップの名前である *tnsnames* を渡します。*yppserv* プログラムは、名前 *payroll* とそれに対応する値 (ネット・サービス名のアドレス) を *tnsnames* マップ内で検索します。このアドレスはクライアントに返され、クライアント・プログラム (またはクライアント・プログラムとしての役割を果たすサーバー) は、このアドレスを使用してデータベース・サーバーに接続します。

システム要件

NIS 外部ネーミングには、SQL*Net 2.2 以上が必要です。

手順 1: NIS サーバーを構成して NIS 外部ネーミングをサポートする

NIS 外部ネーミングをサポートするようにサーバーを構成する前に、Oracle データベース・ネット・サービス名を変換する必要がある NIS サーバー上で NIS が構築され、動作していることを確認してください。具体的には、NIS のマニュアルを参照してください。

NIS マップの既存セットへの tnsnames マップの追加 NIS マップの既存セットに tnsnames マップを追加する手順は、次のとおりです。

1. 6-12 ページの「ローカル・ネーミング・メソッドによるネットワークの構成」の説明に従って、TNSNAMES.ORA ファイルを作成します。

注意： TNSNAMES.ORA ファイルのコピーは、できるだけ \$ORACLE_HOME/network/admin ディレクトリに保管してください。このファイルをあとでもう一度使用して、ネット・サービス名を NIS マップにロードしなければならない場合があります。

2. tns2nis プログラムを使用して、TNSNAMES.ORA ファイルの内容を tnsnames に変換します。

注意： tns2nis プログラムは、Oracle Universal Installer のテープまたはディスク上の NIS 外部ネーミングによって提供されます。

たとえば、引数を 1 つ付けてコマンド行で tns2nis を実行します。

```
tns2nis tnsnames.ora
```

tns2nis は、現行ディレクトリから NATIVE.ORA ファイルを読み込みます。TNSNAMES.ORA が現行ディレクトリにない場合は、フル・パス名を使用してその位置を指定します（たとえば、/etc/native.ora または \$ORACLE_HOME/network/admin/tnsnames.ora ）。

これにより tnsnames マップは、現行の作業ディレクトリに書き込まれます。

3. tnsnames が NIS サーバーにない場合は、tnsnames を NIS サーバーにコピーします。
4. NIS プログラムの makedbm を使用して tnsnames マップをインストールします。詳細は、NIS のマニュアルを参照してください。

注意： この手順は、NIS の管理担当者が行ってください。

makedbm は、tnsnames マップを NIS サーバーが読み取り可能な 2 つのファイルにマップします。これらのファイルの位置は、プラットフォームによって異なります。詳細は、ご使用のプラットフォームに対応するマニュアルを参照してください。

たとえば、Sun Solaris 上で tnsnames マップを生成し、インストールするには、root としてコマンド行から次のように入力します。

```
# makedbm tnsnames /var/yp/'domainname'/tnsnames
```

tnsnames マップが適切にインストールされたことの確認 次の形式でコマンドを入力することにより、NIS サーバーをテストしてマップが適切にインストールされていることを確認できます。

```
ypmatch net_service_name tnsnames
```

たとえば、次のように入力できます。

```
ypmatch payroll.com tnsnames
```

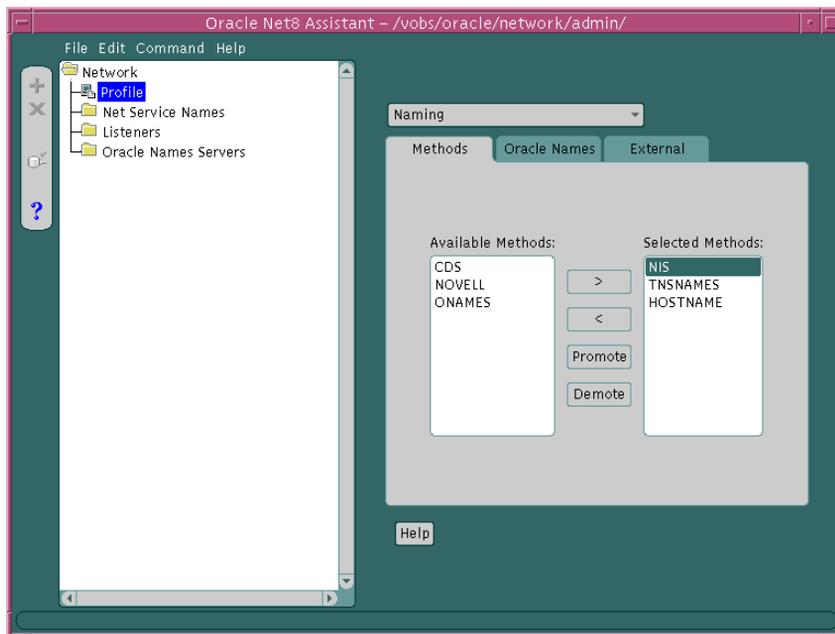
例として、次のようにアドレスの長さ（文字数）のついたアドレスが返されます。

```
99 (description=(address=(protocol=tcp)
(host=garlic)(port=1999)))
(connect_data=(service_name=dirprod)))
```

手順 2: クライアントを構成する

クライアントを構成するには、SQLNET.ORA ファイルの NAMES.DIRECTORY_PATH パラメータに指定する第 1 番目のネーミング・メソッドとして NIS を構成します。このパラメータによって、Net8 が名前解決に使用するネーミング・メソッドの順序を指定します。

1. Oracle Net8 Assistant を起動します。
 - UNIX の場合、\$ORACLE_HOME/bin で netasst を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」-> 「Oracle - HOME_NAME」-> 「Network Administration」-> 「Net8 Assistant」を選択します。
2. 「Profile」アイコンをクリックします。
3. ドロップダウン・リスト・ボックスから「Naming」を選択します。
4. 「Methods」タブをクリックします。
5. 「Available Methods」リストから「NIS」を選択し、「>」をクリックします。
6. 「Selected Methods」リストで「NIS」を選択し、「Promote」ボタンを使用して選択した項目をリストの最上位に移動します。



7. 「File」メニューから「Save Network Configuration」を選択します。

8. 「File」メニューから「Exit」を選択し、Oracle Net8 Assistant アプリケーションを終了します。

SQLNET.ORA には、次のようなエントリが含まれています。

```
names.directory_path=(nis, tnsnames, hostname)
```

NDS の構成

NDS 外部ネーミングは、NDS 固有のネーミング表記法を使用して、Novell NDS を利用できるネットワーク上で Oracle データベースに接続できるようにします。NDS 外部ネーミングをクライアントとサーバー上にインストールすると、次のように入力できます。

```
username/password@net_service_name
```

NDS 外部ネーミングを使用するネットワーク・ユーザーには、次の利点があります。

- クライアントがデータベースに接続するときに簡単な NDS 名 (部分名または完全名) を使用できます。
- TNS アドレスのメンテナンスを簡素化します。すなわち、1 つの変更が NDS 外部ネーミングを使用するすべてのクライアントに作用します。
- リスナーが通知する必要がないので、ネットワーク・トラフィックが軽減されます。

クライアントでの操作

NDS 外部ネーミングはクライアント・ワークステーション上に常駐し、NDS オブジェクト名を TNS アドレスに変換します。クライアント・コードは、名前が Oracle ネット・サービス名と一致する NDS オブジェクトについて NDS ツリーから属性を取得します。この名前は、完全名または部分名のいずれでもかまいません。その名前が部分名であれば、その部分名は現行の名前コンテキストに基づいて修飾されます。

NetWare サーバーでの操作

NDS 外部ネーミングのサーバー側には、次の 3 つの局面があります。

- スキーマの拡張
- SAP (Service Advertising Protocol) の無効化
- ネット・サービス名と、アドレスの NDS への格納

スキーマの拡張 NetWare 4 サーバーでの Oracle インストレーションの最中に、ORACLE:DBInstance と呼ばれるオブジェクト・クラスを挿入するために NDS スキーマが拡張されます。NDS 外部ネーミングを機能させるには、このクラスに ORACLE:TNSAddress と呼ばれる属性が必要です。クラスが存在しない場合は、そのクラスが作成され、TNSAddress 属性が挿入されます。クラスは存在するが、TNSAddress 属性が存在しない場合は、そのク

ラスが変更されます。インストール中に、これを実行する NLM を ORASCHEM.NLM といいます。

SAP (Service Advertising Protocol) の無効化 NetWare の SPX プロトコルは、CONFIG.ORA 内で ORACLE_SAP と呼ばれる値を検索します。このパラメータの値は ON または OFF です。このパラメータを指定しない場合、デフォルトの設定は ON です。これは、SPX ネットワークに関するパフォーマンスに密接な関係があります。ORACLE_SAP が ON であれば、SQL*Net リスナーは SAP (サービス・アドバタイジング・プロトコル) を使用して、そのアドレスを通知します。ORACLE_SAP を OFF に設定すると、SQL*Net リスナーは NDS を使用して情報をそのクライアントに送ります。

ネットワークが NDS 使用可能クライアント (すなわち、NetWare 4 以上を使用するクライアント) のみで構成されている場合は、CONFIG.ORA ファイルを編集して ORACLE_SAP を OFF に設定すると、ネットワークのパフォーマンスが向上します。

ネット・サービス名と、アドレスの NDS への格納 リスナーを起動すると、サーバー上に常駐する Oracle データベース・インスタンスが検索され、NDS データベースにそのリスナーのアドレスが格納されます。

注意： Oracle データベースの NDS オブジェクトをあらかじめ作成しておく必要があります。

その時点で、クライアントは NDS データベースからアドレスをアクセスできます。

システム要件

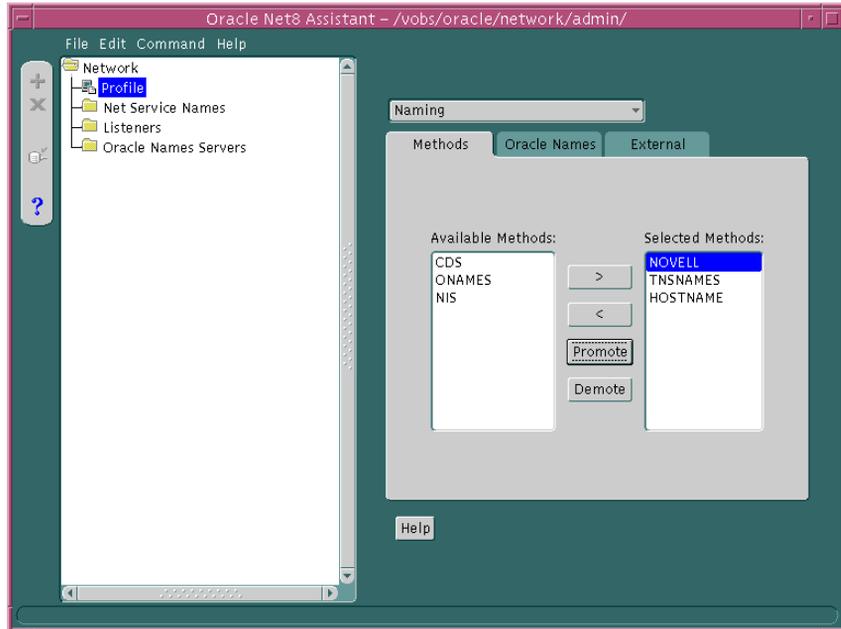
NDS 外部ネーミングには、SQL*Net 2.2 以上と Oracle 7.2 以上が必要です。このアダプタは、Novell ライブラリを実行するクライアントと一緒に使用できますが、サーバー上には NetWare 4.1 以上が必要です。

クライアントの構成

クライアントを構成するには、次の手順に従います。

1. SQLNET.ORA ファイルの NAMES.DIRECTORY_PATH パラメータに指定する第 1 番目のネーミング・メソッドとして NDS を構成します。このパラメータによって、Net8 が名前解決に使用するネーミング・メソッドの順序を指定します。
 - a. Oracle Net8 Assistant を起動します。
 - UNIX の場合、\$ORACLE_HOME/bin で netasst を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」-> 「Oracle - HOME_NAME」-> 「Network Administration」-> 「Net8 Assistant」を選択します。
 - b. 「Profile」アイコンをクリックします。
 - c. ドロップダウン・リスト・ボックスから「Naming」を選択します。
 - d. 「Methods」タブをクリックします。
 - e. 「Available Methods」リストから「NDS」を選択し、「>」をクリックします。

- f. 「Selected Methods」リストで「Novell」を選択し、「Promote」ボタンを使用して選択した項目をリストの最上位に移動します。



- g. 「File」メニューから「Save Network Configuration」を選択します。
 h. 「File」メニューから「Exit」を選択し、Oracle Net8 Assistant アプリケーションを終了します。

これにより SQLNET.ORA ファイルに、NAMES.DIRECTORY_PATH パラメータの先頭に NOVELL を含むエントリが入ります。

```
names.directory_path=(novell, tnsnames, hostname)
```

2. オプションで、タイプなし名またはタイプ付き名を構成します。

- タイプなし名を指定するには、次の手順を実行します。

次のパラメータを SQLNET.ORA ファイルに追加して、名前を解決するためのデフォルトの名前コンテキストを指定します。

```
native_names.nds.name_context
```

注意： このパラメータは SQLNET.ORA ファイルに手動で追加する必要があります。このパラメータは、Oracle Net8 Assistant では作成できません。

たとえば、データベース・オブジェクトの名前が Payroll.Finance.Oracle で、SQLNET.ORA パラメータが NATIVE_NAMES.NDS.NAME_CONTEXT=Finance.Oracle である場合、Payroll という名前は .Payroll.Finance.Oracle のように修飾されます。これはタイプなし名の例です。

注意： 先行するドットは、この名前が完全な NDS の名前であることを示します。SQLNET.ORA 内の名前コンテキスト・パラメータを上書きする場合は、先行するドットを使用して接続文字列内に完全な NDS 名を指定できます。

- タイプ付き名を指定するには、次のように SQLNET.ORA 内にパラメータと値を入力します。

```
native_names.nds.name_context=OU=Finance.O=Oracle
```

構文解析されると、次に示すタイプ付き名が生成されます。

```
CN=Payroll.OU=Finance.O=Oracle.
```

このパラメータの機能は、NET.CFG のパラメータ name context と似ています。SQLNET.ORA の名前コンテキストによって NET.CFG のエントリが上書きされません。SQLNET.ORA パラメータを指定しないと、NET.CFG パラメータが使用されず、いずれのファイルでも名前コンテキストを指定しないと、その名前は [root] にデフォルト設定されます。NET.CFG パラメータの詳細は、Novell クライアントのマニュアルを参照してください。

注意： SQLNET.ORA で指定したデフォルトの名前コンテキストには、先行するドットを挿入できません。このため、エラー・コード 309 (ERR_EXPECTED_IDENTIFIER) が発生します。

```
The parameter being parsed is not typed.
```

サーバーの構成

NetWare サーバーを構成するには、次の手順に従います。

1. SQLNET.ORA ファイルの NAMES.DIRECTORY_PATH パラメータに指定する第 1 番目のネーミング・メソッドとして NDS を構成します。このパラメータによって、Net8 が名前解決に使用するネーミング・メソッドの順序を指定します。
 - a. Oracle Net8 Assistant を起動します。
 - UNIX の場合、\$ORACLE_HOME/bin で netasst を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」-> 「Oracle - HOME_NAME」-> 「Network Administration」-> 「Net8 Assistant」を選択します。
 - b. 「Profile」アイコンをクリックします。
 - c. ドロップダウン・リスト・ボックスから「Naming」を選択します。
 - d. 「Methods」タブをクリックします。
 - e. 「Available Methods」リストから「Novell」を選択し、「>」をクリックします。
 - f. 「Selected Methods」リストで「Novell」を選択し、「Promote」ボタンを使用して選択した項目をリストの最上位に移動します。
 - g. 「File」メニューから「Save Network Configuration」を選択します。
 - h. 「File」メニューから「Exit」を選択し、Oracle Net8 Assistant アプリケーションを終了します。
2. オプションで、\$ORACLE_HOME/nlm にある CONFIG.ORA ファイルに次のパラメータを追加します。

```
oracle_sap=[off|on]
```

ORACLE_SAP は、ON または OFF に設定できます。ORACLE_SAP が ON であれば、リスナーは SAP (サービス・アダプタイジング・プロトコル) を使用して、そのアドレスを通知します。ORACLE_SAP を OFF に設定すると、リスナーは NDS を使用して情報をそのクライアントに送ります。CONFIG.ORA 内で指定しないと、ORACLE_SAP はデフォルトで「ON」に設定されます。すべてのクライアントが NDS を使用するネットワーク上のネットワーク・トラフィックを減らすには、次のオプションを指定します。

```
oracle_sap=off
```

制約

次に、NDS 外部ネーミングを使用する場合のいくつかの制約を示します。

- NDS データベースに格納されている TNS アドレスの長さは、2048 文字を超えることはできません。
- 1 つのデータベース・インスタンスに複数のリスナーを使用することはできません。使用すると、最後に起動するリスナーによってデータベース・オブジェクトに格納されているその他の TNS アドレスが上書きされます。
- SID サポートがサーバー上で使用可能な場合は、データベース・インスタンスに NULL SID を使用できません。インスタンスのうち 1 つでも NULL SID を使用すると、SQL*Net バージョン 2 または Net8 を使用して接続することができません。
- SID サポートが使用可能でない場合は、リスナーの SID_LIST で指定した最後の SID が、使用する SID となります。この場合、SID はユーザーとデータベースに対して透過的ではありません。ユーザーには SID が見えず、データベースは SID を無視します。

注意： SID サポートは、CONFIG.ORA ファイル内の次のパラメータで制御されます。

```
nw_enable_sid_support=[true|false]
```

リスナーの構成

サービスは、リスナーに支援されてセッションを調整します。リスナーは、クライアント・アプリケーションのために接続要求を受け取ります。リスナーは、データベース・サービスまたはデータベース以外のサービスに対し、リスナー構成ファイル (LISTENER.ORA) で指定されたアドレスをリスン・オンするように構成されます。LISTENER.ORA ファイルを構成しておくこと、リスナー制御ユーティリティ (LSNRCTL) を使用してリスナーを管理できます。

Oracle8i のデータベース・インスタンスは、リスナーにそれ自身を登録します。前のリリースでは、データベース SID の記述情報を構成する必要がありました。リリース 8.1 の機能を利用するには、最新のリスナーを使用してください。

デフォルトで、次のリスニング終点がインストール中に自動的に構成されます。

- 外部プロシージャに接続するための IPC
- Oracle データベースに接続する Net8 クライアントのための TCP/IP
- Java オプションに接続する IIOP クライアントのための TCP/IP

このほか、データベースと外部プロシージャ用に SID_LIST が自動的に構成されます。

次に、デフォルトの LISTENER.ORA を内容に対する注釈付きで示します。

```
listener=
  (description_list=
    # ipc listenening endpoint for external procedures' connections
    (description=
      (address_list=
        (address=(protocol=ipc)(key=extproc0))
      )
    )
    # tcp/ip listenening endpoint for java option connections
    (description=
      (protocol_stack=
        (presentation=giop)
        (session=raw)
      )
      (address_list=
        (address=(protocol=tcp)(host=sales-pc)(port=2481))
      )
    )
  )
```

```
# tcp/ip listenening endpoint for database connections
(description=
  (address_list=
    (address=(protocol=tcp)(host=sales-pc)(port=1521))
  )
)
)
# list of services for a listener named LISTENER to receive requests for
sid_list_listener=
(sid_list=
  # service description for external procedures
  (sid_desc=
    (sid_name=plsextproc)
    (oracle_home=c:\%orahome81)
    (program=extproc)
  )
  # service description for the database specified
  (sid_desc=
    (global_dbname=sales.com)
    (oracle_home=c:\%orahome81)
    (sid_name=sales)
  )
)
)
```

この項では、次のリスナー構成項目を取り上げます。

- [LISTENER.ORA ファイル変更後のリスナーの再起動](#)
- [リスナー・プロトコル・アドレスの構成](#)
- [LISTENER 以外のリスナー名の使用](#)
- [大量の接続要求の処理](#)
- [サービス情報によるリスナーの構成](#)

次のページも参照してください。

- [外部プロシージャの構成に関する詳細は、7-63 ページの「外部プロシージャを使用する場合の Net8 の構成」](#)
- [Java オプションと IIOP クライアントの構成に関する詳細は、7-48 ページの「Java オプション接続の構成」](#)

LISTENER.ORA ファイル変更後のリスナーの再起動

LISTENER.ORA の変更後、LSNRCTL ユーティリティを使用してリスナーを一度停止し、再起動する必要があります。

リスナーを停止するには、次のように実行します。

```
lsnrctl
lsnrctl> stop [listener_name]
```

リスナーを起動するには、次のように実行します。

```
lsnrctl
lsnrctl> start [listener_name]
```

listener_name は、LISTENER.ORA ファイルに定義されているリスナーの名前です。LISTENER という名前のデフォルトのリスナーを使用している場合は、リスナーを識別する必要はありません。

詳細情報の参照先

- 8-2 ページの「[Net8 コンポーネント起動の概要](#)」
- A-3 ページの「[リスナー制御ユーティリティ \(LSNRCTL\)](#)」

リスナー・プロトコル・アドレスの構成

ローカル・ネーミングまたは Oracle Names のために構成したリスナー・アドレスは、LISTENER.ORA にも構成しておく必要があります。「Typical」または「Minimal」インストールの実行中、TCP/IP が自動的に LISTENER.ORA に構成されます。これ以外のプロトコルは構成されません。別のプロトコルのリスナー・アドレスを構成するには、次の手順に従います。

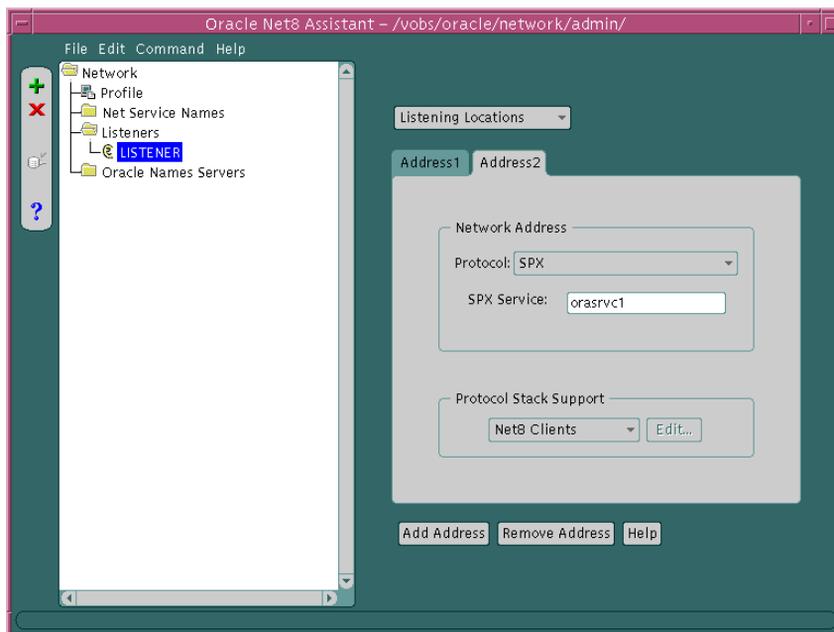
1. Oracle Net8 Assistant を起動します。
 - UNIX の場合、`$ORACLE_HOME/bin` で `netasst` を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」 -> 「Oracle - HOME_NAME」 -> 「Network Administration」 -> 「Net8 Assistant」を選択します。
2. 「Listeners」フォルダをダブルクリックします。
3. 目的のリスナーをクリックします。

注意： インストール中に LISTENER.ORA を構成済みでも、LSNRCTL 制御ユーティリティを使用してリスナーを一度も起動したことがないと、リスナーは「Listeners」フォルダに表示されません。その場合は Oracle Net8 Assistant を終了し、8-2 ページの「[Net8 コンポーネント起動の概要](#)」の説明に従ってリスナーを起動してから再度 Oracle Net8 Assistant を起動してください。

LISTENER.ORA をインストールしていないためにリスナーが存在しない場合は、次の作業を実行します。

- a. 「Edit」メニューの「Create」を選択するか、または「+」ボタンをクリックして、新規に LISTENER.ORA ファイルを作成します。
- b. 「Choose Listener Name」ダイアログ・ボックスに一意のリスナー名を入力します。
4. ドロップダウン・リスト・ボックスから「Listening Locations」を選択します。
5. 「Add Address」をクリックします。

6. 6-76 ページの「**プロトコル・アドレスの構成**」の説明に従って、プロトコルを選択し、要求されたプロトコル・アドレス情報を入力します。



7. データベース・サービスに接続する標準的なクライアントには「Net8 Client」を選択し、Java オプションに接続するクライアントには「IIOP Clients」を選択します。

標準的な Net8 クライアントを TCP/IP 上でリスニングするためにリスナーを構成する場合は、デフォルトのポート 1521 を選択する必要があります。このポートを選択しない場合は、`INITSID.ORA` ファイルに `LOCAL_LISTENER` パラメータを構成し、なんらかのネーミング・メソッドによってリスナー名を解決する必要があります。Net8 クライアントのために `LISTENER.ORA` にデフォルト以外のポート番号を構成する方法の詳細については、6-67 ページの「**デフォルト以外のポートを使用するリスナーの構成**」を参照してください。

「IIOP Clients」クライアントを選択した場合、プロトコルはポート 2481 上の TCP/IP またはポート 2482 上の SSL 付き TCP/IP のいずれかであることを確認してください。これらのポートを選択しない場合は、`INITSID.ORA` ファイルで `LOCAL_LISTENER` パラメータを構成するか、または `MTS_DISPATCHERS` パラメータの `LISTENER` 属性を構成し、ネーミング・メソッドによってリスナー名を解決する必要があります。IIOP クライアントのために `LISTENER.ORA` にデフォルト以外のポート番号を構成する方法の詳細は、7-57 ページの「**独自のリスナーの構成**」の手順に従ってください。

8. 別のプロトコルを構成するには、手順 5 ~ 7 を繰り返します。

9. 「File」メニューから「Save Network Configuration」を選択します。
10. 「File」メニューから「Exit」を選択し、Oracle Net8 Assistant アプリケーションを終了します。

たとえば、デフォルトの TCP/IP アドレスに加えて oraservc1 というサービス名に SPX アドレスを静的に構成した LISTENER.ORA ファイルは、次のようになります。

```
listener=
  (address=
    (protocol= tcp)
    (host= sales-pc)
    (port= 1521)
  )
  (address=
    (protocol= spx)
    (service=orasrvcl)
  )
```

デフォルト以外のポートを使用するリスナーの構成

Net8 クライアント用にデフォルトのポート 1521 を使用しない TCP/IP リスナー・アドレスを構成する場合は、INITSID.ORA ファイルに LOCAL_LISTENER パラメータを設定し、TNSNAMES.ORA ファイルまたは Oracle Names Server のネット・サービス名エントリを使用してこのパラメータを解決する必要があります。

LOCAL_LISTENER パラメータの構成

LOCAL_LISTENER パラメータは、次のように構成します。

```
local_listener=listener_name
```

ディスパッチャがデフォルト以外のリスナーにそれ自身の情報を登録するためには、LOCAL_LISTENER パラメータで指定したリスナー名を、サーバー上の TNSNAMES.ORA ファイルまたは Oracle Names Server などのネーミング・メソッドを使用して解決する必要があります。

このエントリに含めるのはリスナー・アドレスのみで、CONNECT_DATA 部分にあるサービス名情報は含めません。

TNSNAMES.ORA ファイルの構成

Oracle Net8 Assistant では、CONNECT_DATA 情報なしで TNSNAMES.ORA を構成できません。このため、オラクル社では手動で TNSNAMES.ORA を作成することをお薦めしています。

リスナー名が LISTENER1 で、リスニング終点到ポート 1421 の TCP/IP を使用し、ネーミング・メソッドとしてローカル・ネーミング・メソッドを選択した場合、TNSNAMES.ORA のエントリは次のようになります。

```
listener1.us.oracle.com=
(description=
  (address=
    (protocol= tcp)
    (host= sales-pc)
    (port= 1421)
  )
)
```

複数アドレスはサポートされますが、複数アドレスの connect-time failover とクライアント負荷のバランスはサポートされません。Oracle Connection Manager の SOURCE_ROUTE パラメータはサポートされます。

Oracle Names Server の構成

リスナーのエントリを作成するには、Oracle Net8 Assistant で Oracle Names Server を使用します。

1. Oracle Net8 Assistant を起動します。
 - UNIX の場合、\$ORACLE_HOME/bin で netasst を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」 -> 「Oracle - HOME_NAME」 -> 「Network Administration」 -> 「Net8 Assistant」を選択します。
2. Oracle Names Server フォルダを選択します。
3. 目的の Oracle Names Server をクリックします。
4. ドロップダウン・リスト・ボックスから「Manage Data」を選択します。
5. 「Advanced」タブを選択します。
6. 「Add」をクリックし、「Name」フィールドにリスナー名、「Type」フィールドに SMD レコード・タイプ、および「Value」フィールドには次のようにアドレスを入力します。

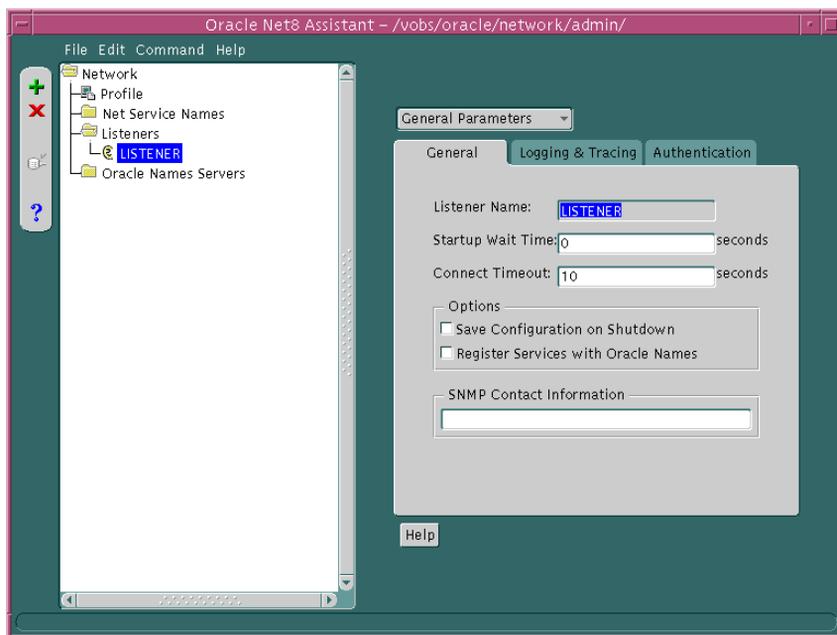
```
(address=(protocol= tcp)(host= sales-pc)(port= 2481))
```
7. 「File」メニューから「Save Network Configuration」を選択します。
8. 「File」メニューから「Exit」を選択し、Oracle Net8 Assistant を終了します。

LISTENER 以外のリスナー名の使用

デフォルトのリスナー名は、LISTENER です。リスナー名は、与えられたノードで一意的でなければなりません。リスナーには、任意の名前を付けることができます。

リスナーに新しい名前を付けるには、次の手順に従います。

1. Oracle Net8 Assistant を起動します。
 - UNIX の場合、\$ORACLE_HOME/bin で netasst を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」-> 「Oracle - HOME_NAME」-> 「Network Administration」-> 「Net8 Assistant」を選択します。
2. 「Listeners」フォルダをダブルクリックします。
3. 目的のリスナーをクリックします。
4. ドロップダウン・リスト・ボックスから「General Parameters」を選択します。



5. 「Listener Name」フィールドを新しいリスナー名に変更します。
6. 「File」メニューから「Save Network Configuration」を選択します。
7. 「File」メニューから「Exit」を選択し、Oracle Net8 Assistant アプリケーションを終了します。

大量の接続要求の処理

多数の接続要求を処理することが予想される場合は、プロセスに対してキューを指定できません。これにより、リスナーは多数の同時接続要求を動的に処理できません。

リスナーのキューサイズを指定するには、リスニング・アドレスの終端にある QUEUE_SIZE パラメータに値を入力します。

```
listener=  
  (address=  
    (protocol= tcp)  
    (host= sales-pc)  
    (port= 1521)  
    (queuesize=20)  
  )
```

注意： 現在は、TCP/IP および DECnet 上で動作するリスナーのキュー・サイズしか設定できません。デフォルトのキューサイズは、使用するシステムによって異なります。Solaris の場合、デフォルトのキューサイズは 5 に設定されます。また、Windows NT 4.0 Workstation のキューサイズは 5 で、Windows NT 4.0 Server のキューサイズは 50 です。

サービス情報によるリスナーの構成

クライアントからの Oracle8 リリース 8.0 や Oracle7 データベースへの接続要求をリスナーが受け入れるには、データベースに関する情報を使用して LISTENER.ORA ファイルを構成する必要があります。また、外部プロシージャや異種間サービスなどのサービス、および Oracle Enterprise Manager などの一部の管理ツールでは、静的な構成が必要です。

外部プロシージャと異種間サービスのためにリスナーを静的に構成する方法の詳細は、[第7章の「Net8 のオプション機能」](#)を参照してください。Oracle Enterprise Manager の構成は、『Oracle Enterprise Manager 構成ガイド』を参照してください。

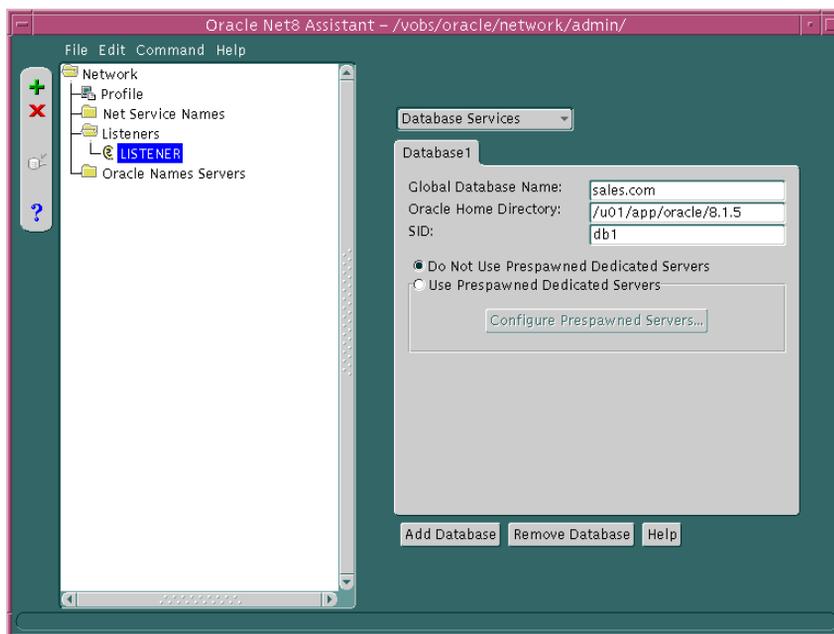
注意： connect-time failover をインプリメントすると、LISTENER.ORA ファイルで静的なサービス構成パラメータを使用できなくなります。ただし静的な構成は、Oracle8 リリース 8.0、または Oracle7 データベースと Oracle Enterprise Manager で必要です。connect-time failover の詳細は 2-29 ページの「[Connect-Time Failover](#)」を、Oracle Enterprise Manager 環境で Oracle Parallel Server に connect-time failover を設定する構成手順の追加情報は『Oracle8i Parallel Server セットアップおよび構成ガイド』を参照してください。

リスナーがサービスを提供しているデータベース・インスタンスに関する情報を構成するには、リスナー構成ファイル内で次の情報を指定する必要があります。

Oracle Net8 Assistant オプション	LISTENER.ORA パラメータ	説明
「System Identifier (SID)」	SID_NAME	システム識別子 (SID) とは、データベース・サーバーに付けられる Oracle システム ID のことをいう。これは、データベース初期化パラメータ・ファイル (INITSID.ORA) の INSTANCE_NAME パラメータで指定されます。 注意： このパラメータの値は、INSTANCE_NAME パラメータの値と一致する必要があります。
「Global Database Name」	GLOBAL_DBNAME	グローバル・データベース名とは、INITSID.ORA ファイルの SERVICE_NAMES パラメータによって指定されるデータベース名とドメイン名をいう。ネットワーク上でグローバル・データベース名を使用してデータベースを参照する場合は、リスナーにそのデータベースのグローバル・データベース名を指定する必要があります。 注意： このパラメータの値は、SERVICE_NAMES パラメータの値と一致する必要があります。
「Oracle Home Directory」	ORACLE_HOME	Oracle ホーム・ディレクトリによって、指定しているデータベースの Oracle ホーム上の場所を識別する。 注意： Windows NT では必要ありません。

リスナーを静的に構成するには、次の手順に従います。

1. Oracle Net8 Assistant を起動します。
 - UNIX の場合、`$ORACLE_HOME/bin` で `netasst` を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」 -> 「Oracle - HOME_NAME」 -> 「Network Administration」 -> 「Net8 Assistant」を選択します。
2. 「Listeners」フォルダをダブルクリックします。
3. 目的のリスナーをクリックします。
4. ドロップダウン・リスト・ボックスから「Database Services」を選択します。



5. 必要に応じて、各フィールドを変更します。
6. 「File」メニューから「Save Network Configuration」を選択します。
7. 「File」メニューから「Exit」を選択し、Oracle Net8 Assistant アプリケーションを終了します。

次に示すのは、SID_LIST_listener_name セクションが静的に構成されている LISTENER.ORA ファイルの一部を抜粋したものです。

```
sid_list_listener=(sid_list=
(sid_desc=
(global_dbname=sales.com)
(sid_name=db1)
(oracle_home=/u01/app/oracle/8.1.5)
)
```

事前に起動または生成される専用サーバーの構成

注意： 事前に生成される専用サーバーは、Windows NT では利用できません。

事前に生成される専用サーバーを作成するには、次の4つのパラメータを LISTENER.ORA ファイルの各 SID_DESC に追加します。

Oracle Net8 Assistant オプション	LISTENER.ORA パラメータ	説明
「Maximum Prespawnd Servers」	PRESPAWN_MAX	リスナーによって事前に生成される専用サーバーの最大数を指定する。この数値は、最小でも各プロトコル用のプール・サイズの合計と同じでなければなりません。事前に生成される専用サーバーが常に新しい接続に使用できるように、この値を大きな値に設定します。
「TCP/IP」 _Ⓜ 、「SPX」 _Ⓜ 「Named Pipes」 _Ⓜ 、「IPC」	PROTOCOL	リスナーによって事前に生成される専用サーバーの対象となるプロトコル。
「Number」	POOL_SIZE	リスナーによって指定のプロトコルに関してメンテナンスされる、未使用の、事前に生成される専用サーバーの数。0（ゼロ）より大きく、PRESPAWN_MAX 値以下の値を選択します。この値は、任意の時点で予測される平均接続数の近似値にしてください。

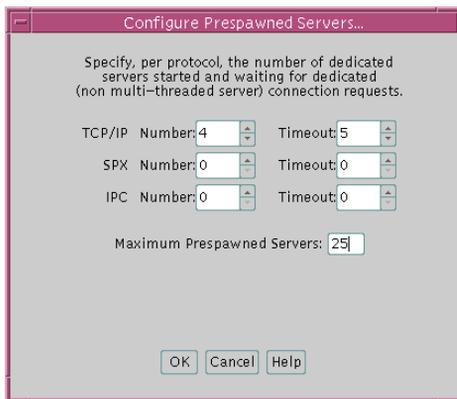
Oracle Net8 Assistant オプション	LISTENER.ORA パラメータ	説明
「Timeout」	TIMEOUT	非活動状態にある事前に生成される専用サーバー・プロセスが次の接続のために待機する時間（分単位）。値は、0（ゼロ）より大きい値でなければなりません。（値0では、非活動状態のシャドウ・プロセスが無限に続くため、マシンのリソースが無駄になります。）短いタイムアウト値を設定してください。タイムアウトは、事前に生成される専用サーバー・プロセスが接続と切断を実行した後でのみアクティブ化されます。つまり、初回の接続のために待機している事前に生成される専用サーバーでは、タイムアウトは発生しません。

注意： これらのパラメータは、Windows NT では利用できません。

事前に生成される専用サーバーを構成するには、次の手順に従います。

1. Oracle Net8 Assistant を起動します。
 - UNIX の場合、\$ORACLE_HOME/bin で `netasst` を実行する
 - Windows NT の場合、「スタート」ボタン->「プログラム」->「Oracle - HOME_NAME」->「Network Administration」->「Net8 Assistant」を選択する
2. 「Listeners」フォルダをダブルクリックします。
3. 目的のリスナーをクリックします。
4. ドロップダウン・リスト・ボックスから「Database Services」を選択します。
5. 「Use Prespawnd Dedicated Servers」を選択し、「Configure Prespawnd Servers」をクリックします。

6. 「Configure Prespawnd Servers」ダイアログ・ボックスが表示されます。



7. 「Configure Prespawnd Servers」ダイアログ・ボックスに必要な情報を入力します。
8. 「File」メニューから「Save Network Configuration」を選択します。
9. 「File」メニューから「Exit」を選択し、Oracle Net8 Assistant アプリケーションを終了します。

次に示すのは、事前に生成される専用サーバーが構成されている LISTENER.ORA ファイルの一部を抜粋したものです。

```
sid_list_listener=(sid_list=
(sid_desc=
(global_dbname=sales.com)
(sid_name=db1)
(oracle_home=/u01/app/oracle/product/8.1.5)
(prespawn_max=25)
(prespawn_list=
(prespawn_desc=
(protocol=tcp)
(pool_size=4)
(timeout=5)
)
)
)
)
```

プロトコル・アドレスの構成

接続が作成されると、要求の受信者（リスナー、Oracle Names Server、Oracle Connection Manager など）とクライアントは同じプロトコルを使用して構成されます。このアドレスの目的は、ネットワーク・オブジェクトのネットワーク・アドレスを識別することです。クライアントは、このアドレスを使用して特定のネットワーク・オブジェクトがある場所に接続要求を送信します。一方、受信者はこのアドレス上でリスニングし、クライアントの情報が自分の持っているアドレス情報に一致するかどうかに基づいて接続を許可します。クライアントと接続受信者にとって、同じプロトコルをインストールするのと同様に、同じアドレスを構成することは重要です。

この項では、次のトピックを取り上げます。

- [アドレスの構成要素](#)
- [プロトコルのパラメータ](#)

アドレスの構成要素

プロトコル・アドレスは、次の構成要素内部に格納されます。

ADDRESS

目的： プロトコル・アドレスを定義します。このパラメータは、ADDRESS_LIST または DESCRIPTION の中に埋め込むことができます。DESCRIPTION は、通常 TNSNAMES.ORA または LISTENER.ORA ファイルでのみ使用されます。

追加情報： 6-78 ページの「[プロトコルのパラメータ](#)」に説明されているように、各プロトコルにはアドレスに必要な特定のパラメータがあります。

例：

```
(address=
(protocol=tcp)
(host=sales-pc)
(port=1521)
)
```

ADDRESS_LIST

目的: 共通の特性セットを共有するアドレス・リストを定義します。

このパラメータは、DESCRIPTION または DESCRIPTION_LIST の中に埋め込むことができます。DESCRIPTION または DESCRIPTION_LIST は、通常 TNSNAMES.ORA または LISTENER.ORA ファイルでのみ使用されます。

DESCRIPTION と DESCRIPTION_LIST の説明は、C-38 ページの「[DESCRIPTIONS および DESCRIPTION_LIST](#)」を参照してください。

注意: TNSNAMES.ORA または LISTENER.ORA ファイルにリストが 1 つしかない場合、ADDRESS_LIST は必要ありません。

例:

```
(address_list=
  (address=
    (protocol=tcp)
    (host=sales-pc)
    (port=1521)
  )
  (address=
    (protocol=tcp)
    (host=hr-pc)
    (port=1521)
  )
)
(address_list=
  (address=
    (protocol=tcp)
    (host=finance-pc)
    (port=1521)
  )
  (address=
    (protocol=tcp)
    (host=sales-pc)
    (port=1521)
  )
)
)
```

プロトコルのパラメータ

リスナー、Oracle Names Server および Oracle Connection Manager は、プロトコル・アドレスによって識別されます。次の表は、サポートしている Oracle プロトコルで使用されるパラメータについて説明しています。

プロトコル	パラメータ	説明
すべて	PROTOCOL	TNS ベースのアプリケーションが存在するネットワークのタイプを示す。
bequeath ¹	PROGRAM	Oracle8i の実行可能ファイルを識別する。
bequeath	ARGV0	サービス名を識別する。
bequeath	ARGS	接続のソース（ローカル・クライアント）を識別する。
IPC	KEY	サーバーの識別方法を示す。オラクル社では、サービス名の使用をお勧めしています。
Named Pipes	SERVER	使用している Oracle8i サーバー・コンピュータの名前を示す。
Named Pipes	PIPE	サーバーへの接続に使用するパイプ名を示す（Named Pipes でサーバーに指定したものと同一 PIPE キーワード）。これには任意の名前を付けることができます。
LU6.2	LU_NAME	Oracle8i サーバーを識別する。完全に修飾された名前にする必要があります。
LU6.2	LLU または LOCAL_LU	ローカルの LU 別名を識別する。このパラメータは、LLU_NAME と一緒に使用できません。
LU6.2	LLU_NAME または LOCAL_LU_NAME	ローカルの LU 名を示す。完全に修飾された名前にする必要があります。このパラメータは、LLU と一緒に使用できません。
LU6.2	MODE または MDN	LU6.2 ネットワークのログ・モード・エントリを識別する。この値は、通常 ORAPLU62 です。
LU6.2	PLU または PARTNER_LU_NAME	Oracle8i サーバーを識別する。完全に修飾された名前にする必要があります。このパラメータは、LLU と一緒に使用できません。
LU6.2	PLU_LA または PARTNER_LU_LOCAL_ALIAS	Oracle7 Server のパートナ LU 別名を識別する。このパラメータは、PLU と一緒に使用できません。

プロトコル	パラメータ	説明
LU6.2	TP_NAME または TPN	ホスト・マシンのトランザクション・プログラム名を識別する。このパラメータは、MVS ホストへの接続では必要ありません。
SPX	SERVICE	ネットワーク上にある TNS ベースのアプリケーションの名前を定義する（サーバーとクライアントには必須）。ネットワーク上に存在する TNS ベースのアプリケーションのサービス名については、ネットワーク管理者に問い合わせてください。
TCP/IP と SSL 付き TCP/IP	HOST	サーバーを識別する。

プロトコル	パラメータ	説明
TCP/IP と SSL 付き TCP/IP	PORT	<p>リスニング・ポート番号を識別する。</p> <ul style="list-style-type: none"> ■ 1521 は、クライアントがデータベース・サービスに接続するためのデフォルトのリスニング・ポート。将来のリリースでは、このポート番号が公式に登録されたポート番号である TCP/IP の 2483 と SSL 付き TCP/IP の 2484 に変更される可能性があります。 ■ 2481 は、クライアントから Java オプションへの TCP/IP 接続でお薦めする、公式に登録されたリスニング・ポート。 ■ 2482 は、クライアントから Java オプションへの SSL 付き TCP/IP を使用した接続にお薦めする、公式に登録されたリスニング・ポート。 ■ 1575 は、TCP/IP または SSL 付き TCP/IP を使用したデフォルトの、公式に登録された Oracle Names Server のリスニング・ポート。 ■ 1630 は、TCP/IP を使用したデフォルトの、公式に登録された Oracle Connection Manager CMGW ゲートウェイ・プロセスのリスニング・ポート。 ■ 1830 は、TCP/IP を使用したデフォルトの、公式に登録された Oracle Connection Manager CMADMIN 管理プロセスのリスニング・ポート。 <p>注意: オラクル社では 1 ~ 65535 までのポート番号を許可していますが、1024 より小さいポート番号は多くのオペレーティング・システムで特権的なプロセスのために使用が制限されています。</p>

¹ bequeath は、TNSNAMES.ORA にあるが LISTENER.ORA にない構成情報を必要とします。

Net8 のオプション機能

この章では、オプション機能について説明します。

内容は次のとおりです。

- ネット・サービス名の変更
- プロファイルの構成
- 接続集中化機能の使用
- 複数プロトコル・サポートの使用
- Net8 アクセス制御の使用
- Oracle Connection Manager 機能を使用する場合のクライアントの構成
- 接続プーリングの使用
- Java オプション接続の構成
- Oracle 以外のデータベース・サービスの構成

ネット・サービス名の変更

この項の内容は、次のとおりです。

- ネット・サービス名の属性の変更
- ネット・サービス名の改名
- ネット・サービス名へのプロトコル・アドレスの追加
- ネット・サービス名からのプロトコル・アドレスの削除
- ネット・サービス名の削除
- ネット・サービス名の接続属性の詳細構成
- 複数アドレス・オプションの構成

これらの変更内容は、UNIX の場合は `$ORACLE_HOME/network/admin`、Windows の場合は `ORACLE_HOME¥network¥admin` に保存されます。

ネット・サービス名の属性の変更

名前、サービス名、SID またはプロトコル・アドレスなどのネット・サービス名の属性を変更するには、次の手順に従います。

1. Net8 Configuration Assistant を起動します。
 - UNIX の場合、`$ORACLE_HOME/bin` の下にある `netca` を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」 -> 「Oracle - HOME_NAME」 -> 「Network Administration」 -> 「Net8 Configuration Assistant」を選択します。
2. 「Local Net8 Service Name configuration」を選択し、「Next」をクリックします。
3. 「Modify」をクリックします。
4. ウィザードのプロンプトやオンライン・ヘルプに従って、属性を変更します。
5. 「Finish」をクリックして構成を保存し、Net8 Configuration Assistant アプリケーションを終了します。

ネット・サービス名の改名

ネット・サービス名を改名するには、次の手順に従います。

1. Net8 Configuration Assistant を起動します。
 - UNIX の場合、`$ORACLE_HOME/bin` の下にある `netca` を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」 -> 「Oracle - *HOME_NAME*」 -> 「Network Administration」 -> 「Net8 Configuration Assistant」を選択します。
2. 「Local Net8 Service Name configuration」を選択し、「Next」をクリックします。
3. 「Rename」をクリックします。
4. ウィザードのプロンプトやオンライン・ヘルプに従って、属性を変更します。
5. 「Finish」をクリックして構成を保存し、Net8 Configuration Assistant アプリケーションを終了します。

ネット・サービス名へのプロトコル・アドレスの追加

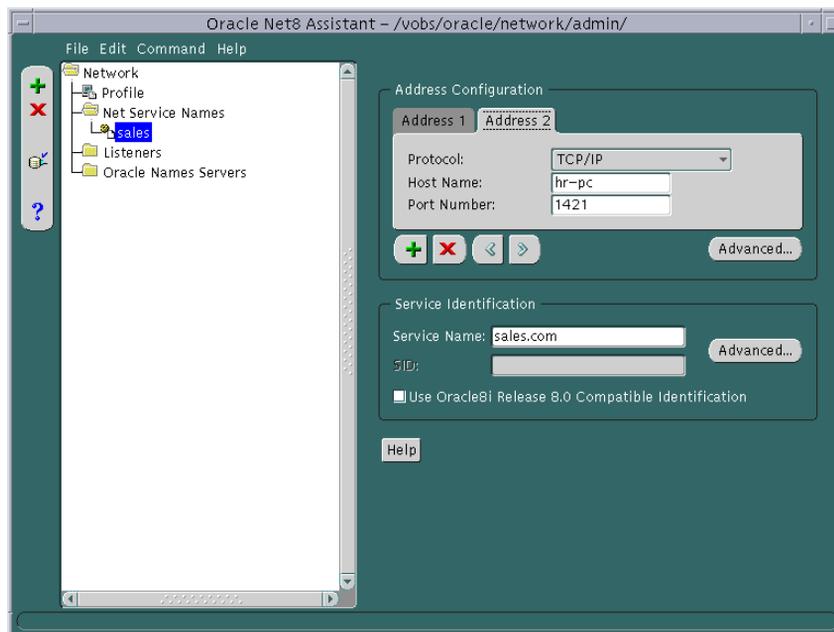
それぞれの接続記述子には、複数のリスナー・アドレスを設定できます。

```
net_service_name=  
(description=  
  (address= (protocol_address_information))  
  (address= (protocol_address_information))  
  (address= (protocol_address_information))  
  (connect_data=  
    (service_name=service_name)  
  )  
)
```

プロトコル・アドレスをネット・サービス名に追加するには、次の手順に従います。

1. Oracle Net8 Assistant を起動します。
 - UNIX の場合、`$ORACLE_HOME/bin` の下にある `netasst` を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」 -> 「Oracle - *HOME_NAME*」 -> 「Network Administration」 -> 「Net8 Assistant」を選択します。
2. ディレクトリ・ツリーで、ネット・サービス名フォルダをダブルクリックします。
3. ネット・サービス名を選択します。
4. 「Address Configuration」グループ・ボックスで「+」をクリックします。

新規のアドレス・タブが表示されます。

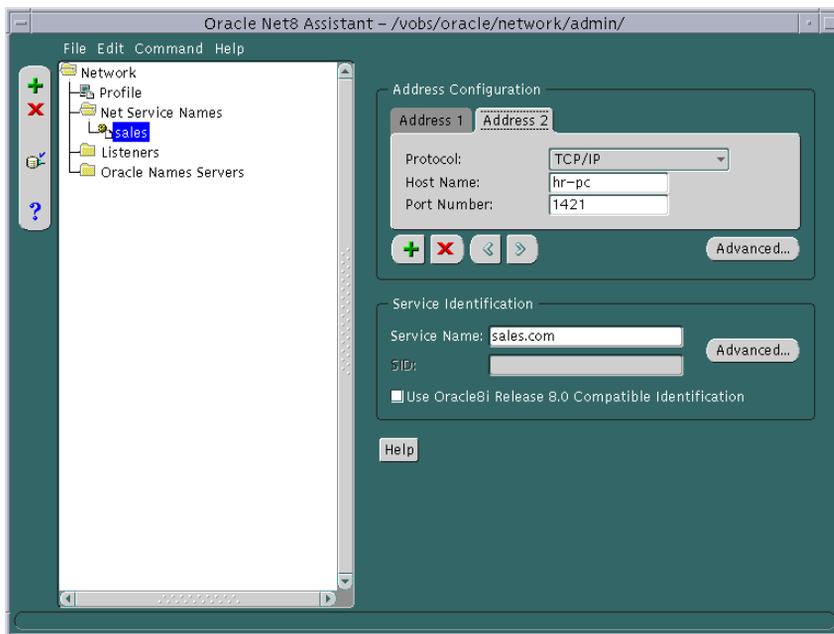


5. プロトコルを選択し、必要なアドレス情報を入力します。プロトコル・アドレス・パラメータの詳細は、6-76 ページの「[プロトコル・アドレスの構成](#)」を参照してください。
6. 「<」ボタンと「>」ボタンを使用して、リスト内の適切な場所にアドレスを配置します。複数アドレス・オプションが構成されていない場合、リスト先頭のアドレスに対して最初に通信が行われます。複数アドレス・オプションの構成方法は、7-11 ページの「[複数アドレス・オプションの構成](#)」を参照してください。
7. 「File」メニューの「Save Network Configuration」を選択します。
TNSNAMES.ORA ファイルが再作成されます。
8. 「File」メニューの「Exit」を選択します。
Oracle Net8 Assistant アプリケーションが終了します。

ネット・サービス名からのプロトコル・アドレスの削除

TNSNAMES.ORA ファイルでネット・サービス名からプロトコル・アドレスを削除するには、次の手順に従います。

1. Oracle Net8 Assistant を起動します。
 - UNIX の場合、\$ORACLE_HOME/bin の下にある netasst を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」-> 「Oracle - HOME_NAME」-> 「Network Administration」-> 「Net8 Assistant」を選択します。
2. ディレクトリ・ツリーで、ネット・サービス名フォルダをダブルクリックします。
3. ネット・サービス名を選択します。
4. 「Address Configuration」グループ・ボックスで「x」をクリックします。



5. 「File」メニューの「Save Network Configuration」を選択します。

TNSNAMES.ORA ファイルが再作成されます。
6. 「File」メニューの「Exit」を選択します。

Oracle Net8 Assistant アプリケーションが終了します。

ネット・サービス名の削除

TNSNAMES.ORA ファイルからネット・サービス名を削除するには、次の手順に従います。

1. Net8 Configuration Assistant を起動します。
 - UNIX の場合、`$ORACLE_HOME/bin` の下にある `netca` を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」 -> 「Oracle - HOME_NAME」 -> 「Network Administration」 -> 「Net8 Configuration Assistant」を選択します。
2. 「Local Net8 Service Name configuration」を選択し、「Next」をクリックします。
3. 「Delete」をクリックします。
4. ウィザードのプロンプトやオンライン・ヘルプに従って、ネット・サービス名を削除します。
5. 「Finish」をクリックして構成を保存し、Net8 Configuration Assistant アプリケーションを終了します。

「Finish」をクリックすると、ネット・サービス名が削除され、TNSNAMES.ORA ファイルが再作成されます。

ネット・サービス名の接続属性の詳細構成

TNSNAMES.ORA の接続記述子の CONNECT_DATA セクションには、オプションとして次の機能を設定できます。

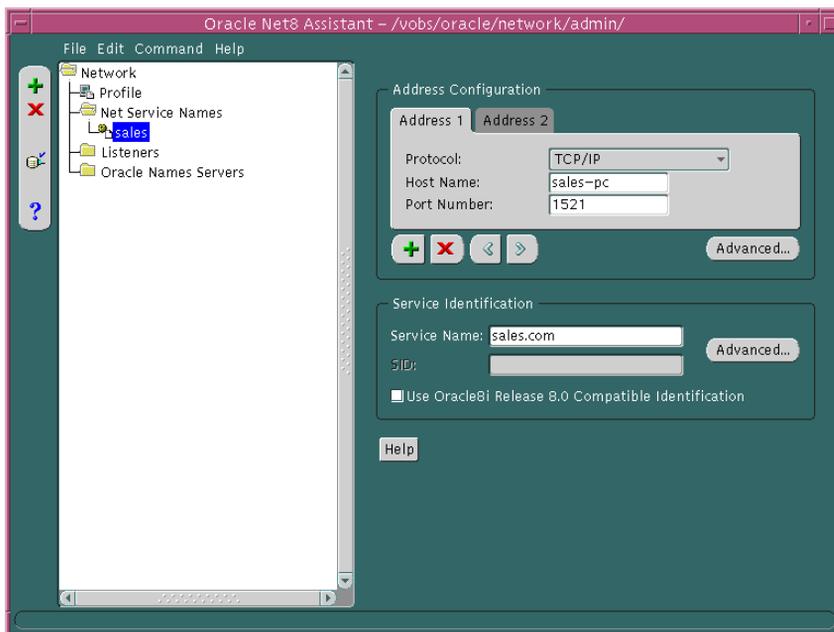
Oracle Net8 Assistant オプション	TNSNAMES.ORA パラメータ	説明
「Instance Name」	INSTANCE_NAME	<p>アクセスするデータベース・インスタンスを識別する。</p> <p>インスタンス名は、INITSID.ORA ファイル内の INSTANCE_NAME パラメータから取得できます。</p> <p>注意: 「Use Oracle8 Release 8.0 Compatible Identification」がオンの場合は使用できません。</p> <p>追加情報: 6-2 ページの「サービス名とインスタンス名の設定」を参照してください。</p>
「Handler Name」	HANDLER_NAME	<p>MTS ディスパッチャなど、アクセスするサービス・ハンドラを識別する。</p> <p>注意: 「Use Oracle8 Release 8.0 Compatible Identification」がオンの場合は使用できません。</p> <p>注意: リリース 8.1.5 では使用できません。</p>
「Session Data Unit (SDU) Size」	SDU	<p>ネットワークを介して送信されるデータ・パケットの転送レートを最適化する場合、セッション・データ・ユニット (SDU) サイズを指定すると、パケットがネットワークを介して送信される状態でパフォーマンスを変更できる。</p> <p>追加情報: 3-13 ページの「セッション・データ・ユニット (SDU) サイズの調整によるデータ転送の最適化」を参照してください。</p>
「Use a Dedicated Server」	SERVER	<p>クライアント接続でマルチスレッド・サーバーに接続せずに専用サーバーを使用する場合は、このオプションをオンにする。</p>
「Use for Heterogeneous Services」	HS	<p>Oracle8i サーバーで Oracle 以外のシステムにアクセスする場合は、このオプションをオンにする。</p> <p>注意: 「Use Oracle8 Release 8.0 Compatible Identification」がオンの場合は使用できません。</p> <p>追加情報: 7-65 ページの「Oracle 異機種間サービスを使用する場合の Net8 の構成」を参照してください。</p>

Oracle Net8 Assistant オプション	TNSNAMES.ORA パラメータ	説明
「Oracle Rdb Settings」		
「Oracle RDB Database」	RDB_DATABASE	Oracle RDB データベースの名前を指定する。 追加情報 : 7-69 ページの「Oracle RDB データベースを使用する場合の Net8 の構成」を参照してください。
「Type of Service」	TYPE_OF_SERVICE	Oracle RDB データベースで使用するサービスのタイプを指定する。 追加情報 : 7-69 ページの「Oracle RDB データベースを使用する場合の Net8 の構成」を参照してください。
「Global Database Name」	GLOBAL_NAME	Oracle RDB データベースを識別する。 追加情報 : 7-69 ページの「Oracle RDB データベースを使用する場合の Net8 の構成」を参照してください。

これらの詳細オプションを構成するには、次の手順に従います。

- Oracle Net8 Assistant を起動します。
 - UNIX の場合、\$ORACLE_HOME/bin の下にある netasst を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」-> 「Oracle - HOME_NAME」-> 「Network Administration」-> 「Net8 Assistant」を選択します。
- ディレクトリ・ツリーで、ネット・サービス名フォルダをダブルクリックします。

3. ネット・サービス名を選択します。



4. 「Service Identification」グループ・ボックスの「Advanced」をクリックします。

「Advanced Service Options」ダイアログ・ボックスが表示されます。



5. 必要に応じてフィールドの入力やオプションの選択を行い、「OK」をクリックします。
6. 「File」メニューの「Save Network Configuration」を選択します。
TNSNAMES.ORA ファイルが再作成されます。
7. 「File」メニューの「Exit」を選択します。
Oracle Net8 Assistant アプリケーションが終了します。

複数アドレス・オプションの構成

1つのネット・サービス名に複数のアドレスが構成されている場合、次の機能を構成できます。

- ソース・ルート (SOURCE_ROUTE)
- クライアント負荷のバランス (LOAD_BALANCE)
- Connect-Time Failover (FAILOVER)

オプション	説明
ソース・ルート (SOURCE_ROUTE)	<p>オンの場合、宛先に到達するまで表示順にアドレスを使用するように指定する。接続先に到達するまでに複数のアドレス終点を通る必要がある場合は、このパラメータが必要です。この機能は通常、Oracle Connection Manager の機能を使用可能にするために使用しません。</p> <p>追加情報: SOURCE_ROUTE パラメータの使用の詳細は、次の項目を参照してください。</p> <ul style="list-style-type: none"> ■ 7-30 ページの「接続集中化機能の使用」 ■ 7-34 ページの「複数プロトコル・サポートの使用」 ■ 7-37 ページの「Net8 アクセス制御の使用」
Connect-Time Failover (FAILOVER)	<p>オンの場合、一定の時間内に最初リスナーと接続できなければ、別のリスナーにフェイルオーバーするように指定する。接続を試行する回数は、リスト内のアドレス数で決まります。オフの場合、接続を試行するアドレスは1つのみです。</p> <p>複数のアドレス・リスト (ADDRESS_LIST) \ 接続記述子 (DESCRIPTION) および複数の接続記述子 (DESCRIPTION_LIST) がある場合、connect-time failover はデフォルトでオンになっています。</p>
クライアント負荷のバラン ス (LOAD_BALANCE)	<p>オンの場合、各リスナーへの負荷のバランスをとるために、リスナー・アドレスのリストから順不同に進行するように指定する。オフの場合、接続が成功するまでアドレスを順に試行するように指定します。</p> <p>複数の接続記述子 (DESCRIPTION_LIST) がある場合、クライアント負荷のバランスはデフォルトでオンになっています。</p>

クライアント負荷のバランスと connect-time failover を有効にすると、次のような利点があります。

Oracle Net8 Assistant オプション	TNSNAMES.ORA パラメータ
「Try each address, in order, until one succeeds」	FAILOVER=ON (リリース 8.1 クライアントの場合) SOURCE_ROUTE=OFF (リリース 8.0 以前のクライアントの場合)
「Try each address, randomly, until one succeeds」 注意: 「Use Options Compatible with Net8 8.0 Clients」がオンの場合は使用できません。	LOAD_BALANCE=ON FAILOVER=ON
「Try one address, selected at random」 注意: 「Use Options Compatible with Net8 8.0 Clients」がオンの場合は使用できません。	LOAD_BALANCE=ON
「Use each address in order until destination reached」	SOURCE_ROUTE=ON
「Use only the first address」 注意: 「Use Options Compatible with Net8 8.0 Clients」がオンの場合は使用できません。	LOAD_BALANCE=OFF FAILOVER=OFF SOURCE_ROUTE=OFF

注意: connect-time failover をインプリメントすると、LISTENER.ORA ファイル内の静的サービス構成パラメータは使用できません。ただし、Oracle8 リリース 8.0 や Oracle7 データベースおよび Oracle Enterprise Manager には静的構成が必要です。リスナーの静的構成方法の詳細は、6-71 ページの「サービス情報によるリスナーの構成」を参照してください。

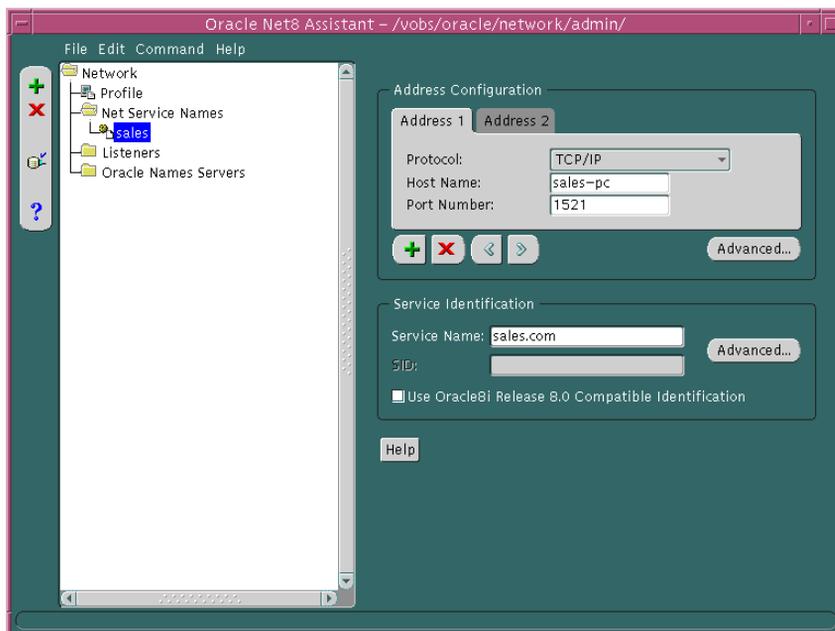
Oracle Enterprise Manager 環境で Oracle Parallel Server の connect-time failover を設定する方法は、『Oracle8i Parallel Server セットアップおよび構成ガイド』を参照してください。

注意: connect-time failover とクライアント負荷のバランスのオプションとパラメータを利用できるのは、リリース 8.1 構成のみです。ソース・ルーティング(「Use each address in order until destination reached」オプション)は、リリース 8.1 より前の構成とリリース 8.1 構成の両方で利用できます。

注意： クライアント負荷のバランスや connect-time failover をソース・ルーティングと併用することはできません。connect-time failover とクライアント負荷のバランスではリストから1つのアドレスを選択しますが、ソース・ルーティングではリスト内の各アドレスに順に接続します。

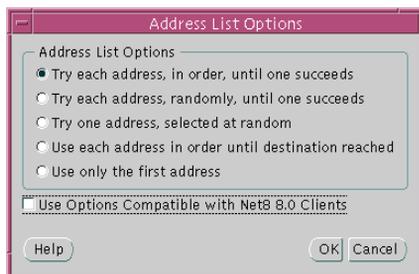
クライアント負荷のバランスや connect-time failover のオンとオフを切り替えるには、Oracle Net8 Assistant を使用します。

1. Oracle Net8 Assistant を起動します。
 - UNIX の場合、\$ORACLE_HOME/bin の下にある netasst を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」-> 「Oracle - HOME_NAME」-> 「Network Administration」-> 「Net8 Assistant」を選択します。
2. ディレクトリ・ツリーで、ネット・サービス名フォルダをダブルクリックします。
3. ネット・サービス名を選択します。



4. 「Address Configuration」グループ・ボックスで「Advanced」をクリックします。

「Address List Options」ダイアログ・ボックスが表示されます。



5. アドレス・リスト・オプションのいずれかを選択し、「OK」をクリックします。
6. 「File」メニューの「Save Network Configuration」を選択します。
TNSNAMES.ORA ファイルが再作成されます。
7. 「File」メニューの「Exit」を選択します。
Oracle Net8 Assistant アプリケーションが終了します。

次の例では、「Try each address, randomly, until one succeeds」オプションでクライアント負荷のバランスが指定された場合の TNSNAMES.ORA です。

```
sales=
  (description=
    (load_balance=on)
    (address=(protocol=tcp)(host=sales-pc)(port=1521)
    )
    (address=(protocol=tcp)(host=hr-pc)(port=1521)
    )
    (connect_data=
      (service_name=sales.com))
  )
```

次の例は、「Try each address, in order, until one succeeds」オプションで connect-time failover が指定された場合の TNSNAMES.ORA です。

```
sales=
  (description=
    (address_list=
      (load_balance=off)
      (failover=on)
      (address=(protocol=tcp)(host=hr-pc)(port=1521)
      )
      (address=(protocol=tcp)(host=finance-pc)(port=1521)
      )
    )
    (connect_data=(service_name=sales.com))
  )
```

プロファイルの構成

クライアントのプロファイルは、機能を設定してネットワーク上でサービスを確立し、維持するために Net8 がどのような役割を果たすかを定義します。プロファイルは、SQLNET.ORA という構成ファイルに格納され、このファイルによりインプリメントされます。このファイルは、UNIX の場合は \$ORACLE_HOME/network/admin、Windows の場合は ORACLE_HOME¥network¥admin にあります。プロファイルで構成される項目は次のとおりです。

- ネーミング・メソッドの優先順位
- トレース機能およびロギング機能
- 特定のプロセスを通じての接続要求
- セキュリティ機能

この項では、次のプロファイル機能について説明します。

- [ドメインの構成](#)
- [ネット・サービス名のネーミング・メソッド解決の優先順位](#)
- [デフォルトの Oracle Names Server の構成](#)
- [プロファイルの詳細情報の構成](#)
- [接続要求のルーティング](#)
- [Oracle Advanced Security の構成](#)

プロファイルに設定できる全パラメータを網羅したリストは、C-7 ページの「[プロファイル・パラメータ \(SQLNET.ORA\)](#)」を参照してください。

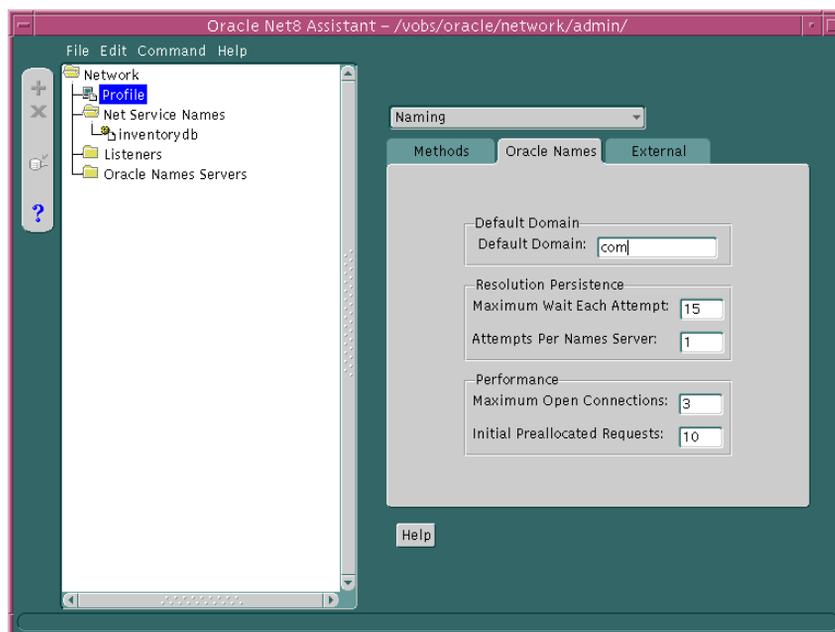
ドメインの構成

SQLNET.ORA の NAMES.DEFAULT_DOMAIN パラメータには、クライアントが最も頻繁に名前を要求するドメインを定義します。このパラメータを設定すると、接続文字列中のネット・サービス名にドメイン名が自動的に付けられます。このパラメータを Oracle Net8 Assistant で設定すると、ドメインを指定せずに作成されたネット・サービス名にそのドメインが自動的に付けられます。たとえば、NAMES.DEFAULT_DOMAIN=COM が設定されている場合、接続文字列 SCOTT/TIGER@ SALES は、TNSNAMES.ORA では SALES.COM として検索されます。接続文字列にドメイン拡張子が含まれる場合 (SCOTT/TIGER@SALES.COM など) ドメインは付加されません。

ドメインを指定するには、Oracle Net8 Assistant を使用します。

1. Oracle Net8 Assistant を起動します。
 - UNIX の場合、\$ORACLE_HOME/bin の下にある `netasst` を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」 -> 「Oracle - HOME_NAME」 -> 「Network Administration」 -> 「Net8 Assistant」を選択します。
2. 「Profile」アイコンをクリックします。
3. ドロップダウン・リスト・ボックスから「Naming」を選択します。

4. 「Oracle Names」タブをクリックします。



5. 「Default Domain」フィールドにドメインを入力します。
6. 「File」メニューの「Save Network Configuration」を選択します。
7. 「File」メニューの「Exit」を選択して Oracle Net8 Assistant アプリケーションを終了します。

SQLNET.ORA には次のようなエントリが追加されます。

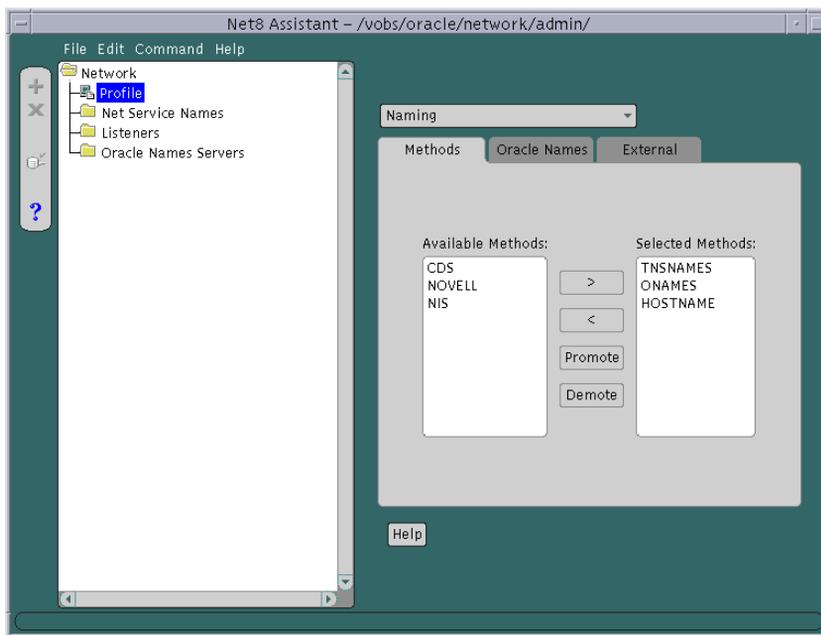
```
names.default_domain=com
```

ネット・サービス名のネーミング・メソッド解決の優先順位

ネーミング・メソッドを構成した後は、第6章の「ネーミング・メソッドとリスナーの構成」に説明されているように、優先順位を設定する必要があります。ネット・サービス名の解決には、まずリストの最初のネーミング・メソッドが使用されます。この方法でネット・サービス名が解決できない場合、リストの2番目の方法が使用されます。

ネーミング・メソッドの順位を指定するには、Oracle Net8 Assistant を使用します。

1. Oracle Net8 Assistant を起動します。
 - UNIX の場合、\$ORACLE_HOME/bin の下にある netasst を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」-> 「Oracle - HOME_NAME」-> 「Network Administration」-> 「Net8 Assistant」を選択します。
2. 「Profile」アイコンをクリックします。
3. ドロップダウン・リスト・ボックスから「Naming」を選択します。
4. 「Methods」タブをクリックします。



5. 「Available Methods」リストから目的のネーミング・メソッドを選択します。
6. ネット・サービス名の解決時に Net8 でどの方法を最初に使用するかを検討し、ネーミング・メソッドを並べ替えます。「Selected Methods」リスト内でネーミング・メソッド

を選択し、「Promote」ボタンや「Demote」ボタンを使用して選択項目をリスト内で移動します。

7. 「File」メニューの「Save Network Configuration」を選択します。
8. 「File」メニューの「Exit」を選択して Oracle Net8 Assistant アプリケーションを終了します。

SQLNET.ORA には次のようなエントリが追加されます。

```
names.directory_path=(onames, tnsnames)
```

使用可能なネーミング・メソッドは次のとおりです。

ネーミング・メソッド	説明
ONAMES	Oracle Names ネット・サービス名は Oracle Names Server により集中的に解決されます。
TNSNAMES	ローカル・ネーミング ネット・サービス名は、クライアントに配置された TNSNAMES.ORA ファイルにより解決されます。
HOSTNAME	ホスト・ネーミング ネット・サービス名は、ホストネーミング・メソッドにより解決されます。ホスト・ネーミングを使用して名前を解決するには、一定の条件を満たす必要があります。
CDS	Cell ディレクトリ・サービス このネーミング・メソッドは、Oracle Advanced Security とともに使用できます。 追加情報： 『Oracle8i Advanced Security 管理者ガイド』を参照してください。
NDS	NetWare ディレクトリ・サービス (NDS)
NIS	ネットワーク情報サービス (NIS)

デフォルトの Oracle Names Server の構成

優先 Oracle Names Server (preferred Oracle Names Server) は、名前解決のためにクライアントでデフォルトとして使用される Oracle Names Server です。通常は、クライアントと物理的に最も近いが、最も安価なネットワーク・リンクで利用できる Oracle Names Server です。とりわけ重要なのは、最も頻繁に使用するサービスのネット・サービス名をデフォルトの Oracle Names Server で解決できることです。

Oracle Names をネーミング・サービスとして使用する場合 (Oracle Names Server をローカル・マシンで実行する必要はありません)、デフォルトの Oracle Names Server の名前とアドレスを指定して、それ以外に利用できる Oracle Names Server のアドレスに優先させることができます。Net8 では、要求が受信されるまでデフォルトの Names Server に対して接続要求を発信します。

注意： デフォルトの Oracle Names Server は、Oracle Net8 Assistant の「Discover Oracle Names Servers」オプションや NAMESCTL 制御ユーティリティの REORDER_NS コマンドを使用するかわりに、ネットワーク上で何が利用できるかについて Oracle Names Server のリストを作成し、応答時間の早い順に順位を付ける新機能です。デフォルトの Oracle Names Server は、検出プロセスで見つかったその他の Oracle Names Server より優先されます。最初にネーム・サーバーを検出し、ネーム・サーバー・リストを生成した後は、デフォルトの Oracle Names Server を削除してもかまいません。

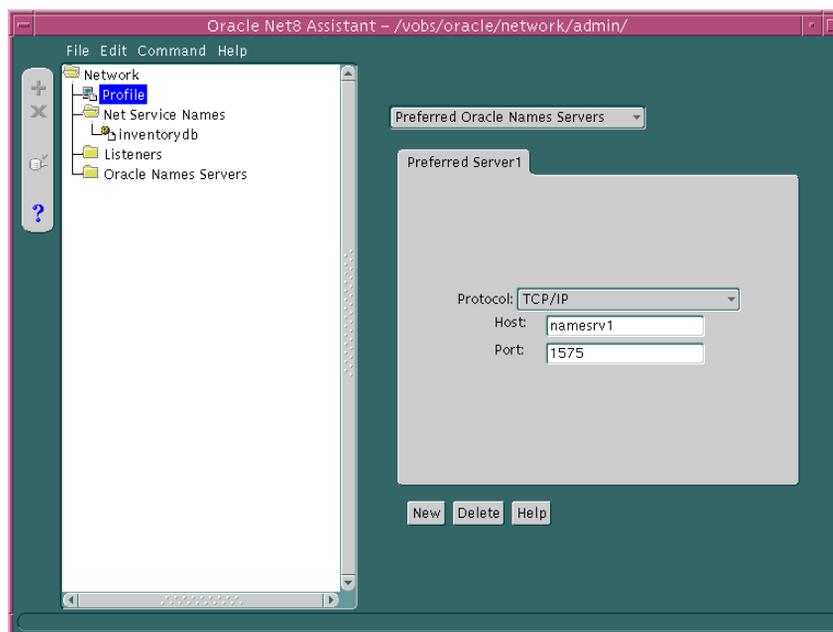
Oracle Names Server リストの作成方法は、2-42 ページの「[Oracle Names のバージョンによる違い](#)」を参照してください。

SQLNET.ORA の NAMES.PREFERRED_SERVERS パラメータには、デフォルトの Oracle Names Server を指定します。デフォルトの Oracle Names Server を構成する場合、NAMES.ORA に構成した Oracle Names リスニング・プロトコル・アドレスと SQLNET.ORA の NAMES.PREFERRED_SERVERS に構成するアドレスは同じでなければなりません。

デフォルトの Oracle Names Server を指定するには、次の手順に従います。

1. Oracle Net8 Assistant を起動します。
 - UNIX の場合、\$ORACLE_HOME/bin の下にある `netasst` を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」 -> 「Oracle - HOME_NAME」 -> 「Network Administration」 -> 「Net8 Assistant」を選択します。
2. 「Profile」アイコンをクリックします。
3. ドロップダウン・リスト・ボックスから「Preferred Oracle Names Servers」を選択します。

4. 「New」をクリックします。



5. 6-76 ページの「**プロトコル・アドレスの構成**」に説明されているように、プロトコルを選択し、そのアドレスで構成する Oracle Names Server に必要なプロトコル・アドレス情報を入力します。
6. 「File」メニューの「Save Network Configuration」を選択します。
7. 「File」メニューの「Exit」を選択して Oracle Net8 Assistant アプリケーションを終了します。

SQLNET.ORA には次のようなエントリが追加されます。

```
names.preferred_servers=  
  (address= (protocol=tcp) (host=namesrv1)(port=1575)
```

プロファイルの詳細情報の構成

プロファイル構成ファイル SQLNET.ORA では、次の詳細機能を構成できます。

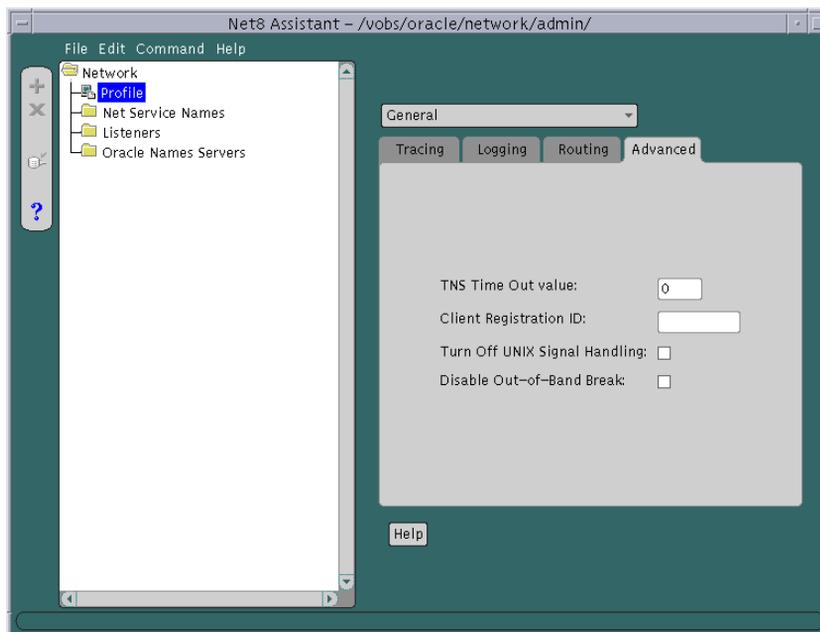
Oracle Net8 Assistant オプション	SQLNET.ORA パラメータ	説明
「TNS Time Out Value」	SQLNET.EXPIRE_TIME	<p>このオプションを使用できるのはサーバーのみ。</p> <p>Net8 では、クライアント / サーバーの接続がまだアクティブであるかどうかを確認するために、プローブを定期的送信します。これは、クライアントの異常終了により、接続が無限にオープンされたままの状態かどうかを確認するために行います。プローブが使用不能接続またはすでに使用されていない接続を検出すると、エラーを返し、それによってサーバー・プロセスが終了します。</p> <p>デッド接続検出機能を使用する上での制約は、次のとおりです。</p> <ul style="list-style-type: none"> ■ デッド接続検出は、bequeath 接続では許可されていない。 ■ プローブ・パケットはごく小さいものですが、ネットワーク・パフォーマンスを低下させるおそれのある余分な通信量を生成する。 ■ サーバーは、どのオペレーティング・システムを使用しているかにより、発生する他のイベントから接続プローブを区別するために、その他の処理を行わなければならない場合がある。このために、ネットワーク・パフォーマンスが低下する場合があります。
「Client Registration ID」	SQLNET.CLIENT_REGISTRATION	<p>クライアント・マシンの一意識別子を設定する。この識別子は接続要求とともにリスナーに渡されます。識別子として、最大 128 文字の任意の文字を使用できます。</p>

Oracle Net8 Assistant オプション	SQLNET.ORA パラメータ	説明
「Turn Off UNIX Signal Handling」	BEQUEATH_DETACH	クライアント・アプリケーションは bequeath プロトコルを介して内部にサーバー・プロセスを子プロセスとして生成するため、クライアント・アプリケーションは子プロセスが終了したときにそのプロセスをクリーン・アップする必要がある。サーバー・プロセスが接続の責任を完遂すると、それは消滅プロセスになります。シグナル・ハンドラは、これらの消滅プロセスをクリーン・アップする必要があります。このパラメータを設定してシグナル・ハンドラを使用禁止にすることにより、消滅プロセスを UNIX init プロセスに渡すようにクライアント・プロファイルを構成できます。
「Disable Out-of-Band Break」	DISABLE_OOB	基礎を形成するプロトコルがバンド外ブレイク機能をサポートしない場合に、この機能を使用禁止にする。

クライアントの詳細機能を設定するには、Oracle Net8 Assistant を使用します。

1. Oracle Net8 Assistant を起動します。
 - UNIX の場合、\$ORACLE_HOME/bin の下にある netasst を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」 -> 「Oracle - HOME_NAME」 -> 「Network Administration」 -> 「Net8 Assistant」を選択します。
2. 「Profile」アイコンをクリックします。
3. ドロップダウン・リスト・ボックスから「General」を選択します。

4. 「Advanced」タブをクリックします。



5. 設定するオプションに値を入力します。
6. 「File」メニューの「Save Network Configuration」を選択します。
7. 「File」メニューの「Exit」を選択して Oracle Net8 Assistant アプリケーションを終了します。

接続要求のルーティング

クライアントと、クライアントとして動作するサーバーは、接続要求をいつでも特定のプロセスに送るように構成できます。Oracle Net8 Assistant の次のオプションまたは SQLNET.ORA ファイルの次のパラメータを指定できます。

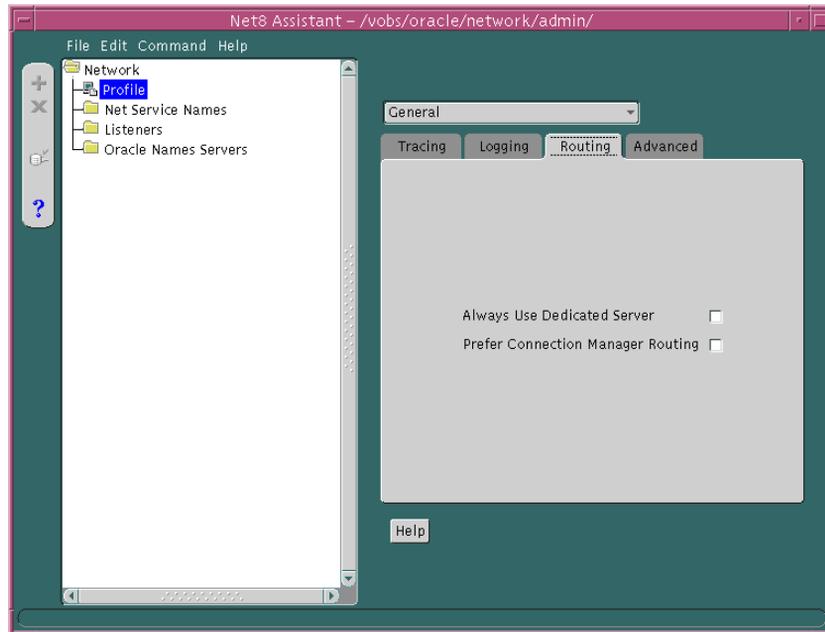
Oracle Net8 Assistant オプション	SQLNET.ORA パラメータ	説明
「Always Use Dedicated Server」	USE_DEDICATED_SERVER	リスナーがこのクライアントから発生するすべてのネットワーク・セッションで専用サーバーを使用するように設定する。
「Prefer Connection Manager Routing」	USE_CMAN	可能な限り Oracle Connection Manager を使用するようにクライアントに対して指定する。Oracle Connection Manager のアドレスが不明な場合は、利用できるリスナー・アドレスを介して接続のルートが設定されます。 注意： Oracle Names とともに Oracle Connection Manager を使用する場合、このオプションはクライアントと Oracle Names Server でオンに設定する必要があります。

注意： Oracle Net8 Assistant には「Use IPC Address for Client」が表示されますが、リリース 8.1 ではこのオプションに対応するパラメータ AUTOMATIC_IPC が使用禁止になっているため、この機能は使用できません。

接続要求のルートを設定するには、Oracle Net8 Assistant を使用します。

- Oracle Net8 Assistant を起動します。
 - UNIX の場合、\$ORACLE_HOME/bin の下にある netasst を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」-> 「Oracle - HOME_NAME」-> 「Network Administration」-> 「Net8 Assistant」を選択します。
- 「Profile」アイコンをクリックします。
- ドロップダウン・リスト・ボックスから「General」を選択します。

4. 「Routing」タブをクリックします。



5. 接続要求のルートを設定するいずれかのオプションをクリックします。
6. 「File」メニューの「Save Network Configuration」を選択します。
7. 「File」メニューの「Exit」を選択して Oracle Net8 Assistant アプリケーションを終了します。

Oracle Advanced Security の構成

Oracle Advanced Security は、Net8 と SQL*Net リリース 2.1.4 以降で動作するオプション製品です。この製品には、次の機能があります。

- セキュリティ・サービス
- 認証サービス
- Oracle DCE Integration

セキュリティ・サービス

Oracle Advanced Security は、データの読み取りまたは変更ができないようにするために、Net8 およびその関連製品によるネットワーク・データの暗号化およびチェックサムの使用を可能にします。これは RSA データ・セキュリティ RC4 またはデータ暗号化規格 (DES) の暗号化アルゴリズムを使用することで、データの未許可表示を防止します。Oracle Advanced Security は、データの修正、削除または再生が行われていないことを保証するために、安全に暗号化されたメッセージの要約を作成し、ネットワークから送信される各パケットに組み込みます。

Oracle Advanced Security は Oracle Connection Manager によってサポートされます。各種プロトコルを使用するクライアントとサーバーは、ネットワーク・プロトコル境界を超えて安全にデータを転送します。たとえば、NetWare (SPX) などの LAN プロトコルを使用しているクライアントは、LU6.2、TCP/IP、DECnet などのプロトコルを使用している大規模サーバーと安全にデータを共有できます。

認証サービス

Oracle Advanced Security には、シングル・サインオンのサポートなどの拡張ユーザー認証サービスが含まれています。これらの認証サービスは、信頼できるユーザー識別を行うことによってネットワーク・アクセス制御ログオン、ロール、監査などの Oracle7 と Oracle8i の既存のセキュリティ機能を拡張できます。アプリケーション側で変更を加える必要はありません。Oracle Advanced Security は、すべてのプロトコル、オペレーティング・システムおよびネーム・サービスで動作します。また、Security Dynamics ACE Server、Kerberos、RADIUS プロトコルや RADIUS 互換サーバー、Secure Sockets Layer (SSL)、DCE Security Server によるトークンの認証と、Identix による生物学的な認証もサポートします。

これらのサービスは、Net8 をインプリメントしているほとんどの製品で使用可能であり、その中には、Oracle8i、Developer 2000 Tools および Net8 をサポートしているその他の Oracle 製品またはサードパーティ製品が含まれています。

Oracle DCE Integration

Oracle Distributed Computing Environment (DCE) Integration は、Net8 と SQL*Net 2.1.6 以降で動作するオプション製品です。ユーザーは、この製品によって透過的に Oracle Tools および Oracle Applications を使用して、DCE 環境内の Oracle7 と Oracle8i のサーバーにアクセスできます。この製品は、転送メカニズムとして認証 RPC (リモート・プロシージャ・コール) を用意しており、これによってベンダー間の相互操作が可能です。DCE セキュリティ・サービスは、DCE にログオンしたユーザーが、ユーザー名またはパスワードを指定しなくても安全にすべての Oracle Application にアクセスできるようにします。これは、以前は "OPSS support" と呼ばれていましたが、現在は "外部認証" と呼ばれることがあります。

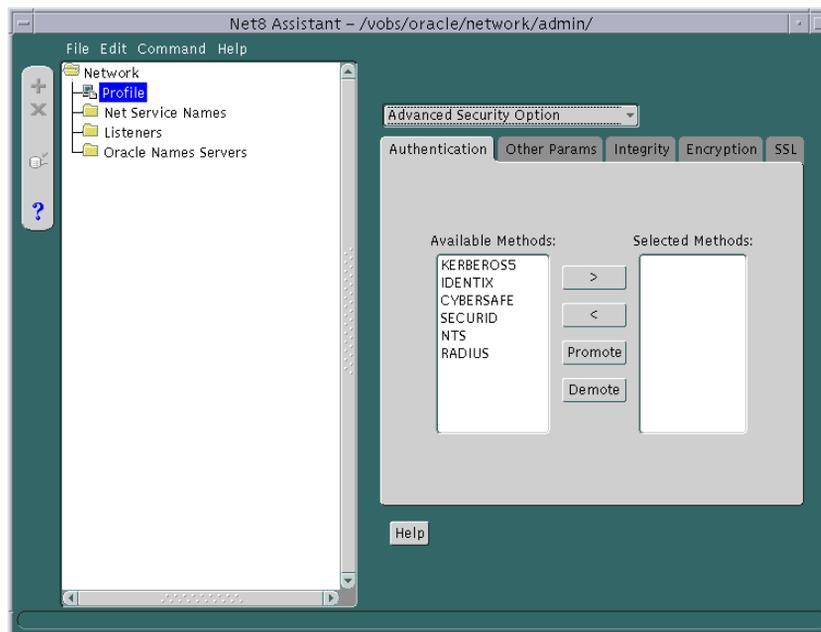
Oracle DCE Integration は、DCE セル・ディレクトリ・サービス (CDS) もサポートしており、これによって DCE 環境全体が Oracle7 と Oracle8i のサービスに透過的にアクセスできます。ユーザーは、よく知っている Oracle サービス名を使用して DCE 環境内の Oracle データベース・サーバーに接続できます。Oracle サービス名は、標準 DCE ツールを使用して 1 箇所から管理できます。詳細は、『Oracle8i Advanced Security 管理者ガイド』およびご使用のプラットフォーム対応のマニュアルを参照してください。

構成

クライアントで Oracle Advanced Security を使用するよう構成するには、次の手順に従います。

1. Oracle Net8 Assistant を起動します。
 - UNIX の場合、`$ORACLE_HOME/bin` の下にある `netasst` を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」-> 「Oracle - HOME_NAME」-> 「Network Administration」-> 「Net8 Assistant」を選択します。
2. 「Profile」アイコンをクリックします。

3. ドロップダウン・リスト・ボックスから「Advanced Security」を選択します。



構成の詳細は、『Oracle8i Advanced Security 管理者ガイド』を参照してください。

4. パラメータを適切に選択または編集します。
5. 「File」メニューの「Save Network Configuration」を選択します。
6. 「File」メニューの「Exit」を選択して Oracle Net8 Assistant アプリケーションを終了します。

接続集中化機能の使用

Oracle Connection Manager では、マルチスレッド・サーバー接続先への単一トランスポート接続を通じて複数の論理クライアント・ネットワーク・セッションを多重化または集中化する Net8 の機能が利用できます。Oracle Connection Manager の接続集中化機能を通じて、この機能が実現されます。

手順 1: Oracle Connection Manager の構成

ゲートウェイ・プロセスである CMGW では、次のデフォルト・リスニング・アドレスで接続要求を受信します。「[Oracle Connection Manager プロセス](#)」

```
cmgw=(address=(protocol=tcp)(host=anyhost)(port=1630))
```

管理プロセスである CMADMIN では、次のリスニング・アドレスでローカルおよびリモートの管理コマンドをリスニングします。

```
cmadmin=(address=(protocol=tcp)(host=anyhost)(port=1830))
```

TCP/IP またはポート 1630 および 1830 を使用しない場合で、アドレス情報が含まれる CMAN.ORA が存在しない場合は、UNIX では \$ORACLE_HOME/network/admin、Windows NT では ORACLE_HOME¥network¥admin に CMAN.ORA を作成して次の情報を指定します。

```
cmgw=(address=(protocol_address_information))
cmadmin=(address=(protocol_address_information))
```

Oracle Net8 Assistant では CMAN.ORA の構成をサポートしていないので、変更は手動で行う必要があります。

次の項目も参照してください。

- CMGW と CMADMIN の詳細は 2-46 ページの「[Oracle Connection Manager プロセス](#)」
- 適切なプロトコル・アドレス・パラメータの使用方法は 6-76 ページの「[プロトコル・アドレスの構成](#)」

手順 2: サーバーの構成

接続先サーバーがマルチスレッド・サーバーとして構成され、多重化機能がオンになっていることを確認します。これは、そのインスタンスのデータベース初期化ファイル (INITSID.ORA) の MTS_DISPATCHERS パラメータに、PROTOCOL 属性と MULTIPLEX 属性を設定することにより確認できます。

例:

```
mts_dispatchers="(protocol=tcp) (multiplex=on)"
```

属性	説明
PROTOCOL (PRO または PROT)	ディスパッチャでリスニング終点を生成するために使用されるネットワーク・プロトコル (上の例では TCP)。
MULTIPLEX (MUL または MULT)	<p>接続集中化を使用可能にするために使用される。</p> <p>1、ON、YES、TRUE または BOTH が指定されると、受信と送信の両方のネットワーク接続で接続集中化が使用可能です。</p> <p>IN が指定されると、クライアントから受信するネットワーク接続で接続集中化が使用可能です。</p> <p>OUT が指定されると、送信するネットワーク接続で接続集中化が使用可能です。</p> <p>0、NO、OFF または FALSE が指定されると、受信と送信の両方のネットワーク接続で接続集中化が使用禁止です。</p>

MTS の構成方法の詳細は、[第 9 章の「マルチスレッド・サーバーの構成」](#)を参照してください。

手順 3: クライアントの構成

クライアントのサポートは、TNSNAMES.ORA ファイルまたは Oracle Names Server を使用して Oracle Connection Manager のアドレスとリスナー・アドレスを設定することで実現します。

- TNSNAMES.ORA ファイルを使用する場合、Oracle Connection Manager のアドレスと SOURCE_ROUTE パラメータを設定する必要があります。このパラメータにより、Oracle Connection Manager から接続先データベースへのアドレスのソース・ルートが作成されます。ローカルネーミング・メソッドの TNSNAMES.ORA ファイルを構成するには、この章の「[Oracle Connection Manager 機能を使用する場合のクライアントの構成](#)」にある手順に従います。
- Oracle Names Server を使用する場合、Oracle Connection Manager により Oracle Names Server 内の既存のアドレスに Oracle Connection Manager のアドレスを挿入することで、アドレスが自動的に更新されます。Oracle Names Server では Oracle Connection Manager に要求を送信するため、クライアント・マシンおよび Oracle Names Server が常駐するすべてのマシンの SQLNET.ORA ファイルに USE_CMAN=TRUE を設定する必要があります。このパラメータを設定するには、7-25 ページの「[接続要求のルーティング](#)」にある手順に従います。

図 7-1 に、通常の TNSNAMES.ORA ファイルと、Oracle Connection Manager を使用するためのエントリ（Connection Manager のエントリは太字部分）が追加された TNSNAMES.ORA ファイルの比較を示します。

図 7-1 Oracle Connection Manager を使用する場合と使用しない場合の TNSNAMES.ORA ファイル

```

sales=
  (description=
    (address=
      (protocol=tcp)
      (host=sales-pc)
      (port=1521)
    )
    (connect_data=
      (service_name=sales,com)
    )
  )
)

sales=
  (description=
    (source_route=yes)
    (address=
      (protocol=tcp)
      (host=cman=pc)
      (port=1630)
    )
    )
  (address=
    (protocol=tcp)
    (host=sales=pc)
    (port=1521)
  )
  (connect_data=
    (service_name=sales.com)
  )
)

```

TNSNAMES.ORA 要素	説明
source_route=yes	Oracle Connection Manager から接続先データベースまでのアドレスのソース・ルートを作成する。
(address= (protocol=tcp) (host=cman=pc) (port=1630))	最初のアドレスは Oracle Connection Manager までのアドレスです。ここから、リスナーを介してサービスに接続します。

複数プロトコル・サポートの使用

注意： 接続パス（2 hop 以上）で複数の Oracle Connection Manager が使用されている場合は、Connection Manager を通じてクライアントに接続するために Oracle Names を使用することはできません。

Oracle Connection Manager には、異なるネットワーク・プロトコルをもつクライアントとサーバーが相互に通信できるようにする複数プロトコルのサポートがあります。Oracle Connection Manager では、Oracle でサポートされるあらゆるプロトコルでリスニングできます。

複数プロトコル・サポートを使用しない場合、SPX を使用するクライアントは、TCP/IP を使用するサーバーに接続できません。Oracle Connection Manager が TCP/IP 用に構成されていれば、クライアントは SPX を使用して Oracle Connection Manager に接続でき、TCP/IP を使用するサーバーには Oracle Connection Manager が接続できます。

手順 1: Oracle Connection Manager の構成

ゲートウェイ・プロセスである CMGW では、次のデフォルト・リスニング・アドレスで接続要求を受信します。

```
cmn=(address=(protocol=tcp)(host=anyhost)(port=1630))
```

管理プロセスである CMADMIN では、次のリスニング・アドレスでローカルおよびリモートの管理コマンドをリスニングします。

```
cmn_admin=(address=(protocol=tcp)(host=anyhost)(port=1830))
```

TCP/IP またはポート 1630 および 1830 を使用しない場合、アドレス情報が含まれる CMAN.ORA が存在しないならば、UNIX では \$ORACLE_HOME/network/admin、Windows NT では ORACLE_HOME¥network¥admin に CMAN.ORA を作成して次の情報を指定します。

```
cmn=(address=(protocol_address_information))
cmn_admin=(address=(protocol_address_information))
```

Oracle Net8 Assistant では CMAN.ORA の構成をサポートしていないので、変更は手動で行う必要があります。

次の項目も参照してください。

- CMGW と CMADMIN の詳細は 2-46 ページの「[Oracle Connection Manager プロセス](#)」
- 適切なプロトコル・アドレス・パラメータ使用方法は 6-76 ページの「[プロトコル・アドレスの構成](#)」

手順 2: クライアントの構成

複数プロトコル・サポートは、TNSNAMES.ORA ファイルまたは Oracle Names Server を使用して Oracle Connection Manager のアドレスとリスナー・アドレスを設定することで実現します。

- TNSNAMES.ORA ファイルを使用する場合、Oracle Connection Manager のアドレスと SOURCE_ROUTE パラメータを設定する必要があります。このパラメータにより、Oracle Connection Manager から接続先データベースへのアドレスのソース・ルートが作成されます。ローカルネーミング・メソッドの TNSNAMES.ORA ファイルを構成するには、この章の「[Oracle Connection Manager 機能を使用する場合のクライアントの構成](#)」にある手順と次の助言に従います。
 - まず、クライアントで利用できるプロトコルを使用して、Oracle Connection Manager のアドレスを持つクライアントを構成する。
 - サーバーで利用できるプロトコルを使用して、リスナーのアドレスを持つクライアントを構成する。
 - Oracle Connection Manager マシンには、クライアント・マシンとサーバー・マシンで使用されるプロトコルがすべて含まれている必要がある。
- Oracle Names Server を使用する場合、Oracle Connection Manager により Oracle Names Server 内の既存のアドレスに Oracle Connection Manager のアドレスを挿入することで、アドレスが自動的に更新されます。Oracle Names Server では Oracle Connection Manager に要求を送信するため、クライアント・マシンおよび Oracle Names Server が常駐するすべてのマシンの SQLNET.ORA ファイルに USE_CMAN=TRUE を設定する必要があります。このパラメータを設定するには、7-25 ページの「[接続要求のルーティング](#)」にある手順に従います。

図 7-2 は、SPX を使用するクライアントが Oracle Connection Manager に接続し、Oracle Connection Manager が TCP/IP を使用してデータベース・サーバーに接続することを示しています。

図 7-2 複数プロトコル・サポートを使用するように構成された CMAN パラメータ

```

cman=
  (description=
    (source_route=yes)
    (address=
      (protocol=spx)
      (service=cman)
    )
    (address=
      (protocol=tcp)
      (host=sales-pc)
      (port=1521)
    )
    (connect_data=
      (service_name=sales.com)
    )
  )

```

TNSNAMES.ORA 要素	説明
source_route=yes	Oracle Connection Manager から接続先データベースまでのアドレスのソース・ルートを作成する。
(address= (protocol=spx) (service=cman))	最初のアドレスは、SPX を使用するクライアントから Oracle Connection Manager までです。ここから、Oracle Connection Manager は TCP/IP を使用するサービスにリスナーを通じて接続します。

Net8 アクセス制御の使用

Oracle Connection Manager には、TCP/IP 環境で指定されたサーバーへのクライアント・アクセスを制御するときに使用できる機能も含まれています。サーバーへの特定のクライアント・アクセスを可能にするか、または制限するフィルタリング・ルールを設定できます。

ネットワーク上にリリース 8.1 のサービスが存在する場合、リリース 8.1 の Oracle Connection Manager がなければこの機能は利用できません。

手順 1: Oracle Connection Manager の構成

Oracle Connection Manager を構成するには、次の手順に従います。

1. アドレス情報が含まれる CMAN.ORA が存在しない場合、UNIX では \$ORACLE_HOME/network/admin、Windows NT では ORACLE_HOME¥network¥admin に、CMAN.ORA ファイルを作成します。Oracle Net8 Assistant では、CMAN.ORA の構成をサポートしていません。

2. CMGW プロセスでデフォルトのリスニング・ポート 1630 を使用しない場合、次のエントリを追加します。

```
cmgw=(address=(protocol=tcp)(host=anyhost)(port=port))
```

CMGW の詳細は、2-46 ページの「**CMGW**」を参照してください。

3. CMADMIN プロセスでデフォルトのリスニング・ポート 1830 を使用しない場合、次のエントリを追加します。

```
cmadmin=(address=(protocol=tcp)(host=anyhost)(port=port))
```

CMADMIN の詳細は、2-47 ページの「**CMADMIN**」を参照してください。

4. CMAN_RULES セクションを追加して次のパラメータを指定します。

```

cman_rules=
  (rule_list=
    (rule=
      (src=shost)
      (dst=dhost)
      (srv=service)
      (act=accept | reject)
    )
  )

```

パラメータ	説明
<i>SHOST</i>	ソースのホスト名または IP アドレス (クライアント)
<i>DHOST</i>	接続先のホスト名または IP アドレス (サーバー)
<i>SERVICE</i>	リリース 8.1 サービスの SERVICE_NAME (INITSID.ORA ファイルから取得) 値またはリリース 8.1 より前のサービスの SID
<i>ACCEPT REJECT</i>	上の 3 つのパラメータに基づく受信要求の受け入れまたは拒否

RULE_LIST 内で、複数の RULE を定義することもできます。最初に適合した RULE のアクション (ACT) が要求に適用されます。CMAN_RULES が存在する場合、Oracle Connection Manager は " 明白に許可されないものは禁止 " という原則に従います。CMAN_RULES が未定義の場合は、すべて許可されます。

次の例は、クライアント・マシン CLIENT1-PC のサービス SALES.COM に対する制限、およびクライアント 144.25.23.45 のサービス DB1 へのアクセスを示します。

```

cman_rules=
  (rule_list=
    (rule=(src=client1-pc)(dst=sales-pc)(srv=sales.com)(act=reject))
    (rule=(src=144.25.23.45)(dst=144.25.187.200)(srv=db1)(act=accept))
  )

```

手順 2: クライアントの構成

クライアントのサポートは、TNSNAMES.ORA ファイルまたは Oracle Names Server を使用して Oracle Connection Manager のアドレスとリスナー・アドレスを設定することで実現します。

- TNSNAMES.ORA ファイルを使用する場合、Oracle Connection Manager のアドレスと SOURCE_ROUTE パラメータを設定する必要があります。このパラメータにより、Oracle Connection Manager から接続先データベースへのアドレスのソース・ルートが作成されます。ローカルネーミング・メソッドの TNSNAMES.ORA ファイルを構成するには、この章の「[Oracle Connection Manager 機能を使用する場合のクライアントの構成](#)」にある手順と助言に従います。
 - まず、TCP/IP のみを使用して、クライアントに Oracle Connection Manager のアドレスを構成する。
 - Oracle Connection Manager で利用できるプロトコルを使用して、クライアントにリスナーのアドレスを構成する。
 - Connection Manager から接続先データベースまでのアドレスのソース・ルートを作成するために、クライアントで SOURCE_ROUTE=YES を構成する。
- Oracle Names Server を使用する場合、Oracle Names Server 内の既存のアドレスに Oracle Connection Manager のアドレスを挿入することで、Oracle Connection Manager によりアドレスが自動的に更新されます。Oracle Names Server では Oracle Connection Manager に要求を送信するため、クライアント・マシンおよび Oracle Names Server が常駐するすべてのマシンの SQLNET.ORA ファイルに USE_CMAN=TRUE を設定する必要があります。このパラメータを設定するには、7-25 ページの「[接続要求のルーティング](#)」にある手順に従います。

Oracle Connection Manager 機能を使用する場合のクライアントの構成

注意： Oracle Connection Manager を使用する場合はアドレス・リストの順序に特定の順序や意味があるので、クライアント負荷のバランスや connect-time failover を設定しないことをお勧めします。SOURCE_ROUTE を設定すると、最初のアドレスはクライアントの Oracle Connection Manager への接続で使用され、2 番目のアドレスは Oracle Connection Manager のリスナーへの接続で使用されます。SOURCE_ROUTE が設定されている場合、クライアント負荷のバランスと connect-time failover は使用禁止です。

注意： 接続パスで複数の Oracle Connection Manager が使用されている場合は、Connection Manager を通じてクライアントに接続するために Oracle Names を使用することはできません。

データベース・サーバーにクライアントの接続要求を集中化させる場合、Connection Manager を通じてのルーティングに関するクライアントを構成します。これは、TNSNAMES.ORA ファイルまたは Oracle Names Server を使用して Oracle Connection Manager のアドレスを設定することで実現します。

- Oracle Names Server を使用する場合、Oracle Names Server 内の既存のアドレスに Oracle Connection Manager のアドレスを挿入することで、Oracle Connection Manager によりアドレスが自動的に更新される。Oracle Names Server では Oracle Connection Manager に要求を送信するため、Oracle Names Server が常駐するすべてのマシンの SQLNET.ORA ファイルに USE_CMAN=TRUE を設定する必要があります。このパラメータを設定するには、7-25 ページの「[接続要求のルーティング](#)」にある手順に従います。
- TNSNAMES.ORA ファイルを使用する場合、Oracle Connection Manager のアドレスと SOURCE_ROUTE パラメータを設定する必要があります。このパラメータにより、Oracle Connection Manager から接続先データベースへのアドレスのソース・ルートが作成されます。ローカルネーミング・メソッドの TNSNAMES.ORA ファイルを構成するには、次の手順に従います。

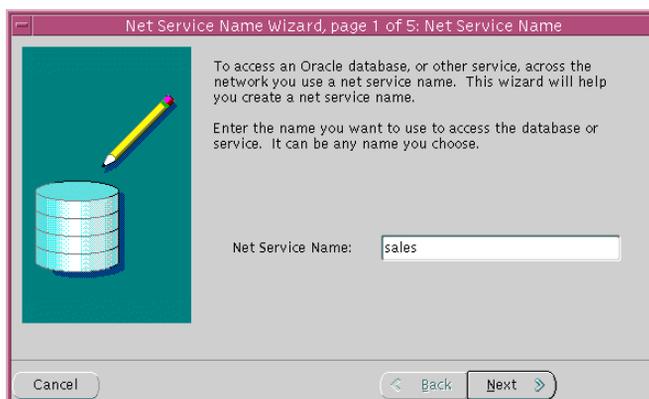
手順 1: Oracle Connection Manager アドレスの構成

手順 2: リスナー・アドレスの構成

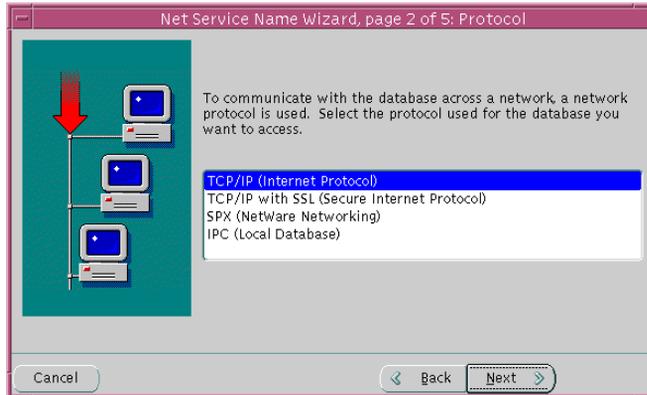
手順 1: Oracle Connection Manager アドレスの構成

TNSNAMES.ORA に Oracle Connection Manager アドレスを構成するには、次の手順に従います。

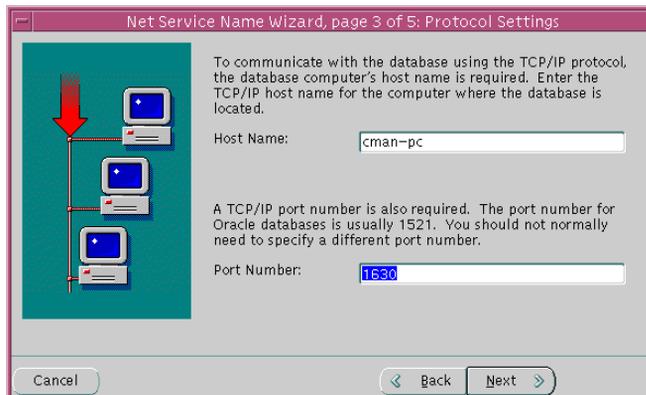
1. Oracle Net8 Assistant を起動します。
 - UNIX の場合、`$ORACLE_HOME/bin` の下にある `netasst` を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」-> 「Oracle - HOME_NAME」-> 「Network Administration」-> 「Net8 Assistant」を選択します。
2. ディレクトリ・ツリーで、ネット・サービス名フォルダをダブルクリックします。
3. ネット・サービス名を新規作成するために、「Edit」メニューの「Create」を選択するか「+」ボタンをクリックします。Net Service Name Wizard が起動します。
4. 「Net Service Name」フィールドに名前を入力し、「Next」をクリックします。



5. CMGW プロセスがリスニング対象としているプロトコルを選択し、「Next」をクリックします。デフォルトのプロトコルは TCP/IP です。

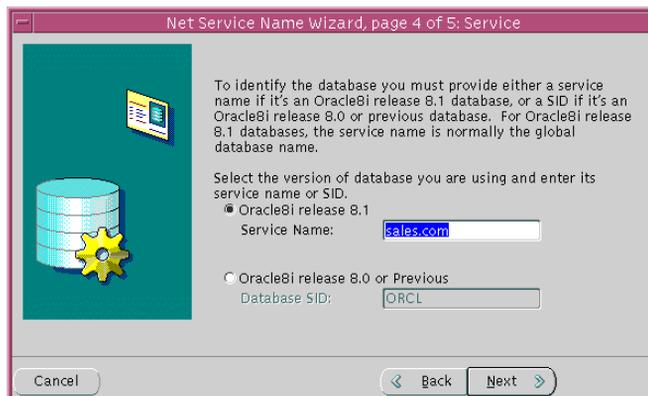


6. Oracle Connection Manager のネットワーク・プロトコル情報を入力し、「Next」をクリックします。TCP/IP を使用している場合、UNIX オペレーティング・システムでのデフォルト・ポートは 1630 です。



プロトコル・パラメータの詳細は、6-76 ページの「[プロトコル・アドレスの構成](#)」を参照してください。

7. 接続先データベースがリリース 8.1 である場合、「Oracle8 Version 8.1」をクリックし、「Service Name」フィールドにサービス名を入力し、「Next」をクリックします。



サービス名は通常、グローバル・データベース名です。使用するサービス名文字列の詳細は、6-2 ページの「サービス名とインスタンス名の設定」を参照してください。

接続先データベースがリリース 8.1 より前である場合、「Oracle8 Version 8.0 or Previous」をクリックし、「Database SID」フィールドに SID を入力し、「Next」をクリックします。

8. 「Next」をクリックします。

注意：「Test」をクリックしないでください。クリックすると接続エラーが発生します。

9. 「Finish」をクリックして構成を保存し、Net Service Name Wizard を終了します。
新規のネット・サービス名と Oracle Connection Manager アドレスが、ネット・サービス名フォルダと TNSNAMES.ORA に追加されます。

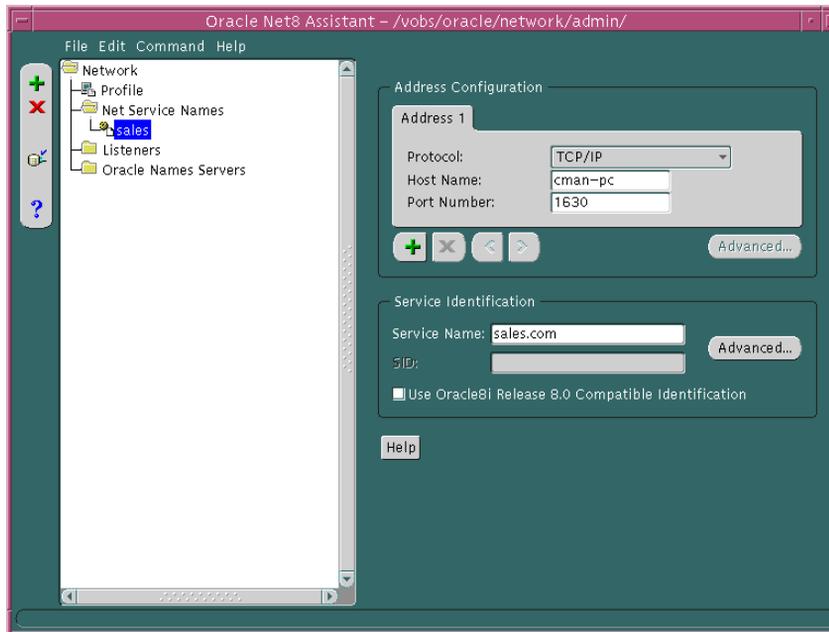
手順 2: リスナー・アドレスの構成

Oracle Connection Manager アドレスを指定した後で、Oracle Connection Manager がサーバーと接続できるように、リスナーのアドレスを作成します。

TNSNAMES.ORA にリスナー・アドレスを構成するには、次の手順に従います。

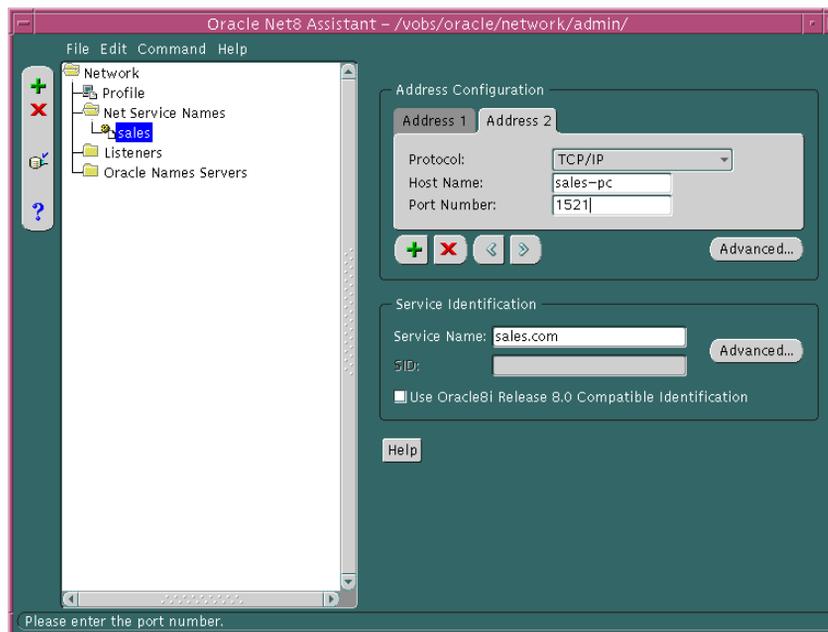
1. ネット・サービス名フォルダをダブルクリックします。
2. 7-41 ページの「[手順 1: Oracle Connection Manager アドレスの構成](#)」で作成したネット・サービス名をクリックします。

「Address 1」タブには Oracle Connection Manager アドレスが表示されることに注意してください。



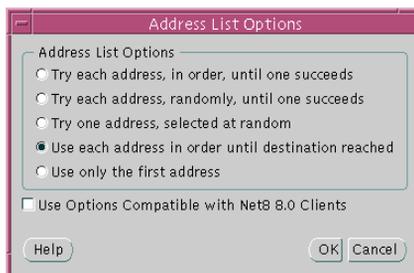
3. 「+」をクリックします。
新規のアドレス・タブが表示されます。

4. プロトコルを選択し、リスナーのプロトコル情報を入力します。TCP/IP を使用している場合、デフォルト・ポート番号は 1521 です。



プロトコル・パラメータの詳細は、6-76 ページの「[プロトコル・アドレスの構成](#)」を参照してください。

5. 「Address Configuration」グループ・ボックスで「Advanced」をクリックします。
「Address List Options」ダイアログ・ボックスが表示されます。



6. 「Use each address in order until destination reached」をオンにします。

このオプションにより、クライアントが最初のアドレス（Oracle Connection Manager）に接続し、最初のアドレスから 2 番目のアドレス（リスナー）に接続するように設定されます。

7. 「OK」をクリックしてダイアログ・ボックスを閉じます。
8. 「File」メニューの「Save Network Configuration」を選択します。
9. 「File」メニューの「Exit」を選択して Oracle Net8 Assistant を終了します。
リスナー・アドレスが TNSNAMES.ORA に追加されます。

接続プーリングの使用

接続プーリングは、マルチスレッド・サーバーに対する物理ネットワーク接続数を最大にすることができるリソース利用機能です。これは、複数のクライアント・プロセス間でディスパッチャの一連の接続を共有またはプールすることにより行います。

接続先サーバーがマルチスレッド・サーバーとして構成され、接続プーリング・パラメータが設定されていることを確認します。これは、データベース初期化ファイル (INITSID.ORA) の MTS_DISPATCHERS パラメータに、PROTOCOL、DISPATCHERS、POOL および CONNECTIONS の各属性を設定することにより確認できます。

例：

```
mts_dispatchers="(protocol=tcp) (dispatchers=2) (pool=yes) (connections=2)"
```

属性	説明
CONNECTIONS (CON または CONN)	各ディスパッチャで許容されるネットワーク接続の最大数。デフォルトは 1022 です。
DISPATCHERS (DIS または DISP)	開始するディスパッチャの初期番号。デフォルトは 1 です。
PROTOCOL (PRO または PROT)	ディスパッチャでリスニング終点を生成するために使用されるネットワーク・プロトコル (上の例では TCP)。
POOL	数字が指定される場合、接続プーリングは送受信のネットワーク接続で使用可能。指定された数は送受信のネットワーク接続でタイムアウトとなるティック数です。 ON、YES、TRUE または BOTH が指定される場合、接続プーリングは送受信のネットワーク接続で使用可能であり、送受信のネットワーク接続でのタイムアウトはデフォルトの 10 が使用されます。
TICKS	ネットワーク・ティックの秒数。デフォルトは 15 秒です。接続プーリングの場合、このパラメータは省略可能です。

MTS の構成方法の詳細は、[第 9 章の「マルチスレッド・サーバーの構成」](#)を参照してください。

Java オプション接続の構成

Oracle8i の Java オプションをプログラミングする方法は次の 3 通りです。

- Java ストアド・プロシージャ - Java (SQLJ) にインプリメントされている従来のデータベース・ストアド・プロシージャ、関数、トリガおよびオブジェクト・メソッド
- Enterprise JavaBeans (EJB) - 標準的なサーバー指向 Java コンポーネント
- CORBA Servers - Java にインプリメントされている標準的な CORBA 2.0 準拠サーバー

EJB と CORBA クライアントは、CORBA Internet Inter-Orb Protocol (IIOP) プロトコルを介して起動されます。

Java ストアド・プロシージャは、専用サーバー・モードかマルチスレッド・サーバー (MTS) モードで動作可能です。EJB と CORBA Servers は MTS 構成でのみ動作します。主に Java ストアド・プロシージャを開発する場合、専用サーバー構成で実行するとよいでしょう。EJB と CORBA Servers を 1 つのアプリケーションで Java ストアド・プロシージャと組み合わせている場合、MTS 構成で EJB と CORBA Servers をサポートし、専用サーバー構成でストアド・プロシージャをサポートするようにデータベースを構成できます。

専用サーバー・モードに Java ストアド・プロシージャを設定し、MTS モードに EJB と CORBA Servers を設定する必要がある場合、9-10 ページの「[マルチスレッド・サーバー・モードと専用サーバー・モードの両方の構成](#)」を参照してください。

この項の内容は、次のとおりです。

- [Java ストアド・プロシージャ構成](#)
- [Enterprise JavaBeans と CORBA 構成](#)
- [リスナーから EJB と CORBA を使用するための Oracle8i デフォルト構成の使用](#)
- [デフォルト以外の構成の使用](#)
- [EJB と CORBA による SSL の使用](#)

Java ストアド・プロシージャ構成

Java ストアド・プロシージャのみを実行するデータベースを構成するには、専用サーバー・モードと MTS モードのどちらでデータベースを実行するかを決める必要があります。Java ストアド・プロシージャを専用サーバー・モードでのみ使用する場合、『Oracle8i Java ストアド・プロシージャ開発者ガイド』に説明されているように、データベースとクライアントを構成する必要があります。

Java ストアド・プロシージャを MTS モードで実行する場合、第 9 章の「マルチスレッド・サーバーの構成」に説明されているように、MTS_DISPATCHERS パラメータを使用して MTS モードでサーバーを構成する手順を実行する必要があります。

Enterprise JavaBeans と CORBA 構成

Enterprise JavaBeans と CORBA Servers を使用するために Oracle8i を構成する場合、次の点に注意する必要があります。

- [IIOP クライアント](#)
- [セッションベースの IIOP と構成についての問題](#)
- [リスナーからのアクセスとディスパッチャからのアクセス（接続負荷のバランス）](#)
- [CORBA と EJB による SSL サポート](#)

IIOP クライアント

クライアントは、Inter-Orb Protocol (IIOP) プロトコルを介してデータベース内の EJB と CORBA Servers にアクセスします。IIOP をサポートするには、General Inter-Orb Protocol (GIOP) プレゼンテーション・プロトコルを使用してデータベースを MTS モードに構成する必要があります (IIOP は GIOP を TCP/IP の上位にインプリメントしたものです)。Oracle8i では、GIOP サービス・インプリメンテーションを提供しています。Oracle8i Java VM は、セッション指向の Java VM です。これは、データベース内の各セッションがプライベート・サーバーとして効率的に固有の VM を取得できることを意味します。Java VM、ORB およびデータベース・プレゼンテーション・レイヤーを組み合わせることで、複数のクライアントから同一または別個のセッションで稼働する EJB および CORBA Servers との CORBA 通信が可能になります。詳細は、『Oracle8i Enterprise JavaBeans と CORBA 開発者ガイド』を参照してください。

セッションベースの IIOP と構成についての問題

Oracle8i のセッション指向の動作をサポートするために、データベースの GIOP 機能が 2 つのプレゼンテーション・プロトコルをサポートするように構成されています。使用するプレゼンテーションに応じて、次のいずれかのプレゼンテーション・ハンドラを選択する必要があります。

- oracle.aurora.server.SgiopServer (サーバーベースの IIOP 接続の場合)。EJB と CORBA は、標準の IIOP 方法に加えて、所属するセッションに基づいて区別されます。セッションベースの IIOP は、標準の IIOP 方法よりはるかに拡張性に富みます。セッションベースの IIOP 接続を使用することをお勧めします。(デフォルト)
- oracle.aurora.server.GiopServer (通常の IIOP 接続の場合)

IIOP をサポートするために、MTS_DISPATCHERS パラメータの PRESENTATION 属性に適切なプレゼンテーションを指定する必要があります。セッションベースの IIOP (デフォルトの動作) を処理するために、MTS_DISPATCHERS パラメータの PRESENTATION 属性には、プレゼンテーション・ハンドラとして oracle.aurora.server.SgiopServer が登録されています。

プレゼンテーションとセッション IIOP の詳細は、『Oracle8i Enterprise JavaBeans と CORBA 開発者ガイド』を参照してください。

リスナーからのアクセスとディスパッチャからのアクセス (接続負荷のバランス)

最後に、IIOP を介してデータベースと通信する EJB と CORBA クライアントは、ディスパッチャに直接アクセスするか、リスナーからディスパッチャにアクセスすることで、データベースと通信できます。この 2 つの方法の長所と短所は次のとおりです。

- リスナーからのアクセス

拡張性の点で、クライアントはリスナーを通じて Java オプションに接続できます。リスナーは一般的なポート番号をリスニングするように構成し、クライアントはこのポート番号を使用してリスナーと通信します。CORBA と EJB をサポートするために、リスナーは TTC プロトコルのポートをリスニングするのみでなく、GIOP やセッションベースの GIOP 接続の処理もするように構成する必要があります。リスナーの負荷のバランスにより、クライアントの要求はディスパッチャ間で分散されます。この構成は複雑ですが、拡張性に富み、1,000 以上の同時接続をサポートできます。

- ディスパッチャへの直接アクセス

ディスパッチャは一般的なポート番号に構成されます。クライアントはこのポート番号を使用して、ディスパッチャに直接接続します。この構成のセットアップは簡単ですが、1,000 を超える同時接続をサポートする必要がある場合は、拡張性に問題が起こる可能性があります。

リスナーからのアクセスをお勧めします。

CORBA と EJB による SSL サポート

Oracle8i では、SSL を両方の GIOP プロトコル（通常の GIOP とセッションベースの GIOP）と組み合わせて使用する場合に必要なプライベート・キーなど、認証データの使用もサポートしています。GIOP で SSL を使用するには、次の 2 つの手順を実行する必要があります。

1. リスナーのアドレス・リストでプロトコルとして TCPS を指定することにより、SSL を受け付けるようにリスナーとディスパッチャを構成します。
2. リスナーとディスパッチャの両方を構成するときに、使用する SSL Wallet を指定します。詳細は、「[EJB と CORBA による SSL の使用](#)」を参照してください。

リスナーから EJB と CORBA を使用するための Oracle8i デフォルト構成の使用

サーバーの「Typical」または「Minimal」インストールを実行中、または Oracle Database Configuration Assistant を使用して「Java VM オプション」を選択した場合、MTS は自動的に、TCP/IP でリスナーを通じてセッションベースの IIOP 接続を使用するように構成されます。

INITSID.ORA ファイルは次のように構成されます。

```
mts_dispatchers="(protocol=tcp)(presentation=oracle.aurora.server.SGiopServer)"
```

デフォルトの属性を次の表に示します。

パラメータ	説明
PROTOCOL (PRO または PROT)	TCP/IP プロトコルを指定する。ディスパッチャでリスニング終点を生成するために使用されます。
PRESENTATION (PRE または PRES)	GIOP のサポートを使用可能にする。GIOP プレゼンテーション値として有効な値は次のとおりです。 <ul style="list-style-type: none">oracle.aurora.server.SGiopServer (セッションベースの GIOP ディスパッチャ接続)。このプレゼンテーションは、TCP/IP と、SSL 付き TCP/IP で有効です。oracle.aurora.server.GiopServer (通常の GIOP ディスパッチャ接続)。このプレゼンテーションは、TCP/IP と、SSL 付き TCP/IP で有効です。 詳細は、『Oracle8i Enterprise JavaBeans と CORBA 開発者ガイド』を参照してください。

LISTENER.ORA ファイルは次のように構成されます。

```
listener=
  (description_list=
    (description=
      (address_list=
        (address=(protocol=tcp)(host=sales-pc)(port=1521))
      )
    )
  )
  (description=
    (protocol_stack=
      (presentation=giop)
      (session=raw)
    )
    (address_list=
      (address=(protocol=tcp)(host=sales-pc)(port=2481))
    )
  )
)
```

LISTENER.ORA 要素	説明
PROTOCOL_STACK	接続で使用される プレゼンテーション・レイヤーとセッション・レイヤー の情報を識別する。
(PRESENTATION=GIOP)	IIOIP クライアントで使用される GIOP のプレゼンテーションを識別する。GIOP は、TCP/IP を使用した oracle.aurora.server.SgiopServer と oracle.aurora.server.GiopServer の両方をサポートしています。
(SESSION=RAW)	セッション・レイヤーを識別する。IIOIP クライアントにセッション・レイヤーはありません。
(ADDRESS=...)	ポート 2481 で TCP/IP が使用されるリスニング・アドレスを指定する。

デフォルト以外の構成の使用

デフォルトの構成を使用できない場合、Java オプションへの IOP 接続をリスナーから行うか、独自の設定でディスパッチャに直接接続するかを構成できます。この項の内容は、次のとおりです。

- リスナーからアクセスするための独自構成
- 独自のリスナーの構成
- ディスパッチャへの直接アクセスの構成

リスナーからアクセスするための独自構成

リスナーから Java オプションへの IOP クライアント接続を構成するには、次の手順に従います。

手順 1: INITSID.ORA ファイルで MTS を使用可能にして IOP を構成

手順 2: リスナーの構成

手順 3: クライアントの構成

手順 1: INITSID.ORA ファイルで MTS を使用可能にして IIOP を構成

MTS は、INITSID.ORA ファイル内で次のように構成する必要があります。

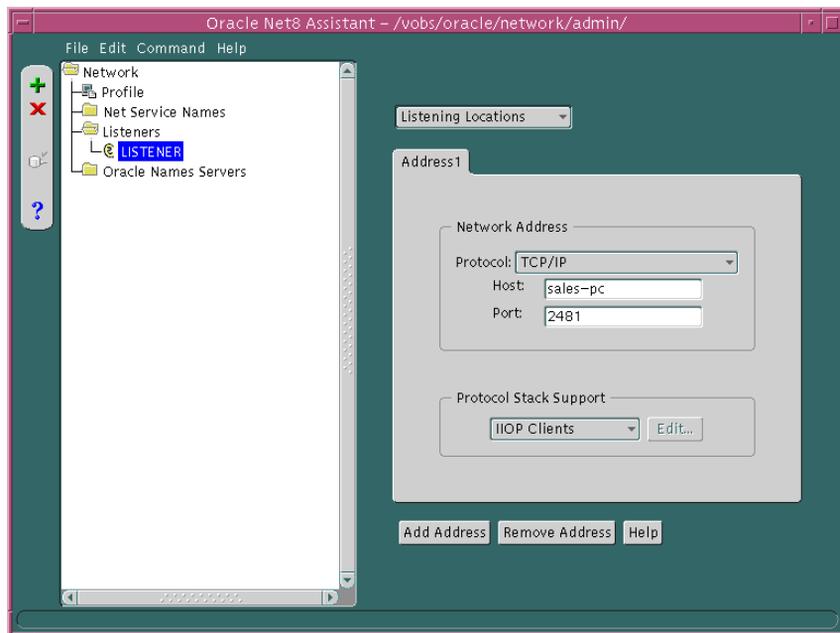
```
mts_dispatchers="(protocol=tcp | tcps)
(presentation=oracle.aurora.server.SGiopServer |
oracle.aurora.server.GiopServer)"
```

属性の説明は次のとおりです。

パラメータ	説明
PROTOCOL (PRO または PROT)	TCP/IP または SSL 付き TCP/IP を指定する。ディスパッチャでリスニング終点を生成するために使用されます。 有効な値: TCP (TCP/IP の場合) または TCPS (SSL 付き TCP/IP)
PRESENTATION (PRE または PRES)	GIOP のサポートを使用可能にする。 GIOP プレゼンテーション値として有効な値は次のとおりです。 <ul style="list-style-type: none"> oracle.aurora.server.SGiopServer (セッションベースの GIOP ディスパッチャ接続)。このプレゼンテーションは、TCP/IP と、SSL 付き TCP/IP で有効です。 oracle.aurora.server.GiopServer (通常の GIOP ディスパッチャ接続)。このプレゼンテーションは、TCP/IP と、SSL 付き TCP/IP で有効です。 <p>詳細は、『Oracle8i Enterprise JavaBeans と CORBA 開発者ガイド』を参照してください。</p>

手順 2: リスナーの構成 Oracle Net8 Assistant を使用してリスナー設定を変更します。

1. Oracle Net8 Assistant を起動します。
 - UNIX の場合、`$ORACLE_HOME/bin` の下にある `netasst` を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」 -> 「Oracle - HOME_NAME」 -> 「Network Administration」 -> 「Net8 Assistant」を選択します。
2. リスナー・フォルダをダブルクリックします。
3. リスナーをクリックします。
4. ドロップダウン・リスト・ボックスから「Listening Locations」を選択します。



5. 「Protocol」ドロップダウン・メニューから「TCP/IP」が「TCP/IP with SSL」を選択します。
6. 「Host」フィールドにデータベースのホスト名を入力します。
7. プロトコルに TCP/IP を選択した場合、「Port」フィールドに 2481 と入力します。SSL 付き TCP/IP を選択した場合、「Port」フィールドに 2482 と入力します。

ポート 2482 や 2483 を使用しない場合、7-57 ページの「[独自のリスナーの構成](#)」の手順に従ってください。

8. 「Client」ドロップダウン・リスト・ボックスから「IIOP Client」を選択します。
9. 「File」メニューの「Save Network Configuration」を選択します。
10. 「File」メニューの「Exit」を選択して Oracle Net8 Assistant を終了します。

手順 3: クライアントの構成 IIOP プレゼンテーションでは、TCP/IP の場合はポート 2481、SSL 付き TCP/IP の場合は 2482 を使用します。詳細は、『Oracle8i Enterprise JavaBeans と CORBA 開発者ガイド』を参照してください。

独自のリスナーの構成

デフォルトのリスナーを使用しない場合、7-54 ページの「[リスナーからアクセスするための独自構成](#)」で説明されている手順に加えて、次の手順を実行します。

- [手順 1: INITSID.ORA ファイルの追加構成](#)
- [手順 2: リスナー名の解決](#)

手順 1: INITSID.ORA ファイルの追加構成 独自のリスナーは、INITSID.ORA に構成する必要があります。構成方法は、ローカル・リスナーの場合とリモート・リスナーの場合とでは異なります。この項では、両方の構成方法を説明します。

ローカル・リスナー

デフォルト以外のローカル・リスナーを通じて接続する場合、INITSID.ORA には LOCAL_LISTENER パラメータか、MTS_DISPATCHERS パラメータの LISTENER 属性を構成する必要があります。

注意： Oracle Database Configuration では、LISTENER 属性を構成できません。LOCAL_LISTENER は手動で構成する必要があります。

LOCAL_LISTENER パラメータ構成は次のとおりです。

```
local_listener=local_listener_name
mts_dispatchers="(protocol=tcp | tcps)
(presentation=oracle.aurora.server.SGiopServer |
oracle.aurora.server.GiopServer)"
```

MTS_DISPATCHERS パラメータの LISTENER 属性の構成は次のとおりです。

```
mts_dispatchers="(protocol=tcp | tcps)
(presentation=oracle.aurora.server.SGiopServer |
oracle.aurora.server.GiopServer)(listener=local_listener_name)"
```

パラメータ	説明
LISTENER (LIS または LIST)	ディスパッチャで登録されるリスナーのネットワーク名を指定する。ネーミング・メソッドで解決される名前を設定する必要があります。

リモート・リスナー

リモート・リスナーを通じて接続する場合、INITSID.ORA には MTS_DISPATCHERS パラメータの LISTENER 属性を構成する必要があります。

```
mts_dispatchers="(protocol=tcp | tcps)
(presentation=oracle.aurora.server.SGiopServer |
oracle.aurora.server.GiopServer)(listener=remote_listener_name)"
```

パラメータ	説明
LISTENER (LIS または LIST)	ディスパッチャで登録されるリスナーのネットワーク名を指定する。ネーミング・メソッドで解決される名前を設定する必要があります。この属性は、リスナーがリモート・マシンに配置される場合または別の Oracle ホームにある場合にのみ指定します。 LISTENER 属性を使用すると、マルチホーム・ホストの管理が容易になります。

手順 2: リスナー名の解決

デフォルト以外のリスナーに関する情報をディスパッチャで登録するために、LOCAL_LISTENER パラメータや MTS_DISPATCHERS パラメータの LISTENER 属性に指定したリスナー名は、サーバーや Oracle Names Server 上の TNSNAMES.ORA などのネーミング・メソッドで解決できる必要があります。

エントリーは、CONNECT_DATA 部分のサービス名情報ではなく、リスナーのアドレスのみを設定します。

TNSNAMES.ORA ファイルの構成

Oracle Net8 Assistant では、CONNECT_DATA 情報を使用しない場合は TNSNAMES.ORA を構成できません。このため、TNSNAMES.ORA を手動で作成することをお勧めします。たとえば、LISTENER1 という名前のリスナーをポート 2481 で TCP/IP を使用するように構成した場合、TNSNAMES.ORA 内のエントリーは次のようになります。

```
listener1.us.oracle.com=
(description=
  (address=
    (protocol= tcp)
    (host= sales-pc)
    (port= 2481)
  )
)
```

複数アドレスがサポートされていますが、複数アドレスの connect-time failover 機能とクライアント負荷のバランス機能はサポートされていません。Oracle Connection Manager の SOURCE_ROUTE パラメータはサポートされています。

Oracle Names Server の構成

リスナーのエントリは、Oracle Net8 Assistant から Oracle Names Server を使用して作成できます。

1. Oracle Net8 Assistant を起動します。
 - UNIX の場合、`$ORACLE_HOME/bin` の下にある `netasst` を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」 -> 「Oracle - *HOME_NAME*」 -> 「Network Administration」 -> 「Net8 Assistant」を選択します。
2. Oracle Names Server フォルダを選択します。
3. Oracle Names Server をクリックします。
4. ドロップダウン・リスト・ボックスから「Manage Data」を選択します。
5. 「Advanced」タブをクリックします。
6. 「Add」をクリックし、「Name」フィールドにリスナー名、「Type」フィールドに A.SMD レコード・タイプ、「Value」フィールドに次のようなアドレスを入力します。

```
(address=(protocol= tcp)(host= sales-pc)(port= 2481))
```
7. 「File」メニューの「Save Network Configuration」を選択します。
8. 「File」メニューの「Exit」を選択して Oracle Net8 Assistant を終了します。

ディスパッチャへの直接アクセスの構成

ディスパッチャから Java オプションへの IIOP クライアント接続を構成するには、次の手順に従います。

- 手順 1: INITSID.ORA ファイルで MTS を使用可能にして IIOP を構成
- 手順 2: クライアントの構成

手順 1: INITSID.ORA ファイルで MTS を使用可能にして IIOP を構成 MTS は、INITSID.ORA ファイル内で次のように構成する必要があります。

```
mts_dispatchers="(address=(protocol=tcp | tcps)(host=server_
host)(port=port))(dispatchers=1)(presentation=oracle.aurora.server.SGiopServer |
oracle.aurora.server.GiopServer)"
```

属性の説明は次のとおりです。

パラメータ	説明
ADDRESS (ADD または ADDR)	ディスパッチャのリスニング対象となるネットワーク・アドレスを指定する。ネットワーク・アドレスには、TCP/IP (TCP) または SSL 付き TCP/IP (TCPS) プロトコル、サーバーのホスト名および GIOP リスニング・ポートを含めることができます。リスニング・ポートは、未使用の任意のポートを指定できます。
DISPATCHERS (DIS または DISP)	開始するディスパッチャの初期番号。
PRESENTATION (PRE または PRES)	GIOP のサポートを使用可能にする。 GIOP プレゼンテーション値として有効な値は次のとおりです。 <ul style="list-style-type: none"> ■ oracle.aurora.server.SGiopServer (セッションベースの GIOP ディスパッチャ接続)。このプレゼンテーションは、TCP/IP と、SSL 付き TCP/IP で有効です。 ■ oracle.aurora.server.GiopServer (通常の GIOP ディスパッチャ接続)。このプレゼンテーションは、TCP/IP と、SSL 付き TCP/IP で有効です。 詳細は、『Oracle8i Enterprise JavaBeans と CORBA 開発者ガイド』を参照してください。

手順 2: クライアントの構成 クライアントでディスパッチャに直接アクセスするために MTS を構成したのと同じポート番号を構成します。詳細は、『Oracle8i Enterprise JavaBeans と CORBA 開発者ガイド』を参照してください。

EJB と CORBA による SSL の使用

Oracle8i では、SSL を両方の GIOP プロトコル (通常の GIOP とセッションベースの GIOP) と組み合わせて使用する場合に必要なプライベート・キーなど、認証データの使用もサポートしています。GIOP で SSL を使用するには、次の 2 つの手順を実行する必要があります。

1. プロトコルとして TCPS を指定して、SSL を受け付けるようにリスナーとディスパッチャを構成します。
 - クライアントがリスナーを通じてデータベースにアクセスするように設定するには、7-54 ページの「[リスナーからアクセスするための独自構成](#)」にある手順に従います。
 - クライアントがディスパッチャを通じて直接データベースにアクセスするように設定するには、7-61 ページの「[ディスパッチャへの直接アクセスの構成](#)」にある手順に従います。これは次の方法で行うこともできます。セッションベースの IIOP および SSL 付き TCP/IP を使用する通常の IIOP のために Oracle Database Configuration Assistant で作成される `INITSID.ORA` ファイル内の次の行をコメント解除します。

```
#mts_dispatchers="(address=(protocol=tcp)(host=server_host)(port=5555))
(dispatchers=1) (presentation=oracle.aurora.server.SGIopServer)"
#mts_dispatchers="(address=(protocol=tcp)(host=server_host)(port=5556))
(dispatchers=1) (presentation=oracle.aurora.server.GiopServer)"
```

これらの行のコメントを解除する場合、クライアントが同じポート番号で構成されることを確認してください。

2. リスナーとディスパッチャの両方で、`SQLNET.ORA` のパラメータを設定して、使用する SSL Wallet を指定します。

- リスナーを構成するには次のように指定します。

```
ssl_client_authentication=false
ssl_version=undetermined
```

- ディスパッチャを構成するには次のように指定します。

```
oss.source.my_wallet=
  (source=
    (method=file)
    (method_data=
      (directory=wallet_location)))
```

SSL Wallet の構成方法は、『Oracle8i Advanced Security 管理者ガイド』を参照してください。

Oracle 以外のデータベース・サービスの構成

Net8 では、Oracle 以外にも次のデータベース・サービスへの接続をサポートしています。

- [外部プロシージャを使用する場合の Net8 の構成](#)
- [Oracle 異機種間サービスを使用する場合の Net8 の構成](#)
- [Oracle RDB データベースを使用する場合の Net8 の構成](#)

外部プロシージャを使用する場合の Net8 の構成

ネットワーク・リスナーも外部プロシージャ・コールをリスニングするように静的に構成できます。PL/SQL や SQL のアプリケーションが外部プロシージャをコールするとき、Net8 ネットワーク・リスナーにより、EXTPROC という、ネットワーク・セッション固有のプロセスが起動されます。PL/SQL からは、ネットワーク・リスナー・サービスを通じて次の情報が EXTPROC に渡されます。

- 共有ライブラリ名
- 外部プロシージャ名
- パラメータ（必要な場合）

次に、EXTPROC により共有ライブラリがロードされ、外部プロシージャが起動されます。

外部プロシージャは、インストール後にデフォルトで構成されます。

外部プロシージャ情報が存在しない環境では、LISTENER.ORA ファイルと TNSNAMES.ORA ファイルを次のように編集します。これらのファイルは、UNIX の場合は `$ORACLE_HOME/network/admin`、Windows プラットフォームの場合は `ORACLE_HOME\network\admin` にあります。

1. TCP/IP または IPC いずれかのリスナー・アドレスを構成します。たとえば、次のように指定します。

```
listener=
  (address=
    (protocol=ipc)
    (key=extproco0)
  )
```

リスナー・プロトコル・アドレスの構成方法は、6-64 ページの「[リスナー・プロトコル・アドレスの構成](#)」を参照してください。

2. サーバーの LISTENER.ORA ファイルに、PLSEXTPROC のシステム ID (SID) 名と EXTPROC のプログラム名を追加する (該当するエントリは太字で示されています)。

```
sid_list_listener_name
  (sid_list=
    (sid_desc=
      (sid_name=plsextproc)
      (oracle_home=d:\orant)
      (program=extproc)
    )
  )
)
```

パラメータ	説明
SID_NAME	外部プロシージャ識別子
ORACLE_HOME	実行可能プログラムの場所
PROGRAM	外部プロシージャ・プログラム名

リスナー・プロトコル・アドレスの構成方法の詳細は、6-71 ページの「[サービス情報によるリスナーの構成](#)」を参照してください。

3. サーバーの TNSNAMES.ORA ファイルに、EXTPROC のネット・サービス名の説明エントリを追加します。CONNECT_DATA セクションの SERVICE_NAME ではなく SID を使用します。このエントリは次のような形式で記述します。

```
extproc_connection_data=
  (description=
    (address=(protocol=ipc)(key=extproc0))

    (connect_data=
      (sid=plsextproc)
    )
  )
```

ネット・サービス名の作成方法の詳細は、6-12 ページの「[ローカル・ネーミング・メソッドによるネットワークの構成](#)」を参照してください。

Oracle 異機種間サービスを使用する場合の Net8 の構成

異機種間サービスは、Oracle8i サーバーに統合されたコンポーネントです。Oracle 以外のシステムから Oracle サーバーにアクセスできる総合テクノロジーを提供します。異機種間サービスを使用すると、次のようなことが可能です。

- Oracle SQL を使用して、データが Oracle サーバーに配置されている場合と同じように、Oracle 以外のシステムに格納されているデータに透過的にアクセスすること
- Oracle プロシージャ・コールを使用して、Oracle 分散環境から Oracle 以外のシステム、サービスまたはアプリケーション・プログラミング・インタフェース (API) に透過的にアクセスすること

異機種間サービスは Oracle8i サーバー内の総合的なテクノロジーを提供しますが、Oracle 以外の特定のシステムにアクセスするには異機種間サービス・エージェントが必要です。

Oracle 以外のシステムへの接続を開始するには、Oracle8i サーバーがゲートウェイ上で Net8 リスナーを通じてエージェント・プロセスを開始します。Oracle8i サーバーがエージェントに接続できる場合、次の手順を実行します。

1. Oracle8i データベースで、ゲートウェイ上のリスナーに接続するためのネット・サービス名を設定します。接続で異機種間サービスを利用できるように、接続記述子にも HS=OK 句を設定する必要があります。
 - a. Oracle Net8 Assistant を起動します。
 - UNIX の場合、\$ORACLE_HOME/bin の下にある `netasst` を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」-> 「Oracle - HOME_NAME」-> 「Network Administration」-> 「Net8 Assistant」を選択します。
 - b. Oracle8i サーバーから Oracle 以外のシステムに接続するために使用するネット・サービス名を作成します。

6-14 ページの「[手順 1: ネット・サービス名を構成する](#)」を参照してください。
 - c. ディレクトリ・ツリーで、ネット・サービス名フォルダをダブルクリックします。
 - d. ネット・サービス名を選択します。
 - e. 「Service Identification」グループ・ボックスの「Advanced」をクリックします。

「Advanced Service Options」ダイアログ・ボックスが表示されます。



- f. 「Use for Heterogeneous Service」をクリックし、「OK」をクリックします。
- g. 「File」メニューの「Save Network Configuration」を選択します。

TNSNAMES.ORA ファイルが再作成されます。

次の例では、TNSNAMES.ORA 内のネット・サービス名が異機種間サービスを使用するように構成されています。

```
megabase6_sales=
  (description=
    (address=
      (protocol=tcp)
      (host=dlsun206)
      (port=1521)
    )
    (connect_data=
      (service_name=sales6)
      (hs=ok)
    )
  )
)
```

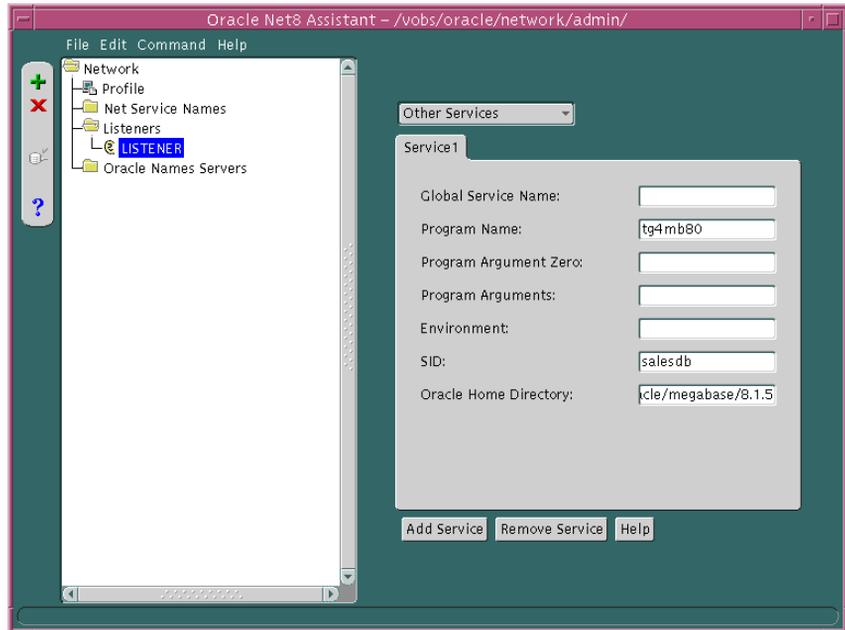
2. ゲートウェイ上のリスナーは、Oracle8i サーバーからの受信要求をリスニングし、異機種間サービス・エージェントを生成するように設定する必要があります。LISTENER.ORA ファイルは、リスナーが異機種間サービス・エージェントを起動するように変更する必要があります。また、リスナーを（再）起動する必要があります。LISTENER.ORA に設定しなければならない機能は次のとおりです。

Oracle Net8 Assistant オプション	LISTENER.ORA パラメータ	説明
「Program Name」	PROGRAM	実行可能エージェントの名前を定義する。
「Oracle Home」	ORACLE_HOME	実行可能エージェントの場所。実行可能エージェントは、\$ORACLE_HOME/bin ディレクトリに常駐させる必要があります。
「SID」	SID_NAME	Oracle 以外のデータベースを表す Oracle システム識別子 (SID) を指定する。

リスナーに新しい名前を付けるには、Oracle Net8 Assistant を使用します。

- a. Oracle Net8 Assistant で、リスナー・フォルダをダブルクリックします。
- b. リスナーをクリックします。
- c. ドロップダウン・リスト・ボックスから「Other Services」を選択します。

- d. 「Add Service」をクリックします。



- e. ゲートウェイの作成を実行するプログラム名、実行可能プログラムの Oracle ホーム、および Oracle 以外のシステムの SID かサービス名を入力します。
- f. 「File」メニューの「Save Network Configuration」を選択します。

LISTENER.ORA ファイルが再作成されます。

SID_LIST_listener_name セクションに、異機種間サービスを使用するように構成された LISTENER.ORA ファイルの例を示します。

```
sid_list_listener=(sid_list=
  (sid_desc=
    (sid_name=salesdb)
    (oracle_home=/home/oracle/megabase/8.1.5)
    (program=tg4mb80)
  )
)
```

- g. 「File」メニューの「Exit」を選択します。

Oracle Net8 Assistant アプリケーションが終了します。

異機種間サービスの詳細は、『Oracle8i 分散システム』を参照してください。

Oracle RDB データベースを使用する場合の Net8 の構成

Oracle RDB は、Digital の 64 ビット・プラットフォームで使用されるデータベースです。Net8 を使用する場合、Oracle RDB サーバーはクライアントにとって Oracle8i データベースと同じように見えます。Oracle RDB には独自のリスナーがあるので、クライアントは Oracle7 と通信するのと同じように Oracle RDB と通信できます。

Oracle RDB との接続を開始するために、Oracle RDB データベースに接続するネット・サービス名を次の機能を使用して設定します。

Oracle Net8 Assistant オプション	TNSNAMES.ORA パラメータ	説明
「RDB Database」	RDB_DATABASE	Oracle RDB データベースの名前を指定する。このパラメータは CONNECT_DATA セクションに記述します。
「Type of Service」	TYPE_OF_SERVICE	Oracle RDB データベースで使用されるサービスのタイプを指定する。RDB インタフェース・ツールで使用されます。この機能は、アプリケーションが Oracle RDB と Oracle データベースの両方をサポートし、アプリケーションを任意に選択する（負荷のバランス）場合にのみ使用します。このパラメータは DESCRIPTION セクションに記述します。
「Global Database Name」	GLOBAL_NAME	Oracle RDB データベースを識別する。省略可能な機能です。このパラメータは CONNECT_DATA セクションに記述します。

構成情報の詳細は、Oracle RDB のマニュアルを参照してください。

Oracle RDB データベースを使用するようにクライアントを構成するには、Oracle Net8 Assistant を使用します。

- a. Oracle Net8 Assistant を起動します。
 - UNIX の場合、`$ORACLE_HOME/bin` の下にある `netasst` を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」-> 「Oracle - HOME_NAME」-> 「Network Administration」-> 「Net8 Assistant」を選択します。
- b. クライアントから Oracle RDB への接続に使用するネット・サービス名を作成します。

6-14 ページの「手順 1: ネット・サービス名を構成する」参照。
- c. ディレクトリ・ツリーで、ネット・サービス名フォルダをダブルクリックします。
- d. ネット・サービス名を選択します。
- e. 「Service Identification」グループ・ボックスの「Advanced」をクリックします。「Advanced Service Options」ダイアログ・ボックスが表示されます。



- f. 「Rdb Database」フィールドに Oracle RDB データベースのファイル名を入力します。
- g. 必要に応じて、「Global Database Name」フィールドにグローバル・データベース名を入力し、「Type of Service」フィールドにサービスのタイプを指定します。これは省略可能です。次に、「OK」をクリックします。
- h. 「File」メニューの「Save Network Configuration」を選択します。

TNSNAMES.ORA ファイルが再作成されます。

次の例では、TNSNAMES.ORA 内のネット・サービス名が Oracle RDB データベースを使用するように構成されています。

```
alpha5=
  (description=
    (address= ...)
    (connect_data=
      (service_name=generic)
      (rbd_database= [.mf]mf_personnel.rdb)
      (global_name= alpha5)))
```

次の例では、TYPE_OF_SERVICE を使用して、Oracle RDB データベース・サービスや Oracle8i データベース・サービスの負荷のバランスをとるように構成されています。

```
alpha5=
  (description_list=
    (description=
      (address= ...)
      (connect_data=
        (service_name=generic)
        (rbd_database= [.mf]mf_personnel.rdb)
        (global_name= alpha5)))
    (description=
      (address= ...)
      (connect_data=
        (service_name=sales.com))
      (type_of_service=oracle8_database))
```

接続の確立とネットワークのテスト

ネットワークの構成が終了したら、接続を確立して各コンポーネントをテストし、ネットワークが正しく機能しているかどうかを確認する必要があります。Net8 には、Oracle Names Server、リスナー、Connection Manager を容易に起動、テストおよび制御できる各種のツールがあります。

この章では、Net8 の制御ユーティリティを使用してネットワークの確立とネットワーク・コンポーネントのテストを行う手順について説明します。内容は次のとおりです。

- データベースへの接続
- 起動と接続の一般的な問題を解決するためのチェックリスト
- ネットワークのテスト

データベースへの接続

データベースに接続するには、ネットワーク・コンポーネントを起動し、`connect username/password@net_service_name` のように、ネット・サービス名を指定した接続文字列を入力する必要があります。

この項の内容は、次のとおりです。

- [Net8 コンポーネント起動の概要](#)
- [接続文字列の入力](#)

Net8 コンポーネント起動の概要

クライアント・ワークステーションとその他のサーバーは、Oracle サーバーにログインするときにネット・サービス名を使用してリスナーに接続します。

すべてのネットワーク・コンポーネントをインストールし、構成した後は、それを起動してネットワークを機能させる必要があります。起動に必要な手順は、次のとおりです。

1. ネットワークで Oracle Names を使用する場合、Oracle Names Server を起動してすべてのクライアント上の Oracle Names Server を検出し、クライアントのキャッシュを実行します。

注意： Oracle Names Server がデータベースを使用してネットワーク情報を格納する場合は、最初にデータベースを起動し、次にリスナーを起動する必要があります。

2. リスナー制御ユーティリティ LSNRCTL を使用してリスナーを起動します。
3. 選択したツールを使用して、データベースを起動します。
4. ネットワークに Oracle Connection Manager が含まれている場合は、接続マネージャ制御ユーティリティ CMCTL を使用して Oracle Connection Manager を起動します。

手順 1: Oracle Names Server の起動

Oracle Names Server ソフトウェアがインストールされ構成されているマシンで、Oracle Net8 Assistant か NAMESCTL 制御ユーティリティを使用して、Oracle Names Server を起動します。

制御ユーティリティ NAMESCTL を使用する場合 Oracle Net8 Assistant を使用する場合

コマンド行から、次のように入力します。

```
namesctl
namesctl> start
```

NAMESCTL の STARTUP コマンドにより Oracle Names Server がメモリにロードされ、実行開始の指示が出されます。Oracle Names Server は、起動時に構成とデータをロードします。

1. Oracle Net8 Assistant を起動します。
-UNIX の場合、\$ORACLE_HOME/bin の下にある netasst を実行します。
-Windows NT の場合、「スタート」ボタン -> 「プログラム」-> 「Oracle - HOME_NAME」-> 「Network Administration」-> 「Net8 Assistant」を選択します。
 2. Oracle Names Server フォルダをダブルクリックします。
 3. Oracle Names Server を選択します。
 4. ドロップダウン・リスト・ボックスから「Manage Server」を選択します。
 5. 「Control」タブをクリックします。
 6. 「Server Operations」フィールドの「Start」ラジオ・ボタンをクリックします。
 7. 「Apply」をクリックします。
-

Windows NT の場合、Oracle Names Server はコントロールパネルから起動することもできます。

1. 「コントロールパネル」ウィンドウの「サービス」アイコンをダブルクリックします。
2. 「OracleHOME_NAMENamesonames_serverService」サービスを選択します。
3. 「開始」をクリックしてサービスを開始します。
4. 「サービス」ウィンドウの「閉じる」をクリックします。

手順 2: Oracle Names クライアント・キャッシュの開始

クライアントが Oracle Names Server 内のアドレス情報を何度も検索しなくすむように、クライアントに情報のキャッシュを作成します。この情報は、CKPCCH.ORA に格納されています。このファイルは、UNIX では \$ORACLE_HOME/network/names、Windows プラットフォームでは ORACLE_HOME¥network¥names に配置されています。

クライアント・キャッシュを作成するには、次の手順に従います。

1. 6-36 ページの「[手順 4: Oracle Names Server を使用するためにクライアントとデータベース・サーバーを構成する](#)」にある説明に従って、Oracle Names Server の検出が実行済みであることを確認します。検出中に収集された情報は、クライアントのキャッシュで使用されます。
2. クライアント上にクライアント・キャッシュを作成して起動します。

```
namesctl  
namesctl> start_client_cache
```

Windows NT の場合、クライアント・キャッシュはコントロールパネルから起動することもできます。

1. 「コントロールパネル」ウィンドウの「サービス」アイコンをダブルクリックします。
2. 「OracleHOME_NAMEClientCache」サービスを選択します。
3. 「開始」をクリックしてサービスを開始します。
4. 「サービス」ウィンドウの「閉じる」をクリックします。

手順 3: リスナーの起動

Net8 がサーバー上の接続を受け付ける場合、サーバーで LSNRCTL 制御ユーティリティを使用して、リスナーを起動する必要があります。

1. コマンド行から、次のように入力します。

```
lsnrctl
lsnrctl> status [listener_name]
```

listener_name は、LISTENER.ORA ファイルに定義されたリスナーの名前です。LISTENER という名前のデフォルトのリスナーを使用している場合は、リスナーを識別する必要はありません。

STATUS コマンドでリスナーが稼働していることが示される場合、手順 2 に進みます。リスナーが稼働していない場合、手順 3 に進みます。

2. リスナーが稼働していない場合でも、リスナーを停止して再起動することをお勧めします。リスナーを停止するには、次のように入力します。

```
lsnrctl> set password password
lsnrctl> stop [listener_name]
```

SET PASSWORD が必要なのは、LISTENER.ORA ファイルにパスワードが設定されている場合のみです。デフォルトのパスワードは、ORACLE です。

3. リスナーを起動します。次のように入力します。

```
lsnrctl> start [listener_name]
```

LSNRCTL は、リスナーが正常に起動されたことを示すステータス・メッセージを表示します。そのリスナーで期待されるすべてのサービスがステータス・メッセージ内のサービス・サマリーの中に記述されていることを確認します。

4. LSNRCTL ユーティリティを終了します。次のように入力します。

```
lsnrctl> exit
```

Windows NT の場合、リスナーはコントロールパネルから起動することもできます。

1. 「コントロールパネル」ウィンドウの「サービス」アイコンをダブルクリックします。
2. 「OracleHOME_NAME_TNSListener」サービス（デフォルトのネットワーク・リスナー名である LISTENER を使用している場合のサービス名）または「OracleHOME_NAME_TNSListenerlsnr」（*lsnr* はデフォルト以外のネットワーク・リスナー名）を選択します。
3. 「開始」をクリックしてサービスを開始します。
4. 「サービス」ウィンドウの「閉じる」をクリックします。

手順 4: データベースの起動

SQL*Plus などのツールを使用して、データベースを起動します。

1. 次のように入力して、データベースに接続せずに SQL*Plus を起動します。

```
sqlplus /nolog
```

2. SYSDBA として Oracle に接続します。

```
sql> connect username/password as sysdba
```

3. STARTUP コマンドを入力するときに、データベース名とパラメータ・ファイルのフル・パスを指定します。

```
sql> startup database_name pfile=initsid.ora
```

PFILE オプションを指定しない場合、標準のパラメータ・ファイルの場所が使用されます。この場所は、UNIX プラットフォームでは \$ORACLE_BASE/admin/db_name/pfile/sid、Windows NT では ORACLE_BASE¥admin¥db_name¥pfile¥sid です。データベース名を指定しない場合、そのインスタンスを起動するパラメータ・ファイル内の DB_NAME の値が使用されます。

データベースの起動方法の詳細は、『Oracle8i 管理者ガイド』を参照してください。

手順 5: Oracle Connection Manager の起動

Oracle Connection Manager がインストールされ構成されている場合、CMCTL 制御ユーティリティを使用して Oracle Connection Manager を起動します。

1. コマンド行から、次のように入力します。

```
cmctl  
cmctl> start cman
```

CMCTL は、Oracle Connection Manager が正常に起動されたことを示すステータス・メッセージを表示します。

2. CMCTL ユーティリティを終了します。次のように入力します。

```
cmctl> exit
```

Windows NT の場合、リスナーはコントロールパネルから起動することもできます。

1. 「コントロールパネル」ウィンドウの「サービス」アイコンをダブルクリックします。
2. Oracle Names を使用している場合、使用可能な Oracle Names Server についての情報を取得するために「OracleHOME_NAMECMAdminService」を選択し、「Start」をクリックします。Oracle Names を使用していない場合、このサービスは起動しないでください。
サービスが起動します。
3. Oracle Connection Manager を起動するために「OracleHOME_NAMECManService」を選択し、「Start」をクリックします。
サービスが起動します。
4. 「Services」ウィンドウの「Close」をクリックします。

接続文字列の入力

ネットワーク・コンポーネントを起動したら、8-2 ページの「[データベースへの接続](#)」にある説明に従って、ネットワークを介した接続を確立できます。接続の確立方法は、[第 6 章の「ネーミング・メソッドとリスナーの構成」](#)で構成したネーミング・メソッドと、接続に使用するツールによって異なります。基本の接続文字列は次のとおりです。

設定する機能	入力する文字列
ホスト・ネーミング・メソッド	sql> CONNECT <i>username/password@alias</i> 別名には通常、IP アドレスの解決方式でのグローバル・データベース名を設定する。クライアントとサーバーが同じドメイン内に存在する場合、入力する必要があるのはグローバル・データベース名のデータベース名のみです。
ローカル・ネーミング・メソッド	sql> CONNECT <i>username/password@net_service_name</i>
Oracle Names 方法	sql> CONNECT <i>username/password@net_service_name</i>
bequeath プロトコル	sql> CONNECT <i>username/password</i>

Oracle Server への接続を開始する方法は多数あります。一般に使用されている方法は、次のとおりです。

- [クライアントをテストするためのオペレーティング・システムからの接続](#)
- [クライアントをテストするためのツール・ログオン画面からの接続](#)
- [クライアントをテストするための 3GL からの接続](#)
- [Tools で特殊コマンドによる接続](#)

具体的な使い方は、それぞれの方法によって多少の違いがあります。ここでは、前述した一般的な方法について簡単に説明します。特定のツールでの使用方法は、そのツールのユーザーズ・ガイドを参照してください。

クライアントをテストするためのオペレーティング・システムからの接続

コマンド行からアプリケーションをデータベース・サーバーに接続させるための一般的な書式を次に示します。

```
tool username/password@net_service_name
```

```
sqlplus system/password@sales
```

ログオン時にパスワードを表示しないようにするには、次のようにコマンド行でパスワード・パラメータを入力しないでおきます。

```
sqlplus system@sales
```

パスワードの入力を求めるメッセージが表示されますが、画面上にパスワードは表示されません。

ほとんどの Oracle Tools は、オペレーティング・システムのコマンド行を使用して接続できますが、他の方法でも接続できます。

クライアントをテストするためのツール・ログオン画面からの接続

一部のツールには、ログオンの代替形式としてログオン画面が用意されています。ユーザーがデータベース・サーバーにログオンするには、ツール・ログオン画面のユーザー名フィールドにユーザー名とネット・サービス名 (*username@net_service_name*) の両方を指定し、パスワード・フィールドに通常どおりにパスワードを入力します。

クライアントをテストするための 3GL からの接続

3GL で作成されたアプリケーションでは、プログラムは次の構文でサーバーへの接続を確立する必要があります。

```
exec sql connect :username identified by :password
```

この接続要求では、*:username* と *:password* は、静的にまたはユーザーの要求でプログラム内に設定できる 3GL 変数です。データベース・サーバーに接続する場合、*:username* 変数の値は、次の書式になります。

```
username@net_service_name
```

:password 変数には、接続先のデータベース・アカウントのパスワードが含まれています。

Tools で特殊コマンドによる接続

一部の Oracle Tools には、そのツールが起動された後、ツールから離れずに代替ユーザー名を使用してデータベースに接続できるコマンドがあります。SQL*Plus では、次の構文で CONNECT コマンドを使用できます。

```
sql> CONNECT username/password@net_service_name
```

たとえば、次のように指定します。

```
sql> CONNECT scott/tiger@serverx
```

これは、オペレーティング・システムのプロンプトではなくツールのプロンプトに応答して入力するという点以外は、オペレーティング・システムのコマンド行を使用する方法と同じです。

他の Oracle Tools では、この説明とは異なる各自の機能やインタフェースに固有の方法が使用されます。たとえば、Oracle CDE Tools では、ログオン・ボタンとユーザー名およびパスワード、リモート・データベース ID フィールドがあるポップアップ・ウィンドウが使用されます。特定のツールによる Oracle への接続の詳細は、そのツールのユーザーズ・ガイドを参照してください。

起動と接続の一般的な問題を解決するためのチェックリスト

次のチェックリストは、Net8 コンポーネントの起動時や接続の確立時に発生する可能性のある一般的な問題点を容易に解決できるようにするためのものです。

問題	解決策
コンポーネントがアクティブになっていない	<ol style="list-style-type: none"> 1. トランスポート・レイヤー・プロトコルとプロトコルをインストールしたことを確認します。 2. 接続するサーバーのリスナーをすべて起動したことを確認します。 3. プロトコル間にセッションを送る場合は、Oracle Connection Manager を起動します。
構成ファイル内の構文エラー	<ol style="list-style-type: none"> 1. 構文エラーの回避には、できるだけ Oracle Net8 Assistant を使用します。構文ファイルを手動で作成または編集する場合は、すべての対応するカッコが所定の位置にあるかどうか、論理構造を示すための行がインデントされているか、入力ミスがないかどうかを注意深くチェックします。 追加情報：構成ファイルの構文の詳細は、C-2 ページの「構成ファイルの構文規則」を参照してください。 2. すべてのネット・サービス名が、すべてのローカル名構成ファイルで接続記述子にマップされていることを確認します。 3. 無効なリスナー名が LSNRCTL START コマンドに入力されていないことを確認します。 4. 入力を確認します。使用しているリスナー名が、リスナー構成ファイル (LISTENER.ORA) で指定された名前と一致していることを確認します。

問題	解決策
ファイルが正しく配置されていない	<ol style="list-style-type: none"> 1. 構成ファイルが見つからないためリスナーが起動できないという指示が出ます。すべての構成ファイルは通常、UNIX では \$ORACLE_HOME/network/admin、Windows NT では ORACLE_HOME¥network¥admin に配置されています。ただし、環境変数やレジストリ値の TNS_ADMIN には別のディレクトリを設定できます。\$TNS_ADMIN を異なるディレクトリに設定すると、Net8 はそのディレクトリ内に構成ファイルが存在すると仮定します。次に、Net8 は ORACLE_HOME ディレクトリ内で構成ファイルを探します。ORACLE_HOME ディレクトリは、オペレーティング・システムによって異なります。ファイル内の位置を示す最も良い方法は、SQLNET.LOG ファイル内を調べるか、またはトレーシングをオンに切り替えるように要求した場合はトレース・ファイル内のヘッダー情報を調べる方法です。 2. Oracle Names Server を使用している場合は、ネーム・サーバーが起動されていることを確認します。また、Names List ファイルが存在するようにデフォルトの Oracle Names Server パラメータが SQLNET.ORA に含まれていることを確認します。 3. NIS などの外部ネーミング・サービスを使用している場合は、クライアントとサーバー上に適切な外部ネーミング・メソッドがインストールされていること、その中にネット・サービス名が正しくロードされていることを確認します。詳細は、オペレーティング・システム固有のマニュアルを参照してください。
アドレスがすでに使用されている	<p>すでに別のプロセスが、リスナー構成ファイル (LISTENER.ORA) に示されているアドレスを使用している可能性があります。TCP/IP、DECnet、OSI などのプロトコルでは、ノード上のネットワーク・サービスは、それぞれ一意のポートまたはソケットを使用する必要があります。SPX や NetBIOS などのネットワーク・プロトコルでは、ネット・サービス名は、ネットワーク全体で一意的のものである必要があります。別のネットワーク・サービスが同じ構成を使用している可能性もあります。ネットワーク管理者に連絡して、ネットワーク・アドレスが利用可能であることを確認してください。</p>
データベースに接続しようとすると、メッセージ「ORA-12203: TNS:Unable to connect to destination」が表示される	<p>リスナー制御ユーティリティ (LSNRCTL) を使用してリスナーを起動します。</p>

問題	解決策
<p>クライアントから接続しよう とすると、メッセージ 「ORA-12154: TNS:Could not resolve service name」が表示される</p>	<ul style="list-style-type: none"> ■ 要求したネット・サービス名が、Oracle Names Server、外部 ネーミング・メソッドで定義されていないか、またはロー カル・ネーミングファイルを予定の場所から検出できませ ん。 ■ Oracle Names を使用している場合、この問題は Oracle Names Server の定義上の問題である可能性があります。ネーミン グ・メソッドをリストするための NAMES.DIRECTORY_ PATH パラメータが、SQLNET.ORA に含まれていることを 確認します。 ■ サーバーが動作していません。 ■ Oracle Names Server リスト・ファイルがありません。 NAMESCTL から REORDER_NS コマンドを発行します。 ■ NAMES.PREFERRED_SERVER パラメータが、 SQLNET.ORA ファイルに正しく構成されていません。パラ メータが SQLNET.ORA に正しく構成されているかどうかを 確認します。
<p>データベースに接続しようと すると、メッセージ 「ORA-1034: Oracle Not Available」または 「ORA-12505: Listener could not resolve SID given in connect descriptor」が表示される</p>	<p>このデータベースは、サーバー・マシン上で実行されていま せん。リスナーのみではデータベース接続はできません。デー タベース・インスタンスも起動されている必要があります。</p>
<p>クライアントからメッセージ 「ORA-12541: No Listener」が戻る</p>	<p>着信が速すぎてリスナーが処理できない接続要求と、リスナー の未処理分 (LISTENER.ORA 内の QUEUESIZE パラメータで決 定される) を超えている接続要求には、「Connection Refused」エラーが戻されます。クライアントにこのエラーが 発生すると、「ORA-12541: No Listener」というメッセージ が戻され、クライアントのログ・ファイルやトレース・ファイ ルには、「Connection Refused」というメッセージが表示さ れます。</p> <p>この問題を修正するには、次の手順を実行します。</p> <ol style="list-style-type: none"> 1. リスナーを停止します。 2. リスナー (LISTENER.ORA) 構成ファイルの QUEUESIZE をもっと大きい値 (予想される同時接続要求に基づく) に 再編成します。 3. リスナーを再起動します。 4. 接続を再試行します。

問題	解決策
リスナーを停止しようとする と、「TNS-01169: The listener has not recognized the password」というメッセー ジが表示される	リスナー制御ユーティリティ (LSNRCTL) 内から SET PASSWORD コマンドを入力し、次にリスナーを停止するために STOP コマンドを入力します。

ネットワークのテスト

ネットワークをテストする望ましい順序は、次のとおりです。

1. 各 Oracle Names Server (ネットワーク・レイアウトに含まれている場合) を起動し、テストします。
2. リスナーを個別に起動し、テストします。
3. 各 Oracle Connection Manager (ネットワーク・レイアウトに含まれている場合) を起動し、テストします。
4. Server でループバック・テストを行います。
5. クライアントで接続できるかどうかテストします。

この項の内容は、次のとおりです。

- [Net8 制御ユーティリティの使用](#)
- [Oracle Names Server のテスト](#)
- [NAMESCTL または Oracle Net8 Assistant によるネットワーク・オブジェクトのテスト](#)
- [リスナーのテスト](#)
- [Oracle Connection Manager のテスト](#)
- [サーバーでの構成のテスト](#)
- [クライアントからのネットワーク接続のテスト](#)

Net8 制御ユーティリティの使用

Net8 には、各ネットワーク・コンポーネントを容易に起動、テスト、制御するために、次のユーティリティがあります。

- Oracle Names 制御ユーティリティ - NAMESCTL
- リスナー制御ユーティリティ - LSNRCTL
- Connection Manager 制御ユーティリティ - CMCTL

Net8 のコンポーネント制御ユーティリティおよびそれらのコマンドの詳細は、[付録 A の「リスナー、Oracle Names Server および Oracle Connection Manager の制御ユーティリティ」](#)を参照してください。

Oracle Names 制御ユーティリティ (NAMESCTL) の使用

Oracle Names 制御ユーティリティ (NAMESCTL) は、Oracle Names Server を起動および制御するためにオペレーティング・システム・プロンプトから実行するツールです。

Oracle Names 制御ユーティリティの一般的な書式は、次のとおりです。

```
namesctl command
```

プログラム・プロンプトで、NAMESCTL コマンドを発行することもできます。コマンド行で NAMESCTL を入力すると、プログラムがオープンします。次に、プログラム・プロンプトから必要なコマンドを入力します。たとえば、次のコマンドは、Oracle Names Server を起動します。

```
namesctl> start
```

リスナー制御ユーティリティ (LSNRCTL) の使用

リスナー制御ユーティリティ (LSNRCTL) は、リスナーを起動および制御するためにオペレーティング・システム・プロンプトから実行するツールです。リスナー制御ユーティリティの一般的な書式は、次のとおりです。

```
lsnrctl command [listener_name] [args]
```

プログラム・プロンプトからリスナー制御コマンドを発行することもできます。コマンド行に LSNRCTL と入力すると、プログラムがオープンします。次に、プログラム・プロンプトから必要なコマンドを入力します。たとえば、次のコマンドは、接続が開始されてからリスナーが有効な接続要求を待つ時間 (秒単位) を決定します。

```
lsnrctl> set connect_timeout 20
```

Connection Manager 制御ユーティリティ (CMCTL) の使用

Connection Manager 制御ユーティリティ (CMCTL) は、Oracle Connection Manager を起動し、制御するためにオペレーティング・システム・プロンプトから実行するツールです。Connection Manager 制御ユーティリティの一般的な書式は、次のとおりです。

```
cmctl command
```

プログラム・プロンプトで、CMCTL コマンドを発行することもできます。コマンド行に CMCTL と入力すると、プログラムがオープンします。次に、プログラム・プロンプトから必要なコマンドを入力します。たとえば、次のコマンドは Oracle Connection Manager を起動します。

```
cmctl> start
```

Oracle Names Server のテスト

Oracle Names Server をテストするには、NAMESCTL PING コマンドを使用します。例として、US.ACME ドメイン内の LABRADOR という Oracle Names Server を PING でテストする 2 つの方法を示します。

NAMESCTL プロンプトから、次のように入力します。

```
namesctl> ping labrador.us.acme
```

次のように入力すると、同じ PING コマンドで複数の Oracle Names Server をテストできます。

```
namesctl> ping huey.uk.acme duey.uk.acme louie.uk.acme
```

Oracle Names Server に接触するためにかかる時間が出力され、肯定応答が戻ります。PING が失敗した場合は、Oracle Names Server が起動されていることを確認するか、Oracle Names Server の構成アドレスを再確認します。

NAMESCTL または Oracle Net8 Assistant によるネットワーク・オブジェクトのテスト

Oracle Names Server に格納されている情報（6-21 ページの「[Oracle Names ネーミング・メソッドによるネットワークの構成](#)」）は、QUERY コマンドで登録を確認するように問い合わせることができます。

QUERY コマンドを使用するには、Oracle Net8 Assistant か NAMESCTL 制御ユーティリティのどちらかを使用します。

Oracle Net8 Assistant を使用する場合	制御ユーティリティ NAMESCTL を使用する場合
<p>1. Oracle Net8 Assistant を起動します。</p> <p>-UNIX の場合、\$ORACLE_HOME/bin の下にある netasst を実行します。</p> <p>-Windows NT の場合、「スタート」ボタン -> 「プログラム」-> 「Oracle - HOME_NAME」-> 「Network Administration」-> 「Net8 Assistant」を選択します。</p> <p>2. Oracle Names Server フォルダをダブルクリックします。</p> <p>3. Oracle Names Server を選択します。</p> <p>4. ドロップダウン・リスト・ボックスから「Manage Data」を選択します。</p> <p>5. 「Advanced」タブをクリックします。</p> <p>6. 「Query」を選択します。</p> <p>7. 「Name」フィールドに、問い合わせるオブジェクトの名前を入力します。</p> <p>8. 必要に応じて、「Name」フィールドにオブジェクトのタイプを入力します。</p> <p>-A.SMD: ネット・サービス名のデータベース・ネットワーク定義など、ネットワーク・アドレス。</p> <p>-CNAME.SMD: 別名。</p> <p>-DL.RDBMS.OMD: グローバル・データベース・リンク。</p> <p>-DLCR.RDBMS.OMD: リンク修飾子。</p> <p>-NS.SMD: Oracle Names Server のアドレス。Oracle Names Server 間で通信する際に使用するシステム・データ。</p> <p>-VIADD.NPO.OMD: SQL*Net バージョン 1 の接続文字列。</p> <p>9. 「Execute」を選択します。</p> <p>10. 「File」メニューの「Save Network Configuration」を選択します。</p> <p>11. 「File」メニューの「Exit」を選択して Oracle Net8 Assistant アプリケーションを終了します。</p>	<p>Oracle Names Server が管理領域に配置されている場合、Oracle Names Server に別名を作成して登録します。</p> <p>全データを表示するには、次のように入力します。</p> <pre>namesctl namesctl> query name *</pre> <p>取り出す情報のタイプも指定できます。一般的なオブジェクトのタイプは次のとおりです。</p> <ul style="list-style-type: none"> ■ A.SMD: ネット・サービス名のデータベース・ネットワーク定義など、ネットワーク・アドレス ■ CNAME.SMD: 別名 ■ DL.RDBMS.OMD: グローバル・データベース・リンク ■ DLCR.RDBMS.OMD: グローバル・リンク修飾子 ■ NS.SMD: Oracle Names Server のアドレス ■ VIADD.NPO.OMD: SQL*Net バージョン 1 の接続文字列 <p>ネット・サービス名のデータベース・アドレスを表示するには、次のように入力します。</p> <pre>namesctl namesctl> query name a.smd</pre> <p>次の例は、ネット・サービス名 SALES への問合せです。</p> <pre>namesctl namesctl> query sales a.smd</pre> <p>QUERY コマンドは、トランザクションに要した時間とネットワーク・オブジェクトに関する情報を戻します。</p>

リスナーのテスト

リスナーをテストするには、8-19 ページの「[サーバーでの構成のテスト](#)」に説明されているように、そのリスナーによって制御されるアクティブ・データベースに対してクライアントから接続を開始します。リスナーにアクセスできるクライアントのみ別のプロトコルを使用する場合は、Oracle Connection Manager を使用してリスナーにアクセスする必要があります。

Oracle Connection Manager のテスト

Oracle Connection Manager をテストするには、ソース・ルート・アドレスが作成されたアクティブなデータベースに対して、クライアントから接続を開始します。

サーバーでの構成のテスト

ネットワークを構成し終わったら、サーバーでループバック・テストを実行して、構成をテストします。

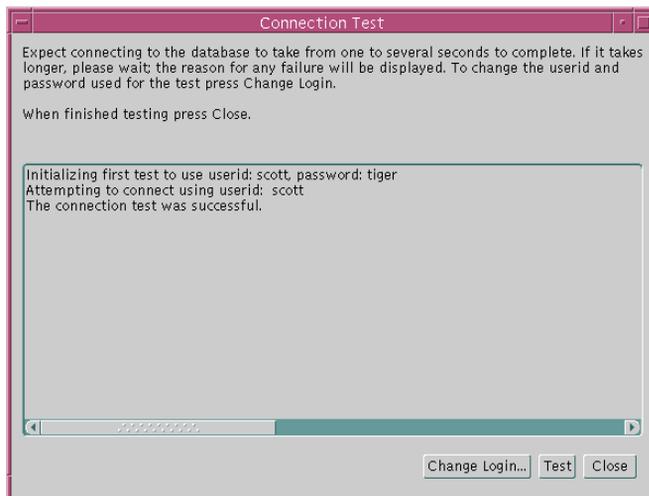
ループバック・テストでは、Net8 を使用してサーバーからプロセス間通信 (IPC) を利用せずにサーバー自体に戻ります。ループバックが正常に終了すると、Net8 がサーバー側で機能していることが確認できます。

ループバック・テストを実行するには、Oracle Net8 Assistant を使用します。

1. Oracle Net8 Assistant を起動します。
 - UNIX の場合、`$ORACLE_HOME/bin` の下にある `netasst` を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」-> 「Oracle - *HOME_NAME*」-> 「Network Administration」-> 「Net8 Assistant」を選択します。
2. ディレクトリ・ツリーで Net Service Names フォルダをクリックします。
3. ネット・サービス名を選択します。
4. 「Command」メニューから「Test Net Service Name」を選択します。

このテストは、リスナーが稼働中であることを前提としています。リスナーが稼働していない場合、8-2 ページの「[Net8 コンポーネント起動の概要](#)」を参照してコンポーネントを起動してください。

テストが正常終了すると、「Connect Test」ダイアログ・ボックスに「The connect test was successful.」というメッセージが表示されます。



テストが成功しなかった場合は、次の処置をとってください。

- データベースとリスナーが稼働していることを確認し、「Test」をクリックします。
 - 「Change Login」をクリックして接続のユーザー名とパスワードを変更し、「Test」をクリックします。
5. 「Close」をクリックして「Connect Test」ダイアログ・ボックスを閉じます。

クライアントからのネットワーク接続のテスト

ネットワーク内でいくつかの異なるクライアントをテストするには、8-8 ページの「[接続文字列の入力](#)」にある手順に従って、それらの各クライアントからサーバーに接続を開始します。

また、Net8 にはネットワーク接続を容易に評価するために、次のツールがあります。

- [TNSPING](#)
- [TRCROUTE](#)
- [Oracle Net8 Assistant](#)

TNSPING

TNSPING は、Net8 ネットワーク上のサービス（Oracle データベースまたは Oracle Names Server、その他の Oracle サービスなど）に正常に到達できるかどうかを判断するためのユーティリティです。

TNSPING を使用してクライアントからサーバーに（または、あるサーバーから別のサーバーに）正常に接続できた場合は、Net8 サービスに到達するために必要な往復時間（ミリ秒単位）の概算が表示されます。

接続に失敗した場合は、発生したエラーを記述するメッセージが表示されます。これによって、データベース接続に対するオーバーヘッドなしで発生中のネットワーク・エラーを参照できます。

TNSPING の使用 次のような手順で、TNSPING ユーティリティを呼び出します。

```
tnsping net_service_name [count]
```

注意： プラットフォームによってインタフェースが異なる場合がありますが、プログラムは同じ引数を受け入れられます。TNSPING を呼び出すと、正しいインタフェース要件が表示されます。

- *net service name*: TNSNAMES.ORA、または NIS や DCE の CDS などのネーム・サービスに存在するものでなければなりません。
- *count* (省略可): プログラムがサーバーへの到達を試行する回数を指定します。

指定されたサービス名がデータベース名の場合、TNSPING は、該当するリスナーへの接触を試行します。データベース自体が実行されているかどうかは、事実上判別されません。データベースへの接続を行うには、Server Manager を使用します。

この後、TNSPING の例をいくつか示します。

データベースへの到達 SPOTDB というネット・サービス名を使用してデータベースに接続するには、次のように入力します。

```
tnsping spotdb
```

これによって、次のメッセージが出力されます。

```
TNS Ping Utility for SunOS:  
Copyright (c) Oracle Corporation 1998. All rights reserved.  
Attempting to contact  
(ADDRESS=(PROTOCOL=TCP)(HOST=spot)(PORT=1521))  
OK (50msec)
```

10 回の試行で STPRD データベースに接続できるかどうかを判別するには、次のコマンドを使用します。

```
tnsping stprd 10
```

このコマンドによって、次のメッセージが出力されます。

```
TNS Ping Utility for SunOS:  
Copyright (c) Oracle Corporation 1998. All rights reserved.  
Attempting to contact (ADDRESS=(PROTOCOL=TCP)(HOST=spot)(PORT=1521))  
OK (290 msec)  
OK (100 msec)  
OK (70 msec)  
OK (70 msec)  
OK (60 msec)  
OK (70 msec)  
OK (70 msec)  
OK (80 msec)  
OK (180 msec)  
OK (340 msec)
```

無効なネット・サービス名 無効なネット・サービス名への接続を試行する TNSPING の例を次に示します。

```
tnsping bad_db
```

この試行では、次のメッセージが出力されます。

```
TNS Ping Utility for SunOS:  
Copyright (c) Oracle Corporation 1998. All rights reserved.  
TNS-03505: Failed to resolve name
```

リスナーのない有効なネット・サービス名 次に、名前は有効であるが、アドレスに変換するためのリスナーが配置されていない（たとえば、リスナーが起動されていない）場合の TNSPING の使用例を示します。

```
tnsping testing
```

次のメッセージが戻ります。

```
TNS Ping Utility for SunOS:  
Copyright (c) Oracle Corporation 1998. All rights reserved.  
Attempting to contact (ADDRESS=(PROTOCOL=tcp)(HOST=spot)(PORT=1521))  
TNS-12541: TNS:no listener
```

Oracle Names Server への到達 Oracle Names Server に到達できるかどうかをチェックするには、次のように Net8 アドレスを使用してコマンドを入力します。

```
tnsping (address=(protocol=tcp)(host=fido)(port=1575))
```

次のようなメッセージがユーザーに戻されます。

```
TNS Ping Utility for SunOS:  
Copyright (c) Oracle Corporation 1998. All rights reserved.  
Attempting to contact (ADDRESS=(PROTOCOL=TCP)(HOST=fido)(PORT=1575))  
OK (70 msec)
```

TRCROUTE

管理者は、Trace Route ユーティリティ (TRCROUTE) を使用して、クライアントからサーバーへのパスまたはルートを知ることができます。TRCROUTE は、問題に遭遇すると、1つのエラーではなくエラー・スタックをクライアントに戻します。これらの追加エラー・メッセージによって、トラブルシューティングが簡単になります。

TRCROUTE は、特殊な種類の接続パケットとして送信される点で TNSPING とは異なります。接続先に向かって移動するとき、TRCROUTE 接続パケットは、経由するすべてのノードの TNS アドレスを収集します。エラーが発生した場合、TRCROUTE は、エラーの発生場所を示すエラー情報を収集します。Trace Route Utility は、収集した情報をクライアント画面に表示します。TRCROUTE の出力は、ファイルにリダイレクトして印刷することもできます。

要件 Trace Route は、Net8 と SQL*Net バージョン 2.3 以降でのみ動作します。クライアントからサーバーへのルートに沿ったすべてのノードは、SQL*Net バージョン 2.3 以降を使用する必要があります。パス上に 2.3 より前のノードがある場合は、次のエラーが表示されます。

```
TNS-03603: Encountered a node with pre-2.3 version of SQL*Net
```

TRCROUTE は、パス上のどのノードに問題があるかを示します。

パフォーマンスへの影響 Trace Route Utility は、最小限のリソースを使用します。これは、特殊な接続パケットの接続データから情報を収集します。標準的な接続パケットには作用しません。

TRCROUTE は、サーバーに作用しません。TRCROUTE 接続パケットの受信と処理はリスナーが行います。これは、拒絶パケットの中に情報を入れることでクライアントに情報を戻します。サーバーは、新しいプロセスを起動したり、ダミー接続を処理する必要はありません。

TRCROUTE の使用 TRCROUTE を呼び出すには、コマンド行から次のように入力します。

```
trcroute net_service_name
```

リスナー負荷のバランスを使用するようにネットワークを構成している場合は、1つのデータベースに対する複数のノード上に1つ以上のリスナーが存在する可能性があります。この場合、Trace Route コーティリティは通常の接続要求と同じように任意のリスナーを使用します。どのリスナー・ノードが使用されたかは、戻される出力によって判断できます。

次に、トレース・ルートの2つの例を示します。

- クライアントからリスナーに移動した正常な Trace Route パケット
- リスナーが起動されていないために、リスナーに到達できなかった問題のある Trace Route パケット

図 8-1 正常な Trace Route

```
%trcroute tcp_direct
Trace Route Utility for Solaris:
Copyright (c) Oracle Corporation 1998. All rights reserved.

Route of TRCROUTE:-----
Node: Client          Time and address of entry into node:
-----
01-DEC-96 13:26:36 ADDRESS= PROTOCOL=TCP Host=shining-sun Port=1581
Node: Server          Time and address of entry into node:
-----
01-DEC-96 13:27:20 ADDRESS= PROTOCOL=TCP Host=setting-sun Port=1521
```

図 8-2 エラーの発生した Trace Route

```
% trcroute tcp_direct

Trace Route Utility for SVR4:
Copyright (c) Oracle Corporation 1996. All rights reserved.

Route of TRCROUTE:-----
Node: Client          Time and address of entry into node:
-----
01-DEC-96 11:12:34 ADDRESS= PROTOCOL=TCP Host=shining-sun Port=1581
TNS-12224: TNS:no listener
TNS-12541: TNS:no listener
TNS-12560: TNS:protocol adapter error
TNS-03601: Failed in route information collection
```

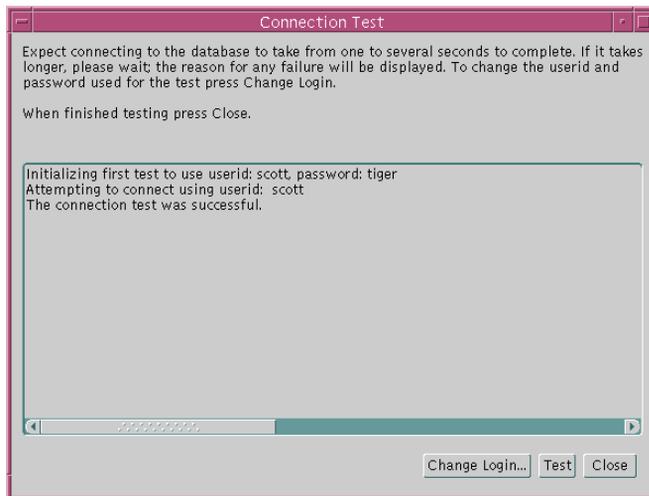
Oracle Net8 Assistant

クライアント・マシンの接続性を検証するには、Oracle Net8 Assistant を使用します。

1. Oracle Net8 Assistant を起動します。
 - UNIX の場合、`$ORACLE_HOME/bin` の下にある `netasst` を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」-> 「Oracle - HOME_NAME」-> 「Network Administration」-> 「Net8 Assistant」を選択します。
2. ディレクトリ・ツリーで Net Service Names フォルダをクリックします。
3. ネット・サービス名を選択します。
4. 「Command」メニューから「Net Service Name」を選択します。

このテストは、リスナーが稼働中であることを前提としています。リスナーが稼働していない場合、8-2 ページの「[Net8 コンポーネント起動の概要](#)」を参照してコンポーネントを起動してください。

テストが正常終了すると、「Connect Test」ダイアログ・ボックスに「The connect test was successful.」というメッセージが表示されます。



テストが成功しなかった場合は、次の処置をとってください。

- データベースとリスナーが稼働していることを確認し、「Test」をクリックします。
 - 「Change Login」をクリックして接続のユーザー名とパスワードを変更し、「Test」をクリックします。
5. 「Close」をクリックして「Connect Test」ダイアログ・ボックスを閉じます。

マルチスレッド・サーバーの構成

この章では、マルチスレッド・サーバー（MTS）の構成方法について説明します。
内容は次のとおりです。

- [概要](#)
- [Oracle のマルチスレッド・サーバー・アーキテクチャ構成](#)
- [マルチスレッド・サーバー・モードと専用サーバー・モードの両方の構成](#)
- [クライアントでのマルチスレッド・サーバー・モードの変更](#)

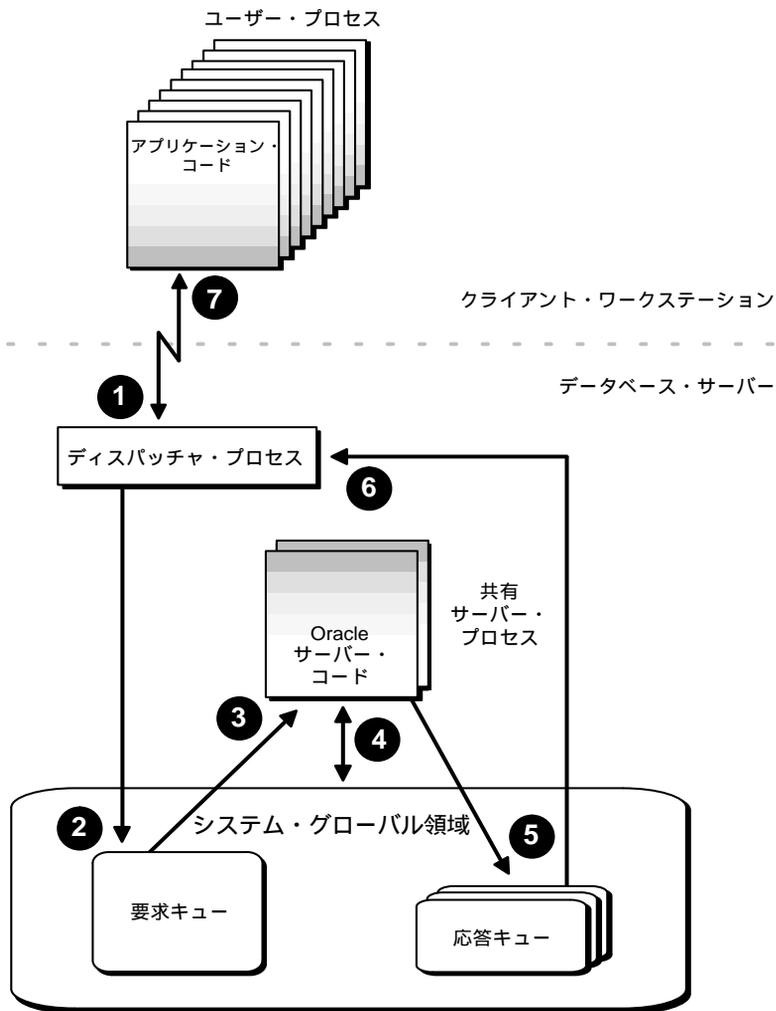
概要

専用サーバー・アーキテクチャを使用する注文入力システムを考えます。顧客の注文を店員がデータベースに入力します。トランザクションの大部分の間、店員は電話で顧客と話し、店員のユーザー処理専用のサーバー・プロセスはアイドル状態です。サーバー・プロセスはトランザクションの大部分の間で不要です。サーバー・プロセスがシステム・リソースを保持しているので、注文を入力している以外の店員にとって、システムは低速です。

MTS アーキテクチャでは、接続のたびに専用サーバー・プロセスを使用する必要はありません（[図 9-1](#) を参照してください）。少数の共有サーバーが、多数の専用サーバーと同じ量の処理を実行できます。また、各ユーザーに必要なメモリー量も比較的小さいので、より少ないメモリーとプロセス管理で、より多くのユーザーをサポートできます。

接続プーリングなどの機能を使用すると、データベースに同時接続できるユーザー数が増加し、ユーザー拡張性が向上します。7-30 ページの「[接続集中化機能の使用](#)」と 2-50 ページの「[接続プーリング](#)」を参照してください。

図 9-1 Oracle マルチスレッド・サーバー・アーキテクチャ



Oracle のマルチスレッド・サーバー・アーキテクチャ構成

システムに MTS 構成を設定するには、リスナー・プロセスを起動し、初期化ファイル (INITSID.ORA) に MTS_DISPATCHERS パラメータを設定します。

注意: Windows NT と一部のプラットフォームでは、Oracle Database Configuration Assistant を使用して MTS_DISPATCHERS パラメータを構成できません。

このパラメータを設定したら、インスタンスを再起動します。再起動の時点で MTS 構成が使用されます。MTS_DISPATCHERS は次の書式で設定します。

```
mts_dispatchers="(attribute=value)"
```

ADDRESS、DESCRIPTOPN、PROTOCOL の属性のいずれか 1 つのみ必要です。

属性	説明
ADDRESS (ADD または ADDR)	ディスパッチャのリスニング対象の終点となるネットワーク・プロトコル・アドレス。 プロトコル・アドレス構文の詳細は、6-76 ページの「 プロトコル・アドレスの構成 」を参照してください。
DESCRIPTION (DES または DESC)	プロトコルなど、ディスパッチャのリスニング対象の終点となるネットワーク設定。構文は次のとおりです。 (description =(address=...))
PROTOCOL (PRO または PROT)	ネットワーク・プロトコルを指定する。ディスパッチャでリスニング終点を生成するために使用されます。

CONNECTIONS、DISPATCHERS、LISTENER、MULTIPLEX、POOL、PRESENTATION、SERVICE、および TICKS の属性は省略可能です。

属性	説明
CONNECTIONS (CON または CONN)	各ディスパッチャで許容されるネットワーク接続の最大数。デフォルト値は、オペレーティング・システム固有です。
DISPATCHERS (DIS または DISP)	開始するディスパッチャの初期番号。デフォルトは 1 です。
LISTENER (LIS または LIST)	<p>ディスパッチャで登録されるリスナーのネットワーク名を指定する。ネーミング・メソッドで解決される名前を設定する必要があります。この属性は、リスナーが次のいずれかである場合にのみ指定します。</p> <ul style="list-style-type: none"> ■ デフォルト以外のローカル・リスナーである場合 ■ リモート・マシンに配置されている場合 <p>デフォルト以外のローカル・リスナーは、LOCAL_LISTENER パラメータで指定することもできます。Oracle Database Configuration Assistant では LOCAL_LISTENER パラメータの構成がサポートされないため、LISTENER 属性を使用することをお勧めします。</p> <p>重要: ディスパッチャでデフォルト以外のリスナーに情報を登録するために、指定するリスナー名は、サーバー上の TNSNAMES.ORA や Oracle Names Server などのネーミング・メソッドを使用して解決する必要があります。エントリーは、CONNECT_DATA 部分のサービス名情報ではなく、リスナーのアドレスのみを設定します。</p> <p>たとえば、リスナー名が LIST1 で、リスニング終点ではポート 1521 で TCP/IP を使用し、指定されたネーミング・メソッドがローカルネーミング・メソッドである場合、TNSNAMES.ORA 内のエントリーは次のようになります。</p> <pre>listener1.us.oracle.com= (description= (address= (protocol= tcp) (host= sales2-pc) (port= 1521)))</pre> <p>Oracle Net8 Assistant では、CONNECT_DATA 情報を使用しない場合は TNSNAMES.ORA を構成できません。このため、TNSNAMES.ORA を手動で作成することをお勧めします。</p>

属性	説明
MULTIPLEX (MUL または MULT)	<p>Oracle Connection Manager の接続集中化機能を使用可能にする。</p> <p>1、ON、YES、TRUE または BOTH が指定されると、受信と送信の両方のネットワーク接続で接続集中化が使用可能です。</p> <p>IN が指定されると、クライアントから受信するネットワーク接続で接続集中化が使用可能です。</p> <p>OUT が指定されると、クライアントから送信するネットワーク接続で接続集中化が使用可能です。</p> <p>0、NO、OFF または FALSE が指定されると、受信と送信の両方のネットワーク接続で接続集中化が使用禁止です。</p> <p>追加情報 : 7-30 ページの「接続集中化機能の使用」を参照してください。</p>
POOL	<p>数字が指定される場合、接続プーリングは送受信のネットワーク接続で使用可能。指定された数は送受信のネットワーク接続でタイムアウトとなるティック数です。</p> <p>ON、YES、TRUE または BOTH が指定される場合、接続プーリングは送受信のネットワーク接続で使用可能であり、送受信のネットワーク接続でのタイムアウトはデフォルトの 10 が使用されます。</p> <p>追加情報 : 7-47 ページの「接続プーリングの使用」と 2-50 ページの「接続プーリング」を参照してください。</p>

属性	説明
PRESENTATION (PRE または PRES)	<p>次の特定の プレゼンテーション・プロトコル・サポートを使用可能にする。</p> <ul style="list-style-type: none"> ■ 標準の Net8 および Java スタアド・プロシージャ接続の場合、Two-Task Common (TTC) (デフォルト) ■ Internet Inter-Orb Protocol (IIOB) を使用する Enterprise JavaBeans と CORBA Server 通信の場合、General Inter-Orb Protocol (GIOP) <p>IIOB は、TCP/IP または SSL 付き TCP/IP の上位に GIOP をインプリメントしたものです。GIOP プレゼンテーション値として有効な値は次のとおりです。</p> <ul style="list-style-type: none"> ■ oracle.aurora.server.SGiopServer (セッションベースの GIOP ディスパッチャ接続)。このプレゼンテーションは、TCP/IP と、SSL 付き TCP/IP で有効です。 ■ oracle.aurora.server.GiopServer (通常の GIOP ディスパッチャ接続)。このプレゼンテーションは、TCP/IP と、SSL 付き TCP/IP で有効です。 <p>プレゼンテーションごとに個別の MTS_DISPATCHERS エントリが必要です。</p> <p>追加情報: 7-48 ページの「Java オプション接続の構成」を参照してください。</p>
SERVICE (SER または SERV)	<p>ディスパッチャが Net8 リスナーに登録するサービス名。SERVICE 属性は、SERVICE_NAMES パラメータに優先します。</p>
SESSIONS (SES または SESS)	<p>各ディスパッチャで許容されるネットワーク・セッションの最大数。デフォルト値は、オペレーティング・システム固有です。</p>
TICKS	<p>ネットワーク・ティックの秒数。デフォルトは 15 秒です。このパラメータは、接続プーリングのオプションです。</p>

MTS パラメータのチューニング方法の詳細は、『Oracle8i チューニング』を参照してください。

ディスパッチャの初期数の設定

インスタンス起動時に開始されるディスパッチャの数は、パラメータ `MTS_DISPATCHERS` で制御されます。インスタンスの起動前にネットワーク・プロトコルごとに起動するディスパッチャ数を見積もります。

注意： 共有サーバーとは異なり、ディスパッチャ数は動的に変更されません。ディスパッチャ数は、`ALTER SYSTEM` コマンドを使用して明示的に変更する必要があります。この方法では、`MTS_MAX_DISPATCHERS` の上限までディスパッチャ数を変更できます。

`MTS_MAX_DISPATCHERS` のデフォルト値は 5 です。セットアップ時に構成されたディスパッチャの初期数、つまり複数の `MTS_DISPATCHERS` 変数に構成された全ディスパッチャの合計が `MTS_MAX_DISPATCHERS` に指定された値より大きい場合、`MTS_MAX_DISPATCHERS` はシステムにより大きい数に自動的に設定し直されます。

各インスタンスに適切なディスパッチャ数は、データベースに期待するパフォーマンス、プロセス（オペレーティング・システムに依存）ごとの接続数に関するホストのオペレーティング・システムの制限、およびネットワーク・プロトコルごとに必要な接続数によって異なります。

インスタンスは、データベース・システム上の同時実行ユーザーと同じ数を用意する必要があります。インスタンスの起動後は、必要に応じてディスパッチャを追加起動できます。

ディスパッチャの初期数の計算

オペレーティング・システムでプロセスごとに可能な接続数がわかれば、次の式を使用して、インスタンスの起動時に作成するディスパッチャの初期数をネットワーク・プロトコルごとに計算します。

$$\text{ディスパッチャ数} = \text{CEIL} \left(\frac{\text{同時セッションの最大数}}{\text{ディスパッチャごとの接続数}} \right)$$

たとえば、通常時に 900 ユーザーが TCP/IP 経由、600 ユーザーが SPX 経由で同時接続し、プロセスごとに 255 の接続をサポートするシステムがあるとします。この場合、TCP/IP の DISPATCHERS 属性には 4 以上のディスパッチャ、SPX の DISPATCHERS 属性には 3 以上のディスパッチャを設定する必要があります。

```
mts_dispatchers="(protocol=tcp) (dispatchers=4)"
mts_dispatchers="(protocol=spx) (dispatchers=3)"
```

例

例 1 ディスパッチャで使用される IP アドレスを設定するには、次のように入力します。

```
mts_dispatchers="(address=(partial=true) (protocol=tcp) (host=144.25.16.201))
(dispatchers=2)"
```

これにより、HOST=144.25.16.201 でリスニングするディスパッチャが 2 つ起動します。このホストは、ディスパッチャにアクセス可能なカードである必要があります。

例 2 ディスパッチャの正確な位置を設定するには、次のように PORT を追加します。

```
mts_dispatchers="(address=(partial=true) (protocol=tcp)
(host=144.25.16.201) (port=5000)) (dispatchers=1)"
mts_dispatchers="(address=(partial=true) (protocol=tcp)
(host=144.25.16.201) (port=5001)) (dispatchers=1)"
```

注意： INITSID.ORA ファイルには複数の MTS_DISPATCHERS を指定できますが、1 箇所にまとめて記述する必要があります。また、MTS_DISPATCHERS のデフォルト値は 1 です。

マルチスレッド・サーバー・モードと専用サーバー・モードの両方の構成

Oracle8i リリース 8.1.5 では、GIOP と TTC プレゼンテーションを使用する MTS をサポートしています。(IOP は GIOP を TCP/IP の上にインプリメントしたものです。) MTS 接続を有効にするには、`INITSID.ORA` で各プレゼンテーションに固有の `MTS_DISPATCHERS` エントリが必要なので、IOP クライアント用の MTS と TTC クライアント用の専用サーバーを構成できます。これは、`INITSID.ORA` ファイルで `MTS_DISPATCHERS` パラメータに GIOP プレゼンテーションを設定することで実現します。専用サーバーはデフォルトの動作なので、TTC を使用するための追加構成は不要です。専用サーバーは、すべての TTC クライアント接続で使用されます。

次の例では、2 つの `MTS_DISPATCHERS` エントリに、`PRESENTATION` 属性を指定して別個の GIOP プレゼンテーションを構成しています。

```
mts_dispatchers="(protocol=tcp)
(presentation=oracle.aurora.server.SGiopServer)"
mts_dispatchers="(protocol=tcp)
(presentation=oracle.aurora.server.GiopServer)"
```

IOP クライアントと TTC クライアント用に MTS を構成することもできます。これは、プレゼンテーションごとにエントリを分けて設定することで実現します。次の例では、3 つの `MTS_DISPATCHERS` エントリに次のプレゼンテーションを構成します。

- Net8 クライアント用の TTC
TTC はデフォルトのプレゼンテーションなので、明示的エントリはありません。
- IOP クライアント用の 2 つの GIOP プレゼンテーション

```
mts_dispatchers="(protocol=tcp)"
mts_dispatchers="(protocol=tcp)
(presentation=oracle.aurora.server.SGiopServer)"
mts_dispatchers="(protocol=tcp)
(presentation=oracle.aurora.server.GiopServer)"
```

クライアントでのマルチスレッド・サーバー・モードの変更

ネットワークが MTS 用に構成され、特定のクライアントで専用サーバー・モードが必要な場合、専用サーバーを使用するネット・サービス名を TNSNAMES.ORA ファイルに構成できます。これは、接続記述子の CONNECT_DATA セクション内に (SERVER=DEDICATED) を指定することで実現します。

```
net_service_name=  
(description=  
  (address= (protocol_address_information))  
  (connect_data=  
    (service_name=service_name)  
    (server=dedicated)  
  )  
)
```

SERVER パラメータの設定方法は、7-7 ページの「[ネット・サービス名の接続属性の詳細構成](#)」を参照してください。

プログラマのための Net8 拡張機能

Net8 には、Net8 OPEN と呼ばれるアプリケーション・プログラム・インタフェース (API) があり、このインタフェースでプログラマはデータベース・アプリケーションとデータベース以外のアプリケーションの両方を開発できます。また、Net8 には、UNIX クライアント・プログラミング、シグナル・ハンドラとアラーム・プログラミング、bequeath プロトコルおよび子プロセスの終了など、プログラマにとっていくつかの新しい利点があります。

内容は次のとおりです。

- [Net8 OPEN](#)
- [UNIX クライアントのプログラム](#)

Net8 OPEN

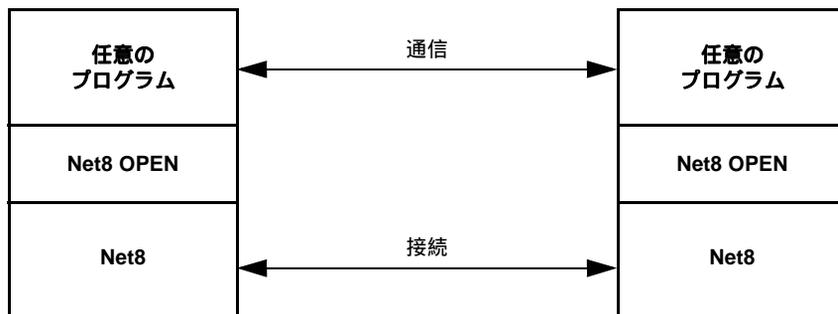
Net8 には、Net8 OPEN と呼ばれるアプリケーション・プログラム・インタフェース（API）があり、このインタフェースでプログラマは次のことが行えます。

- 環境内ですでに採用されている Net8 ネットワークを使用するデータベース・アプリケーションとデータベース以外のアプリケーションの両方を開発する
- ネットワーク・インタフェースへのコールを変更しないで各マシン上で開発されたアプリケーションを実行する

Net8 OPEN は、すべての業界標準ネットワーク・プロトコルに対して単一の共通インタフェースをアプリケーションに提供します。

Net8 OPEN と他の製品との関係を [図 10-1](#) に示します。

図 10-1 Net8 OPEN



Net8 OPEN を使用して、次に示すさまざまな問題を解決できます。

- 3層の接続（クライアント / エージェント / サーバー） - アプリケーションを使用してエージェントと通信します。エージェントは、Oracle、またはリモート情報サーバーのような Oracle 以外のデータ・ソースとの同時接続を可能にするアプリケーション・サーバーになります。
- 分散アプリケーション - データベースまたはその他ミドルウェアを必要とせずに、既存の Oracle ネットワーク上で実行できる分散アプリケーションを構築します。
- 機能拡張されたクライアント - 非 SQL 情報を SQL アプリケーションと統合します。たとえば、プロセス制御アプリケーションによるセンサーなどの非 SQL アプリケーションとの交信が可能です。

Net8 OPEN API 関数コール

リモート・プロシージャ・コール・インタフェースとは異なり、Net8 OPEN はデータを送受信する基本的なアプリケーションを開発するためのバイト・ストリーム指向の API を備えています。Net8 OPEN で開発されたアプリケーションでは、ネットワークを介して送信された値は、受信側で正しく解釈されることを保証する必要があります。

Net8 OPEN API は 5 つの関数コールから構成されます。

- [TNSOPEN](#)
- [TNSCLOSE](#)
- [TNSSEND](#)
- [TNSRECV](#)
- [TNSCONTROL](#)

TNSOPEN

説明:	<p>Net8 OPEN API の接続ハンドルを初期化します。この関数は、ユーザーが行う最初の Net8 OPEN コールでなければなりません。tnsopen() では接続は確立されないことに注意してください。接続は、必要に応じて送信または受信コールによって確立されます。</p> <p>クライアント・プログラム（接続を初期化する）を記述する場合、name には TNSNAMES.ORA ファイルと同じ形式のネット・サービス名を指定します。</p> <p>サーバー・プログラムを記述する場合、name は NULL にします。接続は、初回の tnsrecv() コールによるデータの受信時にサーバー・プログラムによって自動的にピックアップされます（tnsrecv に関する項を参照してください）。</p>
構文:	<pre>int tnsopen(handlep, name) void **handlep; const char *name;</pre>
パラメータ:	<p>handlep (IN/OUT) - Net8 接続ハンドルを受信するためのアドレス</p> <p>name (IN) - ネット・サービス名</p>
前提条件:	handlep パラメータは、NULL にしないでください。
戻り値:	正常終了の場合、ゼロ値が戻ります。エラーが発生した場合、正の Net8 API エラー番号が戻されます。

TNSCLOSE

説明:	接続を停止します。ユーザーは、接続をクローズし、ハンドルを正常に解放するためにこの関数をコールする必要があります。
構文:	<pre>int tnsfclose(handlep) void **handlep;</pre>
パラメータ:	handlep(IN/OUT) - Net8 接続ハンドルへのポインタのアドレス
前提条件:	handlep パラメータは、NULL にしないでください。
戻り値:	正常終了の場合、ゼロ値が戻り、*handlep に NULL が設定されます。エラーが発生した場合、正の Net8 API エラー番号が戻されます。

TNSSEND

説明:	<p>Net8 接続ハンドルにデータを送信します。</p> <p>クライアント側で初めて <code>tnssend()</code> をコールすると、ハンドルにデータが送信される前に接続が確立されます。クライアント・アプリケーションは、サーバーとの接続を確立するために、<code>tnsopen()</code> に続くコールとして <code>tnssend()</code> をコールする必要があります。クライアント・アプリケーションが <code>tnsrecv()</code> を最初にコールしたり、サーバー・プログラムが <code>tnssend()</code> を最初にコールすることは正しくありません。</p> <p>これは、<code>tnssend()</code> コールによって接続の確立に関するエラーが戻る場合があることも意味することに注意してください。したがって、たとえば不正な TNS アドレスを指定したことが最初に認識できるのは、期待するような <code>tnsopen()</code> コールではなく、最初の <code>tnssend()</code> コールでエラーが発生した時点です。</p>
構文:	<pre>int tnssend(handle, data, length) void *handle; const void *data; size_t *length;</pre>
パラメータ:	<p><code>handle</code>(IN/OUT) - <code>tnsopen()</code> によって戻される Net8 接続ハンドルへのポインタ</p> <p><code>data</code>(IN) - 送信されるデータへのポインタ</p> <p><code>length</code>(IN/OUT) - 送信予定のデータの長さ (バイト単位) と実際に書き込まれたバイト数に対するポインタ</p>
前提条件:	パラメータは、NULL にしないでください。
戻り値:	正常終了の場合、ゼロ値と、実際に書き込まれたバイト数が <code>length</code> パラメータによって指示された値として戻ります。エラーが発生した場合、正の Net8 API エラー番号が戻されます。

TNSRCV

説明:	<p>Net8 接続ハンドルからデータを受信します。</p> <p>サーバー側での <code>tnsrecv()</code> に対する最初のコールで、ハンドルからデータを受信する前に接続が確立されます。サーバーは、クライアントからの接続を受諾するために、<code>tnsopen()</code> に続く最初のコールとして <code>tnsrecv()</code> をコールする必要があります。</p> <p>着信する接続は Net8 リスナー (<code>tnslsnr</code>) によって受諾され、必要に応じてサーバー・プログラムのコピーが自動的に起動され、それに新しい接続が渡されます。これを発生させるために、サーバー・プログラムのロケーションにネットワーク・リスナーを構成する必要があります。後述する構成に関する項を参照してください。</p>
構文:	<pre>int tnsrecv(handle, data, length) void *handle; void *data; size_t *length;</pre>
パラメータ:	<p><code>handle</code> (IN/OUT) - <code>tnsopen()</code> によって戻される Net8 接続ハンドルへのポインタ</p> <p><code>data</code> (IN/OUT) - データを受信するバッファへのポインタ</p> <p><code>length</code> (IN/OUT) - データを受信する予定のバッファの長さとして実際に受信したバイト数に対するポインタ</p>
前提条件:	すべてのパラメータは、NULL にしないでください。
戻り値:	正常終了の場合、ゼロ値が戻り、実際に受信したバイト数が <code>length</code> パラメータによって指示された値として戻ります。エラーが発生した場合、正の Net8 API エラー番号が戻されます。

TNSCONTROL

説明:	接続をブロック・モードまたは非ブロック・モードに設定します。
構文:	<pre>int tnscontrol(handle, cmd) void *handle; int *cmd;</pre>
パラメータ:	<p>handle(IN) - tnsopen() によって戻される Net8 接続ハンドルへのポインタ</p> <p>cmd(IN) - 接続に適用されるオプション。現在サポートされている値を次に示します。</p> <p>TNSAPINONBLOCKING - 接続を非ブロック・モードに設定する</p> <p>TNSAPIBLOCKING - 接続をブロック・モードに設定する</p>
前提条件:	handle は NULL ではなく、cmd はサポートされているコマンドにしてください。
戻り値:	オプションが問題なく設定された場合はゼロ値が戻ります。

Net8 OPEN アプリケーション・プログラム・インタフェースの検索

アプリケーション・プログラム・インタフェースは、標準の Net8 インストールの一部として用意されています。そのインタフェースを使用するには、次のものを用意する必要があります。

- TNSAPI.H - API インタフェースとエラーが記述されているヘッダー・ファイルです。このファイルは、UNIX では \$ORACLE_HOME/network/public、Windows NT では *ORACLE_HOME*\network\tnsapi\include にあります。
- Net8 OPEN ライブラリ - 他の Oracle ネットワーキング・ライブラリとともに設定され、TNSAPI という名前が含まれています。なお、ライブラリの名前は、プラットフォームごとに異なります。UNIX の場合、これは \$ORACLE_HOME/network/lib ディレクトリに配置され、LIBTNSAPIA という名前になります。Windows プラットフォームでは、TNSAPI.DLL と TNSAPI.LIB というファイルが *ORACLE_HOME*/network/tnsapi/lib ディレクトリに格納されています。
- サンプル Make ファイル - ネットワーク・ディレクトリ内にプラットフォーム用として、このファイルが用意されています。これらのファイルを使用して、アプリケーションを構築するための適切なリンク行を判断できます。

独自のアプリケーションの開発

Net8 OPEN 関数を参照するモジュールには、次に示すように、TNSAPI.H が含まれる必要があります。

```
#include <tnsapi.h>
```

Make ファイル（または build コマンドの他のロケーション）は、TNSAPI.H を見つけられるようにインクルード・パスが正しく設定されていることを確認する必要があります。インストールで用意されるサンプル Make ファイルを参照してください。

Net8 OPEN アプリケーションを使用するためのシステムの構築

Net8 OPEN アプリケーションを認識するように Net8 を構築するには、次のようにします。

1. リスナー構成ファイル (LISTENER.ORA) にサーバー・プログラムのロケーションを追加すると、接続要求を受信した場合にネットワーク・リスナーがサーバーの起動を認識できます。

注意: リリース 8.1 構成では、データベース・インスタンス登録によりサービス情報がリスナーに登録されるため、この操作は不要です。

これを行うには、Oracle データベースと同じようなサービスのシステム識別子 (SID) 名を選択します。データベースと同じ SID は選択しないでください。

たとえば、"chat" プログラムを構成する場合は、SID "chatsid" をコールできます。プログラムを Oracle サーバーの実行可能プログラムと同じディレクトリに配置します。このディレクトリは通常、UNIX では \$ORACLE_HOME/bin、Windows NT では ORACLE_HOME¥bin です。

次のように、リスナー構成ファイルに次のエントリを設定します。

```
sid_list_listener =
  (sid_list =
    (sid_desc =
      (sid_name = chatsid)/*your SID name*/
      (oracle_home = /usr/oracle)/*$ORACLE_HOME bin directory*/
      (program = chatsvr)/*the name of your server program*/)
```

リスナーが新しいサービスを認識するように、再起動する必要があります。

2. アプリケーション・サーバーのアドレスをローカル名構成ファイル (TNSNAMES.ORA) に追加します。

たとえば、リスナーが次のアドレスでリスニングしているとします。

```
(description=(address=(protocol=tcp)(host=unixhost)(port=1521)))
```

また、上で作成したサービスを "chat" として参照させるとします。

リリース 8.1 構成の場合、次のパラメータをローカル名構成ファイルに追加します。

```
chat=  
(description=  
  (address=(protocol=tcp)(host=unixhost)(port=1521)  
  )  
  (connect_data=(service_name=chatsid)))
```

リリース 8.1 より前の構成の場合、次のパラメータをローカル名構成ファイルに追加します。

```
chat=  
(description=  
  (address=(protocol=tcp)(host=unixhost)(port=1521)  
  )  
  (connect_data=(sid=chatsid)))
```

前述の LISTENER.ORA ファイルで構成した SID がアドレスに記述されていることに注意してください。行が続いていることを示すために 2 行目が最低 1 つのスペース文字で始まっていることにも注意してください。

ネットワーク内にドメインがある場合は、それに応じてサービス名を付ける必要があります。たとえば、ドメインが acme.com の場合は chat.acme.com とします。この場合も、TNSNAMES.ORA ファイルをテンプレートとして使用しますが、他のネット・サービス名がすべてドメインで終了する場合は、サービスにも同様に名前を付ける必要があります。

3. Oracle サーバーの実行可能プログラムと同じディレクトリ内に、サービスの実行可能プログラムを配置します。UNIX プラットフォームでは、LISTENER.ORA ファイルで指定されている \$ORACLE_HOME/bin ディレクトリに、実行可能プログラムを配置します。この例では、ロケーション /usr/oracle/bin/chatsvr にプログラム "chatsvr" を保存します。

オペレーティング・システムによっては、プログラムの実行許可があるかどうかを確認する必要があります。

サンプル・プログラム

Net8 OPEN には、2 つのサンプル・アプリケーションが用意されています。

- `finger` - サーバーに接続し、ログインしているユーザーの情報を取り出すユーティリティです。このユーティリティには、分散アプリケーションの構築に関する基本的な手順を例示する 1 対のプログラムが含まれています。このクライアント・プログラムは、Solaris と Windows NT の両方で実行されます。サーバーは UNIX 固有です。
- `tftp` - このサンプル・クライアント・プログラムとサンプル・サーバー・プログラムは、UNIX にインプリメントされており、`tftp` プロトコルを使用する単純なファイル転送を支援します。

Net8 OPEN API エラー

この項では、前述の関数コールに失敗したときに戻される可能性があるエラー番号の一覧を示します。送信または受信コールが行われた時点で接続が確立されていなかった場合は、これらのコールから接続関連エラーが戻る可能性があることに注意してください。

20002 - SDFAIL_TNSAPIE - The underlying "send" command failed in tnssend().
20003 - RECVFAIL_TNSAPIE - The underlying "receive" command failed in tnsrecv().
20004 - INVSVROP_TNSAPIE - Operation is invalid as the server.
20005 - INVCLIOP_TNSAPIE - Operation is invalid as the client.
20006 - HDLUNINI_TNSAPIE - The connection should be initialized by calling tnsopen().
20007 - INHFAIL_TNSAPIE - Server failed in inheriting the connection from the listener.
20008 - ACPTFAIL_TNSAPIE - Server failed in accepting the connection request from the client.
20009 - NULHDL_TNSAPIE - A null handle was passed into the call, which is not allowed.
20010 - INVOP_TNSAPIE - An invalid operation called was passed into the call.
20011 - MALFAIL_TNSAPIE - A malloc failed in TNS API call.
20012 - NLINIFAIL_TNSAPIE - Failed in NL initialization.
20013 - NMTOOLONG_TNSAPIE - Service name is too long.
20014 - CONFALL_TNSAPIE - Client connect request failed.
20015 - LSNFAIL_TNSAPIE - Server failed to listen for connect request.
20016 - ANSFAIL_TNSAPIE - Server failed to answer connect request.
20017 - NMRESFAIL_TNSAPIE - Failed to resolve service name.
20018 - WOULDBLOCK_TNSAPIE - Operation would block.
20019 - CTLFAIL_TNSAPIE - Control call failed.
20020 - TNSAPIE_ERROR - TNS error occurred.
20021 - INVCTL_TNSAPIE - Invalid operation request in control call.

UNIX クライアントのプログラム

UNIX のイベント・プログラミングでは、UNIX シグナルを使用する必要があります。イベントが発生すると、シグナルによってプロセスに通知されます。プロセスは、生成された特定のシグナルに関連するコードを実行します。UNIX では、特定のシグナル・コールのための 1 つのプロセスが 1 つ以上のシグナル・ハンドラまたはアラームを設定することは許されていません。プロセスが SIGCHLD (子プロセスのステータス変更に関するシグナル) のようなシグナルに対して次のシグナル・ハンドラまたはアラーム要求を設定した場合、UNIX は SIGCHLD に対する直前の要求を無効化し、それを失います。

作成したアプリケーションからこのような要求が出された場合、シグナル処理またはアラームのためにシステムが特定の要求を消失し、応答がなくなる可能性があります。要求されたシグナルによっては、シグナル・ハンドラ上の問題のためにシステムは消滅したプロセスを正常にクリーン・アップしない可能性があります。

Net8 には、Oracle がこれらの要求を使用するとともにシグナル・ハンドラとアラームを使用できるようにする 2 つの解決策が用意されています。

- [シグナル・ハンドラとアラームのプログラミング](#)
- [Bequeath](#)

シグナル・ハンドラとアラームのプログラミング

Net8 には、各シグナルに対するすべてのシグナル・ハンドラ要求またはアラーム要求のテーブルを維持するオペレーティング・システム依存 (OSD) コールがあります。シグナル・ハンドラまたはアラームを使用するすべてのプログラムは、この Oracle OSD コールを使用する必要があります。OSD コールには、特定のコールに対して複数のシグナル・ハンドラまたはアラームを設定することが許されない UNIX 環境のプログラマのための解決策が用意されています。シグナル・ハンドラまたはアラームを使用するすべてのプログラムは、Oracle OSD コールを使用する必要があります。ただし、これは、現在内部でしか使用できません。近い将来、クライアント・アプリケーション用の OSD コールの外部用バージョンがリリースされる予定です。

それまでは、データベース接続を行う前に、クライアントのすべてのシグナル・ハンドラを設定すると、OSD コールがシグナルに対して設定された最後のシグナル・ハンドラを記憶し、それをシグナル・ハンドラ・テーブルに追加します。なお、これを行うと、シグナル・ハンドラを使用禁止にできなくなります。

Oracle OSD シグナル処理規則

すべての SIGCHLD コール用のテーブル駆動型共有 OSD シグナル・ハンドラを使用する場合は、次の規則に従う必要があります。

- 子プロセス ID を認識して、正しいプロセスをクリーン・アップできるようにする。
- 適切な子プロセス ID に対して、`wait()` ではなく、`waitpid()` コールを使用する。
- `waitpid()` コールは非ブロックで行う必要がある。

Bequeath

この項の対象者は、子プロセスのステータスの変化を SIGCHLD コールによって把握する部分に UNIX シグナル・ハンドラを使用し、アプリケーションのネットワーク部分に Net8 を使用する UNIX アプリケーション・プログラマです。

クライアント・アプリケーションは、同じマシン上の Oracle データベースと通信するように命令されると、`bequeath` プロトコルを使用して接続を確立します。`bequeath` プロトコルは、クライアントがリスナーを使用せずにデータベースから情報を検索できるようにします。`bequeath` プロトコルは、クライアント・アプリケーションごとにサーバー・プロセスを内部的に起動します。ある意味で、`bequeath` プロトコルはリモート・リスナーが接続に対して行う同じ操作をローカルで実行します。

子プロセスの終了

クライアント・アプリケーションは bequeath プロトコルを介して内部にサーバー・プロセスを子プロセスとして生成するため、クライアント・アプリケーションは子プロセスが終了したときにそのプロセスをクリーン・アップする必要があります。サーバー・プロセスが接続の責任を完遂すると、それは消滅プロセスになります。シグナル・ハンドラは、これらの消滅プロセスをクリーン・アップする必要があります。別の方法として、シグナル・ハンドラを使用禁止にすることにより、このプロセスを UNIX init プロセスに渡すようにクライアントの SQLNET.ORA ファイルを構成できます。

Oracle Net8 Assistant を使用して、UNIX シグナル・ハンドラを使用禁止にするようにクライアントを構成します。使用禁止にするように設定された SQLNET.ORA パラメータは、次のとおりです。

```
bequeath_detach=yes
```

このパラメータを選択すると、すべての子プロセスは UNIX init プロセス (pid = 1) に渡されます。init プロセスは、消滅した子プロセスを自動的にチェックして終了します。

bequeath は、シグナル・ハンドラの使用を自動的に選択して子プロセスのステータスの変化を捕捉します。作成したアプリケーションでシグナル・ハンドラを使用していない場合、このデフォルトによる影響はありません。

Net8 のトラブルシューティング

Net8 には、ログ・ファイルとトレース・ファイルを使用してネットワーク上の問題点を理解して解決する方法があります。これらのファイルは、エラーが発生すると、ネットワーク・コンポーネント間の対話を追跡し記録します。この情報を評価すると、非常に複雑なネットワーク上の問題点でも容易に診断してトラブルシューティングできます。

この章では、一般的なネットワーク・エラーとそれらのエラーを解決する手順について説明します。また、さらに複雑なネットワーク上の問題点を診断してトラブルシューティングするためにエラー情報をロギングしてトレースする方法についても説明します。内容は次のとおりです。

- Net8 の診断
- 一般的なエラー・メッセージの解決
- フィールドからのトラブルシューティングのヒント
- ログ・ファイルとトレース・ファイルを使用したネットワーク上の問題点のトラブルシューティング
- エラー情報のロギング
- エラー情報のトレース
- オラクル社サポート・サービスへの連絡

Net8 の診断

Net8 のインストールと構成を終了し、基本的なピアツーピア（単一プロトコル・ネットワーク）接続を確立しようとして ORA ERROR が戻された場合、この項を読むことにより、問題の原因を診断できることがあります。

基底にあるエラーが Net8 によりエラー番号やエラー・メッセージで報告されますが、これらは実際の問題を指しているとは限りません。この項は、Net8 の正常に機能する部分と機能していない部分を切り分けるのに役立ちます。また、エラーが次のどの項目に属するかを判断するのに役立ちます。

- Oracle ソフトウェア
- オペレーティング・システム・レイヤー
- その他のネットワーク・レイヤー

ほとんどの場合、各種ネットワーク・レイヤーを順にテストすることで問題を発見できます。

サーバーの診断

次の質問に教えてください（この項の指示に従うにはサーバー管理者の助けが必要になることもあります）。

- 他のシステム（ワークステーション / サーバー）は Net8 を使用してサーバーに接続できるか
- サーバー、データベースまたはリスナーの構成が最近変更されたか

上の質問に1つでも当てはまるものがある場合、この項をスキップして 11-4 ページの「[クライアントの診断](#)」に進んでください。

不明な場合、またはいずれにも当てはまらない場合、このまま続けてください。

サーバー上の Net8 を診断するには次の作業を行います。

- [作業 1: データベースの実行状態の確認](#)
- [作業 2: ループバック・テストの実行](#)

作業 1: データベースの実行状態の確認

データベースが稼働していることを確認するには、次の手順を実行します。

- 次のように入力して、有効なユーザー名とパスワードを使用してデータベースにログオンおよび接続します。

```
sqlplus system/manager
```

データベースに接続していることを確認するメッセージが表示されます。次のエラーが表示される場合は、データベース管理者に連絡してください。

- ORA-1017: invalid U/P
- ORA-1034: Oracle not available

作業 2: ループバック・テストの実行

ループバック・テストを実行するには、次の手順に従います。

1. LISTENER.ORA、TNSNAMES.ORA および SQLNET.ORA の各ファイルが、UNIX では `$ORACLE_HOME/network/admin`、Windows NT では `ORACLE_HOME¥network¥admin` に存在することを確認します。
2. ループバック・テストを実行するには、8-19 ページの「[サーバーでの構成のテスト](#)」にある手順に従います。
 - ループバック・テストでエラーが発生する場合、次の手順に進みます。
 - ループバック・テストが正常に終了する場合、下の「[クライアントの診断](#)」にスキップします。
3. エラー情報の詳細については、データベースの問題および解決を行うウェブサイト <http://support.oracle.com> を確認するか、またはオラクル社サポート・サービスに連絡してください。

クライアントの診断

この時点で、次の最低 1 項目が確認されているので、Net8 サーバー側のリスナーは正常に機能していることになります。

- ネットワークが機能していることを示すサーバーでのループバック・テストが正常終了した
- 他のマシン（サーバー、ワークステーション）も Net8 を使用して同一の Oracle Server に接続している
- このマシンに変更を加える（新たな製品のインストールやネットワーク構成の変更など）前は、このワークステーションからの接続が機能していた

クライアントの診断を行うには、次の手順に従います。

1. サーバーにインストールされているのと同じ Oracle プロトコルがインストールされていることを確認します。UNIX の場合、`adapters` プログラムを実行することで、このことを確認できます。`$ORACLE_HOME/bin` で `adapters` を実行します。

次のような出力が表示される。

```
Installed Net8 Transport Protocols are:
```

```
IPC
TCP/IP
BEQueath
SSL
RAW
...
```

2. 基底にあるネットワーク・トランスポートの基本的な接続性を確認します（この確認は、ネットワーク管理者に依頼しなければならないことがあります）。Net8 テクノロジを使用して正常に接続できるかどうかは、基底のネットワークに依存します。

プロトコル	確認方法
TCP/IP	ワークステーションから、リスナーやデータベースが配置されているサーバーに対して、ファイル転送やターミナル・エミュレーション・ユーティリティ（FTP、TELNET および PING）を使用する。
SPX	<ul style="list-style-type: none"> ■ データベースを実行しているマシンに Netware ログインを実行する ■ ネットワーク上のドライブをマップできること、または Print Servers や File Servers などのその他の Novell サービスを使用できることを確認する ■ SPX ネットワーク上の Novell Server や Novell File Server から DISPLAY_SERVERS を実行して、リスナー・サービスがブロードキャストを行うことを確認する
Named Pipes	<ul style="list-style-type: none"> ■ マイクロソフト・ネットワーク上のその他のコンピュータやサーバーをみる ■ マイクロソフト・ネットワーク内のドライブを共有できることを確認する

3. Net8 クライアントと適切なプロトコルの両方が存在することを確認するために、すべての Net8 ソフトウェアがインストールされていることを確認します。
4. クライアント・マシンで、TNSNAMES.ORA ファイルと SQLNET.ORA ファイルが、UNIX では `$ORACLE_HOME/network/admin`、Windows プラットフォームでは `ORACLE_HOME¥network¥admin` に存在することを確認します。

SQLNET.ORA と TNSNAMES.ORA の検索順は、次のとおりです。

- a. アプリケーションが実行される現行の作業ディレクトリ
- b. TNS_ADMIN 環境変数
TNS_ADMIN 環境変数が Windows プラットフォームに定義されていない場合、TNS_ADMIN はレジストリに設定されていることがあります。
- c. TNS_ADMIN が定義されていない場合、UNIX では `$ORACLE_HOME/network/admin`、Windows プラットフォームでは `ORACLE_HOME¥network¥admin`

選択した Oracle データベースに Net8 を使用して接続している他の稼働中のクライアント・マシンがある場合、既存のファイルのバックアップをとって、稼働中のマシンの TNSNAMES.ORA ファイルと SQLNET.ORA ファイルを稼働していないワークステー

ションにコピーします。これにより、ファイルにエラーが含まれている可能性が排除されます。

5. 8-21 ページの「[クライアントからのネットワーク接続のテスト](#)」に従って、Net8 レイヤーをテストします。

注意： TNSPING は使用しないでください。TNSPING は TCP/IP PING ユーティリティと同様に機能し、ソケットの作成やオープンは行われず、リスナーとも接続しません。これで確認できるのは、リスナーがサーバー側に存在することのみです。

6. なおも接続がうまくいかない場合、次の作業を行います。
 - 11-15 ページの項「[ログ・ファイルとトレース・ファイルを使用したネットワーク上の問題点のトラブルシューティング](#)」に従って、トレースを使用します。

一般的なエラー・メッセージの解決

ネットワーク通信が複雑であるため、さまざまな理由で各種ソースからネットワーク・エラーが発生する場合があります。エラーが発生した場合、SQL*Plus や SQL*Forms などのアプリケーション (Net8 からのネットワーク・サービスにより異なる) には、通常、エラー・メッセージが発生します。

次のリストは、最も一般的なネットワーク・エラー・メッセージです。

- `ORA-12154: TNS:could not resolve service name`
- `ORA-12198: TNS:could not find path to destination`
- `ORA-12203 TNS:unable to connect to destination`
- `ORA-12224:TNS:no listener`
- `ORA-12500: TNS:listener failed to start a dedicated server process`
- `ORA-12533: TNS:illegal ADDRESS parameters`
- `ORA-12545: TNS:name lookup failure`
- `ORA-12560: TNS:protocol adapter error`
- `ORA-3113: TNS:End of file on communication channel`
- `ORA-3121: No interface driver connection - function not performed`

次の表は、各ネットワーク・エラーと、それらのエラーをトラブルシューティングする手順を示したものです。

エラー番号：メッセージ	説明/トラブルシューティングの手順
ORA-12154: TNS:could not resolve service name	<p>原因: Net8 が TNSNAMES.ORA 構成ファイルで指定されたネット・サービス名を検出できない。</p> <p>処置:</p> <ol style="list-style-type: none"> 1. TNSNAMES.ORA ファイルが存在し、アクセスできることを確認します。 2. TNSNAMES.ORA ファイルのコピーが複数存在しないことを確認します。 3. 接続文字列に指定したネット・サービス名が TNSNAMES.ORA ファイル内で接続記述子にマップされていることを、TNSNAMES.ORA で確認します。また、ファイルに構文エラーがないことを確認します。 4. SQLNET.ORA ファイルの重複コピーがないことを確認します。 5. ドメイン名を使用している場合、SQLNET.ORA ファイルに NAMES.DEFAULT_DOMAIN パラメータが含まれていることを確認します。このパラメータが存在しない場合は、接続文字列にドメイン名を指定する必要があります。 ドメイン名を使用しておらず、このパラメータが存在する場合は、このパラメータを削除するか、またはコメントにして使用禁止にします。 6. ログイン・ダイアログ・ボックスから接続する場合は、接続するネット・サービス名の前に @ 記号を付けていないことを確認します。 7. クライアントのトレースをオンにして、操作を再度実行します。
ORA-12198: TNS:could not find path to destination	<p>原因: クライアントが目的のデータベースを見つけられない。</p> <p>処置:</p>
ORA-12203 TNS:unable to connect to destination	<ol style="list-style-type: none"> 1. 到達するネット・サービス名を正しく入力したかどうかを確認します。 2. TNSNAMES.ORA ファイルの接続記述子に指定したネット・サービス名 ADDRESS パラメータが正しいかどうかを確認します。 3. TNSNAMES.ORA ファイルが正しいディレクトリに格納されていることを確認します。 4. リモート・ノード上のリスナーが起動し、動作していることを確認します。動作していない場合は、リスナー制御ユーティリティを使用してリスナーを起動します。 5. ログイン・ボックスから接続する場合は、接続するネット・サービス名の前に @ 記号を付けていないことを確認します。

エラー番号：メッセージ	説明 / トラブルシューティングの手順
ORA-12203 TNS:unable to connect to destination	<p>ORA-12203 エラーは総合的なエラーで、2 次的なエラーを隠している場合があります。このため、2 次的な ORA メッセージがないかどうか最新の SQLNET.LOG ファイルを確認します。このファイルは、UNIX では \$ORACLE_HOME/network/log、Windows プラットフォームでは ORACLE_HOME¥network¥log にあります。このログ・ファイルを解析して 2 次的なエラーがないことが判明した場合、次の状況のいずれかが問題の原因である可能性があります。</p> <p>原因: 選択したネットワーク・プロトコルには適切でない Oracle プロトコルがインストールされている。プロトコル・サポート・ドライバが欠落している場合は通常、SQLNET.LOG やクライアント・トレース・ファイルに次のエラーが作成されます。</p> <ul style="list-style-type: none">■ ORA-12203■ ORA-12538■ ORA-00508 <p>処置: 適切な Oracle プロトコルがインストールされていることを確認します。UNIX の場合、adapters プログラムを実行することで、このことを確認できます。\$ORACLE_HOME/bin で adapters を実行します。</p> <p>次のような出力が表示される。</p> <p>Installed Net8 Transport Protocols are:</p> <pre>IPC TCP/IP BEQueath SSL RAW ...</pre>
ORA-12203 続き	<p>原因: 接続文字列に無効なネット・サービス名が指定されている。</p> <p>処置: 接続文字列に指定されたネット・サービス名が TNSNAMES.ORA ファイルに存在すること、およびそのネット・サービス名の ADDRESS 情報が有効であることを確認します。チェック項目は次のとおりです。</p> <ul style="list-style-type: none">■ HOST 名や SERVICE 名は正しいか■ 指定した PORT は正しいか

エラー番号：メッセージ	説明 / トラブルシューティングの手順
ORA-12203 続き	<p>原因：Net8 は指定された接続記述子を TNSNAMES.ORA ファイルで見つけることができない。</p> <p>処置：データベースが稼働していることを確認してから、次の項目を確認します。</p> <ol style="list-style-type: none"> リスナーが実行中であることを確認します。次のように入力します。 <pre>lsnrctl lsnrctl> status listener_name</pre> <p><i>listener_name</i> は、LISTENER.ORA ファイルに定義されたリスナー名です。LISTENER という名前のデフォルト・リスナーを使用している場合、リスナーを識別する必要はありません。</p> <p>リスナーが実行中でないことが出力により示される場合、次のコマンドでリスナーを起動します。</p> <pre>lsnrctl> start listener_name</pre> TNSNAMES.ORA ファイルが適切な場所に配置されていることを確認します。この場所は、UNIX では \$ORACLE_HOME/network/admin、Windows プラットフォームでは <i>ORACLE_HOME</i>\network\admin です。
ORA-12203 続き	<p>原因：接続先システムのリスナーがリスニングしていない。</p> <p>処置：リモート・システムのリスナーが実行中であることを確認します。次のように入力します。</p> <pre>lsnrctl lsnrctl> status listener_name</pre> <p><i>listener_name</i> は、LISTENER.ORA ファイルに定義されたリスナー名です。LISTENER という名前のデフォルト・リスナーを使用している場合、リスナーを識別する必要はありません。</p> <p>リスナーが実行中でないことが出力により示される場合、次のコマンドでリスナーを起動します。</p> <pre>lsnrctl> start listener_name</pre>
ORA-12203 続き	<p>原因：基底のネットワーク・トランスポートに問題がある。</p> <p>処置：使用しているネットワーク・トランスポートにより提供されるユーティリティで、プロトコル自体が機能していることを確認します。たとえば、TCP/IP の場合、リモート・システムに対して PING を実行します。</p>
ORA-12203 続き	<p>原因：TNSNAMES.ORA ファイルが適切なディレクトリに配置されていない。</p> <p>処置：TNSNAMES.ORA ファイルが適切な場所に配置されていることを確認します。ファイルの場所は、UNIX では \$ORACLE_HOME/network/admin、Windows プラットフォームでは <i>ORACLE_HOME</i>\network\admin、または 11-4 ページの「クライアントの診断」に説明されている代替パスです。</p>

エラー番号：メッセージ	説明 / トラブルシューティングの手順
ORA-12203 続き	<p>原因: TCP/IP アドレスの (HOST=<i>server_name</i>) パラメータまたは SPX の (SERVICE=<i>tns_application</i>) パラメータがクライアント・マシンとサーバー・マシンとで一致していない。</p> <p>処置: これらのパラメータの値がサーバーとクライアントで同じであることを確認します。</p> <p>TCP/IP セットアップの場合、サーバー上の LISTENER.ORA ファイルの HOST パラメータと、クライアントの TNSNAMES.ORA ファイルの HOST パラメータが、同じ名前または各システムで同一の IP アドレスに変換される名前を指していることを確認します。これは特に、サーバーが複数の IP アドレスを持ち、それぞれがサーバー上の各種ネットワーク・インタフェースに割り当てられている場合に重要です。</p> <p>SPX セットアップの場合、名前がサーバー・マシンとクライアント・マシンで同じである必要がある。</p>
ORA-12224:TNS:no listener	<p>原因: リスナーが実行されていないため、接続要求を完了できない。</p> <p>処置:</p> <ol style="list-style-type: none">1. 指定した接続先アドレスが、リスナーで使用されているアドレスの1つと一致していることを確認します。2. このエラーが、バージョンの互換性に起因する問題でないかどうかを確認します。
ORA-12500: TNS:listener failed to start a dedicated server process	<p>原因: リスナーが、ユーザーをデータベース・サーバーに接続するプロセスの起動に失敗した。</p> <p>処置:</p> <ol style="list-style-type: none">1. LISTENER.ORA ファイル内の SID_LIST セクションと、TNSNAMES.ORA ファイルの CONNECT DATA セクションにあるシステム識別子 (SID) が正しいかどうかを確認します。2. SID_LIST セクションの PROGRAM パラメータに指定されたプログラム名が /bin ディレクトリに存在することを確認するか、またはプログラム名のパスを指定します。3. データベースにアクセスする十分な権限がユーザーにあることを確認します。
ORA-12533: TNS:illegal ADDRESS parameters	<p>原因: TNSNAMES.ORA ファイルで指定された接続記述子の ADDRESS セクションにあるプロトコル固有のパラメータが正しくない。</p> <p>処置: プロトコル固有のキーワードについての詳細は、ご使用のプラットフォーム用 Oracle オペレーティング・システム対応のマニュアルを参照。</p>

エラー番号: メッセージ	説明/トラブルシューティングの手順
ORA-12545: TNS:name lookup failure	<p>原因: リモート・ノード上のリスナーに接続できない。</p> <p>処置:</p> <ol style="list-style-type: none">1. TNSNAMES.ORA ファイルまたは LISTENER.ORA ファイル内の ADDRESS が正しいかどうかを確認します。2. リモート・ノード上のリスナーが起動されたことを確認します。リスナー制御ユーティリティの STATUS コマンドでその状態をチェックし、必要であれば START コマンドでそのリスナーを起動します。
ORA-12560: TNS:protocol adapter error	<p>原因: リスナーが、ユーザーをデータベース・サーバーに接続するプロセスの起動に失敗した。</p> <p>処置:</p> <ol style="list-style-type: none">1. トレースをオンにして、操作を再度実行します。2. 問題を診断するために、トレース・ファイルの内容を検証します。
ORA-3113: TNS:End of file on communication channel	<p>原因: 通信チャネル上で、予測しなかったファイルの終わりが処理された。これは、通信リンクが少なくとも一時的にダウンした可能性があることを示す場合があります。また、サーバーがダウンしていることを示す場合もあります。</p> <p>処置: 再送カウントを変更しなければならない場合があります。このエラーのトラブルシューティングについての詳細は、ご使用の Oracle オペレーティング・システム対応のマニュアルを参照してください。</p>
ORA-3121: No interface driver connection - function not performed	<p>原因: 接続文字列で使用される SQL*Net バージョン 1 の接頭辞に誤りがある。</p> <p>処置: 接続文字列内で次の接頭辞は使用してはいけません。</p> <ul style="list-style-type: none">■ T:■ X:■ P: <p>原因: Oracle データベースがインストールされていないクライアント・マシンからは、ユーザー名とパスワードのみが指定された。</p> <p>処置: 接続文字列を指定します。</p>

フィールドからのトラブルシューティングのヒント

ネットワークの問題の診断に困ったときは、次のヒントが参考になることがあります。

- **構成中はサーバー・マシン名ではなくノードやネットワーク・アドレスを使用する** - これにより内部的な検索の問題が除去され、接続がやや高速になります。

TCP/IP - TCP/IP アドレスのホスト名ではなく、198.32.3.5 のようなインターネット・アドレスを使用します。たとえば、TNSNAMES.ORA ファイルの (HOST=*server_name*) 行を (HOST=198.32.3.5) のようにインターネット・アドレスに変更します。
- **SPX 接続の問題を考慮する** - リモート・リスナーでの接続を要求するワークステーションは、まず NetWare IPX ネットワークでの SPX サービスのロケーションを認識する必要があります。

クライアント・ワークステーションは、SPX サービスの検索要求を発行します。サービスが見つからない場合、ワークステーションにエラーが戻されます。
- **ループバック・テストを実行する** - 8-19 ページの「[サーバーでの構成のテスト](#)」の説明に従って、サーバーでループバック・テストを実行します。テストが正常終了する場合、FTP を使用して TNSNAMES.ORA ファイルと SQLNET.ORA ファイルをクライアントに送信します。
- **作業マシンとサーバーの間をチェックする** - ワイド・エリア・ネットワーク (WAN) の場合、すべての中間システムが正常に機能していることを確認します。すべてのマシンが正常である場合、タイミングに問題がある可能性があります。
- **タイミングに問題がないかどうかを確認する** - タイミングの問題は、クライアント・ログ・ファイルの ORA-12293、ORA-12535 または ORA-12547 のエラーに関係します。

これを解決するには、名前ではなく正確なアドレスを使用して接続のスピード・アップを図り、LISTENER.ORA ファイルの CONNECT_TIMEOUT パラメータの値を増やします。このパラメータのデフォルト値は 10 秒です。
- **どの Oracle アプリケーションがエラーを発生しているのかを判断する** - SQL*Plus は機能していても、CASE tools は機能していない可能性があります。データ量に問題があることが判明した場合、基本的な接続を使用してサイズの大きい (5MB) ファイルを転送してみます。

トラブルシューティング時の質問への回答

問題を診断する際に、次の項目をチェックしてください。

- すべてのマシンに問題があるのか、1台のマシンのみか

同じソフトウェア（Oracle およびサードパーティ製品）をインストールしているにもかかわらず、機能するマシンと機能しないマシンがある場合、距離が近ければ、ネットワーク・ケーブルを取り替えて問題が進展するかどうかを確認します。状況に変化があった場合、問題はクライアント/サーバー接続に関するもので、PC に限定的なものではないことを示しています。

- クライアントとサーバー間のリンクの種類（X.25、ISDN、トークン・リングまたは専用線など）は何か。

Sniffers や LAN Analyzers は、断続的な接続エラーやタイムアウトおよび再送信パケットの検出を行うのに便利です。また、会話のどちら側が応答を待っているのかを知ることができます。

- Oracle のアプリケーションは機能しているにもかかわらず、サードパーティ・アプリケーションにエラーが発生するか

ログ・ファイルとトレース・ファイルを使用したネットワーク上の問題点のトラブルシューティング

Net8 では、問題が発生する順を追って、問題点の原因の内容について詳細を示します。この情報は、ログ・ファイルとトレース・ファイル内に作成、格納されます。エラー情報のロギングとトレースのプロセスにより、ネットワーク上の問題点を容易に診断、解決できます。

サーバーとリスナーの場合、ログ・ファイルのデフォルトの場所は、UNIX では \$ORACLE_HOME/network/log、Windows プラットフォームでは `ORACLE_HOME\network\log`、トレース・ファイルのデフォルトの場所は、UNIX では \$ORACLE_HOME/network/trace、Windows プラットフォームでは `ORACLE_HOME\network\trace` です。クライアントの場合、ログ・ファイルとトレース・ファイルのデフォルトの場所は、現行の作業ディレクトリです。

エラー情報のロギング

Oracle ネットワーク製品で発生するすべてのエラーは、ネットワーク管理者やデータベース管理者が評価できるように、ログ・ファイルに追加記録されます。画面に表示されるエラー・メッセージが不適切でエラーを理解できない場合、管理者はログ・ファイルから追加情報を得ることができます。エラー・スタックとしてのログ・ファイルは、各種レイヤーでのソフトウェアの状態を示します。

すべてのエラーが記録されていることを保証しているため、クライアントまたはネーム・サーバーに対するロギングを使用禁止にすることはできません。さらに、ログ・ファイルを置換したり消去できるのは管理者のみです。リスナーのログ・ファイルには、すべてのクライアント接続要求に関する監査証跡と大半のリスナー制御コマンドが含まれています。

この項の内容は、次のとおりです。

- [エラー・スタック](#)
- [ログ・ファイル名](#)
- [ログ・パラメータの理解と設定](#)
- [制御ユーティリティ実行時のログ設定](#)
- [ログ・ファイルの使用](#)
- [リスナーのログ監査証跡](#)
- [Oracle Connection Manager ログの理解](#)

エラー・スタック

ログ・ファイルには、エラー・スタックに含まれている情報があります。エラー・スタックとは、ネットワーク・エラーの結果として Oracle 通信スタック内に各レイヤーごとに作成される情報です。

図 11-1 は、エラー・スタックに表示される可能性のある Oracle ネットワーク製品間の関係を示したものです。

図 11-1 ネットワーク製品とエラー・スタック・コンポーネント

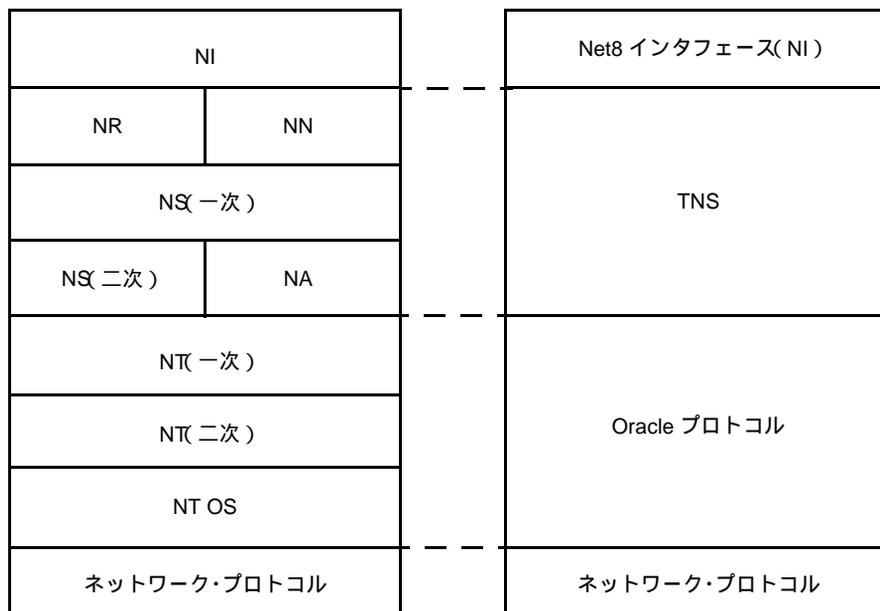


図 11-1 のレイヤーは次のとおりです。

NI	Net8 インタフェース・レイヤー
NR	ネットワーク・ルーティング
NN	ネットワーク・ネーミング (Oracle Names)
NS	ネットワーク・セッション (一次および二次層)
NA	ネットワーク認証 (NA) とネットワーク暗号化 (NAE) など、固有のサービス
NT	ネットワーク・トランスポート (一次、二次およびオペレーティング・システム層)

ネットワークによって、これらすべてのコンポーネントが含まれている場合と、含まれていない場合があります。

エラー例

たとえば、クライアント・アプリケーションのユーザーが Net8 と TCP/IP を使用してデータベース・サーバーとの接続を確立しようとする場合は、次のように入力します。

```
sqlplus scott/tiger@hrserver.com
```

SQL*Plus バナーが画面に表示され、次のエラーが表示されます。

```
ORA-12203: TNS:Unable to connect to destination
```

このメッセージは、データベースに連絡できなかったために、サーバーへの接続に失敗したことを示します。アプリケーションでは 1 行のエラー・メッセージしか表示されませんが、ネットワーク層によってログ・ファイルの中に情報量の多いエラー・スタックが記録されます。クライアント側では、SQLNET.LOG と呼ばれるログ・ファイルに次のような ORA-12203 エラーに対応するエラー・スタックが含まれています。

図 11-2 一般的なエラー・スタック

```
*****
```

```
Fatal OSN connect error 12203, connecting to:
  (DESCRIPTION=(CONNECT_DATA=(SID=trace)(CID=(PROGRAM=)
    (HOST=lala)(USER=sviavant)))(ADDRESS_LIST=(ADDRESS=
    (PROTOCOL=ipc)(KEY=trace))(ADDRESS=(PROTOCOL=tcp)
    (HOST=lala)(PORT=1521))))
```

```
VERSION INFORMATION:
```

```
TNS for SunOS:
```

```
Oracle Bequeath NT Protocol Adapter for SunOS:
```

```
Unix Domain Socket IPC NT Protocol Adaptor for SunOS:
TCP/IP NT Protocol Adapter for SunOS:
  Tracing to file: /home/sviavant/trace_admin.trc
  Tns error struct:
    nr err code: 12203
    TNS-12203: TNS:unable to connect to destination
    ns main err code: 12541
    TNS-12541: TNS:no listener
    ns secondary err code: 12560
    nt main err code: 511
    TNS-00511: No listener
    nt secondary err code: 61
    nt OS err code: 0
```

ログ・ファイル名

各 Net8 コンポーネントは、独自のログ・ファイルを作成します。次の表は、デフォルトのファイル名とそれに含まれている情報の説明を示したものです。

ログ・ファイル	エラー情報の内容
SQLNET.LOG	クライアントまたはサーバー、またはその両方
LISTENER.LOG	リスナー
NAMES.LOG	Oracle Names Server
CMAN_PID.TRC	Oracle Connection Manager CMGW ゲートウェイ・プロセス
CMADM_PID.TRC	Oracle Connection Manager CMADMIN 管理プロセス

ログ・パラメータの理解と設定

ロギングされた情報の種類と量、ファイルが格納されている位置など、ロギングを制御するパラメータは、次のように各ネットワーク・コンポーネントの構成ファイル内に設定されます。

ネットワーク・コンポーネント	パラメータを設定する構成ファイル
クライアント	SQLNET.ORA
サーバー	SQLNET.ORA
リスナー	LISTENER.ORA
Oracle Names Server	NAMES.ORA
Oracle Connection Manager プロセス	CMAN.ORA

この項の内容は、次のとおりです。

- [SQLNET.ORA パラメータ](#)
- [LISTENER.ORA パラメータ](#)
- [NAMES.ORA パラメータ](#)
- [CMAN.ORA パラメータ](#)
- [構成ファイルのログ・パラメータ設定](#)

この項で説明するパラメータの詳細は、[付録 C の「構成パラメータ」](#)を参照してください。

SQLNET.ORA パラメータ

SQLNET.ORA に設定されるパラメータは次のとおりです。

SQLNET.ORA パラメータ	Oracle Net8 Assistant オプション	説明
LOG_DIRECTORY_CLIENT	Client Information: Log Directory	クライアント・ログ・ファイルの作成先となるディレクトリを設定する。デフォルトのクライアント・ディレクトリは、現行の作業ディレクトリです。
LOG_FILE_CLIENT	Client Information: Log File	クライアントのログ・ファイルの名前を設定する。デフォルトのログ・ファイル名は SQLNET.LOG です。
LOG_DIRECTORY_SERVER	Server Information: Log Directory	サーバー・ログ・ファイルの作成先となるディレクトリを設定する。デフォルトのサーバー・ディレクトリは、UNIX では \$ORACLE_HOME/network/log、Windows NT では ORACLE_HOME¥network¥log です。
LOG_FILE_SERVER	該当なし	サーバーのログ・ファイルの名前を設定する。デフォルトのログ・ファイル名は SQLNET.LOG です。

LISTENER.ORA パラメータ

LISTENER.ORA に設定されるパラメータは次のとおりです。

LISTENER.ORA パラメータ	Oracle Net8 Assistant オプション	説明
LOG_DIRECTORY_ <i>listener_name</i>	ログ・ファイル	リスナー・イベントで自動生成されるファイルの作成先となるディレクトリとログ・ファイルを設定する。デフォルトのディレクトリは、UNIX では \$ORACLE_HOME/network/log、Windows NT では ORACLE_HOME\network\log です。また、ファイル名は LISTENER.LOG です。
LOG_FILE_ <i>listener_name</i>		

NAMES.ORA パラメータ

NAMES.ORA に設定されるパラメータは次のとおりです。

NAMES.ORA パラメータ	Oracle Net8 Assistant オプション	説明
NAMES.LOG_DIRECTORY	Log Directory	ログ・ファイルの作成先となるディレクトリを設定する。デフォルトのディレクトリは、UNIX では \$ORACLE_HOME/network/log、Windows NT では ORACLE_HOME\network\log です。
NAMES.LOG_FILE	ログ・ファイル	クライアントのログ・ファイルの名前を設定する。デフォルトのログ・ファイル名は NAMES.LOG です。

CMAN.ORA パラメータ

CMAN.ORA に設定されるパラメータは次のとおりです。

CMAN.ORA パラメータ	説明
LOG_LEVEL	<p>ロギングのレベルを設定する。</p> <ul style="list-style-type: none">■ 範囲は 0 から 4 <p>レベル 0 - ロギングなし レベル 1 - 基本的なレポートの作成 レベル 2 - RULE_LIST 一致検索レポートの作成 レベル 3 - リレー・ブロッキング・レポートの作成 レベル 4 - リレー I/O カウント・レポートの作成</p> <p>CMGW ゲートウェイ・プロセスと CNADMIN 管理プロセスではそれぞれ、CMAN_PID.LOG および CMADM_PID.LOG というログ・ファイルを作成する。これらのログ・ファイルは、UNIX では \$ORACLE_HOME/network/log、Windows NT では ORACLE_HOME¥network¥log にあります。</p>

構成ファイルのログ・パラメータ設定

SQLNET.ORA、LISTENER.ORA および NAMES.ORA のロギング・パラメータは、Oracle Net8 Assistant を使用して設定できます。CMAN.ORA のロギング・パラメータは、C-77 ページの「Oracle Connection Manager パラメータ (CMAN.ORA)」にある説明に従って、手動で CMAN.ORA に設定する必要があります。ロギング・パラメータを設定するには、次の手順に従います。

1. Oracle Net8 Assistant を起動します。
 - UNIX の場合、\$ORACLE_HOME/bin の下にある netasst を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」-> 「Oracle - HOME_NAME」-> 「Network Administration」-> 「Net8 Assistant」を選択します。
2. ログ・パラメータを指定します。

ログ・ファイル	ロギング・パラメータの設定方法
SQLNET.LOG	<ol style="list-style-type: none"> 1. 「Profile」アイコンをクリックします。 2. ドロップダウン・リスト・ボックスから「General」を選択します。 3. 「Logging」タブをクリックします。 4. 設定を指定します。
LISTENER.LOG	<ol style="list-style-type: none"> 1. Listeners フォルダをクリックします。 2. リスナーを選択します。 3. ドロップダウン・リスト・ボックスから「General Parameters」を選択します。 4. 「Logging and Tracing」タブをクリックします。 5. 設定を指定します。
NAMES.LOG	<ol style="list-style-type: none"> 1. Oracle Names Servers フォルダをクリックします。 2. Oracle Names Server を選択します。 3. ドロップダウン・リスト・ボックスから「Configure Server」を選択します。 4. 「Advanced」タブをクリックします。 5. ログ・ディレクトリとファイル名を指定します。

3. 「File」メニューの「Save Network Configuration」を選択します。
4. 「File」メニューの「Exit」を選択します。

Oracle Net8 Assistant アプリケーションが終了します。

制御ユーティリティ実行時のログ設定

ロギングは、制御ユーティリティの実行中に設定できます。制御ユーティリティによるロギング設定では、*.ORA ファイルにパラメータを設定しません。設定は、制御ユーティリティの現在のセッションでのみ有効です。

- リスナーの場合、LSNRCTL 制御ユーティリティから SET LOG_FILE コマンドや SET LOG_DIRECTORY コマンドを使用します。実行方法は A-3 ページの「[リスナー制御ユーティリティ \(LSNRCTL\)](#)」を参照してください。
- Oracle Names Server の場合、A-30 ページの「[Oracle Names 制御ユーティリティ \(NAMESCTL\)](#)」にある説明に従って、NAMESCTL 制御ユーティリティから SET LOG_FILE_NAME コマンドを使用するか、Oracle Net8 Assistant から次のように設定します。
 - a. Oracle Net8 Assistant を起動します。

UNIX の場合、\$ORACLE_HOME/bin の下にある netasst を実行します。

Windows NT の場合、「スタート」ボタン -> 「プログラム」 -> 「Oracle - HOME_NAME」 -> 「Network Administration」 -> 「Net8 Assistant」を選択します。
 - b. Oracle Names Servers フォルダをクリックします。
 - c. Oracle Names Server を選択します。
 - d. ドロップダウン・リスト・ボックスから「Manage Server」を選択します。
 - e. 「Logging」タブをクリックします。
 - f. ログ・ディレクトリとファイル名を指定します。

注意： Oracle Connection Manager の場合、ロギングの設定は、C-77 ページの「[Oracle Connection Manager パラメータ \(CMAN.ORA\)](#)」にある説明に従って、Oracle Connection Manager 構成ファイルから行う以外に方法はありません。

- g. 「File」メニューの「Save Network Configuration」を選択します。
 - h. 「File」メニューの「Exit」を選択します。
- Oracle Net8 Assistant アプリケーションが終了します。

ログ・ファイルの使用

ログ・ファイルを使用してネットワーク・エラーを診断するには、次の手順に従います。

1. アプリケーションから受信した最新のエラー番号のログ・ファイルを見直します。ほとんどの場合、これがログ・ファイルの最後のエントリです。
2. ファイルの一番下から順にたどり、エラー・レポートで0以外の最初のエントリを探します。通常は、これが実際の原因です。
3. そのエラーが希望の情報を示していない場合は、正しいエラー情報が見つかるまでスタック内の次のエラーを見直します。
4. それでもエラーの原因が明らかにならない場合は、トレースをオンにし、エラー・メッセージが発生したステートメントを再度実行します。

リスナーのログ監査証跡

リスナー・ログ・ファイルには、ネットワーク利用度の統計を収集および分析できる監査証跡情報と、次の内容を示す情報があります。

- クライアントの接続要求
- リスナー制御ユーティリティが発行した start または stop、status、reload、service の各コマンド

この機能をオフに切り替えることはできません。

リスナーのログ監査証跡の書式

監査証跡は、次の各フィールド、すなわち、*Timestamp*、*Connect Data*、*[Protocol Info]*、*event*、*[SID | service]*、*Return Code* 内のテキストを書式設定します。監査証跡のプロパティは次のとおりです。

- 各フィールドは、アスタリスク (*) で区切ります。
- PROTOCOL INFO と SID | SERVICE は、接続が試行された場合にのみ現れます。
- 成功した接続またはコマンドには、コード 0 が戻ります。
- 失敗した場合は、エラー・メッセージにマップされるコードが生成されます。

reload 要求に対するログ・ファイルへの一般的な出力は次のようになります。

図 11-3 成功した reload 要求の一般的な監査証跡情報

```
28-dec-98 14:12:18 * (connect_data=(service=sales.com)(cid=
(program=)(host=sales=pc)(user=system)) (command=reload) (arguments=64)
(service=listener) (version=(version=135282688))* reload * 0
```

connection 要求に対するログ・ファイルへの一般的な出力は次のようになります。

図 11-4 成功した connection 要求の一般的な監査証跡情報

```
28-dec-98 14:16:21 * (connect_data=(service=sales.com)(cid=
(program=c:\ora81\bin\sqlplus.exe)
(host=windowspc)(user=dsteiner))) * (address=(protocol=tcp)
(host=144.25.23.246)(port=3366))
* establish * sales.com * 0
```

接続に使用されたプラットフォームおよびプロトコル、ソフトウェアとともにユーザー ID が記録されることに注意してください。

監査証跡情報の使用

監査証跡情報を表に保存した後、それをレポート形式に書式設定することで、傾向とユーザー・アクティビティを検証できます。データを表にインポートするには、SQL*Loader などのインポート・ユーティリティを使用します。

Oracle Connection Manager ログの理解

Oracle Connection Manager の CMGW ゲートウェイ・プロセスと CNADMIN 管理プロセスではそれぞれ、CMAN_PID.LOG および CMADM_PID.LOG というログ・ファイルを作成します。これらのログ・ファイルは、UNIX では \$ORACLE_HOME/network/log、Windows NT では ORACLE_HOME¥network¥log にあります。

図 11-5 CMAN_PID.LOG

```
(TIMESTAMP=30-DEC-98 18:03:10)(EVENT=10)(VERSION=8.1.4.0.0)
(TIMESTAMP=30-DEC-98 18:03:10)(EVENT=36)(rule_list=
(rule=(src=spcstn)(dst=x)(srv=x)(act=accept)))
(TIMESTAMP=30-DEC-98 18:03:10)(EVENT=32)(PARAMETER_LIST=(MAXIMUM_RELAYS=1024)(RELAY_
STATISTICS=no)(AUTHENTICATION_LEVEL=0)(LOG_LEVEL=1)(SHOW_TNS_INFO=no)(ANSWER_
TIMEOUT=0)(MAXIMUM_CONNECT_DATA=1024)(USE_ASYNC_CALL=yes)(TRACING=no)(TRACE_
DIRECTORY=default)(MAX_FREELIST_BUFFERS=0))
(TIMESTAMP=30-DEC-98 18:03:10)(EVENT=34)(ADDRESS_LIST=
(ADDRESS=(PROTOCOL=tcp)(HOST=)(PORT=1610)(QUEUE_SIZE=32)))
(TIMESTAMP=30-DEC-98 18:03:12)(EVENT=38)(COMMAND=2)
(TIMESTAMP=30-DEC-98
18:03:27)(EVENT=26)(RLYNO=0)(SRC=(ADDRESS=(PROTOCOL=tcp)(HOST=spcstn.us.oracle.com)(
PORT=34758)))(DST=(ADDRESS=(PROTOCOL=tcp)(HOST=144.25.187.89)(PORT=1581)))
(TIMESTAMP=30-DEC-98 18:03:43)(EVENT=28)(RLYNO=0)(SINCE=30-DEC-98
18:03:27)(STATISTICS=(IN=(BYTES=0)(PACKETS=0)(DCDS=0)(OUT=(BYTES=0)(PACKETS=0)(DCDS=
0))))
```

図 11-6 CMADM_PID.LOG

```
(TIMESTAMP=30-DEC-98 18:03:09)(EVENT=Sent Admin Status to UI)
(TIMESTAMP=30-DEC-98 18:03:10)(EVENT=CMan Registration)
```

CMAN_PID.LOG のイベント・コード

CMAN_PID.LOG では、イベント・コード番号を使用してイベントを報告します。この項では、それぞれのイベント・コードについて説明します。報告されるイベント・コードは CMAN.ORA の LOG_LEVEL パラメータに設定されるログ・レベルによって異なります。

ログ・レベル1のイベント

コード	説明
10	ゲートウェイの起動中
12	ゲートウェイのシャットダウン中
14	TNS アドレスでリスニングを実行中
18	応答エラー 追加情報: 11-29 ページの「 イベント・コード 18 の原因 」を参照してください。
20	受信コールを拒否 追加情報: 11-29 ページの「 イベント・コード 20 の原因 」を参照してください。
26	リレーは現在オープンしている
28	リレーが現在クローズしている
30	統計情報レポート
32	<parameter_list>
34	<address_list>
36	<parameter_list>
38	CMCTL コマンド
40	ゲートウェイがビジーのため、CMCTL コマンドを拒否
42	デッド接続を検出
44	リレーのタイムアウト
11	<address_list> の引数に誤りがある
13	<parameter_list> の引数に誤りがある
15	<rule_list> の引数に誤りがある
23	CMCTL レコードの不良
25	コマンド行引数が長すぎる

コード	説明
27	メモリの割当てエラー
29	TNS エラー
31	CMCTL の実行中に TNS エラーが発生

イベント・コード 18 の原因 応答エラーには次の原因が考えられます。

コード	説明
1	タイムアウト
2	接続データ・バッファの容量不足
3	TNS による拒否
4	TNS パケットのチェックサム・エラー

イベント・コード 20 の原因 受信コールが拒否されるのは次の場合です。

コード	説明
1	ゲートウェイのシャットダウン中
3	受信コールに接続データが含まれていない
4	受信コールの接続データに誤りがある
5	すべてのリレーが使用中
6	リレー・バッファを取得できない
7	TNS の致命的エラー
8	ASO サービスが使用不可
9	フィルタ規則による拒否
10	送信コールエラー
11	Net8/TNS による拒否
12	リスナーが動作していない
13	リスナーに到達できていない
14	ホスト名検索エラー
15	プロトコル・アダプタ (およびプロトコル・スタック) がロードされていない

コード	説明
16	TNSNAMES.ORA または Oracle Names Server に SOURCE_ROUTE が設定されていない
17	規則または接続文字列データ不良のため拒否

ログ・レベル2のイベント

コード	説明
102	受信コールへの応答
104	送信コールの作成
105	受信コールの受入れ
106	規則合致レポート

ログ・レベル3のイベント

コード	説明
202	コールがブロックされる（非同期 TNS サポートされていない）
204	リレーのブロック 追加情報：「イベント・コード 204 の原因」を参照してください。
206	バッファに残余データが含まれている

イベント・コード 204 の原因

コード	説明
1	書き込み側の準備を待機中
2	書き込み側の未処理分消去を待機中
3	回復時の EWOULDBLOCK エラー
4	TNS パケットのチェックサム・エラー
5	送信時の EWOULDBLOCK または EPARTIAL エラー

ログ・レベル4のイベント

コード	説明
302	このバイト数を読み取る
304	このバイト数を書き込む
306	再試行時にこのバイト数を書き込む

エラー情報のトレース

トレース機能は、実行されたネットワーク・イベントが記述される一連の詳細ステートメントを生成します。操作をトレースすると、ログ・ファイルにある情報より多くの Net8 コンポーネントの内部操作に関する情報が得られます。この情報は、エラーの原因となったイベントの確認で評価できるファイル出力です。

注意： トレース機能には大量のディスク容量を消費するため、システムのパフォーマンスに大きな影響を与える可能性があります。したがって、必要なときにのみトレースを行えるようにしてください。

この項の内容は、次のとおりです。

- [トレース・ファイルの名前](#)
- [トレース・パラメータの理解と設定](#)
- [制御ユーティリティ実行時のトレース設定](#)
- [Net8 トレースの評価](#)
- [Trace Assistant を使用したトレース・ファイルの検査](#)

トレース・ファイルの名前

各 Net8 コンポーネントは、独自のトレース・ファイルを作成します。次の表は、デフォルトのファイル名とそれに含まれている情報の説明を示したものです。

ログ・ファイル	エラー情報の内容
SQLNET.TRC	クライアント
SVR_PID.TRC	サーバー
LISTENER.TRC	リスナー
NAMES.TRC	Oracle Names Server
CMAN_PID.TRC	Oracle Connection Manager CMGW ゲートウェイ・プロセス
CMADM_PID.TRC	Oracle Connection Manager CMADMIN 管理プロセス

トレース・パラメータの理解と設定

ロギングされた情報の種類と量、ファイルが格納されている位置など、トレース機能を制御するパラメータは、次のように各ネットワーク・コンポーネントの構成ファイル内に設定されます。

トレース対象のネットワーク・コンポーネント	パラメータを設定する構成ファイル
クライアント	SQLNET.ORA
サーバー	SQLNET.ORA
リスナー	LISTENER.ORA
Oracle Names Server	NAMES.ORA
Oracle Connection Manager プロセス	CMAN.ORA

この項の内容は、次のとおりです。

- [SQLNET.ORA パラメータ](#)
- [LISTENER.ORA パラメータ](#)
- [NAMES.ORA パラメータ](#)
- [CMAN.ORA パラメータ](#)
- [構成ファイルのトレース・パラメータ設定](#)

この項で説明するパラメータの詳細は、[付録 C の「構成パラメータ」](#)を参照してください。

SQLNET.ORA パラメータ

SQLNET.LOG に設定されるパラメータは次のとおりです。

SQLNET.ORA パラメータ	Oracle Net8 Assistant オプション	説明
TRACE_LEVEL_CLIENT	Client Information: Trace Level	<p>トレース機能によりクライアントで記録される詳細レベルを示す。トレース・レベルは、0 から 16 までの数値 (0 はトレースなし、16 は最大限のトレースを表す) か、OFF、ADMIN、USER または SUPPOER の値で指定します。</p> <ul style="list-style-type: none"> ■ OFF (0 と同じ) はトレースを実行しません。 ■ USER (4 と同じ) はユーザーによるエラー条件を特定するためのトレースを実行します。 ■ ADMIN (6 と同じ) はインストール固有の問題を特定するためのトレースを実行します。 ■ SUPPORT (16 と同じ) はサポートに対してトラブルシューティングのためのトレース情報を提供します。
TRACE_DIRECTORY_CLIENT	Client Information: Trace Directory	<p>クライアント・トレース・ファイルの作成先となるディレクトリを設定する。デフォルトのクライアント・ディレクトリは、UNIX では \$ORACLE_HOME/network/trace、Windows NT では ORACLE_HOME¥network¥trace です。</p>
TRACE_FILE_CLIENT	Client Information: Trace File	<p>クライアントのトレース・ファイルの名前を設定する。デフォルトのトレース・ファイル名は SQLNET.TRC です。</p>

SQLNET.ORA パラメータ	Oracle Net8 Assistant オプション	説明
TRACE_UNIQUE_CLIENT	Client Information: Unique Trace File Name	一意なトレース・ファイルが作成されているかどうかを判別する。デフォルト値は OFF です。すべてのクライアントで同一のトレース・ファイルが使用されます。値が OFF の場合、クライアントにトレース・ファイルが新規作成されると、既存のファイルに上書きされます。値を ON に設定すると、複数のトレース・ファイルが共存できるように、各トレース・ファイル名にはプロセス識別子が付加されます。 このパラメータで作成されるトレース・ファイルの一意名は、SQLNETPID.TRC です。
TRACE_LEVEL_SERVER	Server Information: Trace Level	トレース機能によりサーバーで記録される詳細レベルを示す。トレース・レベルは、0 から 16 までの数値 (0 はトレースなし、16 は最大限のトレースを表す) か、OFF、ADMIN、USER または SUPPORT の値で指定します。 <ul style="list-style-type: none"> ■ OFF (0 と同じ) はトレースを実行しません。 ■ USER (4 と同じ) はユーザーによるエラー条件を特定するためのトレースを実行します。 ■ ADMIN (6 と同じ) はインストレーション固有の問題を特定するためのトレースを実行します。 ■ SUPPORT (16 と同じ) はサポートに対してトラブルシューティングのためのトレース情報を提供します。
TRACE_DIRECTORY_SERVER	Server Information: Trace Directory	サーバー・トレース・ファイルの作成先となるディレクトリを設定する。デフォルトのサーバー・ディレクトリは、UNIX では \$ORACLE_HOME/network/trace、Windows NT では ORACLE_HOME\network\trace です。
TRACE_FILE_SERVER	Server Information: Trace File	クライアントのトレース・ファイルの名前を設定する。デフォルトのログ・ファイル名は、SVR_PID.TRC です。

TNSPING ユーティリティでは、SQLNET.ORA に次のオプション・トレース・パラメータを手動で追加することもできます。TNSPING では、Net8 ネットワーク上のサービス（データベース、Oracle Names Server またはその他の TNS サービスなど）が正常に到達できるかどうかを判定します。

SQLNET.ORA パラメータ	説明
TNSPING.TRACE_LEVEL	<p>トレース機能により TNSPING ユーティリティで記録される詳細レベルを示す。トレース・レベルは、0 から 16 までの数値（0 はトレースなし、16 は最大限のトレースを表す）か、OFF、ADMIN、USER または SUPPORT の値で指定します。</p> <ul style="list-style-type: none">■ OFF（0 と同じ）はトレースを実行しません。■ USER（4 と同じ）はユーザーによるエラー条件を特定するためのトレースを実行します。■ ADMIN（6 と同じ）はインストール固有の問題を特定するためのトレースを実行します。■ SUPPORT（16 と同じ）はサポートに対してトラブルシューティングのためのトレース情報を提供します。
TNSPING.TRACE_DIRECTORY	<p>TNSPING トレース・ファイルの作成先となるディレクトリを設定する。デフォルトのディレクトリは、UNIX では <code>\$ORACLE_HOME/network/trace</code>、Windows NT では <code>ORACLE_HOME\network\trace</code> です。</p>

LISTENER.ORA パラメータ

LISTENER.ORA に設定されるパラメータは次のとおりです。

LISTENER.ORA パラメータ	Oracle Net8 Assistant オプション	説明
TRACE_LEVEL_ <i>listener_name</i>	Trace Level	<p>トレース機能によりリスナーで記録される詳細レベルを示す。トレース・レベルは、0 から 16 までの数値 (0 はトレースなし、16 は最大限のトレースを表す) か、ADMINISTRATOR、USER または SUPPORT の値で指定します。</p> <ul style="list-style-type: none"> ■ USER (4 と同じ) はユーザーによるエラー条件を特定するためのトレースを実行します。 ■ ADMINISTRATOR (6 と同じ) はインストレーション固有の問題を特定するためのトレースを実行します。 ■ SUPPORT (16 と同じ) はサポートに対してトラブルシューティングのためのトレース情報を提供します。
TRACE_ DIRECTORY_ <i>listener_name</i>	トレース・ ファイル	<p>トレース・ファイルの作成先となるディレクトリとファイルを設定する。デフォルトのディレクトリは、UNIX では \$ORACLE_ HOME/network/trace、Windows NT では ORACLE_HOME¥network¥trace です。また、 ファイル名は LISTENER.TRC です。</p>
TRACE_FILE_ <i>listener_name</i>		

NAMES.ORA パラメータ

NAMES.ORA に設定されるパラメータは次のとおりです。

NAMES.ORA パラメータ	Oracle Net8 Assistant オプション	説明
NAMES.TRACE_DIRECTORY	Trace Directory	トレース・ファイルの作成先となるディレクトリを設定する。デフォルトのディレクトリは、UNIX では \$ORACLE_HOME/network/trace、Windows NT では ORACLE_HOME¥network¥trace です。
NAMES.TRACE_FILE	トレース・ファイル	トレース・ファイルの名前を設定する。デフォルトのログ・ファイル名は NAMES.TRC です。
NAMES.TRACE_UNIQUE	Make Trace File Unique	値を ON に設定すると、複数のトレース・ファイルが共存できるように、トレース・セッションごとに一意なファイル名が作成される。生成されるトレース・ファイルの名前には、プロセス識別子が付加されます。 このパラメータで作成されるトレース・ファイルの一意名は、NAMESPID.TRC です。

CMAN.ORA パラメータ

CMAN.ORA に設定されるパラメータは次のとおりです。

CMAN.ORA パラメータ	説明
TRACING	トレース機能を ON または OFF に設定する。
TRACE_DIRECTORY	トレース・ファイルの作成先となるディレクトリを設定する。デフォルトのディレクトリは、UNIX では \$ORACLE_HOME/network/trace、Windows NT では ORACLE_HOME¥network¥trace です。

構成ファイルのトレース・パラメータ設定

SQLNET.ORA、LISTENER.ORA および NAMES.ORA のロギング・パラメータは、Oracle Net8 Assistant を使用して設定できます。CMAN.ORA のロギング・パラメータは、C-77 ページの「[Oracle Connection Manager パラメータ \(CMAN.ORA\)](#)」にある説明に従って、手動で設定する必要があります。ロギング・パラメータを設定するには、次の手順に従います。

1. Oracle Net8 Assistant を起動します。
 - UNIX の場合、\$ORACLE_HOME/bin の下にある netasst を実行します。
 - Windows NT の場合、「スタート」ボタン -> 「プログラム」 -> 「Oracle - HOME_NAME」 -> 「Network Administration」 -> 「Net8 Assistant」を選択します。
2. トレース・パラメータを指定します。

ログ・ファイル	トレース・パラメータの設定方法
SQLNET.TRC (クライアント用) SVR_PID.TRC (サーバー用)	<ol style="list-style-type: none"> 1. 「Profile」アイコンをクリックします。 2. ドロップダウン・リスト・ボックスから「General」を選択します。 3. 「Tracing」タブをクリックします。 4. 設定を指定します。
LISTENER.TRC	<ol style="list-style-type: none"> 1. Listeners フォルダをクリックします。 2. リスナーを選択します。 3. ドロップダウン・リスト・ボックスから「General Parameters」を選択します。 4. 「Logging and Tracing」タブをクリックします。 5. 設定を指定します。
NAMES.TRC	<ol style="list-style-type: none"> 1. Oracle Names Servers フォルダをクリックします。 2. Oracle Names Server を選択します。 3. ドロップダウン・リスト・ボックスから「Configure Server」を選択します。 4. 「Logging」タブをクリックします。 5. 設定を指定します。

3. 「File」メニューの「Save Network Configuration」を選択します。
4. 「File」メニューの「Exit」を選択します。

Oracle Net8 Assistant アプリケーションが終了します。

制御ユーティリティ実行時のトレース設定

トレースは、制御ユーティリティの実行中に設定できます。制御ユーティリティによるトレース設定では、*.ORA ファイルにパラメータを設定しません。設定は、制御ユーティリティの現在のセッションでのみ有効です。

- リスナーの場合、LSNRCTL 制御ユーティリティから SET TRC_FILE コマンドや SET TRC_DIRECTORY コマンドを使用します。実行方法は A-3 ページの「[リスナー制御ユーティリティ \(LSNRCTL\)](#)」を参照してください。
- Oracle Names Server の場合、A-30 ページの「[Oracle Names 制御ユーティリティ \(NAMESCTL\)](#)」にある説明に従って、NAMESCTL 制御ユーティリティから SET TRACE_FILE_NAME コマンドや SET TRACE_LEVEL コマンドを使用するか、次のように Oracle Net8 Assistant から設定します。
 - a. Oracle Net8 Assistant を起動します。

UNIX の場合、\$ORACLE_HOME/bin の下にある netasst を実行します。

Windows NT の場合、「スタート」ボタン->「プログラム」->「Oracle - HOME_NAME」->「Network Administration」->「Net8 Assistant」を選択します。
 - b. Oracle Names Servers フォルダをクリックします。
 - c. Oracle Names Server を選択します。
 - d. ドロップダウン・リスト・ボックスから「Manage Server」を選択します。
 - e. 「Logging」タブをクリックします。
 - f. トレース・レベル、ディレクトリおよびファイル名を指定します。

注意： Oracle Connection Manager の場合、トレースの設定は、C-77 ページの「[Oracle Connection Manager パラメータ \(CMAN.ORA\)](#)」にある説明に従って、Connection Manager 構成ファイルから行う以外に方法はありません。

- g. 「File」メニューの「Save Network Configuration」を選択します。
 - h. 「File」メニューの「Exit」を選択します。
- Oracle Net8 Assistant アプリケーションが終了します。

Net8 トレースの評価

手動で、または Trace Assistant Tool を使用して、トレース・ファイルを評価すると、次のことが理解でき、ネットワークの診断やトラブルシューティングが容易に行えます。

- ネットワーク・ノード間のパケットの流れ
- Net8 のどのコンポーネントが障害に陥っているか
- 関連するエラー・コード

ネットワーク・ノード間のデータ・パケットの流れの理解

Net8 は、データ・パケットを送受信することにより、その機能を実行します。SUPPORT のトレース・レベルを指定すると、トレース・ファイル内の Net8 パケットの実際の内容を参照できます。送受信したパケット・タイプの順序により、接続を確立した方法を確認できます。

データ・パケット形式の理解 トレース・ファイルの各行は、後ろにメッセージが続くプロシージャで始まります。各プロシージャの後には、実際のデータを示す 16 進データの行があります。パケットの内部をフローする実際のデータは、16 進データの右側に表示される場合があります。

次の表は、Net8 パケット・キーワードの一覧と、それらのキーワードが表すパケット・タイプを示したものです。

キーワード	パケット・タイプ
NSPTCN	Connect
NSPTAC	Accept
NSPTRF	Refuse
NSPTRS	Resend
NSPDA	Data
NSPCNL	Control
NSPTMK	Marker

注意： Oracle ネットワーク製品または EBCDIC データを通じて暗号化を使用している場合、このデータは表示されません。

たとえば、次の行はネットワークを通じて NSPTCN パケットを送る `nscon` と呼ばれるプロシージャを記述しています。

```
nscon: sending NSPTCN packet
```

各パケットには、パケット・タイプを示すキーワードがあります。すべてのパケット・タイプは、接頭辞 NSP から始まります。特定のパケット情報についてトレース・ファイルを検討する場合、このことを覚えておくと便利です。

図 11-7 は、一般的なパケット情報を示したものです。

図 11-7 パケット情報

```

nscon: entry
nscon: doing connect handshake...
nscon: sending NSPTCN packet
npsend: entry
npsend: plen=187, type=1
npsend: 187 bytes to transport
npsend:packet dump
npsend:00 BB 00 00 01 00 00 00 |.....|
npsend:01 33 01 2C 0C 01 08 00 |.3.,....|
npsend:7F FF 7F 08 00 00 00 01 |.....|
npsend:00 99 00 22 00 00 08 00 |..."....|
npsend:01 01 28 44 45 53 43 52 |..(DESCR|
npsend:49 50 54 49 4F 4E 3D 28 |IPTION=(|
npsend:43 4F 4E 4E 45 43 54 5F |CONNECT_|
npsend:44 41 54 41 3D 28 53 49 |DATA=(SI|
npsend:44 3D 61 70 33 34 37 64 |D=ap347d|
npsend:62 31 29 28 43 49 44 3D |b1)(CID=|
npsend:28 50 52 4F 47 52 41 4D |(PROGRAM|
npsend:3D 29 28 48 4F 53 54 3D |=)(HOST=|
npsend:61 70 32 30 37 73 75 6E |ap207sun|
npsend:29 28 55 53 45 52 3D 6D |)(USER=m|
npsend:77 61 72 72 65 6E 29 29 |warren))|
npsend:29 28 41 44 44 52 45 53 |)(ADDRES|
npsend:53 5F 4C 49 53 54 3D 28 |S_LIST=(|
npsend:41 44 44 52 45 53 53 3D |ADDRESS=|
npsend:28 50 52 4F 54 4F 43 4F |(PROTOCO|
npsend:4C 3D 74 63 70 29 28 48 |L=tcp)(H|
npsend:4F 53 54 3D 61 70 33 34 |OST=ap34|
npsend:37 73 75 6E 29 28 50 4F |7sun)(PO|
npsend:52 54 3D 31 35 32 31 29 |RT=1521)|
npsend:29 29 29 00 00 00 00 00 |))).....|
npsend: normal exit
nscon: exit (0)

```

関連エラー出力の理解

Net8 に関して問題が発生する都度、エラー・コードが <ERROR> または <FATAL> の接頭辞を持つトレース・ファイルにログが記録されます。図 11-8 は、一般的なトレース・ファイルのエラー出力を示したものです。

図 11-8 トレース・ファイルのエラー出力

```
nspsend: entry
nspsend: plen=244, type=6
ntpwr: entry
ntpwr: exit
-<ERROR>- nspsend: transport write error
nspsend: error exit
nserror: entry
-<ERROR>- nserror: nsres: id=0, op=65, ns=12541, ns2=12560; nt[0]=511,
nt[1]=61,nt[2]=0
-<ERROR>- nsopen: unable to open transport
nricdt: Call failed...
nricdt: exit
-<ERROR>- osnqper: error from nricall
-<ERROR>- osnqper: nr err code: 12203
-<ERROR>- osnqper: ns main err code: 12541
-<ERROR>- osnqper: ns (2) err code: 12560
-<ERROR>- osnqper: nt main err code: 511
-<ERROR>- osnqper: nt (2) err code: 61
-<ERROR>- osnqper: nt OS err code: 0
osnqme: entry
osnqme: reporting nr (1) error: (12203) as rdbms err (12203)
osnqme: exit
-<ERROR>- onstns: Couldn't connect, returning 12203
nricall: Exiting NRICALL with following termination result -1
nricall: exit
osnqme: entry
osnqme: reporting nr (1) error: (12203) as rdbms err (12203)
osnqme: exit
-<ERROR>- onstns: Couldn't connect, returning 12203
-<ERROR>- osnqper: error from nricall
```

エラー・コードを評価する最も能率的方法は、記録されている最新の NS エラー・コードを評価することです。これは、セッション・レイヤーが接続を制御しているためです。最も重要なエラー・メッセージは、ファイルの最後部にあるメッセージです。それらは最新のエラーであり、接続上の問題の原因を示しています。

特定のリターン・コードについての詳細は、Oracle UNIX エラー・ツール `oerr` を使用してください。コマンド行プロンプトで次のように入力し、`oerrTool` を使用して Net8 リターン・コードに関するさらに詳細な情報を検索してください。

```
oerr tns error_number
```

Trace Assistant を使用したトレース・ファイルの検査

Net8 には、既存のトレース・ファイル・テキスト行をさらに読みやすいパラグラフに変換して、トレース・ファイルの情報を理解しやすくする Trace Assistant というツールがあります。Trace Assistant は、レベル 16 (SUPPORT) の SQL*Net または Net8 トレース・ファイルにのみ対応します。

Trace Assistant を実行するには、コマンド行プロンプトで次のように入力します。

```
trcasst [options] filename
```

オプションの説明は次のとおりです。

オプション	説明
-o	接続性と Two Task Common (TTC) 情報を表示する。-o に続いて、次のオプションを使用できます。 <ul style="list-style-type: none"> ■ c (サマリー接続情報) ■ d (詳細接続情報) ■ u (サマリー TTC 情報) ■ t (詳細 TTC 情報) ■ q (サマリー TTC 情報を拡張する SQL コマンドを表示する)
-p	オラクル社の内部使用オプション。
-s	統計情報を表示する。
-e	エラー情報の表示を可能にする。-e に続いて、次のエラー・レベルを指定できます。 <ul style="list-style-type: none"> ■ 0 またはなし (nerror 機能からダンプされた NS エラー番号を変換し、他のエラーすべてをリストする) ■ 1 (nerror 機能から NS エラー変換のみを表示する) ■ 2 (エラー番号を変換しないで表示する)

オプションが指定されない場合のデフォルトは、-odt -e -s (エラー・レベル 0 で、詳細接続情報および詳細 Two-Task Common 情報、統計情報を表示) です。

図 11-9 は、Trace Assistant がどのようにトレース・ファイル情報をさらに読みやすい形式に変換するかを示したものです。

図 11-9 Trace Assistant の一般的な変換結果

トレース・ファイル	オプション -e0 または -e1 を指定した Trace Assistant による変換結果
<pre>nsc2addr: normal exit nsopen: entry nsmal: 404 bytes at 0x10d5a48 nsopen: opening transport... -<ERROR>- ntus2err: sd=13, op=1, resnt[0]=511, resnt[1]=2, resnt[2]=0 -<ERROR>- nerror: nres: id=0, op=65, ns=12541, ns2=12560; nt[0]=511, nt[1]=2, nt[2]=0 -<ERROR>- nsopen: unable to open transport</pre>	<pre>Error found.Error Stack follows: id: 00000 Operation code: 00065 NS Error 1:12541 NS Error 2:12560 NT Generic Error: 00511 Protocol Error: 00146 OS Error: 00000 NS & NT Errors Translation 12541, 00000, "TNS: リスナーはありません。" // *Cause: The connection request could not be completed because the listener // is not running. // *Action: Ensure that the supplied destination address matches one of // the addresses used by the listener - compare the TNSNAMES.ORA entry with // the appropriate LISTENER.ORA file (or TNSNAV.ORA if the connection is to // go by way of an Interchange).Start the listener on the remote machine. / 12560, 00000, "TNS: プロトコル・アダプタ・エラー " // *Cause: A generic protocol adapter error occurred. // *Action: Check addresses used for proper protocol specification.Before // reporting this error, look at the error stack and check for lower level // transport errors.For further details, turn on tracing and re-execute the // operation.Turn off tracing when the operation is complete. / 00511, 00000, " リスナーがありません。" // *Cause: The connect request could not be completed because no application // is listening on the address specified, or the application is unable to // service the connect request in a sufficiently timely manner. // *Action: Ensure that the supplied destination address matches one of // the addresses used by the listener - compare the TNSNAMES.ORA entry with // appropriate LISTENER.ORA file. // Start the listener on the remote machine.</pre>

ただし、トレース・ファイル内に nerror 機能からログに記録されなかった他のエラーが存在する場合があります。

Net8 パケットでネットワークを横断する情報の理解

Trace Assistant では、Net8 と Two Task Common 通信レイヤーの両方からデータ・パケットを表示できます。Trace Assistant には、これらのパケットを表示するための2つのオプションがあります。

- サマリー接続性 (-oc オプションを使用)
- 詳細接続性 (-od オプションを使用)

Net8 パケットの例

次の例は、さまざまなトランザクションで各種パケットが Net8 レイヤーとの間で送受信される時、Trace Assistant がどのようにそれらのパケットを表示するかを示したものです。

- bequeath 接続
- リダイレクト接続
- データ・パケット

ここで、送受信されるパケットには、このノードが *nnn* 個のバイトを持つある種のパケットを送受信していることを示す接頭辞 "--->Send*nnn* bytes" または "<--- Received *nnn* bytes" があることに注意してください。これらの接頭辞によって、ノードがクライアントまたはサーバーのどちらであるかが判断できます。接続要求は常にクライアントによって送信され、サーバー（またはリスナー）によって受信されます。

図 11-10 bequeath 接続で送られたサマリー・データ・パケット

trcasst -oc <filename> を使用する場合

この例には2つのパケットが示されています。最初のパケットは、クライアントからリスナーに送信される接続パケットです。2番目のパケットは、サーバーから戻される受諾パケットです。

```
---> Send      192 bytes - Connect packet
      Connect data length: 142
      (description=(address=(protocol=tcp)(host=dlsun)(port=1521))(connect_
data=(sid=dbl)(cid=(program=)(host=dlsun)(user=use1))))
<--- received 24 bytes - accept packet
      Accept data length: 0
```

図 11-11 bequeath 接続で送られた詳細データ・パケット**trcasst -od <filename> を使用する場合**

この例は、接続交渉中に接続データとともに送信されたすべての詳細を示しています。

```
---> Send      50 bytes - Connect packet
Current NS version number is: 309.
Lowest NS version number can accommodate is: 300.
Global options for the connection:
    can receive attention
    no attention processing
    Don't care
    Maximum SDU size: 2048
    Maximum TDU size: 5120
    NT protocol characteristics:
        Test for more data
        Spawner is running
        Hang on to Listener connection
        Full duplex I/O
        Urgent data support
        Generate SIGURG signal
        Handoff connection to another
    Line turnaround value: 0
    Connect data length: 234
    Connect data offset: 50
    Connect data maximum size: 2048
        Native Services wanted
        Native Services wanted
Cross facility item 1: 0
    Cross facility item 2: 0
    Connection id: 0x0000000000000000
    Packet data is in the following data packet
---> Send      244 bytes - Data packet
(DESCRIPTION=(ADDRESS=(PROTOCOL=beq)(PROGRAM=/private/oracle/bin/oracle)(ARGV0=oracle)(A
RGS='(DESCRIPTION=(LOCAL=YES) (ADDRESS=(PROTOCOL=beq))))(DETACH=NO))(CONNECT_
DATA=(CID=(PROGRAM=)(HOST=dlsun)(USER=use1))))
<--- Received 24 bytes - Accept packet
    Accepted NS version number is: 307.
Global options for the connection:
    no attention processing
    Don't care
    Accepted maximum SDU size: 2048
    Accepted maximum TDU size: 4096
    Connect data length: 0
        Native Services wanted
        Native Services wanted
```

図 11-12 リダイレクト接続で送られたサマリー・データ・パケット

trcasst -oc <filename> を使用する場合

```
---> Send      187 bytes - Connect packet
      Connect data length: 153
      (DESCRIPTION=(CONNECT_
DATA=(SID=ap347db1)(CID=(PROGRAM=)(HOST=apsun)(USER=use2)))(ADDRESS_
LIST=(ADDRESS=(PROTOCOL=tcp)(HOST=apsun)(PORT=1521))))
<--- Received 8 bytes - Resend packet
---> Send      187 bytes - Connect packet
      Connect data length: 153
      (DESCRIPTION=(CONNECT_DATA=(SID=apdb1)(CID=(PROGRAM=)(HOST=apsun)(USER=use2)))(ADDRESS_
LIST=(ADDRESS=(PROTOCOL=tcp)(HOST=apsun)(PORT=1521))))
<--- Received 24 bytes - Accept packet
      Accept data length: 0
```

図 11-13 データ・パケット

trcasst -oc <filename> または -od <filename> を使用する場合

接続が確立された後、他のノードに送信されるデータが Two-Task Common 層から Net8 に与えられます。サマリー・ビューと詳細ビューはともに、同じサマリー情報を生成します。

```
---> Send      30 bytes - Data packet
<--- Received 201 bytes - Data packet
---> Send      439 bytes - Data packet
<--- Received 400 bytes - Data packet
```

Two Task Common パケットの例

Two-Task Common は、データベースを対象とするカーソルのオープン、行の選択、行の更新などの要求を処理します。すべての要求は、サーバーによって応答されます。ログオンを要求した場合は、その要求を完了させたデータベースから応答が戻ります。11-48 ページの [図 11-14](#) から 11-51 ページの [図 11-16](#) は、予測不可能な情報のタイプを示したものです。

Two-Task Common のサマリー情報は、各行に 1 つのパケットでなく 2 つのパケットを表示するという点で、他の表示とは異なります。これを行うのは、Two-Task Common が動作するときの要求 / 応答のペアリング・プロセスをミラー化するためです。

図 11-14 Two Task Common のサマリー情報**trcasst -ou <filename> を使用する場合**

(O3LOGA)	1st half of challenge-response logon	80	78
(O3LOGON)	2nd half of challenge-response logon	97	59
(OOPEN)	# 1	21	16
(OPARSEX)	# 1	245	59
(OCLOSE)	# 1	17	11
(OVERSION)		29	16
(OOPEN)	# 2	21	16
(OALL7)	# 2 Parse Can Defn=2 Exec Fetch "SELECT A.V	268	100
(OOPEN)	# 3	21	16
(OALL7)	# 3 Parse Exec=1 "SELECT USER FROM SYS.DUAL	152	70
(OALL7)	# 3 Defn=1 Fetc	117	88
(OCLOSE)	# 3	17	11

各行では、最初に表示される項目が実際に作成される要求です。2 番目の項目は、それがトランザクションに関連する場合はカーソル番号です。3 番目の項目は、フラグ・リストまたは応答中の SQL コマンドです。フラグは、要求に次の特性があることを示します。

!PL/SQL=PL/SQL 要求ではない

COM= コミット

IOV=I/O ベクター取得

DEFN= 定義

EXEC= 実行

FETCH= フェッチ

CAN= 取消

DESCSEL= 選択の記述

DESCBND= バインドの記述

BND= バインド

PARSE= 解析

EXACT= 強制

右端には、送受信されたバイト数が表示されます。

3行目の OOPEN は、出力がどのように要求 / 応答ペアを表示するかを示す最初の例です。OOPEN は後ろに #1 を付けて表示されますが、それはクライアントからカーソルのオープン要求が送信され、サーバーがオープンしたカーソルの番号である 1 で応答していることを示します。要求 / 応答のペアは 1 行に配置されるため、このオプションと他の接続オプションを組み合わせず使用しないでください。

図 11-15 Two Task Common のサマリー情報

trcasst -ot <filename> を使用する場合

```
start of user function (TTIFUN)
  1st half of challenge-response logon (O3LOGA)
    Username: applsys
    Terminal: ttyp5
    Machine: ap207sun
    System User: mwarren
    Process: 24459
    Program: aiap45@ap207sun (TNS interface)
return opi parameter (TTIRPA)
  OPI parameter: 3309B1A977A62A3C
start of user function (TTIFUN)
  2nd half of challenge-response logon (O3LOGON)
    Username: applsys
    Terminal: ttyp5
    Machine: ap207sun
    System User: mwarren
    Process: 24459
    Program: aiap45@ap207sun (TNS interface)
ORACLE function complete (TTIOER)
start of user function (TTIFUN)
  Open a cursor
return opi parameter (TTIRPA)
  Cursor #: 1
start of user function (TTIFUN)
  Parse and Execute (OPARSEX) Cursor # 1
alter session set nls_language= 'AMERICAN' nls_territory= 'AMERICA' nls_currency=
'$' nls_iso_currency= 'AMERICA' nls_numeric_characters= '.,' nls_date_format=
'DD-MON-YY' nls_date_language= 'AMERICAN' nls_sort= 'BINARY'
ORACLE function complete (TTIOER)
start of user function (TTIFUN)
  Close cursor (OCLOSE) Cursor # 1
V6 Oracle func complete (TTISTA)
  Succeeded
```

図 11-16 サマリー Two-Task の最上部にある詳細 SQL 情報

trcasst -ouq <filename> を使用する場合

(O3LOGA)	1st half of challenge-response logon	180	78
(O3LOGON)	2nd half of challenge-response logon	197	59
(OOPEN)	# 1	21	16
(OPARSEX)	# 1 alter session set nls_language= 'AMERICAN' nls_ territory= 'AMERICA' nls_currency= '\$' nls_iso_ currency= 'AMERICA' nls_numeric_characters= ',' nls_ date_format= 'DD-MON-YY' nls_date_language= 'AMERICAN' nls_sort= 'BINARY'	245	59
(OCLOSE)	# 1	17	11
(O71SESOPN)	(get session ID)	47	18
(OOPEN)	# 1	21	16
(OVERSION)	Oracle7 Server Release 8.1.3.0.0 - Production Release with the distributed and parallel query optionsPL/SQL Release 2.1.4.0.0 - Production	29	157
(O71SESOPN)	(get session ID)	47	18

詳細 Two-Task オプションに自動的に与えられる詳細 SQL 情報を表示するには、サマリー Two-Task コマンドに q を追加します。

適切な統計に収集されたデータの分析

収集される統計の種類は、ネットワーク・パートナー間で送受信されたコール (TTC)、パケットおよびバイトの数に関するものです。次の例は、代表的なトレース・ファイルの統計を示したものです。

図 11-17 一般的なトレース・ファイルの統計

trcasst -s <filename> を使用する場合

```
=====
Trace File Statistics:
-----
SQL*Net:
  Total Calls:      466 sent,      491 received,      423 upi
  Total Bytes:     119214 sent,    86614 received
  Average Bytes:   255 sent,      176 received
  Maximum Bytes:   2048 sent,      2048 received
GRAND TOTAL PACKETS sent:  466      received:  491
-----
```

トレース・ファイルの例

次の例は、デコードされた完全なトレース・ファイルを示したものです。この例は、Oracle Client Tool SVRMGRL で次の要求により作成されました。

```
connect scott/tiger@june
```

メッセージ「ORA-12154: TNS:could not resolve service name」が画面に表示されました。

図 11-18 トレース・ファイルの例

説明	トレース・ファイルの情報
Trace Configuration Information セクション内のトレース・ファイルのトレース・レベルと位置に注意してください。	<pre>-- TRACE CONFIGURATION INFORMATION FOLLOWS --- New trace stream is "C:¥ORANT¥network¥trace¥sqlnet7.trc" New trace level is 16 --- TRACE CONFIGURATION INFORMATION ENDS ---</pre>
Oracle Names コンポーネントは、ネット・サービス名 JUNE.COM を検出できません。クライアントがネット・サービス名 JUNE に拡張子 .COM を追加することに注意してください。	<pre>nnfotran: tnsname.ora entry for name "june.com" not found nnftqrm: Error querying june.com of attribute A.SMD errcode 406 nnfgrwsp: Query unsuccessful, skipping to next adapter</pre>
クライアントは、ネット・サービス名アドレスを解決するためにネーム・サーバー (oranamesrvr0) にアクセスしようとしています。	<pre>nnfgrwsp: Switching to ONAMES adapter nnfgrwsp: Original name: june nnfgrwsp: Qualified name: june.com nngsget_get_stream: looking for "(DESCRIPTION=(CONNECT_ DATA=(RPC=ON))(ADDRESS=(PROTOCOL=tc)(HOST=oranamesrvr0)(PORT=1575)))" nngsget_get_stream: cache miss, opening new stream nngsnad_new_stream_addr: "(DESCRIP TION=(CONNECT_ DATA=(RPC=ON))(ADDRESS=(PROTOCOL =tc)(HOST=oranamesrvr0)(PORT=1575)))" nngsget_get_stream: no caller address will be sent to callee</pre>

図 11-18 トレース・ファイルの例

説明	トレース・ファイルの情報
ネットワーク・ルーティング (NR) は、Oracle Names Server (oramesrvr0) へのルーティングを行います。	<pre> nricall: entry nric2a: entry nric2a: Getting local community information nriglp: entry nriglp: Looking for local addresses setup by nrigla nriglp: No addresses in the preferred address list nriglp: exit nric2a: TNSNAV.ORA is not present. No local communities entry. nrigla: entry nrigla: Getting local address information nrigla: Simple address... nrigla: No community component so just use straight address nrigla: exit nridst: entry nridst: Resolving address to use to call destination or next hop nridst: Found destination address nridst: Local address nridst: Local destination community found nridst: exit nric2a: This is a local community access nric2a: exit nricall: Got routable address information nricall: Making call with following address information: (DESCRIPTION=(CONNECT_ DATA=(RPC=ON))(ADDRESS=(PROTOCOL=tcp)(HOST=oramesrvr0)(PORT =1575))) nricdt: entry nricdt: Calling with outgoing connect data (DESCRIPTION=(CONNECT_ DATA=(RPC=ON))(ADDRESS=(PROTOCOL=tcp)(HOST=oramesrvr0)(PORT =1575))) </pre>
ネットワーク・セッション (NS) は、Oracle Names Server にセッションを設定します。	<pre> nscall: entry nscall: connecting... nsc2addr: entry nsc2addr: (DESCRIPTION=(CONNECT_ DATA=(RPC=ON))(ADDRESS=(PROTOCOL=tcp)(HOST=oramesrvr0)(PORT =1575))) </pre>

図 11-18 トレース・ファイルの例

説明	トレース・ファイルの情報
ネットワーク・トランスポート (NT) は、トランスポート・セッションを設定します。	<pre> nttbnd2addr: entry nttbnd2addr: port resolved to 1575 nttbnd2addr: looking up IP addr for host: oranamestrvr0 nttbnd2addr: exitnsopen: entry nsmal: entry nsmal: 330 bytes at 0x30d76e74 nsmal: normal exit nsopen: opening transport... nttcon: entry nttcon: toc = 1 nttcnp: entry nttcnp: creating a socket. nttcnp: exit nttcni: entry nttcni: trying to connect to socket 1. ntt2err: entry </pre>
ネットワーク・トランスポート (NT) は、ネーム・サーバーが動作していないのでエラー「no listener」を戻します。	<pre> -<ERROR>- ntt2err: soc 1 error - operation=1, ntresnt[0]=511, ntresnt[1]=61 ntresnt[2]=0 ntt2err: exit nttcni: exit nttcon: exit nserror: entry </pre>
エラーは、次のレイヤー (NS) に伝えられます。	<pre> -<ERROR>- nserror: nsres: id=0, op=65, ns=12541, ns2=12560; nt[0]=511, nt[1]=61,nt[2]=0 -<ERROR>- nsopen: unable to open transport nsmfr: entry nsmfr: 330 bytes at 0x30d76e74 nsmfr: normal exit nsopen: error exit nscall: error exit nricdt: Call failed... nricfg: entry nricfg: exit nricdt: Call made to destination nricdt: exit nricall: Failed to copy originating community name value binding nricall: Exiting NRICALL with following termination result -1 nricall: exit nngsfad_free_stream_addr: "(DESCRIPTION=(CONNECT_ DATA=(RPC=QN))(ADDRESS=(PROTOCOL=tcp)(HOST=oranamestrvr0)(PORT =1575)))" -<ERROR>- nngsget_get_stream: open failure, error stack follows </pre>

図 11-18 トレース・ファイルの例

説明	トレース・ファイルの情報
エラーは、次のレイヤー (TNS) に伝えられます。	<pre>TNS-12224: TNS:no listener TNS-12541: TNS:no listener TNS-12560: TNS:protocol adapter error TNS-00511: No listener mfggrwsp: Query unsuccessful, skipping to next adapter</pre>
Oracle Names Server が使用できないので、アドレスは Oracle Names Server 上で見つけられません。	<pre>nnfun2a: address for name "june" not found nngsfad_free_stream_addr: "(DESCRIPTION = (CONNECT_DATA= (RPC=ON)) (ADDRESS=(PROTOCOL=tcp) (HOST=oranamesrvr0) (PORT=1575)))" nngtdei_deinit_msg: free message pool block nngtfms_free_msg: message ID -10429 nngtfms_free_msg: message was a request nngtfms_free_msg: message free, type 100 nngtfoa_free_objarr: free message object array nngtfmt_free_msg_type: type-specific message free, type 100 nngtfoa_free_objarr: free message object array nngtfms_free_msg: message ID 0 nngtfms_free_msg: message was a response nngtfms_free_msg: message free, type 0 nngsdei_deinit_streams: deinit nngscls_close_stream: UID 11 not established, ignored nngscls_close_stream: UID 0 not established, ignored osnqrn: Return code from nnfsn2a is 409</pre>
エラーがユーザーに戻されます。	<pre>--<ERROR>- onstns: Couldn't connect, returning 12154 onstns: exit osnqtg: Count in the OSN global area is now 0 nrigbd: entry nrigbd: exit osnqtg: Count in the NL global area is now 0</pre>

トレース・ファイル例のサマリー

このトレース・ファイルは、エラー「ORA-12154: Could not resolve service name」が発生したときに、Net8 で何が起こったかについてサマリーを示します。この例では、クライアントがネット・サービス名 JUNE への接続に失敗しました。これは、NAMES.DEFAULT_DOMAIN = COM パラメータが SQLNET.ORA ファイルに存在するからです。このパラメータは、ネット・サービス名 JUNE など、要求されたネット・サービス名に拡張子 .COM を追加します。ただし、このサービス名は、クライアントの TNSNAMES.ORA ファイル内でも、Oracle Names Server 内でも定義されません。この問題を解決するには、次のようにします。

- SQLNET.ORA ファイルを編集して NAMES.DEFAULT_DOMAIN 構成パラメータを削除する
- JUNE ではなく JUNE.COM への接続を要求する

Oracle社サポート・サービスへの連絡

この章を読んだ後でも問題が解決できない場合、またはOracle社サポート・サービスに連絡してエラーを報告するように指示された場合は、次の情報を手元にご用意ください。

- アプリケーションが稼働しているハードウェアとオペレーティング・システムのリリース番号
- 現在の問題に関係するすべての Oracle ネットワーク製品の 5 桁のリリース番号
- 使用しているサードパーティ・ベンダーとバージョン
- 複数のエラー・コードやメッセージが表示された場合は、正確なコード番号およびメッセージ・テキスト、それらの表示順序
- クライアントとサーバー間に存在するリンクの種類
- 機能する部分の説明
- 正確なエラー・メッセージ (ある場合)
- トレースを実行した場合、Net8 トレース、ない場合はログ・ファイル

第 III 部

リファレンス

第 III 部は、このガイドの参照資料です。次の付録があります。

- 付録 A の「リスナー、Oracle Names Server および Oracle Connection Manager の制御ユーティリティ」
- 付録 B の「構成ファイルのサンプル」
- 付録 C の「構成パラメータ」

リスナー、Oracle Names Server および Oracle Connection Manager の制御ユーティリティ

Net8 には、リスナー、Oracle Names および Oracle Connection Manager を制御できるユーティリティが用意されています。この付録では、前提条件またはパスワード、構文または引数の規則、使用上の注意、それらを適用するときに役に立つ例など、各ユーティリティで使用できるコマンドを示します。

この付録では、次の 3 つの制御ユーティリティについて説明します。

- リスナー制御ユーティリティ (LSNRCTL)
- Oracle Names 制御ユーティリティ (NAMESCTL)
- Oracle Connection Manager 制御ユーティリティ (CMCTL)

操作モード

制御ユーティリティは、ランタイム・セッションの間動作します。ユーティリティは、次の3つのモードで実行できます。

- インタプリタ・モード - 制御ユーティリティが制御ユーティリティのコマンド行からロードされます。ロードされると、次のプロンプトが表示されます。

```
lsnrctl>
```

- コマンド行モード - 完全なコマンドをプログラムに対するパラメータとして指定して制御ユーティリティを実行すると、ほとんどのコマンドをオペレーティング・システムのコマンド行から実行することもできます。この場合は、制御ユーティリティがコマンドをロードおよび実行してから、オペレーティング・システム・プロンプトに戻ります。たとえば、次のコマンドを使用します。

```
namesctl start  
namesctl status sales.com
```

- バッチ・コマンド・モード - 複数のコマンドを標準テキスト・ファイル内で組み合わせて、それらを一連のコマンドとして実行できます。コマンドをバッチ・モードで実行する場合は、次のフォーマットを使用します。

```
control_utility @file_name
```

- REM または # を使用して、バッチ・スクリプト内でコメントを示すことができます。他のすべての行はコマンドと見なされます。バッチ実行中は、コマンドの確認は不要です。

リスナー制御ユーティリティ (LSNRCTL)

リスナー制御ユーティリティ (LSNRCTL) はリスナーを管理します。リスナー制御ユーティリティの一般構文は、次のとおりです。

```
lsnrctl command [listener_name]
```

listener_name は、LISTENER.ORA ファイルで定義されたリスナーの名前です。デフォルトのリスナー (LISTENER) を使用している場合は、必ずしもリスナーを識別する必要はありません。

LSNRCTL には、次に示すタイプのコマンドが含まれています。

- 操作コマンド (START、STOP など)
- 修飾子コマンド (SET *property* など)
- 情報コマンド (STATUS、SHOW *property* など)
- コマンド・ユーティリティ操作コマンド (EXIT、QUIT、HELP など)

このユーティリティを使用して 1 つ以上のリスナーで、基本的な管理機能を実行できます。また、Oracle Names Server のパラメータ設定も表示および変更できます。

SET および SHOW 修飾子

修飾子 SET を使用して、リスナーのパラメータ値や、LSNRCTL 制御ユーティリティ・セッション中の LSNRCTL 環境を変更できます。

セキュリティ

PASSWORDS.*listener_name* パラメータが LISTENER.ORA ファイルで設定されている場合、LSNRCTL 制御ユーティリティはリスナーの起動や停止などの重要な操作に対して、SET PASSWORD コマンドを必要とします。

LSNRCTL コマンド

次のコマンドは、リスナー制御ユーティリティ (LSNRCTL) で使用できます。

CHANGE_PASSWORD

目的:	このコマンドを使用して、リスナーの暗号化されたパスワードを動的に変更します。リスナー構成ファイルで確立済みの非暗号化パスワードは変更されません。このコマンドでは、新しいパスワードを確立するか、リスナー構成ファイル内で暗号化されているパスワードのみを変更します。
前提条件:	なし
パスワード:	必要 パスワードが設定されている場合、このコマンドの前に SET PASSWORD コマンドを実行する必要があります。
構文:	オペレーティング・システムからは、次のように入力します。 <code>lsnrctl change_password [listener_name]</code> LSNRCTL プログラムからは、次のように入力します。 <code>lsnrctl> change_password [listener_name]</code>
引数:	[listener_name]
使用上の注意:	古いパスワードと新しいパスワードの入力を促すプロンプトが表示されます。次に、新しいパスワードを再入力すると、古いパスワードが変更されます。操作中は、古いパスワードも新しいパスワードも表示されません。
例:	<pre>lsnrctl> change_password Old password: New password: Reenter new password: Connecting to (ADDRESS=(PROTOCOL=ipc)(KEY=iris)) Password changed for LISTENER The command completed successfully</pre>

DBSNMP_START

目的:	同じノードで稼働している Oracle データベースの SNMP サブエージェントを起動します。
前提条件:	ローカルに実行する必要があります。
パスワード:	不要
構文:	オペレーティング・システムからは、次のように入力します。 <code>lsnrctl dbsnmp_start</code> LSNRCTL プログラムからは、次のように入力します。 <code>lsnrctl> dbsnmp_start</code>
引数:	なし
使用上の注意:	なし
例:	<code>lsnrctl> dbsnmp_start</code>

DBSNMP_STATUS

目的:	Oracle データベースの SNMP サブエージェントが稼働しているかどうかを確認します。
前提条件:	なし
パスワード:	不要
構文:	オペレーティング・システムからは、次のように入力します。 <code>lsnrctl dbsnmp_status</code> LSNRCTL プログラムからは、次のように入力します。 <code>lsnrctl> dbsnmp_status</code>
引数:	なし
使用上の注意:	Oracle データベースが存在するノード上で実行する必要があります。
例:	<code>lsnrctl> dbsnmp_status</code>

DBSNMP_STOP

目的:	同じノードで稼働している Oracle データベースの SNMP サブエージェントを停止します。
前提条件:	ローカルに実行する必要があります。
パスワード:	不要
構文:	オペレーティング・システムからは、次のように入力します。 <code>lsnrctl dbsnmp_stop</code> LSNRCTL プログラムからは、次のように入力します。 <code>lsnrctl> dbsnmp_stop</code>
引数:	なし
使用上の注意:	なし
例:	<code>lsnrctl> dbsnmp_stop</code>

EXIT

目的:	LSNRCTL を終了して、オペレーティング・システム・プロンプトに戻ります。
前提条件:	なし
パスワード:	必要 パスワードが設定されている場合、このコマンドの前に SET PASSWORD コマンドを実行する必要があります。
構文:	オペレーティング・システムからは、次のように入力します。 <code>lsnrctl exit</code> LSNRCTL プログラムからは、次のように入力します。 <code>lsnrctl> exit</code>
引数:	なし
使用上の注意:	このコマンドは QUIT と同じです。
例:	<code>lsnrctl> exit</code>

HELP

目的: 使用可能な LSNRCTL コマンドの一覧を表示します。

前提条件: なし

パスワード: 不要

構文: オペレーティング・システムからは、次のように入力します。
lsnrctl help [*command*]
LSNRCTL プログラムからは、次のように入力します。
lsnrctl> help [*command*]

引数: help コマンド

例: lsnrctl> help
The following operations are available
An asterisk (*) denotes a modifier or extended command:
change_password
dbsnmp_start
dbsnmp_stop
dbsnmp_status
exit
quit
reload
services
set*
show*
spawn
start
status
stop
trace
version

QUIT

目的:	LSNRCTLを終了して、オペレーティング・システム・プロンプトに戻ります。
前提条件:	なし
パスワード:	必要 パスワードが設定されている場合、このコマンドの前に SET PASSWORD コマンドを実行する必要があります。
構文:	オペレーティング・システムからは、次のように入力します。 <code>lsnrctl quit</code> LSNRCTL プログラムからは、次のように入力します。 <code>lsnrctl> quit</code>
引数:	なし
使用上の注意:	このコマンドは EXIT と同じです。
例:	<code>lsnrctl> quit</code>

RELOAD

目的:	リスナー・アドレス以外のものをすべて停止して、LISTENER.ORA ファイルを再読み込みします。このコマンドで、実際にリスナーを停止せずにサービスを追加または変更できます。
前提条件:	このコマンドは有効なノードには作用しません。この場合は、リスナーを停止して再起動する必要があります。
パスワード:	必要 パスワードが設定されている場合、このコマンドの前に SET PASSWORD コマンドを実行する必要があります。
構文:	オペレーティング・システムからは、次のように入力します。 <code>lsnrctl reload [listener_name]</code> LSNRCTL プログラムからは、次のように入力します。 <code>lsnrctl> reload [listener_name]</code>
引数:	リスナーの名前

RELOAD

- 使用上の注意 : リスナー構成ファイルにパスワードがある場合は、RELOAD コマンドを使用する前に SET PASSWORD コマンドを使用する必要があります。
- パスワードは LSNRCTL プログラムの中から設定する必要があります。オペレーティング・システムのコマンド行からは、パスワードを設定できません。パスワードを設定する方法は、暗号化パスワード機能を使用しているかどうかによって決まります。暗号化されたパスワードを使用しない場合は、LSNRCTL コマンド行にそのパスワードを入力します。
- 例 : `lsnrctl> reload`

SAVE_CONFIG

- 目的 : リスナーのトレース・レベル、トレース・ファイル、トレース・ディレクトリ、および接続タイムアウトなどの現在の構成状態を、LISTENER.ORA ファイルと比較します。変更はすべて、形式、コメントおよび大文字と小文字をできるだけ維持したまま、LISTENER.ORA に保存されます。LISTENER.ORA を変更する前に、バックアップとして LISTENER.BAK が作成されます。
- パスワード : 必要
- パスワードが設定されている場合、このコマンドの前に SET PASSWORD コマンドを実行する必要があります。
- 構文 : オペレーティング・システムからは、次のように入力します。
- ```
lsnrctl save_config [listener_name]
```
- LSNRCTL プログラムからは、次のように入力します。
- ```
lsnrctl> save_config [listener_name]
```
- 引数 : リスナーの名前
- 使用上の注意 : 管理者がオンラインで変更した構成箇所をリスナー構成ファイルに保存する場合は、このコマンドを使用します。このコマンドは特に、変更された暗号化パスワードの保存に役立ちます。
- 例 : `lsnrctl> save_config listener`

SERVICES

目的:	リスナーがリスニングしているサービスの詳細情報を表示します。たとえば、接続がいくつ確立され、いくつ確立されなかったかなどです。このコマンドは、3種類のサービス (LISTENER.ORA ファイルの専用サーバー、ディスパッチャ情報、事前に生成されたシャドウ) を表示します。
前提条件:	なし
パスワード:	必要 パスワードが設定されている場合、このコマンドの前に SET PASSWORD コマンドを実行する必要があります。
構文:	オペレーティング・システムからは、次のように入力します。 <code>lsnrctl services [listener_name]</code> LSNRCTL プログラムからは、次のように入力します。 <code>lsnrctl> services [listener_name]</code>
引数:	リスナーの名前
使用上の注意:	データベース管理者がリスナーのサービスに関する情報を取得する場合は、このコマンドを使用します。 SET DISPLAY および SET RAWMODE コマンドを使用すると、出力の形式を変更できます。

SERVICES

```
例: lsnrctl> services listener_name
LSNRCTL SERVICES コマンドの出力は次のようになります。
Connecting to (ADDRESS=(PROTOCOL=IPC)(KEY=EXTPROC))
Services Summary...
  PLSExtProc          has 1 service handler(s)
    DEDICATED SERVER established:0 refused:0
    LOCAL SERVER
  db1                 has 3 service handler(s)
    DEDICATED SERVER established:0 refused:0
    LOCAL SERVER
    DISPATCHER established:0 refused:0 current:0 max:254
state:ready
    D000 <machine: dlsun1770, pid: 1289>

    (ADDRESS=(PROTOCOL=tcp)(DEV=14)(HOST=144.25.17.193)(PORT=2222
    ))
    DISPATCHER established:0 refused:0 current:0 max:254
state:ready
    D001 <machine: dlsun1770, pid: 1291>

    (ADDRESS=(PROTOCOL=tcp)(DEV=14)(HOST=144.25.17.193)(PORT=2223
    ))
The command completed successfully
```

SET

目的:	SET コマンドで設定できるパラメータ値を表示します。
前提条件:	なし
パスワード:	必要 パスワードが設定されている場合、このコマンドの前に SET PASSWORD コマンドを実行する必要があります。
構文:	オペレーティング・システムからは、次のように入力します。 <code>lsnrctl set [command]</code> LSNRCTL プログラムからは、次のように入力します。 <code>lsnrctl> set [command]</code>
引数:	set コマンド
使用上の注意:	リスナー構成パラメータ <code>PASSWORDS_listener_name</code> に、このコマンドが表示されている場合、このコマンドを使用するには有効なパスワードを設定しておく必要があります。 リスナー構成ファイルにパスワードがある場合は、SET コマンドを使用する前に SET PASSWORD コマンドを使用する必要があります。 パスワードは LSNRCTL プログラムの中から設定する必要があります。オペレーティング・システムのコマンド行からは、パスワードを設定できません。パスワードを設定する方法は、暗号化パスワード機能を使用しているかどうかによって決まります。暗号化されたパスワードを使用しない場合は、LSNRCTL コマンド行にそのパスワードを入力します。
例:	<code>lsnrctl> set</code> set の後には、次の操作が行えます。 <code>connect_timeout</code> <code>current_listener</code> <code>displaymode</code> <code>log_file</code> <code>log_directory</code> <code>log_status</code> <code>password</code> <code>raw_mode</code> <code>save_config_on_stop</code> <code>startup_waittime</code> <code>trc_file</code> <code>trc_directory</code> <code>trc_level</code> <code>use_plugandplay</code>

SET CONNECT_TIMEOUT

目的:	接続が開始されてからリスナーが有効な接続要求を待つ時間を決定します。
前提条件:	なし
パスワード:	不要
構文:	オペレーティング・システムからは、次のように入力します。 <pre>lsnrctl set connect_timeout time</pre> LSNRCTL プログラムからは、次のように入力します。 <pre>lsnrctl> set connect_timeout time</pre>
引数:	秒数
使用上の注意:	なし
例:	<pre>lsnrctl> set connect_timeout 20</pre> Connecting to (ADDRESS=(PROTOCOL=ipc)(KEY=iris)) LISTENER parameter "connect_timeout" set to 20 The command completed successfully

SET CURRENT_LISTENER

目的:	複数のリスナーに対するパラメータを設定または表示します。
前提条件:	SET CURRENT_LISTENER は LSNRCTL ユーティリティの中から入力します。
パスワード:	不要
構文:	オペレーティング・システムからは、次のように入力します。 <pre>lsnrctl set current_listener [listener_name]</pre> LSNRCTL プログラムからは、次のように入力します。 <pre>lsnrctl> set current_listener [listener_name]</pre>
引数:	リスナーの名前
デフォルト引数	LISTENER

SET CURRENT_LISTENER

使用上の注意:

ノード上に複数のリスナーがある場合は、別のリスナーを設定しない限り、LSNRCTL コマンドはデフォルト・リスナー (LISTENER) に適用されます。

同じ LSNRCTL セッション内で LSNRCTL 以後のコマンドは、CURRENT_LISTENER をリセットしない限り、2 番目のリスナーに適用されます。

LSNRCTL の SHOW CURRENT_LISTENER コマンドを使用して、現リスナーを表示することもできます。

SET CURRENT_LISTENER は LSNRCTL ユーティリティの中から入力します。ユーティリティを終了すると、設定が失われます。

例:

```
lsnrctl> set current_listener [listener_name]
```

SET DISPLAYMODE

目的:	LSNRCTL SERVICE および LSNRCTL STATUS コマンドの出力を、追加情報で変更します。
パスワード:	不要
構文:	オペレーティング・システムからは、次のように入力します。 <pre>lsnrctl set displaymode argument</pre> LSNRCTL プログラムからは、次のように入力します。 <pre>lsnrctl> set displaymode argument</pre>
引数:	RAW (SET RAWMODE と同じ) COMPAT (旧バージョン用) NORMAL VERBOSE (サポート用) 注意: NORMAL および VERBOSE は、現在のリリースでは一部機能しません。
例:	<pre>lsnrctl> set displaymode compat lsnrctl> status Connecting to (ADDRESS=(PROTOCOL=IPC)(KEY=EXTPROC)) STATUS of the LISTENER ----- Alias LISTENER Version TNSLSNR for Solaris: Version 8.1.5.0.0 Start Date 07-DEC-98 14:28:23 Uptime 0 days 0 hr. 30 min. 16 sec Trace Level off Security OFF SNMP OFF Listener Parameter File /private/dsteiner/installtest2/network/admin/listener.ora Listener Log File /private/dsteiner/installtest2/network/log/listener.log Services Summary... PLSExtProc has 1 service handler(s) sales has 3 service handler(s) The command completed successfully</pre>

SET LOG_DIRECTORY

目的:	リスナー・プロセスが書き込まれたログ・ファイルが存在する、デフォルトのディレクトリを変更できます。
前提条件:	なし
パスワード:	不要
構文:	オペレーティング・システムからは、次のように入力します。 <code>lsnrctl set log_directory [directory]</code> LSNRCTL プログラムからは、次のように入力します。 <code>lsnrctl> set log_directory [directory]</code>
引数:	ディレクトリ・パス
例:	<code>lsnrctl> set log_directory /usr/oracle/admin</code> Connecting to (ADDRESS=(PROTOCOL=ipc)(KEY=iris))LISTENER parameter "log_directory" set to /usr/oracle/admin The command completed successfully

SET LOG_FILE

目的:	ログ・ファイルの非デフォルト名を設定します。
前提条件:	なし
パスワード:	不要
構文:	オペレーティング・システムからは、次のように入力します。 <code>lsnrctl set log_file [file_name]</code> LSNRCTL プログラムからは、次のように入力します。 <code>lsnrctl> set log_file [file_name]</code>
引数:	ファイル名
例:	<code>lsnrctl> set log_file list.log</code> Connecting to (ADDRESS=(PROTOCOL=ipc)(KEY=iris))LISTENER parameter "log_file" set to list.log The command completed successfully

SET LOG_STATUS

目的:	リスナーのロギングをオンまたはオフにします。
前提条件:	なし
パスワード:	不要
構文:	オペレーティング・システムからは、次のように入力します。 lsnrctl set set log_status [on off] LSNRCTL プログラムからは、次のように入力します。 lsnrctl> set set log_status [on off]
引数:	ON OFF
例:	lsnrctl> set log_status on

SET PASSWORD

目的:	LSNRCTL ユーティリティからリスナー・プロセスに認証用に送信されるパスワードを変更します。リスナー自体のパスワードを変更するには、CHANGE_PASSWORD コマンドを使用します。
構文:	オペレーティング・システムからは、次のように入力します。 lsnrctl set password LSNRCTL プログラムからは、次のように入力します。 lsnrctl> set password 非暗号化パスワードの構文は次のとおりです。 set password password
引数:	パスワード
使用上の注意:	シェルの起動時またはセッション中のいつでも、このコマンドを入力できます (リスナーを停止する前に、SET PASSWORD コマンドを入力する必要があります)。 パスワードをより安全に入力するには、対話モードを使用します。リスナーは暗号化パスワードと非暗号化パスワードをサポートしています。
例:	lsnrctl> set password enter listener password: password

SET RAWMODE

目的: LSNRCTL SERVICE および LSNRCTL STATUS コマンドの出力を、追加情報で変更します。

パスワード: 不要

構文: オペレーティング・システムからは、次のように入力します。

```
lsnrctl set rawmode argument
```

LSNRCTL プログラムからは、次のように入力します。

```
lsnrctl> set rawmode argument
```

引数: ON | OFF

例:

```
lsnrctl> set rawmode on
lsnrctl> status
Connecting to (ADDRESS=(PROTOCOL=IPC)(KEY=EXTPROC))
STATUS of the LISTENER
-----
Alias                LISTENER
Version              TNSLSNR for Solaris: Version 8.1.5.0.0
Start Date           07-DEC-98 14:28:23
Uptime               0 days 0 hr. 9 min. 6 sec
Trace Level          off
Security             OFF
SNMP                 OFF
Listener Parameter File
/private/dsteiner/installtest2/network/admin/listener.ora
Listener Log File
/private/dsteiner/installtest2/network/log/listener.log
Services Summary
(SERVICE=(SERVICE_NAME=PLSExtProc)(INSTANCE=(INSTANCE_
NAME=PLSExtProc)(NUM=1)(INSTANCE_CLASS=ORACLE)))
(SERVICE=(SERVICE_NAME=dlsun17)(INSTANCE=(INSTANCE_
NAME=db1)(NUM=3)(INSTANCE_CLASS=ORACLE)))
The command completed successfully
```

SET SAVE_CONFIG_ON_STOP

目的:	パラメータを ON に設定すると、LSNRCTL SET コマンドで行った変更が LISTENER.ORA に保存されます。リスナーが終了する直前にすべてのパラメータが保存されます。パラメータの形式、コメント、大文字と小文字は、可能な限り維持されます。すべてのパラメータをすぐに保存する場合は、SAVE_CONFIG コマンドを使用します。
パスワード:	必要 パスワードが設定されている場合、このコマンドの前に SET PASSWORD コマンドを実行する必要があります。
構文:	オペレーティング・システムからは、次のように入力します。 lsnrctl set save_config_on_stop [on off] LSNRCTL プログラムからは、次のように入力します。 lsnrctl> set save_config_on_stop [on off]
引数:	ON OFF
例:	lsnrctl> set save_config_on_stop on

SET STARTUP_WAITTIME

目的:	リスナーが START コマンドに応答するまで休止する時間を指定します。
前提条件:	なし
パスワード:	不要
構文:	オペレーティング・システムからは、次のように入力します。 lsnrctl set startup_waittime [time] LSNRCTL プログラムからは、次のように入力します。 lsnrctl> set startup_waittime [time]
引数:	秒数
例:	LSNRCTL> set startup_waittime 10 Connecting to (ADDRESS=(PROTOCOL=ipc)(KEY=iris))LISTENER parameter "startup_waittime" set to 10 The command completed successfully

SET TRC_DIRECTORY

目的:	リスナー・プロセスを書き込むトレース・ファイルの、デフォルトの場所を変更できます。
前提条件:	なし
パスワード:	不要
構文:	オペレーティング・システムからは、次のように入力します。 <code>lsnrctl set trc_directory [directory]</code> LSNRCTL プログラムからは、次のように入力します。 <code>lsnrctl> set trc_directory [directory]</code>
引数:	ディレクトリ・パス
例:	<code>lsnrctl> set trc_directory /usr/oracle/admin</code> Connecting to (ADDRESS=(PROTOCOL=ipc)(KEY=iris)) LISTENER parameter "trc_directory" set to /usr/oracle/admin The command completed successfully

SET TRC_FILE

目的:	トレース・ファイルの非デフォルト名を設定します。
前提条件:	なし
パスワード:	不要
構文:	オペレーティング・システムからは、次のように入力します。 <code>lsnrctl set trc_file [file_name]</code> LSNRCTL プログラムからは、次のように入力します。 <code>lsnrctl> set trc_file [file_name]</code>
引数:	ファイル名
例:	<code>lsnrctl> set trc_file list.trc</code> Connecting to (ADDRESS=(PROTOCOL=ipc)(KEY=iris)) LISTENER parameter "trc_file" set to list.trc The command completed successfully

SET TRC_LEVEL

目的:	リスナーのトレース機能をオンにします。
前提条件:	なし
パスワード:	必要 パスワードが設定されている場合、このコマンドの前に SET PASSWORD コマンドを実行する必要があります。
構文:	オペレーティング・システムからは、次のように入力します。 <pre>lsnrctl set trc_level level</pre> LSNRCTL プログラムからは、次のように入力します。 <pre>lsnrctl> set trc_level level</pre>
引数:	OFF USER ADMIN SUPPORT 0 4 10 16 <ul style="list-style-type: none">■ OFF または 0 - トレース出力しない■ USER または 4 - ユーザー・トレース情報■ ADMIN または 10 - 管理トレース情報■ SUPPORT または 16 - オラクル社サポート・サービス・トレース情報
使用上の注意:	USER を選択すると、特定の情報のみがトレースされます。 ADMIN を選択すると、より詳細な情報がトレースされます。このコマンドを使用すると、LISTENER.ORA ファイルで設定した値が上書きされます。 LISTENER.ORA ファイルのパラメータ <code>PASSWORDS_listener_name</code> にこのコマンドが表示されている場合、このコマンドを使用するには有効なパスワードを設定しておく必要があります。 このコマンドは TRACE と同じです。
例:	<pre>lsnrctl> set trc_level admin</pre>

SET USE_PLUGANDPLAY

目的:	リスナーに、データベース情報を Oracle Names Server に登録するよう指示します。
パスワード:	必要 パスワードが設定されている場合、このコマンドの前に SET PASSWORD コマンドを実行する必要があります。
構文:	オペレーティング・システムからは、次のように入力します。 <code>lsnrctl set use_plugandplay [on off]</code> LSNRCTL プログラムからは、次のように入力します。 <code>lsnrctl> set use_plugandplay [on off]</code>
引数:	ON OFF
例:	<code>lsnrctl> set use_plugandplay on</code>

SHOW

目的:	SET PASSWORD を除くすべての SET コマンドには、同等の SHOW コマンドがあります。LSNRCTL は SHOW コマンドのうちの 1 つにตอบสนองして、そのパラメータに対するリスナーの現行の設定を表示します。
前提条件:	なし
パスワード:	同等の SET コマンドを参照してください。
構文:	オペレーティング・システムからは、次のように入力します。 <pre>lsnrctl show [listener_name] [command]</pre> LSNRCTL プログラムからは、次のように入力します。 <pre>lsnrctl> show [listener_name] [command]</pre>
引数:	リスナー名、show コマンド
使用上の注意:	LSNRCTL を通じて SHOW で表示することはできませんが、設定できないパラメータがあります。 <pre>show [listener_name] snmp_visible</pre> SNMP クライアントがリスナーにアクセスできるかどうかを表示します。
例:	<pre>lsnrctl> show</pre> show の後には、次の操作が行えます。 An asterisk (*) denotes a modifier or extended command: <pre>connect_timeout current_listener displaymode log_file log_directory log_status password raw_mode save_config_on_stop startup_waittime trc_file trc_directory trc_level use_plugandplay</pre>

SPAWN

目的:	リスナーが稼働しているマシンに格納されていて、 LISTENER.ORA ファイルに別名でリストされているプログラムを 起動します。
前提条件:	なし
パスワード:	不要
構文:	オペレーティング・システムからは、次のように入力します。 <code>lsnrctl spawn [listener_name] alias (arguments=arg1,arg2,)</code> LSNRCTL プログラムからは、次のように入力します。 <code>lsnrctl> spawn [listener_name] alias (arguments=arg1,arg2,)</code>
引数:	<ul style="list-style-type: none"> ■ リスナーの名前 ■ リスナー構成ファイルにリストされているプログラムの別名 ■ 作成するプログラムに送られる引数
例:	<code>lsnrctl> spawn nstest_alias (ARGUMENTS='')</code>

START

目的:	指定したリスナーを起動します。
前提条件:	リスナーは停止している必要があります。
パスワード:	不要
構文:	オペレーティング・システムからは、次のように入力します。 <code>lsnrctl start [listener_name]</code> LSNRCTL プログラムからは、次のように入力します。 <code>lsnrctl> start [listener_name]</code>
引数:	リスナーの名前 リスナー名を入力しないと、デフォルトで LISTENER が起動され ます。

START

使用上の注意 :

LISTENER.ORA ファイル内で LISTENER 以外の名前で構成されているリスナーを起動するには、そのリスナーの名前を指定します。
たとえば、リスナー名が TCP_LSNR の場合は、次のように入力します。

```
lsnrctl start tcp_listener
```

または、LSNRCTL プログラム・プロンプトから、次のように入力します。

```
lsnrctl> start tcp_listener
```

例 :

```
lsnrctl> start
```

```
Starting /private/dsteiner/sales/bin/tnslsnr: please wait...
```

```
TNSLSNR for Solaris: Version 8.1.4.0.0 - Beta
System parameter file is
/private/dsteiner/sales/network/admin/listener.ora
Log messages written to
/private/dsteiner/sales/network/log/listener.log
Trace information written to
/private/dsteiner/sales/network/trace/listener.trc
Listening on:
(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=dlsun1770)(PORT=1521))
(PROTOCOL_STACK=(PRESENTATION=ITC)(SESSION=NS)))
```

```
Connecting to
(ADDRESS=(PROTOCOL=TCP)(HOST=dlsun1770)(PORT=1521))
STATUS of the LISTENER
```

```
-----
Alias                LISTENER
Version              TNSLSNR for Solaris: Version
8.1.4.0.0 - Beta
Start Date           08-JAN-99 15:50:10
Uptime               0 days 0 hr. 0 min. 0 sec
Trace Level          user
Security             OFF
SNMP                 OFF
```

```
Listener Parameter File
/private/dsteiner/sales/network/admin/listener.ora
Listener Log File
/private/dsteiner/sales/network/log/listener.log
Listener Trace File
/private/dsteiner/sales/network/trace/listener.trc
Services Summary...
```

```
  PLSExtProc          has 1 service handler(s)
The command completed successfully
```

STATUS

目的:	バージョン、開始時刻、アップタイム、使用する LISTENER.ORA ファイル、トレースをオンにするかどうかなど、基本的な情報を表示します。
前提条件:	なし
パスワード:	不要
構文:	オペレーティング・システムからは、次のように入力します。 <pre>lsnrctl status [listener_name]</pre> LSNRCTL プログラムからは、次のように入力します。 <pre>lsnrctl> status [listener_name]</pre>
引数:	リスナーの名前
使用上の注意:	status コマンドで、次のことができます。 <ul style="list-style-type: none">■ ログイン・オプションとトレース・オプションに対する現在の設定をチェックする。■ このリスナーを通じて使用できるデータベース SID のリストをチェックする。これらのデータベース SID は、LISTENER.ORA ファイルの SID マッピングで定義されています。■ LISTENER.ORA でパスワードを暗号化するかどうかをチェックする (リスナー・パスワードを暗号化する場合は、1 つのパスワードしか設定できません)。■ ネットワーク・リスナーが、SNMP ベースのネットワーク管理システムからの問合せに回答できるかどうかをチェックする。■ TNSLSNR がリスニングするアドレスをチェックする。 SET DISPLAY および SET RAWMODE コマンドを使用すると、出力の形式を変更できます。

STATUS

```
例 : lsnrctl> status
      STATUS of the LISTENER
      -----
      Alias                LISTENER
      Version              TNSLSNR for Solaris: Version 8.1.5.0.0
      Start Date           07-DEC-98 14:28:23
      Uptime                0 days 0 hr. 9 min. 6 sec
      Trace Level          off
      Security              OFF
      SNMP                  OFF
      Listener Parameter File
      /private/dsteiner/installtest2/network/admin/listener.ora
      Listener Log File
      /private/dsteiner/installtest2/network/log/listener.log
      Services Summary...
      PLSExtProc           has 1 service handler(s)
      db1                   has 3 service handler(s)
      The command completed successfully
```

STOP

目的:	指定したリスナーを停止します。
前提条件:	リスナーが稼働している必要があります。
パスワード:	必要 パスワードが設定されている場合、このコマンドの前に SET PASSWORD コマンドを実行する必要があります。
構文:	オペレーティング・システムからは、次のように入力します。 <pre>lsnrctl stop [listener_name]</pre> LSNRCTL プログラムからは、次のように入力します。 <pre>lsnrctl> stop [listener_name]</pre>
引数:	リスナーの名前
使用上の注意:	すでにパスワードを構成している場合は、STOP コマンドを使用する前に SET PASSWORD コマンドを使用する必要があります。 パスワードは LSNRCTL プログラムの中から設定する必要があります。オペレーティング・システムのコマンド行からは、パスワードを設定できません。パスワードを設定する方法は、暗号化パスワード機能を使用しているかどうかによって決まります。暗号化されたパスワードを使用しない場合は、LSNRCTL コマンド行にそのパスワードを入力します。 リスナーを停止するときは注意してください。プラットフォームとプロトコルによっては、リスナーを停止すると、現在稼働している Net8 接続がすべてシャットダウンされます。接続が継続される場合もありますが、実行中のプロセスをクローズするまでは、リスナーを再起動できません。リスナーの停止前に、すべてのネットワーク・ユーザーに警告メッセージを送信することをお勧めします。
例:	<pre>lsnrctl> stop</pre>

TRACE

目的:	リスナーのトレース機能をオンにします。
前提条件:	有効なパスワードが必要です。
パスワード:	必要 パスワードが設定されている場合、このコマンドの前に SET PASSWORD コマンドを実行する必要があります。
構文:	オペレーティング・システムからは、次のように入力します。 <code>lsnrctl trace off user admin support [listener_name]</code> LSNRCTL プログラムからは、次のように入力します。 <code>lsnrctl> trace off user admin support [listener_name]</code>
引数:	[OFF USER ADMIN SUPPORT] [リスナー名] <ul style="list-style-type: none"> ■ OFF - トレース出力しない ■ USER - ユーザー・トレース情報 ■ ADMIN - 管理トレース情報 ■ SUPPORT - オラクル社サポート・サービス・トレース情報
使用上の注意:	USER を選択すると、特定の情報のみがトレースされます。 ADMIN を選択すると、より詳細な情報がトレースされます。このコマンドを使用すると、LISTENER.ORA ファイルで設定した値が上書きされます。 このコマンドは SET TRC_LEVEL と同様に機能します。
例:	<code>lsnrctl> trace admin listener</code>

VERSION

目的:	現行の TNS リスナーとプロトコルのバージョンを表示します。
前提条件:	なし
パスワード:	不要
構文:	オペレーティング・システムからは、次のように入力します。 <code>lsnrctl version [listener_name]</code> LSNRCTL プログラムからは、次のように入力します。 <code>lsnrctl> version [listener_name]</code>
引数:	リスナーの名前
例:	<code>lsnrctl> version listener1</code>

Oracle Names 制御ユーティリティ (NAMESCTL)

Oracle Names 制御ユーティリティ (NAMESCTL) は、Oracle Names Server を操作、制御するためにオペレーティング・システム・プロンプトから実行するツールです。このユーティリティには、次に示すタイプのコマンドが含まれています。

- 操作コマンド (START、STOP、RESTART など)
- 修飾子コマンド (SET *property* など)
- 情報コマンド (STATUS、SHOW *property*、PING など)
- コマンド・ユーティリティ操作コマンド (EXIT、QUIT、HELP など)

NAMESCTL 制御ユーティリティを使用すると、基本的な管理機能を 1 つ以上の Oracle Names Server で実行できます。さらに、RESET_STATS_INTERVAL、TRACE_LEVEL などの Oracle Names Server パラメータの設定を表示および変更できます。

パラメータ・オプション

NAMESCTL をロードするときに、有効なパラメータ設定をプログラムに渡して、デフォルト設定または構成済みの設定を上書きすることができます。たとえば、次のように指定します。

```
namesctl namesctl.trace_level=admin
```

NAMESCTL をロードし、トレースを ADMIN レベルに設定しますが、NAMESCTL.TRACE_LEVEL の現在構成されている値とは無関係です。

SET および SHOW 修飾子

修飾子 SET を使用して、Oracle Names Server のパラメータ値や、NAMESCTL 制御ユーティリティ・セッション中の NAMESCTL 環境を変更できます。たとえば、次のコマンドで、トレース・レベルを制御および変更するためのノードを設定します。

```
namesctl> set server dolphin.com  
namesctl> set trace_level admin
```

最初の修飾子でノードを DOLPHIN.COM に設定します。これ以降のコマンドは DOLPHIN.COM に送られ、2 番目の修飾子で DOLPHIN.COM のトレース・レベルをサーバーに設定します。サーバーは ADMIN レベルでトレースを開始します。

NAMESCTL の分配操作

Oracle Names 制御ユーティリティは、Oracle Names Server をネットワーク内の他の Oracle Names Server と同じマシンで処理します。1 人の管理者がリージョン内のすべての Oracle Names Server を管理する場合や、特定の Oracle Names Server の可用性をチェックする場合は、NAMESCTL ユーティリティが非常に役立ちます。

大半のコマンドは、コマンドを実行する対象の Oracle Names Server を指定するために、最後の引数に Oracle Names Server の名前が入ります。サーバーの名前を指定しないと、SET で設定してある Oracle Names Server にコマンドが適用されます。たとえば、次のように指定します。

```
show system_queries dolphin.acme
```

このコマンドで、Oracle Names Server の DOLPHIN.ACME に対するシステム問合せ、およびそれらの問合せが次にいつ行われるかが表示されます。個々の Oracle Names Server に対して一連のコマンドを実行する前に、次のように入力してください。

```
namesctl> set server onames_server
```

セキュリティ

NAMES.PASSWORD パラメータが NAMES.ORA ファイルに設定されている場合、NAMESCTL 制御ユーティリティは Oracle Names Server の起動や停止などの重要な操作に対して、SET PASSWORD コマンドを必要とします。

NAMESCTL.SERVER_PASSWORD パラメータが NAMESCTL を実行しているノードの SQLNET.ORA ファイルに設定されている場合は、保護された操作を実行するたびに SET PASSWORD コマンドを要求されることはありません。

Oracle Names Server のパスワードを、管理者のクライアント SQLNET.ORA ファイル内で明示的に指定することでセキュリティの問題が心配になる場合は、パラメータを指定せずに、必ず次のコマンドを使用してください。

```
namesctl> set password
```

パスワードがネットワーク上に送信される場合、SQLNET.ORA ファイルで NAMESCTL.INTERNAL_ENCRYPT_PASSWORD を FALSE に設定しない限り、パスワードは NAMES.ORA ファイルの設定にかかわらず必ず暗号化されます。このパラメータを FALSE に設定すると、パスワードは暗号化されません。

NAMESCTL の確認モード

一部の NAMESCTL コマンドは、実行する前に確認が必要です。コマンドを発行すると、次のプロンプトが表示されます。

```
confirm:[yes or no]
```

コマンドを実行するときは「yes」と入力し、コマンドを取り消すときは「no」と入力します。

SQLNET.ORA ファイルで NAMESCTL.NOCONFIRM パラメータを TRUE に設定すると、確認モードをオフにできます。このパラメータを OFF に設定すると、すべてのコマンドが確認なしで実行されます。

NAMESCTL コマンド

次のコマンドは、Oracle Names 制御ユーティリティ (NAMESCTL) で使用できます。

DELEGATE_DOMAIN

目的:	現行のリージョンのサブリージョンの先頭としてドメインを定義します。
前提条件:	なし
パスワード:	不要
構文:	<p>オペレーティング・システム・プロンプトからは、次のように入力します。</p> <pre>namesctl delegate_domain domain_name onames_server [description=(address_list=)(address= (protocol_address_ information))([])]</pre> <p>NAMESCTL プログラムからは、次のように入力します。</p> <pre>delegate_domain domain_name onames_server [(description=address_list=] (address= (protocol_ address_information))([])]</pre>
引数:	<p><i>domain_name</i>、<i>onames_server</i>、およびアドレスは必須です。<i>domain_name</i> と <i>onames_server</i> は有効なドメイン名でなければなりません。アドレスは、有効な TNS アドレスでなければなりません。</p>
使用上の注意:	<p>このコマンドを使用して、名前領域を動的に細分化することができます。</p> <p>ドメイン <i>domain_name</i> は、Oracle Names Server 名として <i>onames_server</i> で定義されます。<i>onames_server</i> は、そのアドレスとして定義されます。</p> <p>ドメインがリージョンから委任されていない場合は、そのリージョン内の Oracle Names Server がすべてのサブドメインに対する権限をもちます。ドメインを委任するには、まず新しいリージョンを作成する必要があります。</p> <p>ドメインが委任されると、現行のリージョンの Oracle Names Server は以後の操作を、Oracle Names Server に管理されているドメインであるサブリージョンに転送します。</p>
例:	<pre>namesctl> delegate_domain webwidgets.acme.com ns1.webwidgets.acme.com (address=(protocol=tcp) (host=fred.webwidgets.acme.com) (port=1575))</pre>

DOMAIN_HINT

目的 :	Oracle Names Server がルート・リージョンにない場合に、ルート・リージョンの Oracle Names Server の場所を識別します。
前提条件 :	なし
パスワード :	不要
構文 :	オペレーティング・システムからは、次のように入力します。 <pre>namesctl domain_hint domain_name onames_server [description=(address_list=(address= (protocol_address_ information))))</pre> NAMESCTL プログラムからは、次のように入力します。 <pre>domain_hint domain_name onames_server [description=(address_list=(address= (protocol_address_ information))))</pre>
引数 :	<i>domain_name</i> 、 <i>onames_server</i> 、およびアドレスは必須です。 <i>domain_name</i> と <i>onames_server</i> は有効なドメイン名でなければなりません。アドレスは、有効な TNS アドレスでなければなりません。
使用上の注意 :	このコマンドを使用すると、名前領域内の他のリージョンに至るパスを動的に定義できます。 ドメイン <i>domain_name</i> は、Oracle Names Server 名として <i>onames_server</i> で定義されます。 <i>onames_server</i> は、そのアドレスとして定義されます。 ルート・リージョンでないリージョンに対しては、このコマンドを使用して少なくともルート・リージョンを定義しないと、他のリージョン内のオブジェクトを探ることができません。ローカル Oracle Names Server が他の特定のリージョンに直接アクセスできるように、追加のヒントを提供することができます。
例 :	<pre>namesctl> domain_hint acme.com ns0.acme.com (address=(protocol=tcp) (host=top.acme.com) (port=1575))</pre>

DUMP_TNSNAMES

目的:	ローカル・リージョンで定義されているすべてのアドレスの名前とアドレスを、TNSNAMES.ORA ファイルに書き出します。
前提条件:	なし
パスワード:	不要
構文:	オペレーティング・システムからは、次のように入力します。 <code>namesctl dump_tnsnames</code> NAMESCTL プログラムからは、次のように入力します。 <code>dump_tnsnames</code>
引数:	なし
使用上の注意:	ローカル・リージョンで定義されているすべてのアドレスの名前とアドレスを、TNSNAMES.ORA ファイルに書き出します (アドレス型レコード (A.SMD) をすべて)。 コマンドは、TNSNAMES.ORA ファイルを作成するか、一部を書き換えます。既存の TNSNAMES.ORA ファイルにある、Oracle Names Server で定義されていないエントリは、そのまま残ります。Oracle Names Server にも定義されている TNSNAMES.ORA ファイルのエントリは上書きされます。Oracle Names Server で定義されていて、TNSNAMES.ORA で定義されていないエントリは追加されます。
例:	<code>namesctl> dump_tnsnames</code>

EXIT

目的 :	NAMESCTL プログラムを終了します。
前提条件 :	NAMESCTL プログラムがロードされている必要があります。
パスワード :	不要
構文 :	NAMESCTL プログラムからは、次のように入力します。 EXIT
引数 :	なし
使用上の注意 :	EXIT コマンドは、Oracle Names Server には適用されず、 NAMESCTL プログラムのみに適用されます。 EXIT コマンドは QUIT コマンドと同じです。
例 :	<pre>namesctl> exit namesctl finished.</pre>

FLUSH

目的:	格納されている無許可データを Oracle Names Server キャッシュからすべて削除します。
前提条件:	複数のリージョンで構成されている環境のみに適用されます (データを集中管理する環境には、無許可データがありません)。
パスワード:	必要 パスワードが設定されている場合、このコマンドの前に SET PASSWORD コマンドを実行する必要があります。
構文:	オペレーティング・システム・プロンプトからは、次のように入力します。 <pre>namesctl flush [onames_server] ...</pre> NAMESCTL プログラムからは、次のように入力します。 <pre>flush [onames_server] ...</pre>
引数:	Oracle Names Server 名を空白で区切って指定します。引数を指定しない場合は、現行の Oracle Names Server のキャッシュからのみ外部名が削除されます。
使用上の注意:	FLUSH は、キャッシュに格納されている外部データすべてを消去します。通常は、次の場合に外部データをキャッシュから削除します。 <ul style="list-style-type: none">- ネットワーク内で大量のデータを変更して、通常の TTL エージング・メカニズムの所要時間が長すぎる場合。- キャッシュに格納されている外部データの名前を変換するときに、識別不能なエラーが発生した場合。キャッシュからすべての外部データを削除すると、次にそのデータが要求されたときに、再び検索されます。
例:	<pre>namesctl>flush Confirm [yes or no]: yes</pre>

FLUSH_NAME

目的 :	1 つ以上の無許可の名前を現行の Oracle Names Server のキャッシュから削除します。
前提条件 :	複数のリージョンで構成されている環境のみに適用されます (データを集中管理する環境には、無許可データがありません)。
パスワード :	必要 パスワードが設定されている場合、このコマンドの前に SET PASSWORD コマンドを実行する必要があります。
構文 :	オペレーティング・システム・プロンプトからは、次のように入力します。 <pre>namesctl flush_name name</pre> NAMESCTL プログラムからは、次のように入力します。 <pre>flush_name name</pre>
引数 :	1 つの名前
使用上の注意 :	FLUSH_NAME は、Oracle Names Server のリージョンの外部からキャッシュに書き込まれたデータ (つまり、無許可データ) のみを削除します。通常は、ソース・データが変更された可能性を示唆するように名前が異常に動作する場合に、外部データを削除します。 FLUSH_NAME は、現在の外部データ・キャッシュ、および現リージョンと正当なリージョンの間にある他のすべてのサーバーから名前を削除します。 名前は現行の Oracle Names Server から削除されます。現行の Oracle Names Server は、デフォルトの優先 Oracle Names Server、または SET SERVER コマンドで設定した Oracle Names Server です。
例 :	<pre>namesctl>flush_name mountain.acme.com</pre>

HELP

目的:	NAMESCTL コマンドの詳しい説明を表示します。
前提条件:	なし
パスワード:	不要
構文:	オペレーティング・システム・プロンプトからは、次のように入力します。 <code>namesctl help [command]</code> NAMESCTL プログラムからは、次のように入力します。 <code>help [command]</code>
引数:	help コマンド
使用上の注意:	HELP コマンドを使用すると、NAMESCTL の各コマンドの機能を素早く表示できます。引数を指定しないと、有効なコマンドのリストが表示されます。 引数を指定すると、そのコマンドの機能が 1 行で表示されます。
例:	<code>namesctl> help</code> 次の操作が行えます。 An asterisk (*) denotes a modifier or extended command: exit flush flush_name log_stats ping query quitreload repeat* reset_stats restart set* show shutdown start startup status stop version

LOAD_TNSNAMES

目的:	1 つ以上の TNSNAMES.ORA ファイルで定義されているすべてのネット・サービス名とアドレスを、Oracle Names Server にロードします。
前提条件:	なし
パスワード:	不要
構文:	オペレーティング・システム・プロンプトからは、次のように入力します。 namesctl load_tnsnames <i>file_name</i> [<i>file_name</i>] ... NAMESCTL プログラムからは、次のように入力します。 load_tnsnames <i>file_name</i> [<i>file_name</i>] ...
引数:	TNSNAMES.ORA 形式で、' <i>name = address</i> ' の定義を含む 1 つ以上のファイル名
使用上の注意:	一般的にこのコマンドは、TNSNAMES.ORA ファイルを使用した後、Oracle Names の使用を開始するときに 1 度実行します。Oracle Network Manager で管理されていたデータベース・テーブルをアップグレードする別の方法です。各リージョンでこのコマンドを 1 度実行してください。この操作中に定義された名前は永続的に定義され、1 つのサーバーからすべてのリージョンに伝達されます。
例:	namesctl>load_tnsnames network/admin/tnsnames.ora Name: koala.lab.npd.us.oracle.com Response status: normal, successful completion Name: devdd.rdbms.us.oracle.com Response status: normal, successful completion Name: envyd.lab.npd.us.oracle.com Response status: normal, successful completion Name: stealth.npd.us.oracle.com Response status: normal, successful completion Name: null.us.oracle.com Response status: normal, successful completion Name: slime.lab.npd.us.oracle.com Response status: normal, successful completion Name: felix.hp.us.oracle.com Response status: normal, successful completion Name: dtnet1.dec.oracle.com Response status: normal, successful completion Name: devds.rdbms.us.oracle.com Response status: normal, successful completion

LOG_STATS

目的:	現在の統計を、その Oracle Names Server の構成済みログ・ファイルに記録します。
前提条件:	なし
パスワード:	必要 パスワードが設定されている場合、このコマンドの前に SET PASSWORD コマンドを実行する必要があります。
構文:	オペレーティング・システム・プロンプトからは、次のように入力します。 <code>namesctl log_stats [onames_server] ...</code> NAMESCTL プログラムからは、次のように入力します。 <code>log_stats [onames_server] ...</code>
引数:	Oracle Names Server 名を空白で区切って指定します。引数を指定しないと、現行の Oracle Names Server の統計のみがリセットされません。
使用上の注意:	STATUS コマンドの結果または他の動作から、データをログに記録する必要があると判断した場合は、統計を記録できます。LOG_STATS は、現在の LOG_STATS_INTERVAL には影響しません。
例:	<code>namesctl> log_stats</code> Statistics counters logged.

PASSWORD

目的 :	Oracle Names Server の特権操作 (RELOAD、STOP など) を実行するのに必要なパスワードを登録します。
前提条件 :	NAMESCTL プログラムがロードされている必要があります。
パスワード :	N/A
構文 :	NAMESCTL プログラムからは、次のように入力します。 <code>password [password]</code>
引数 :	NAMES.ORA の NAMES.PASSWORD パラメータで暗号化されている値と一致するテキスト文字列
使用上の注意 :	<p>PASSWORD を使用すると、Oracle Names Server のパスワードは変更されず、NAMESCTL 変数のみが設定されます。この変数の値がコマンド要求時に NAMESCTL から Oracle Names Server に送信され、Oracle Names Server 上で構成されている値と比較されます。これらの値が一致している場合は、パスワードを必要とする操作が許可されます。</p> <p>特権操作、つまり Oracle Names Server の機能を変更する操作のみが、このコマンドの影響を受けます。SHOW や STATUS などは特権操作と見なされないため、パスワードは不要です。</p> <p>PASSWORD コマンドの引数としてパスワードを指定します。引数を指定しないと、パスワードの入力を促すプロンプトが表示されません。入力したパスワードは画面に表示されません。</p> <p>パスワードがネットワークで送信される場合は、パスワードがどのように設定されていても、必ず暗号化されます。</p>
例 :	<pre>namesctl> password open_sesame namesctl> password Enter name server password:</pre>

PING

目的:	現行の Oracle Names Server または指定したサーバーに接続し、要求 / 応答時間を表示します。
前提条件:	なし
パスワード:	不要
構文:	オペレーティング・システム・プロンプトからは、次のように入力します。 <code>namesctl ping [onames_server] ...</code> NAMESCTL プログラムからは、次のように入力します。 <code>ping [onames_server] ...</code>
引数:	Oracle Names Server 名を空白で区切って指定します。引数を指定しないと、現行の Oracle Names Server のみが ping されます。
使用上の注意:	ping は、Oracle Names Server が機能しているかどうかを確認してから、NAMESCTL ユーザーの場所から Oracle Names Server までの通常の応答時間を表示します。
例:	<code>namesctl> ping nserver.com</code> Round trip time is 0.04 seconds

QUERY

目的:	Oracle Names Server に格納されているネットワーク・オブジェクトの内容をテストまたは取り出します。
前提条件:	なし
パスワード:	不要
構文	オペレーティング・システム・プロンプトからは、次のように入力します。 <pre>namesctl query object_name [object_type] [modifiers]</pre> NAMESCTL プログラムからは、次のように入力します。 <pre>query object_name [object_type] [modifiers]</pre>

一般的なオブジェクト型

- A.SMD - データベース・サービス定義またはネット・サービス名にあるネットワーク・アドレス。
- CNAME.SMD - 別名 (" 標準名 " という場合もあります)。
- DL.RDBMS.OMD - グローバル・データベース・リンク。
- DLCR.RDBMS.OMD - グローバル・データベース・リンク修飾子。
- NS.SMD - Oracle Names Server アドレス。Oracle Names Server 間で通信する際に使用するシステム・データです。
- VIADD.NPO.OMD - SQL*Net バージョン 1 の接続文字列。

有効な修飾子

- AUTHORITY - データがローカル・キャッシュに格納されていても、問合せをデータのソース (データがローカルとみなされる管理リージョン内) で変換します。データがソースで変更された疑いがある場合に、この修飾子を使用できます。
- NOFORWARD - データを問い合わせますが、要求は送りません。データがローカル・データでないときに NOFORWARD を指定すると、問合せが変換されません。
- TRACE - 応答へのパスのトレースを実現します。要求の転送先 Oracle Names Server を調べる場合に役立ちます。

引数:	必須ネットワーク・オブジェクト名とネットワーク・オブジェクト型
-----	---------------------------------

QUERY

使用上の注意 :

QUERY を使用して、定義済みのデータを見つけられるかどうか、およびその内容が正しいかどうかをテストすることができます。

QUERY コマンドは、常にデフォルトの現行の Oracle Names Server または SET SERVER コマンドで指定した Oracle Names Server 上で動作します。

名前のみをパラメータに指定して QUERY コマンドを使用する場合、Oracle Names Server はその名前をもつデータの個数と操作の実行時間を知らせてきます。

QUERY コマンドの引数として名前と型を指定すると、特定の名前が検索されて、ユーザーに戻されます。

QUERY コマンドでは、複数の引数を指定できます。たとえば、次のように指定します。

```
query sales.com a.smd authority trace
```

例 :

```
namesctl> query bones.dem.medicine a.smd
Total response time:0.04 seconds
Response status:normal, successful completion
Authoritative answer:yes
Number of answers:1
Canonical name:bones.dem.medicine
TTL: 1 day
Alias translations:
    from:bones.dem.medicine
    to: bones.dem.medicine
Answers:
    data type is "a.smd"
    Syntax is ADDR:... (DESCRIPTION=(ADDRESS=
(PROTOCOL=TCP)(Host=bones-pc)
(Port=1521))(CONNECT_DATA=(SERVICE_NAME=bones.dem.medicine)))
```

QUIT

目的:	NAMESCTL プログラムを終了します。
前提条件:	NAMESCTL プログラムがロードされている必要があります。
パスワード:	不要
構文:	NAMESCTL プログラムからは、次のように入力します。 QUIT
引数:	なし
使用上の注意:	QUIT コマンドは、Oracle Names Server には適用されず、 NAMESCTL プログラムのみに適用されます。 QUIT コマンドは EXIT コマンドと同様に機能します。
例:	<pre>namesctl> quit nl-00851: namesctl finished</pre>

REGISTER

目的:	Oracle Names Server にネットワーク・オブジェクトを登録します。
パスワード:	不要
構文:	NAMESCTL プログラムからは、次のように入力します。 <pre>register object_name [-t service_type] [-d description=(address_list=(address= (protocol_address_ information)))] [-h onames_server] [-l listener_name]</pre>
	サービスの種類
	<ul style="list-style-type: none">■ ORACLE_DATABASE■ ORACLE_LISTENER■ ORACLE_NAMESERVER
引数:	オブジェクト名は必須です。サービスおよびアドレス・データ、ホストを指定しなくても、登録プロセスを実行できます。ただし、登録したオブジェクトを使用するには、これらのデータが必要です。つまり、アドレスを指定しないで登録したオブジェクト名は使用できません。

REGISTER

使用上の注意 : REGISTER は、サービスおよびサービスのタイプ、ホスト名、アドレスを手動で登録するためのコマンドです。サービスのタイプとデータを有効な文字列で指定できますが、一般的な登録では、サービスの種類として database または listener を指定し、データとして TNS アドレスを指定します。

オブジェクトの登録は、リージョン内の他のすべての予約済み Oracle Names Server に伝えられます。

例 :

```
namesctl> register parts -t oracle_database -d (DESCRIPTION=
(AADDRESS= (PROTOCOL=TCP)(HOST=nineva)(PORT=1575)) (CONNECT_
DATA=(SERVICE_NAME=db3)))
```

RELOAD

目的 : サーバーの管理リージョン内でデータが変更されたかどうかを直ちにチェックします。変更されたデータがある場合は、すべてのデータベース・サービス名、ネット・サービス名、グローバル・データベース・リンク、および別名を再ロードします。

前提条件 : なし

パスワード : 必要
パスワードが設定されている場合、このコマンドの前に SET PASSWORD コマンドを実行する必要があります。

構文 : オペレーティング・システム・プロンプトからは、次のように入力します。

```
namesctl reload [onames_server] ...
```

NAMESCTL プログラムからは、次のように入力します。

```
reload [onames_server] ...
```

引数 : Oracle Names Server 名を空白で区切って指定します。引数を指定しないと、現行の Oracle Names Server のみが再ロードされます。

使用上の注意 : すべての Oracle Names Server は、NAMES.ORA ファイルの NAMES.ADMIN_REGION 構成パラメータで指定されているデータベースから、データを直接ロードします。

複数のリージョンで構成されている環境では、RELOAD が現管理リージョンのデータのみ適用されます。キャッシュ内の外部データはすべて変更されません。

例 :

```
namesctl> reload
Server reloaded.
```

REORDER_NS

目的:	ローカル Oracle Names Server とそのアドレスがリストされたファイルを作成します。
前提条件:	なし
パスワード:	不要
構文:	オペレーティング・システム・プロンプトからは、次のように入力します。 <pre>namesctl reorder_ns [description=(address_list= (address=(protocol_address_information)))]</pre> NAMESCTL プログラムからは、次のように入力します。 <pre>reorder_ns [description=(address_list= (address= (protocol_address_information)))]</pre>
引数:	オプションの Oracle Names Server アドレスが、接続するための最初のサーバーとして使用されます。
使用上の注意:	このコマンドは、クライアントが名前を調べるために Oracle Names Server に接続できるように、Oracle Names Server の名前とアドレスが定義されたファイルを生成します。 REORDER_NS コマンドは、次のことを実行します。 <ol style="list-style-type: none">1. Oracle Names Server を検索します。SQLNET.ORA で優先 Oracle Names Server パラメータを探します。続いて、予約済みのサーバーを呼び出すか、ポート 1575 の TCC/IP を使用してローカル Oracle Names Server を検索します。アドレス引数がある場合は、その引数が使用されます。2. ローカル・リージョン内のすべての Oracle Names Server に対して問合せを送ります。3. これらのサーバーのそれぞれに PING を送ります。4. サーバーのリストを応答時間の短い順にソートします。5. Oracle Names Server リストにソートした名前とアドレスのリストを書き込みます。
例:	<pre>namesctl> reorder_ns (address=(protocol=tcp)(host=nineva)(port=1575))</pre>

REPEAT

目的:	QUERY または REGISTER、TIMED_QUERY、UNREGISTER を複数回実行して、平均戻り速度を計算します。
前提条件:	なし
パスワード:	不要
構文:	オペレーティング・システム・プロンプトからは、次のように入力します。 <code>namesctl repeat number query type</code> NAMESCTL プログラムからは、次のように入力します。 <code>repeat number query type</code> <i>number</i> は整数、 <i>type</i> は QUERY コマンドに表示されているとおりです。
引数:	なし
使用上の注意:	REPEAT コマンドは、要求の平均応答時間を知るのに役立ちます。 実行回数として、あまり大きな数を指定しないでください。実行を繰り返している間、NAMESCTL プログラムは他の作業を実行できません。
例:	<pre>namesctl> repeat 10 query manatee a.smd Number of requests: 10 Average response time: 0.01 seconds Minimum response time: 0.01 seconds Maximum response time:0.04 seconds Total response time:0.14 seconds Response status:normal, successful completion Authoritative answer:yes Number of answers: 1 TTL: 1 day Answers: data type is "a.smd" Syntax is ADDR:(DESCRIPTION=(ADDRESS= (PROTOCOL=TCP)(Host=salmon) (Port=1522))(CONNECT_DATA=(SID=otter)))</pre>

RESET_STATS

目的 :	起動時に、Oracle Names Server の統計を Oracle Names Server の元の値にリセットします。
前提条件 :	なし
パスワード :	必要 パスワードが設定されている場合、このコマンドの前に SET PASSWORD コマンドを実行する必要があります。
構文 :	オペレーティング・システム・プロンプトからは、次のように入力します。 <code>namesctl reset_stats [onames_server] ...</code> NAMESCTL プログラムからは、次のように入力します。 <code>reset_stats [onames_server] ...</code>
引数 :	Oracle Names Server 名を空白で区切って指定します。引数を指定しないと、現行の Oracle Names Server の統計のみがリセットされません。
使用上の注意 :	統計が直ちにリセットされることを除けば、RESET_STATS は、RESET_STATS_INTERVAL (NAMES.ORA パラメータ) が終了するのを待つと同じことです。
例 :	<code>namesctl> reset_stats</code> Confirm [yes or no]: yes Server statistics reset.

RESTART

目的:	起動時に Oracle Names Server を元の状態にリセットします。
前提条件:	なし
パスワード:	必要 パスワードが設定されている場合、このコマンドの前に SET PASSWORD コマンドを実行する必要があります。
構文:	オペレーティング・システム・プロンプトからは、次のように入力します。 <code>namesctl restart [onames_server] ...</code> NAMESCTL プログラムからは、次のように入力します。 <code>restart [onames_server] ...</code>
引数:	Oracle Names Server 名を空白で区切って指定します。引数を指定しないと、現行の Oracle Names Server のみが再起動されます。
使用上の注意:	Oracle Names Server がすでに稼働しているという点を除いて、RESTART は STARTUP と同じです。 データが再ロードされ、統計がリセットされ、すべての外部データが削除されます。有効な外部キャッシュ・データ（つまり、TTL が 0 より大きいデータ）が、チェックポイント・ファイルから取り出されず（TTL の値は、0 より大きく設定する必要があります）。
例:	<code>namesctl> restart</code> Confirm [yes or no]: yes Server restarted.

SET

目的:	SET コマンドで設定できるパラメータ値を表示します。
前提条件:	なし
パスワード:	必要 パスワードが設定されている場合、このコマンドの前に SET PASSWORD コマンドを実行する必要があります。
構文:	namesctl> set
引数:	なし
使用上の注意:	<p>リスナー構成パラメータ <code>PASSWORDS_listener_name</code> に、このコマンドが表示されている場合、このコマンドを使用するには有効なパスワードを設定しておく必要があります。</p> <p>リスナー構成ファイルにパスワードがある場合は、SET コマンドを使用する前に SET PASSWORD コマンドを使用する必要があります。</p> <p>パスワードは LSNRCTL プログラムの中から設定する必要があります。オペレーティング・システムのコマンド行からは、パスワードを設定できません。パスワードを設定する方法は、暗号化パスワード機能を使用しているかどうかによって決まります。暗号化されたパスワードを使用しない場合は、LSNRCTL コマンド行にそのパスワードを入力します。</p>
例:	<pre>namesctl> set</pre> <p>set の後には、次の操作が行えます。</p> <p>An asterisk (*) denotes a modifier or extended command:</p> <pre>cache_checkpoint_interval default_domain forwarding_available log_file_name log_stats_interval namesctl_trace_level password requests_enabled reset_stats_interval save_config_interval save_config_on_stop server trace_file_name trace_level</pre>

SET CACHE_CHECKPOINT_INTERVAL

目的: 外部のリージョンについて収集したすべての情報を、Oracle Names Server のキャッシュ・ファイルに保存する頻度を設定します。

パスワード: 不要

構文: オペレーティング・システムからは、次のように入力します。

```
namesctl set cache_checkpoint_interval time
```

NAMESCTL プログラムから、次のように入力します。

```
set cache_checkpoint_interval time
```

引数: time は秒単位です。

たとえば、CACHE_CHECKPOINT_INTERVAL を 36 時間に設定する場合は、次のように入力します。

```
set cache_checkpoint_interval 129600
```

使用上の注意: 最小値: 10 秒

最大値: 259200 (3 日)

デフォルト: 0 (キャッシュ頻度なし)

例:

```
namesctl> set cache_checkpoint_interval 10
```

SET DEFAULT_DOMAIN

目的:	NAMESCTL クライアントのデフォルト・ドメインを設定または変更します。
前提条件:	NAMESCTL プログラムがロードされている必要があります。
パスワード:	不要
構文:	NAMESCTL プログラムからは、次のように入力します。 <pre>set default_domain domain_name</pre>
引数:	1つのドメイン名
使用上の注意:	<p>DEFAULT_DOMAIN パラメータを設定すると、そのドメイン内の名前を非修飾名で指定できます。たとえば、DEFAULT_DOMAIN を US.ACME に設定してある場合は、グローバル名 WIDE.US.ACME を次のようにして問い合わせることができます。</p> <pre>namesctl> query wide</pre> <p>DEFAULT_DOMAIN の初期値は、SQLNET.ORA ファイルの NAMES.DEFAULT_DOMAIN パラメータから、NAMESCTL プログラムを開始するときに設定されます。</p> <p>引数を指定しないと、SQLNET.ORA からデフォルトが読み込まれて割り当てられます。</p> <p>SET DEFAULT_DOMAIN を使用すると、ドメインごとに名前を処理する作業が容易になります。</p>
例:	<pre>namesctl> set default_domain us.acme Default domain is now "US.ACME"</pre>

SET FORWARDING_AVAILABLE

目的:	Oracle Names Server の名前要求転送機能をオンまたはオフにします。
前提条件:	Oracle Names Server が稼働している必要があります。
パスワード:	必要 パスワードが設定されている場合、このコマンドの前に SET PASSWORD コマンドを実行する必要があります。
構文:	オペレーティング・システムからは、次のように入力します。 <code>namesctl set forwarding_available on off yes no</code> NAMESCTL プログラムからは、次のように入力します。 <code>set forwarding_available off</code>
引数:	ON OFF YES NO
デフォルト値:	OFF
使用上の注意:	<p>この設定が適用されるのは、ローカル・クライアントをもたず、外部 Oracle Names Server からの要求のみを処理する Oracle Names Server のみです。通常は、クライアントまたはサービスをもたないルート・リージョン内の Oracle Names Server のみに適用されます。このようなサーバーがリージョン間の要求処理でボトルネックの原因となる場合は、転送機能をオフにして、その Oracle Names Server の作業負荷を半減できます。転送機能がオフに設定された Oracle Names Server は、要求を転送して応答を戻すかわりに、要求に応答できる Oracle Names Server のアドレスを要求側に知らせます。この場合、作業負荷の全体量は減少しないで、転送機能のない Oracle Names Server から要求側の Oracle Names Server に作業負荷が移動するのみです。</p> <p>警告: SET FORWARDING_AVAILABLE をオフに設定すると、その Oracle Names Server に直接依存しているすべてのクライアントは外部名を変換できません。クライアントは、Oracle Names Server のように要求をリダイレクトできません。NAMES.PREFERRED_SERVERS 構成パラメータに他の Oracle Names Server がリストされていても、クライアントの要求は失敗します。</p>
例:	<code>namesctl> set forwarding_available off</code> Request processing is now disabled.

SET LOG_FILE_NAME

目的:	ログ・ファイルの名前を変更します。デフォルトの名前は NAMES.LOG です。
前提条件:	なし
パスワード:	必要 パスワードが設定されている場合、このコマンドの前に SET PASSWORD コマンドを実行する必要があります。
構文:	オペレーティング・システムからは、次のように入力します。 <code>namesctl set log_file_name file_name</code> NAMESCTL プログラムからは、次のように入力します。 <code>set log_file_name file_name</code>
引数:	ログ・ファイルのファイル名
使用上の注意:	LOG_FILE_NAME は、すべてのロギング・メッセージの宛先を変更します。
例:	<code>namesctl> set log_file_name namesvr1</code>

SET LOG_STATS_INTERVAL

目的:	統計がログ・ファイルに記録される頻度を変更します。
前提条件:	なし
パスワード:	必要 パスワードが設定されている場合、このコマンドの前に SET PASSWORD コマンドを実行する必要があります。
構文:	オペレーティング・システムからは、次のように入力します。 <code>namesctl set log_stats_interval time</code> NAMESCTL プログラムからは、次のように入力します。 <code>namesctl> set log_stats_interval time</code>
引数:	time は、秒数、または [<i>n</i> > DAY[S]] [<i>hh</i> >:<mi>:<ss>] で指定します。 たとえば、LOG_STATS_INTERVAL を 36 時間に設定する場合は、次のように入力します。 <code>set log_stats_interval 129600</code> <code>set log_stats_interval 1 day 12:00:00</code> 日数と時間 : 分 : 秒を組み合わせで指定するか、日数を指定しないで時間のみを指定できます。
制限:	最小値 : 10 秒 最大値 : 制限なし 特殊値 : 0 (リセットされないことを意味する) デフォルト値 : 0 (記録しない)
使用上の注意:	LOG_STATS_INTERVAL の初期値は、Oracle Network Manager で構成されている値、または Oracle Names Server のロード時に SQLNET.ORA ファイルにある NAMES.LOG_STATS_INTERVAL の値に基づいて設定されます。デフォルト値は 0 (記録しない) です。SET LOG_STATS_INTERVAL は、サーバーの稼働中にこの値を上書きするためのコマンドです。
例:	<code>namesctl> set log_stats_interval 7200</code> Statistic counter logging interval is now 2 hours

SET NAMESCTL_TRACE_LEVEL

目的:	NAMESCTL プログラムのトレース・レベルを指定します。
前提条件:	なし
パスワード:	必要 パスワードが設定されている場合、このコマンドの前に SET PASSWORD コマンドを実行する必要があります。
構文:	オペレーティング・システムからは、次のように入力します。 <code>namesctl set namesctl_trace_level off user admin support</code> NAMESCTL プログラムからは、次のように入力します。 <code>namesctl> set namesctl_trace_level off user admin support</code>
引数:	OFF USER ADMIN SUPPORT
使用上の注意:	トレースは、NAMESCTL プログラムの処理で発生する予測不能な障害または識別不能な障害を診断するのに役立ちます。トレースは、通常の NAMESCTL 処理で行われる一連のイベントをオペレーティング・システム・ファイルに書き込んで、管理者が検査できるようにします。 トレース出力は、OFF (なし)、USER (基本的情報) または ADMIN という 3 段階のレベルで指定します。 引数を指定しないと、クライアントの SQLNET.ORA ファイル内の値にリセットされます。デフォルト設定は OFF です。
例:	<code>namesctl> set namesctl_trace_level admin</code> Controller's local trace level changed from 0 to 4

SET PASSWORD

目的:	Oracle Names Server の特権操作 (RELOAD、STOP など) を実行するのに必要なパスワードを登録します。
前提条件:	NAMESCTL プログラムがロードされている必要があります。
パスワード:	N/A
構文:	NAMESCTL プログラムからは、次のように入力します。 <pre>namesctl> set password [password]</pre>
引数:	現行の Oracle Names Server のパラメータ NAMES.PASSWORD に格納されている値と一致するテキスト文字列
使用上の注意:	<p>SET PASSWORD では、Oracle Names Server のパスワードは変更されません。この変数の値が、NAMESCTL コマンドで Oracle Names Server に送信され、Oracle Names Server 上で構成されている値と比較されます。これらの値が一致している場合は、パスワードを必要とする操作が許可されます。</p> <p>特権操作、つまり Oracle Names Server の機能を変更する操作のみが、このコマンドの影響を受けます。SHOW や STATUS などは特権操作と見なされないため、パスワードは不要です。</p> <p>SET PASSWORD コマンドの引数としてパスワードを指定します。引数を指定しないと、パスワードの入力を促すプロンプトが表示されます。入力したパスワードは画面に表示されません。</p> <p>パスワードがネットワークで送信される場合は、パスワードがどのように設定されていても、必ず暗号化されます。</p>
例:	<pre>namesctl> set password open_sesame namesctl> set password enter name server password:</pre>

SET REQUESTS_ENABLED

目的 :	現行の Oracle Names Server が要求に応答するかどうかを決定します。
前提条件 :	なし
パスワード :	必要 パスワードが設定されている場合、このコマンドの前に SET PASSWORD コマンドを実行する必要があります。
構文 :	オペレーティング・システムからは、次のように入力します。 <code>namesctl requests_enabled on off</code> NAMESCTL プログラムからは、次のように入力します。 <code>namesctl> set requests_enabled on off</code>
引数 :	ON OFF
使用上の注意 :	このプロパティを OFF に設定すると、ネーム・サーバーに名前を要求するすべてのクライアントが拒否されます。Oracle Names Server が予期せず動いているときは、この設定を使用してネーム・サーバーを診断できます。
例 :	<code>namesctl> set requests_enabled off</code> Confirm [yes or no]: yes General request processing is now disabled

SET RESET_STATS_INTERVAL

目的:	統計がゼロまたは現サーバーの初期値にリセットされる間隔を変更します。
前提条件:	なし
パスワード:	必要 パスワードが設定されている場合、このコマンドの前に SET PASSWORD コマンドを実行する必要があります。
構文:	オペレーティング・システムからは、次のように入力します。 <code>namesctl set reset_stats_interval time</code> NAMESCTL プログラムからは、次のように入力します。 <code>namesctl> set reset_stats_interval time</code>
引数:	time は、次のいずれかのフォーマットで指定します。 秒数 <code>[nDAY[S]] [hh:mi:ss]</code> たとえば、RESET_STATS_INTERVAL を 72 時間に設定する場合は、次のように入力します。 <code>set reset_stats_interval 259200</code> <code>set reset_stats_interval 3 days</code>
制限:	最小値: 10 秒 最大値: 制限なし デフォルト値: 0 (リセットしない)
使用上の注意:	RESET_STATS_INTERVAL の初期値は、Oracle Names Server をロードするときの NAMES.RESET_STATS_INTERVAL パラメータに基づいて設定されます。SET RESET_STATS_INTERVAL は、Oracle Names Server の稼働中にこの値を上書きするためのコマンドです。
例:	<code>namesctl> set reset_stats_interval 1 day</code> Statistic counter reset interval is now 24 hours

SET SAVE_CONFIG_INTERVAL

目的:	NAMESCTL SET コマンドで行った変更が、一定間隔で NAMES.ORA に保存されます。
パスワード:	必要 パスワードが設定されている場合、このコマンドの前に SET PASSWORD コマンドを実行する必要があります。
構文:	オペレーティング・システムからは、次のように入力します。 <code>namesctl set save_config_interval time</code> NAMESCTL プログラムからは、次のように入力します。 <code>set save_config_interval time</code>
引数:	time は秒単位です。
例:	<code>namesctl> set save_config_interval 10</code>

SET SAVE_CONFIG_ON_STOP

目的:	パラメータを ON に設定すると、NAMESCTL SET コマンドで行った変更が NAMES.ORA に保存されます。NAMESCTL が終了する直前にすべてのパラメータが保存されます。パラメータの形式、コメント、大文字と小文字は、可能な限り維持されます。
パスワード:	必要 パスワードが設定されている場合、このコマンドの前に SET PASSWORD コマンドを実行する必要があります。
構文:	オペレーティング・システムからは、次のように入力します。 <code>namesctl set save_config_on_stop on off</code> NAMESCTL プログラムからは、次のように入力します。 <code>namesctl> set save_config_on_stop on off</code>
引数:	ON OFF
例:	<code>namesctl> set save_config_on_stop on</code>

SET SERVER

目的:	現行の Oracle Names Server を変更します。
前提条件:	NAMESCTL プログラムがロードされている必要があります。
パスワード:	不要
構文:	NAMESCTL プログラムからは、次のように入力します。 <pre>namesctl> set server [<i>onames_server</i>] [description=(address_list= (address= (<i>protocol_address_</i> <i>information</i>))))]</pre>
引数:	有効な Oracle Names Server または有効な Oracle Names Server アドレス
使用上の注意:	SET SERVER を使用すると、NAMESCTL ユーティリティの実行中に、Oracle Names Server を切り替えることができます。修飾子として指定できるのは、現行の Oracle Names Server のメモリーに定義されている名前、または任意の Oracle Names Server の TNS アドレスです。 指定した Oracle Names Server 名は、通常の名前検索で変換されます。現行の Oracle Names Server がアドレスを知っているか取り出せる場合のみ、別の Oracle Names ネーム・サーバーを設定できます。現行の Oracle Names Server が設定されていない場合は、TNS アドレスを入力しないとこのコマンドを実行できません。引数を指定しない場合は、NAMES.PREFERRED_SERVERS を使用します。
例:	<pre>namesctl> set server server1.us.acme</pre>

SET TRACE_FILE_NAME

目的:	トレース情報を書き込むファイルの名前を変更します。
前提条件:	なし
パスワード:	不要
構文:	オペレーティング・システムからは、次のように入力します。 <code>namesctl set trace_file_name file_name</code> NAMESCTL プログラムから、次のように入力します。 <code>namesctl> set trace_file_name file_name</code>
引数:	ファイル名
例:	<code>namesctl> set trace_file_name namesvr1</code>

SET TRACE_LEVEL

目的:	現行の Oracle Names Server をトレースするための TRACE_LEVEL を変更します。
前提条件:	なし
パスワード:	必要 パスワードが設定されている場合、このコマンドの前に SET PASSWORD コマンドを実行する必要があります。
構文:	オペレーティング・システムからは、次のように入力します。 namesctl set trace_level off user admin support NAMESCTL プログラムからは、次のように入力します。 namesctl> set trace_level off user admin support
引数:	OFF USER ADMIN SUPPORT <ul style="list-style-type: none">■ OFF - トレース出力しない■ USER - ユーザー・トレース情報■ ADMIN - 管理トレース情報■ SUPPORT - オラクル社サポート・サービス・トレース情報
使用上の注意:	トレースは、現行の Oracle Names Server の処理で発生する予測不能な障害、または識別不能な障害を診断するのに役立ちます。トレースは、通常の Oracle Names Server 処理で行われる一連のイベントをオペレーティング・システム・ファイルに書き込んで、管理者が検査できるようにします。 トレース出力は、OFF (なし)、USER (基本的情報)、ADMIN (最大量の情報) または SUPPORT (サポート・サービスに必要な情報) という 4 段階のレベルで指定します。 TRACE_LEVEL を設定すると、トレースが直ちに開始されます。トレース・レベルを OFF にリセットするまで、すべての操作がトレースされます。 トレース・ファイルは非常に大きくなります。問題を診断した後は、トレース・レベルを必ずオフにしてください。
例:	namesctl> set trace_level admin Trace level is now 6.

SHOW CACHE_CHECKPOINT INTERVAL

目的:	Oracle Names Server のキャッシュが、チェックポイント・ファイルに書き込まれる頻度を表示します。
前提条件:	なし
パスワード:	不要
構文:	オペレーティング・システムからは、次のように入力します。 <pre>namesctl show cache_checkpoint_interval</pre> NAMESCTL プログラムからは、次のように入力します。 <pre>namesctl> show cache_checkpoint_interval</pre>
引数:	なし
使用上の注意:	CACHE_CHECKPOINT_INTERVAL の初期値は、NAMES.ORA ファイルにある NAMES.CACHE_CHECKPOINT_INTERVAL の値に設定されます。デフォルト値は0 (ゼロ) で、cache_checkpoint は機能しません。キャッシュ・チェックポイント・ファイルに書き込まれるデータには、ネット・サービスの名前とアドレス、およびクライアントにかわって Oracle Names Server が外部リージョンに問合せを転送して取得した Oracle Names Server アドレスなどがあります。
例:	<pre>namesctl> show cache_checkpoint_interval</pre> Cache checkpoint interval is currently 8 minutes 20 seconds

SHOW FORWARDING_AVAILABLE

目的:	外部名の要求を、Oracle Names Server が転送するかリダイレクトするかを表示します。
前提条件:	なし
パスワード:	不要
構文:	オペレーティング・システムからは、次のように入力します。 <code>namesctl show forwarding_available</code> NAMESCTL プログラムからは、次のように入力します。 <code>namesctl> show forwarding_available</code>
引数:	Oracle Names Server 名を空白で区切って指定します。名前を指定しないと、現ネーム・サーバーの設定が表示されます。
使用上の注意:	デフォルトでは、すべての Oracle Names Server が外部名の要求を転送します。転送が使用禁止になっている場合は、要求された名前に対して権限をもつリージョン内の Oracle Names Server に、外部名の要求がリダイレクトされます。 要求転送を使用禁止にすると、特定のサーバーの負荷が軽減されません。また、そのサーバーに直接依存しているクライアントは、外部名を変換できません。クライアントはリダイレクトされず、他の Oracle Names Server のみがリダイレクトされます。SET FORWARDING_AVAILABLE も参照してください。
例:	<code>namesctl> show forwarding_available</code> Request forwarding is currently enabled

SHOW DEFAULT_DOMAIN

目的:	NAMESCTL クライアントのデフォルト・ドメインを表示します。
前提条件:	なし
パスワード:	不要
構文:	オペレーティング・システムからは、次のように入力します。 <pre>namesctl show default_domain</pre> NAMESCTL プログラムからは、次のように入力します。 <pre>namesctl> show default_domain</pre>
引数:	なし
使用上の注意:	DEFAULT_DOMAIN パラメータを設定すると、そのドメイン内の名前を非修飾名で指定できます。たとえば、DEFAULT_DOMAIN を ACME.COM に設定すると、グローバル名 WIDE.ACME.COM を次のようにして問い合わせることができます。 <pre>namesctl> query wide</pre> <pre>Total response time: 0.20 seconds</pre> <pre>Response status: normal, successful completion</pre> <pre>Authoritative answer: yes</pre> <pre>Number of answers: 0</pre> <pre>TTL: 1 day</pre> DEFAULT_DOMAIN の初期値は、SQLNET.ORA ファイルの NAMES.DEFAULT_DOMAIN パラメータから NAMESCTL プログラムを開始するときに設定されます。 ユーザーが現デフォルト・ドメインがわからない場合、または現構成のデフォルトを知りたい場合に、SHOW DEFAULT_DOMAIN を使用します。
例:	<pre>namesctl> show default_domain</pre> <pre>Current default domain is "com"</pre>

SHOW LOG_FILE_NAME

目的:	Oracle Names Server がロギング情報を書き込むファイルの名前を表示します。
前提条件:	なし
パスワード:	不要
構文:	オペレーティング・システムからは、次のように入力します。 <code>namesctl show log_file_name</code> NAMESCTL プログラムからは、次のように入力します。 <code>namesctl> show log_file_name</code>
引数:	なし
使用上の注意:	LOG_FILE_NAME の初期値は、NAMES.ORA ファイルにある NAMES.LOG_FILE_NAME の値に設定されます。デフォルト値は NAMES.LOG で、UNIX では \$ORACLE_HOME/network/trace、Windows NT では <i>ORACLE_HOME</i> ¥network¥trace に置かれます。
例:	<code>namesctl> show log_file_name</code> Log file name is currently /private/ora23/network/names.log

SHOW LOG_STATS_INTERVAL

目的:	統計がログ・ファイルに記録される頻度を表示します。
前提条件:	なし
パスワード:	不要
構文:	オペレーティング・システムからは、次のように入力します。 <pre>namesctl show [onames_server] ... log_stats_interval</pre> NAMESCTL プログラムから、次のように入力します。 <pre>namesctl> show [onames_server] ... log_stats_interval</pre>
引数:	Oracle Names Server 名を空白で区切って指定します。名前を指定しないと、現ネーム・サーバーの設定が表示されます。
使用上の注意:	LOG_STATS_INTERVAL の初期値は、NAMES.ORA ファイルにある NAMES.LOG_STATS_INTERVAL の値に設定されます。デフォルト値は 0 (記録しない) です。
例:	<pre>namesctl> show log_stats_interval Statistic counter logging is currently disabled</pre>

SHOW NAMESCTL_TRACE_LEVEL

目的:	NAMESCTL プログラムのトレース・レベルを表示します。
前提条件:	なし
パスワード:	不要
構文:	オペレーティング・システムからは、次のように入力します。 <code>namesctl show namesctl_trace_level</code> NAMESCTL プログラムからは、次のように入力します。 <code>namesctl> show namesctl_trace_level</code>
引数:	なし
使用上の注意:	トレースは、NAMESCTL プログラムの処理で発生する予測不能な障害または識別不能な障害を診断するのに役立ちます。トレースは、通常の NAMESCTL 処理で行われる一連のイベントをオペレーティング・システム・ファイルに書き込んで、管理者が検査できるようにします。 トレース出力は、OFF (なし)、USER (基本的情報) または ADMIN (最大量の情報) という 3 段階のレベルで指定します。 SHOW NAMESCTL_TRACE_LEVEL は、NAMESCTL プログラムに対する現在のトレース・レベルを確認するための唯一の方法です。
例:	<code>namesctl> show namesctl_trace_level</code> Controller's trace level is currently 0

SHOW REQUESTS_ENABLED

目的 :	Oracle Names Server が要求に応答しているかどうかを表示します。
前提条件 :	なし
パスワード :	不要
構文 :	オペレーティング・システムからは、次のように入力します。 <pre>namesctl show [onames_server] ... requests_enabled</pre> NAMESCTL プログラムからは、次のように入力します。 <pre>namesctl> show [onames_server] ... requests_enabled</pre>
引数 :	Oracle Names Server 名を空白で区切って指定します。名前を指定しないと、現ネーム・サーバーの設定が表示されます。
使用上の注意 :	REQUESTS_ENABLED がオフになっている場合は、Oracle Names Server に対するすべての要求が拒否されます。このパラメータは診断用にもみ使用します。
例 :	<pre>namesctl> show requests_enabled General request processing is currently enabled</pre>

SHOW RESETS_STATS_INTERVAL

目的:	統計をログ・ファイルにダンプする間隔を表示します。
前提条件:	なし
パスワード:	不要
構文:	オペレーティング・システムからは、次のように入力します。 <code>namesctl show reset_stats_interval</code> NAMESCTL プログラムからは、次のように入力します。 <code>namesctl> show reset_stats_interval</code>
使用上の注意:	RESET_STATS_INTERVAL の初期値は、NAMES.RESET_STATS_INTERVAL の値に設定されます。デフォルト値は 0 (リセットしない) です。この場合は、Oracle Names Server が稼働している全時間の統計が蓄積されます。たとえば、統計が毎日リセットされる場合、統計はサーバーが動作していた総時間ではなく当日の合計を表します。
例:	<code>namesctl> show reset_stats_interval</code> Statistic counter reset interval is currently 5 minutes

SHOW SERVER

目的:	現行の Oracle Names Server 名を表示します。
前提条件:	なし
パスワード:	不要
構文:	オペレーティング・システムからは、次のように入力します。 <code>namesctl show server</code> NAMESCTL プログラムからは、次のように入力します。 <code>namesctl> show server</code>
引数:	なし
使用上の注意:	SHOW SERVER は、コマンドが適用される現行の Oracle Names Server を表示します。
例:	<code>namesctl> show server</code> currently managing name server "NameServer.us.oracle.com Version banner is "Oracle Names for SunOS: Version 8.1.5.0.0"

SHOW STATUS

目的 :	Oracle Names Server の一般的なステータス情報を表示します。
前提条件 :	なし
パスワード :	不要
構文 :	オペレーティング・システムからは、次のように入力します。 <pre>namesctl show [onames_server] ... status</pre> NAMESCTL プログラムからは、次のように入力します。 <pre>namesctl> show [onames_server] ... status</pre>
引数 :	Oracle Names Server 名を空白で区切って指定します。名前を指定しないと、現ネーム・サーバーの設定が表示されます。
使用上の注意 :	Oracle Names Server の現在の状態を表示します。 STATUS コマンドと同じです。
例 :	<pre>namesctl> show status Version Banner is "Oracle Names for SunOS: Version 8.1.5.0.0" Server has been running for:1 day 2 hours 3 minutes 35.16 seconds....</pre>

SHOW SYSTEM_QUERIES

目的:	すべてのシステム問合せの次のオカレンスを表示します。
前提条件:	このコマンドは分散構成のみに適用されます。1つの管理リージョンのみで構成されている環境では、システム問合せが実行されません。
パスワード:	不要
構文:	オペレーティング・システムからは、次のように入力します。 <code>namesctl show system_queries</code> NAMESCTL プログラムからは、次のように入力します。 <code>namesctl> show system_queries</code>
引数:	なし
使用上の注意:	システム問合せは、Oracle Names Server 間の情報を最新の状態に保つために、ときどき実行されます。 システム問合せとしてリストされているアクティビティを変更できるアクションは特にありません。システム問合せを表示することによって、管理者はシステムの変更がいつ行われるかを理解できません。また、RESTART を実行してシステム・データを強制的に再ロードする必要があるかどうかを判断することもできます。
例:	<code>namesctl> show system_queries</code> System query index number:1 Query ID:49824 Query next issued in:2 hours 55 min 3.84 seconds Query state:2 Name:"" Desired data type:ns.smd

SHOW TRACE_FILE_NAME

目的:	現行の Oracle Names Server の、トレース・ファイル名とパスを表示します。
前提条件:	なし
パスワード:	不要
構文:	オペレーティング・システムからは、次のように入力します。 <code>namesctl show trace_file_name</code> NAMESCTL プログラムからは、次のように入力します。 <code>namesctl> show trace_file_name</code>
引数:	なし
使用上の注意:	TRACE_FILE_NAME の初期値は、NAMES.ORA ファイルにある NAMES.TRACE_FILE_NAME の値に設定されます。デフォルト値は NAMES.TRC で、UNIX では \$ORACLE_HOME/network/trace、Windows NT では ORACLE_HOME¥network¥trace に置かれます。このファイルは有効なファイル名をもち、Oracle Names Server がデータを書き込めるファイルでなければなりません。 NAMES.TRACE_LEVEL でトレースを使用可能にしてある場合のみ、このファイルが使用されます。
例:	<code>namesctl> show trace_file_name</code> Trace file name is currently /private/ora23/network/names.trc

SHOW TRACE_LEVEL

目的:	現行の Oracle Names Server をトレースする、トレース・レベルを表示します。
前提条件:	なし
パスワード:	不要
構文:	オペレーティング・システムからは、次のように入力します。 <code>namesctl show trace_level</code> NAMESCTL プログラムからは、次のように入力します。 <code>namesctl> show trace_level</code>
引数:	なし
使用上の注意:	トレースは、現行の Oracle Names Server の処理で発生する予測不能な障害、または識別不能な障害を診断するのに役立ちます。トレースは、通常の Oracle Names Server 処理で行われる一連のイベントをオペレーティング・システム・ファイルに書き込んで、管理者が検査できるようにします。 トレース出力は、OFF (なし)、USER (基本的情報) または ADMIN (最大量の情報) という 3 段階のレベルで指定します。 SHOW TRACE_LEVEL は、現ネーム・サーバーのトレース・レベルを確認するための唯一の方法です。Oracle Names Server 構成ファイルで TRACE_LEVEL が構成されていても、NAMESCTL プログラムからのコールによってその値が上書きされる場合があります。
例:	<code>namesctl> show trace_level</code> Trace level is currently 0

SHOW VERSION

目的 :	Oracle Names Server の現在のバージョンと名前を表示します。
前提条件 :	なし
パスワード :	不要
構文 :	オペレーティング・システムからは、次のように入力します。 <pre>namesctl show [onames_server] ... version</pre> NAMESCTL プログラムからは、次のように入力します。 <pre>namesctl> show [onames_server] ... version</pre>
引数 :	Oracle Names Server 名を空白で区切って指定します。名前を指定しないと、現ネーム・サーバーの設定が表示されます。
使用上の注意 :	このコマンドで、サーバーの名前とバージョンを確認します。この情報は、小さな問題を解決するときに役立ちます。NAMESCTL をサーバーに接続するたびに、このコマンドが使用可能になります。
例 :	<pre>namesctl> show version Currently managing Oracle Names server "NameServer.com" Version banner is "Oracle Names for SunOS: Version 8.1.5.0.0"</pre>

SHUTDOWN

目的 :	1 つ以上の Oracle Names Server を停止します。
前提条件 :	Oracle Names Server が起動している必要があります。
パスワード :	必要 パスワードが設定されている場合、このコマンドの前に SET PASSWORD コマンドを実行する必要があります。
構文 :	オペレーティング・システムからは、次のように入力します。 <code>namesctl shutdown [onames_server] ...</code> NAMESCTL プログラムからは、次のように入力します。 <code>namesctl> shutdown [onames_server] ...</code>
引数 :	Oracle Names Server 名を空白で区切って指定します。引数を指定しないと、現行の Oracle Names Server のみが停止します。
使用上の注意 :	SHUTDOWN は、現行の Oracle Names Server を停止して、メモリーからプログラムを削除します。Oracle Names Server を停止する必要があるのは、アップグレードやマシン保守などの運用上の理由がある場合のみです。Oracle Names Server を停止および起動するときは、ネットワーク内の任意の場所から実行できる RESTART コマンドを使用するようお勧めします。SHUTDOWN と START を個々に実行する場合は、Oracle Names Server マシン上で実行する必要があります。 SHUTDOWN は STOP と同じです。
例 :	<code>namesctl> shutdown</code> Confirm [yes or no] yes Server shut down.

START

目的 :	Oracle Names サービス・プログラムをロードして、システム・データとローカル管理リージョン・データのロードを開始します。
前提条件 :	Oracle Names Server が停止している必要があります。
パスワード :	不要
構文 :	オペレーティング・システムからは、次のように入力します。 <pre>namesctl start</pre> NAMESCTL プログラムからは、次のように入力します。 <pre>namesctl> start</pre>
引数 :	なし
使用上の注意 :	<p>START は、Oracle Names Server を初めてメモリーにロードするときに使用するコマンドです。Oracle Names Server は起動時に、操作パラメータ設定のための構成ファイルを読み込み、管理リージョンのデータをすべてロードします。</p> <p>Oracle Names Server 起動時のセキュリティは、Oracle Names がインストールされているオペレーティング・システムによって保証されます。Oracle Names Server はローカル・セッションから起動する必要がありますので、ネットワーク・セキュリティが損なわれることはありません。</p> <p>START は STARTUP と同じです。</p>

START

```
例 : namesctl> start
Starting "/private/dsteiner/sales/bin/names"...server
successfully started

Currently managing name server "namesrv1.us.oracle.com"
Version banner is "Oracle Names for Solaris: Version
8.1.5.0.0"

Server name:
namesrv1.us.oracle.com
Server has been running for:                0.16 seconds
Request processing enabled:                  yes
Request forwarding enabled:                  yes
Requests received:                           0
Requests forwarded:                          0
Foreign data items cached:                   0
Region data next checked for reload in:      not set
Region data reload check failures:           0
Cache next checkpointed in:                  not set
Cache checkpoint interval:                   not set
Cache checkpoint file name:
/private/dsteiner/sales/network/names/ckpcch.ora
Statistic counters next reset in:            not set
Statistic counter reset interval:            not set
Statistic counters next logged in:           not set
Statistic counter logging interval:          not set
Trace level:                                 0
Trace file name:
/private/dsteiner/sales/network/trace/names_10841.trc
Log file name:
/private/dsteiner/sales/network/log/names.log
System parameter file name:
/private/dsteiner/sales/network/admin/names.ora
Command-line parameter file name:           ""
Administrative region name:                  ""
Administrative region description:           ""
ApplTable Index:                            0
Contact:                                     ""
Operational Status                           0
Save Config on Stop                          yes
```

START_CLIENT_CACHE

目的:	クライアント・キャッシュ・デーモン・プロセスを起動します。
前提条件:	クライアント・キャッシュ・デーモン・プロセスが停止している必要があります。 クライアント・キャッシュ・デーモン・プロセスを実行する前に、Oracle Names Server リストが必要です。
パスワード:	不要
構文:	オペレーティング・システムからは、次のように入力します。 <code>namesctl start_client_cache</code> NAMESCTL プログラムからは、次のように入力します。 <code>namesctl> start_client_cache</code>
引数:	なし
使用上の注意:	起動すると、クライアント・キャッシュ・デーモン・プロセスは、Oracle Names Server から受信したすべての情報を格納し、検索を高速に行えるようにします。
例:	<code>namesctl> start_client_cache</code>

STARTUP

目的:	このコマンドは START と同じです。
前提条件:	Oracle Names Server が停止している必要があります。
パスワード:	不要
構文:	オペレーティング・システムからは、次のように入力します。 <code>namesctl startup names.parameter = value</code> NAMESCTL プログラムからは、次のように入力します。 <code>namesctl> start names.parameter = value</code>
引数:	なし
使用上の注意:	STARTUP は START と同じです。
例:	START の例を参照してください。

STATUS

目的:	1 つ以上の Oracle Names Server の統計と内部設定の多くを表示します。
前提条件:	Oracle Names Server が起動している必要があります。
パスワード:	不要
構文:	オペレーティング・システムからは、次のように入力します。 <code>namesctl status [onames_server] ...</code> NAMESCTL プログラムから、次のように入力します。 <code>namesctl> status [onames_server] ...</code>
引数:	Oracle Names Server 名を空白で区切って指定します。引数を指定しないと、現行の Oracle Names Server の状態のみが表示されます。
使用上の注意:	STATUS は、Oracle Names Server 稼働中のアクティビティ、および一時点における Oracle Names Server の状態を表示します。
例:	<pre>namesctl> status Version banner is "Oracle Names for SunOS: 8.1.5.0.0" Server name:NSEVER.com Server has been running for:1 day 20 hours</pre>

STOP

目的 :	1 つ以上の Oracle Names Server を停止します。
前提条件 :	Oracle Names Server が起動している必要があります。
パスワード :	必要 パスワードが設定されている場合、このコマンドの前に SET PASSWORD コマンドを実行する必要があります。
構文 :	オペレーティング・システムからは、次のように入力します。 <code>namesctl stop [onames_server] ...</code> NAMESCTL プログラムからは、次のように入力します。 <code>namesctl> stop [onames_server] ...</code>
引数 :	Oracle Names Server 名を空白で区切って指定します。引数を指定しないと、現行の Oracle Names Server のみが停止します。
使用上の注意 :	STOP は、現行の Oracle Names Server を停止して、メモリーからプログラムを削除します。Oracle Names Server を停止する必要があるのは、アップグレードやマシン保守などの運用上の理由がある場合のみです。Oracle Names Server を停止および起動するときは、ネットワーク内の任意の場所から実行できる RESTART コマンドを使用するようお勧めします。STOP と START を個々に実行する場合は、Oracle Names Server マシン上で実行する必要があります。 STOP は SHUTDOWN と同じです。
例 :	<code>namesctl> stop</code> <code>Confirm [yes or no]: yes</code> <code>Server shut down</code>

TIMED_QUERY

目的:	Oracle Names Server のキャッシュにあるすべてのデータを表示します。
パスワード:	不要
構文:	オペレーティング・システムからは、次のように入力します。 <code>namesctl timed_query</code> NAMESCTL プログラムからは、次のように入力します。 <code>namesctl> timed_query [time]</code>
使用上の注意:	TIMED_QUERY は、引数 [<i>time</i>] で表される時刻以降のデータをすべて返します。 <i>time</i> 引数で、所定の時刻以降に登録されたすべてのオブジェクトが戻されます。 <i>time</i> 引数を使用するには、起動後に登録されたすべての情報を最初の TIMED_QUERY でダンプします。このダンプが終了すると、最新ダンプ情報の終了時点を示す最終タイムスタンプが表示されます。その時点以降に記録されたすべてのデータを表示するには、 <i>time</i> 引数で最終タイムスタンプ番号を指定します。

UNREGISTER

目的 :	ネットワーク・オブジェクトを Oracle Names Server から削除します。
前提条件 :	なし
パスワード :	不要
構文 :	オペレーティング・システムからは、次のように入力します。 <pre>namesctl unregister <i>object_name</i> -d description=(address_list= (address= (<i>protocol_address_</i> <i>information</i>)))) [-h <i>hostname</i>] [-l <i>listener_name</i>]</pre> NAMESCTL プログラムからは、次のように入力します。 <pre>namesctl> unregister <i>object_name</i> (-d <i>address_data</i>) [-h <i>hostname</i>] [-l <i>listener_name</i>]</pre>
引数 :	オブジェクト名 (必須) およびネーム・サーバーの登録されているオブジェクトのアドレスまたはリスナー名、ホスト名
使用上の注意 :	登録されているサービスを手動で抹消するためのコマンドです。そのオブジェクトの定義が、リージョン内の Oracle Names Server から削除されます。オブジェクトが、アドレスまたはリスナー名、ホスト名で登録されている場合は、アドレスまたはリスナー名、ホスト名をコマンド行で指定しないと、オブジェクトを抹消できません。
例 :	<pre>namesctl> unregister parts -t oracle_database -d (description= (address= (protocol=tcp)(host=nineva)(port=1575)) (connect_ data=(service_name=db3)))</pre>

VERSION

目的:	Oracle Names Server の現在のバージョンと名前を表示します。
前提条件:	なし
パスワード:	不要
構文:	オペレーティング・システムからは、次のように入力します。 <code>namesctl version</code> NAMESCTL プログラムからは、次のように入力します。 <code>namesctl> version</code>
引数:	Oracle Names Server 名を空白で区切って指定します。名前を指定しないと、現ネーム・サーバーの設定が表示されます。
使用上の注意:	このコマンドで、サーバーの名前とバージョンを確認します。この情報は、小さな問題を解決するときに役立ちます。NAMESCTL をサーバーに接続するたびに、このコマンドが使用可能になります。
例:	<code>namesctl> version</code> Currently managing Oracle Names server "NameServer.com" Version banner is "Oracle Names for SunOS: Version 8.1.5.0.0"

Oracle Connection Manager 制御ユーティリティ (CMCTL)

Oracle Connection Manager 制御ユーティリティ (CMCTL) は、Oracle Connection Manager を起動し、制御するためにオペレーティング・システム・プロンプトから実行するツールです。Oracle Connection Manager 制御ユーティリティの一般的な書式は、次のとおりです。

```
CMCTL command [process_type]
```

ここで、*process_type* はコマンドを実行するときのプロセスのタイプです。これらを次に示します。

- `cman` (CMGW ゲートウェイおよび CMADMIN 管理プロセスの両方)
- `adm` (CMADMIN プロセスのみ)
- `cm` (CMGW プロセスのみ)

たとえば、管理プロセスとメイン・プロセスの両方を起動する場合は、次のように入力します。

オペレーティング・システムからは、次のように入力します。

```
cmctl start cman
```

CMCTL プログラムからは、次のように入力します。

```
cmctl> start cman
```

CMCTL コマンド

次のコマンドは、Oracle Connection Manager 制御ユーティリティ (CMCTL) で使用できません。

EXIT

目的: CMCTL ユーティリティ・プログラムを終了します。

前提条件: なし

例: `cmctl> exit`

START

目的: Oracle Connection Manager を起動します。

前提条件: Oracle Connection Manager が動作してはなりません。

引数: `cman` - CMGW と CMADMIN プロセスの両方を起動する (デフォルト)
`cm` - CMGW プロセスのみを起動する
`adm` - CMADMIN プロセスのみを起動する

例: `cmctl> start cm`

STATS

- 目的： リレー件数合計、アクティブなリレー件数、合計のほとんどを占めるリレー件数、拒否されたリレー件数など、基本的な統計情報を表示します。
- 前提条件： なし
- 引数： cman - CMGW と CMADMIN プロセスの両方の統計を表示する (デフォルト)
cm - CMGW プロセスの統計を表示する
adm - CMADMIN プロセスの統計を表示する
- 例：
cmctl> stats
(statistics=(total_relays=8)(active_delays=4)(most_relays=6)(out_of_relay=5)(total_refused=5)
- TOTAL_RELAYS: Oracle Connection Manager が、起動後に確立した接続の総数
 - ACTIVE_DELAYS: 現在アクティブな接続の数
 - MOST_RELAYS: Oracle Connection Manager の起動後に、同時に確立された接続の最大数
 - OUT_OF_RELAY: Oracle Connection Manager の起動後に、リレーの範囲外により拒否された接続要求の総数
 - TOTAL_REFUSED: Oracle Connection Manager の起動後に、拒否された接続要求の総数

STATUS

目的:	バージョン、起動時刻、アップタイムなどの、基本的な情報を表示します。
前提条件:	なし
引数:	cman - CMGW と CMADMIN プロセスの両方のステータス情報を表示する (デフォルト) cm - CMGW プロセスのステータス情報を表示する adm - CMADMIN プロセスのステータス情報を表示する
例:	<pre>cmctl> status CMAN status: status=(version=8.1.5.0.0)(started=01-DEC-98 11:41:33)(state-running) ADMIN status: status=(version=8.1.5.0.0)(started=01-DEC-98 11:41:29)(state-running)</pre>

STOP

目的:	Oracle Connection Manager プロセスを、対話モードで停止します。
前提条件:	Oracle Connection Manager が動作している必要があります。
引数:	cman - CMGW と CMADMIN プロセスの両方を停止する (デフォルト) cm - CMGW プロセスのみを停止する adm - CMADMIN プロセスのみを停止する
使用上の注意:	接続がアクティブな状態のままに停止コマンドを発行すると、停止の確認を求められます。
例:	<pre>cmctl> stop cm</pre>

STOP NOW

目的 :	すべての Oracle Connection Manager プロセスを停止します。プロセスが実行中の場合、終了を確認するプロンプトは表示されません。
前提条件 :	Oracle Connection Manager が動作している必要があります。
引数 :	なし
例 :	<code>cmctl> stop now</code>

構成ファイルのサンプル

ここでは、Net8 で使用する構成ファイルのサンプルを紹介します。内容は次のとおりです。

- プロファイル (SQLNET.ORA)
- ローカル名構成ファイル (TNSNAMES.ORA)
- リスナー構成ファイル (LISTENER.ORA)
- Oracle Names Server 構成ファイル (NAMES.ORA)
- Oracle Connection Manager 構成ファイル (CMAN.ORA)

プロファイル (SQLNET.ORA)

プロファイル (SQLNET.ORA) には、クライアントまたはサーバーが Net8 の機能を使用する方法についての作業環境を指定するパラメータが含まれています。SQLNET.ORA は、UNIX では \$ORACLE_HOME/network/admin、Windows では ORACLE_HOME¥network¥admin に置かれます。各パラメータの詳細は、C-7 ページの「[プロファイル・パラメータ \(SQLNET.ORA\)](#)」を参照してください。

```
names.default_domain = com
names.initial_retry_timeout = 30
names.max_open_connections = 3
names.message_pool_start_size = 10
names.preferred_servers = (address_list =
    (address=(protocol=ipc)(key=n23))
    (address=(protocol=tcp)(host=nineva)(port=1383))
    (address=(protocol=tcp)(host=cicada)(port=1575))
)
names.request_retries = 2
names.directory_path = (tnsnames, onames, hostname)
namesctl.trace_directory = /oracle/network/trace
namesctl.trace_file = namesctl.trc
namesctl.trace_level = admin
namesctl.trace_unique = true
namesctl.no_initial_server = false
namesctl.internal_use = true
namesctl.noconfirm = true
namesctl.server_password = mangler
namesctl.internal_encrypt_password = false
names.dce.prefix = ./:/subsys/oracle/names
names.nds.name_context = personnel.acme
names.nis.meta_map=sqlnet.maps
sqlnet.authentication_services=(none)
sqlnet.authentication_services=(beq, oss)
sqlnet.kerberos5_cc_name=/tmp/mycc
sqlnet.kerberos5_clockskew=600
sqlnet.kerberos5_conf=/tmp/mykrb.conf
sqlnet.kerberos5_realms=/tmp/mykrb.realms
sqlnet.kerberos5_keytab=/tmp/myv5srvtab
sqlnet.authentication_kerberos5_service=acme
sqlnet.authentication_gssapi_service=acme/asriniva.us.oracle.com@us.oracle.com
sqlnet.identix_fingerprint_method=oracle
sqlnet.identix_fingerprint_database=ofm
sqlnet.identix_fingerprint_database_user=<username>
sqlnet.identix_fingerprint_database_password=<password>
sqlnet.authentication_gssapi_service=acme/scott.us.oracle.com@us.oracle.com
oss.source.my_wallet
    =(source
```

```
        =(method=file)
          (method_data=/dve/asriniva/oss/wallet)
        )
oss.source.encrypted_private_key
  =(source
    =(method=oracle)
      (method_data=
        (username=andre_security_service)
        (password=andre_security_service)
        (sqlnet_address=andreoss)
      )
    )
  )
oss.source.certificates
  =(source
    =(method=oracle)
      (method_data=
        (username=scott_security_service)
        (password=ascott_security_service)
        (sqlnet_address=andreoss)
      )
    )
  )
oss.source.attributes
  =(source
    =(method=oracle)
      (method_data=
        (username=scott_oracle_security_service)
        (password=scott_oracle_security_service)
        (sqlnet_address=andreoss)
      )
    )
  )
sqlnet.crypto_checksum_client = required
sqlnet.encryption_client = required
sqlnet.crypto_checksum_types_client = required
sqlnet.crypto_checksum_types_server = required
sqlnet.encryption_types_client = required
sqlnet.encryption_types_server = required
sqlnet.crypto_seed = "4fhfguweotcadsfdsafjkdsfq5f201p45mxskdlfdasf"
sqlnet.crypto_checksum_server = required
sqlnet.encryption_server = required
trace_level_client = admin
trace_directory_client = /oracle/network/trace
trace_file_client = /oracle/network/trace/cli.trc
trace_unique_client = on
log_directory_client = /oracle/network/log
log_file_client = /oracle/network/log/sqlnet.log
log_directory_server = /oracle/network/trace
trace_directory_server = /oracle/network/trace
```

```
trace_file_server = /orace/network/trace/svr_<pid>.trc
trace_level_server = admin
use_dedicated_server = on
use_cman = true
tnsping.trace_directory = /oracle/network/trace
tnsping.trace_level = admin
sqlnet.expire_time = 10
sqlnet.client_registration = <unique_id>
bequeath_detach = yes
automatic_ipc = off
disable_oob = on
```

ローカル名構成ファイル (TNSNAMES.ORA)

ローカル名構成ファイル (TNSNAMES.ORA) には、サービスの別名が含まれます。データベースのサービスの別名は、ネットワーク上でデータベースの位置を示す接続記述子にマップされます。TNSNAMES.ORA は、クライアントと分散データベース・サーバーが接続先とサーバーを識別するときに使用されます。TNSNAMES.ORA は、UNIX では \$ORACLE_HOME/network/admin、Windows では `ORACLE_HOME\network\admin` に置かれます。各パラメータの詳細は、C-31 ページの「[ローカル・ネーミング・パラメータ \(TNSNAMES.ORA \)](#)」を参照してください。

```
tcpnew1 = (description=
            (source_route=yes)
            (address=(protocol=tcp)(port=1610)(host=spcstn))
            (address=(protocol=tcp)(port=1580)(host=spcstn)))
        )

spx2tcp = (description=
            (source_route=yes)
            (address=(protocol=spx)(service=orasrvcl))
            (address=(protocol=tcp)(port=1580)(host=spcstn)))
        (connect_data=(service_name=sales2))
    )
```

より複雑なサンプルは、C-31 ページの「[ローカル・ネーミング・パラメータ \(TNSNAMES.ORA \)](#)」を参照してください。

リスナー構成ファイル (LISTENER.ORA)

リスナー構成ファイル (LISTENER.ORA) には、ネットワーク・リスナーの動作について作業環境を指定するパラメータが含まれています。LISTENER.ORA は、UNIX では \$ORACLE_HOME/network/admin、Windows NT では ORACLE_HOME¥network¥admin に置かれます。各パラメータの詳細は、C-50 ページの「リスナー・パラメータ (LISTENER.ORA)」を参照してください。

```
listener=
  (address= # default tcp listening address
    (protocol=tcp)
    (port=1521)
    (host=mudshark)
  )
  (address= # non-default ipc listening address
    (protocol=ipc)
    (key=salesdb)
  )
)
sid_list_listener_name=(sid_list=
  (sid_desc=
    (sid_name=sales)
    (global_dbname=salesdb.mycompany)
    (oracle_home=/private1/app/oracle/product/8.0.3)
    (prespawn_max=20)
    (prespawn_list=
      (prespawn_desc=(protocol=tcp)(pool_size=2)(timeout=5))
      (prespawn_desc=(protocol=ipc)(pool_size=3)(timeout=2))
    )
  )
)
trace_level_listener=admin
trace_directory_listener=/private1/app/oracle/product/8.0.2/network/trace
trace_file_listener=listener
logging_listener=on
log_directory_listener=/private1/app/oracle/product/8.0.2/network/log
log_file_listener=listener
save_config_on_stop_listener=true
startup_wait_time_listener=0
```

注意： SID_LIST_listener_name セクションは、バージョン 8 またはバージョン 7 のデータベースに接続するときのみ必要です。

Oracle Names Server 構成ファイル (NAMES.ORA)

Oracle Names Server 構成ファイル (NAMES.ORA) には、各 Oracle Names Server の場所、ドメイン情報およびオプション構成パラメータを指定するパラメータが含まれています。NAMES.ORA は、UNIX では \$ORACLE_HOME/network/admin、Windows NT では *ORACLE_HOME*\network\admin に置かれます。各パラメータの詳細は、C-65 ページの「Oracle Names パラメータ (NAMES.ORA)」を参照してください。

```
names.addresses = (address=(protocol=tcp)(host=oranamesrvr0)(port=1575))
names.server_name = onsl.com
names.domains = (domain_list=
                  (domain=
                    (name=com)
                    (min_ttl=86400)
                  )
                  (domain=
                    (name=org)
                    (min_ttl=86400)
                  )
                )
names.admin_region= (region=
                    (name= local_region.com)
                    (type= rosdB)
                    (userid = names)
                    (password = names)
                    (description =
                      (address = (protocol = tcp)(host = nineva)(port = 1387))
                      (connect_data = (service = em))
                    )
                    (docname = sbox)
                    (version = 34619392) # 2.1.4
                    (refresh = 14400)
                    (retry = 600)
                    (expire = 259200)
                  )
names.authority_required = false
names.auto_refresh_expire = 259200
names.auto_refresh_retry = 180
names.cache_checkpoint_file = cache.chkp
names.cache_checkpoint_interval = 7200
names.config_checkpoint_file = config.chkp
names.default_forwarders=
  (forwarder_list=
    (forwarder=
      (name= rootserv1.com)
      (address=(protocol=tcp)(port=42100)(host=roothost))))
names.default_forwarders_only = true
```

```
names.domain_hints =
  (hint_desc=
    (hint_list=
      (hint=(name=rootserv1.com)
        (address=(protocol=tcp)(host=nineva)(port=42100))))))
names.forwarding_available = true
names.forwarding_desired = true
names.log_directory = /oracle/network/log
names.log_file = names.log
names.log_stats_interval = 3600
names.log_unique = false
names.max_open_connections = 10
names.max_reforwards = 2
names.message_pool_start_size = 24
names.no_modify_requests = false
names.no_region_database = false
names.password = 625926683431aa55
names.reset_stats_interval = 3600
names.region_checkpoint_file = reg.ckp
names.trace_directory = /oracle/network/trace
names.trace_file = names.trc
names.trace_level = admin
names.trace_unique = true
```

Oracle Connection Manager 構成ファイル (CMAN.ORA)

Connection Manager 構成ファイル (CMAN.ORA) には、Oracle Connection Manager を使用するための作業環境を指定するパラメータが含まれています。CMAN.ORA は、UNIX では \$ORACLE_HOME/network/admin、Windows NT では *ORACLE_HOME*\network\admin に置かれます。各パラメータの詳細は、C-77 ページの「[Oracle Connection Manager パラメータ \(CMAN.ORA\)](#)」を参照してください。

```
cmn = (address = (protocol=tcp)(host=anyhost)(port=1630))
cmn_admin = (address=(protocol=tcp)(host=anyhost)(port=1830))

cmn_profile = (parameter_list=
    (maximum_relays=512)
    (log_level=1)
    (tracing=yes)
    (trace_directory=/oracle/network/trace)
    (relay_statistics=yes)
    (show_tns_info=yes)
    (use_async_call=yes)
    (authentication_level=0)
)
# the following specifies a rule for single access control #
cmn_rules = (rule_list=
    (rule=(src=spcstn)(dst=x)(srv=x)(act=accept))
)

```

構成パラメータ

すべての Net8 構成パラメータを網羅したリストを参考として示します。内容は次のとおりです。

- 構成ファイルの構文規則
- ADDRESS と ADDRESS_LIST
- プロファイル・パラメータ (SQLNET.ORA)
- ローカル・ネーミング・パラメータ (TNSNAMES.ORA)
- リスナー・パラメータ (LISTENER.ORA)
- Oracle Names パラメータ (NAMES.ORA)
- Oracle Connection Manager パラメータ (CMAN.ORA)
- プロトコル固有のパラメータ (PROTOCOL.ORA)

構成ファイルの構文規則

Net8 ネットワークの構成ファイルは、キーワードと値が対となるパラメータの集まりです。キーワードと値の対をカッコで囲みます。

```
parameter=(keyword=value)
```

一部のキーワードは、他のキーワードと値の対をそのキーワードの値とします。

```
(keyword=  
  (keyword=value)  
  (keyword=value)  
)
```

たとえば、ローカル名構成ファイル (TNSNAMES.ORA) のアドレス部に、次の行を含めることができます。

```
(address=  
  (protocol=tcp)  
  (host=max)  
  (port=1521)  
)
```

どのキーワードが、他のキーワードと値の対の親または所有者であるかをインデントで反映するように構成ファイルを設定します。インデントの使用は義務付けられていませんが、このフォーマットによって読みやすく理解しやすいファイルになります。

このようにファイルをインデントしない場合でも、行を折り返すときは空白を1つ以上入れないと、折り返し行が新しいパラメータとして解釈されてしまいます。次のレイアウトは受け入れられます。

```
(address=(protocol=tcp)  
  (host=max)(port=1521))
```

次のレイアウトは受け入れられません。

```
(address=(protocol=tcp)  
(host=max)(port=1521))
```

構成ファイルの構文規則の詳細

次の規則が構成ファイルの構文に適用されます。

- キーワードと値の対を1つ以上含むパラメータで始まる構成ファイルのキーワードは、一番左の列にある必要がある。キーワードの前に1つ以上の空白がある場合は、前の行の継続行として解釈されます。
- すべての文字は、ネットワーク・キャラクタ・セットに属している必要がある（次の項を参照）。
- キーワードでは大/小文字が区別されない。値は、オペレーティング・システムおよびプロトコルによって、大/小文字を区別する場合と区別しない場合があります。
- キーワードと値の対の中で、"="の前後の空白はオプションとする。
- キーワードの階層がある場合、キーワードの後ろには必ず他のキーワードが続く。階層のどのレベルでも、キーワードを任意の順序で入力できます。たとえば、次の構文はどちらも有効です。

```
(address =  
  (protocol=tcp)  
  (host=martha)  
  (port=1521)  
)
```

```
(address =  
  (protocol=tcp)  
  (port=1521)  
  (host=martha)  
)
```

- キーワードに空白を含めることはできない。値に空白を含めるときは、二重引用符(")で囲む必要があります。
- 接続記述子の最大長は4KB。
- コメントを入力するときは、行の先頭でシャープ記号(#)を使用する。シャープ記号の後から行の終わりまでが、コメントとみなされます。
- キーワードと値の対で等号の前または後に1つの語または連結語を入力する場合は、カッコが不要となる。

ネットワーク・キャラクタ・セット

キーワードと値の対で使用できるネットワーク・キャラクタ・セットには、次の文字があります。接続記述子は1バイト文字にしてください。

A-Z、a-z

0-9

()<>/¥

. . ; ' " = - _

\$ + * # & ! % ? @

このキャラクタ・セットの中では、次の記号が予約済みです。

()=¥" '#

キーワードまたは値を引用符で囲まない場合は、予約記号をキーワードまたは値の一部としてではなくデリミタとして使用します。単一引用符または二重引用符を使用して、予約記号を含む値を囲むことができます。引用符で囲んだ値の中に引用符を含めるには、別のタイプの引用符を使用します。円記号(¥)をエスケープ文字として使用します。

予約記号の使用例として、アドレス内の数値 DECnet オブジェクトがあります。OBJECT を ABC や #123 などの名前にすることができます。これらは次の形式で入力します。

(OBJECT=ABC)

または

(OBJECT=¥#123)

符号 "#" は予約されている記号なので、文字の前に円記号を付ける必要があります。

次の文字は、接続記述子の中では使用できますが、キーワードまたは値の中では使用できません。

<space> <tab> <CR> <newline>

キャラクタ・セット

リスナー名、ネット・サービス名および Oracle Connection Manager 名は、次のキャラクタ・セットに制限されています。

[a...z] [A...Z] [0...9] _

最初の文字は、アルファベットの文字にしてください。一般的に、最大 64 文字が受け付けられます。データベース・サービス名はデータベース管理者が定義したグローバル・データベース名と一致する必要があります。グローバル・データベース名は、データベース名（最初は 8 文字に制限されている）とデータベース・ドメインで構成されています。サービス名とグローバル・データベース名は、大 / 小文字を区別しません。

ADDRESS と ADDRESS_LIST

接続時に、リスナー、Oracle Names Server、Oracle Connection Manager などの要求の受信側とクライアントは、同じプロトコル・アドレスで構成されます。アドレスの目的は、ネットワーク・オブジェクトのネットワーク・アドレスを識別することです。クライアントはこのアドレスを使用して、接続要求を特定のネットワーク・オブジェクトの場所へ送信します。受信側はこのアドレスで要求をリスニングし、クライアント情報に対応するアドレス情報に基づいて、接続を許可します。クライアントと接続の受信側に同じプロトコルをインストールし、同じアドレスを構成することが重要です。

プロトコル・アドレスは、2 つの要素で構成されます。

ADDRESS

目的: プロトコル・アドレスを定義します。このパラメータは、ADDRESS_LIST または DESCRIPTION の下に埋め込まれる場合があります。DESCRIPTION は、一般的に TNSNAMES.ORA または LISTENER.ORA ファイルのみで使用されます。

追加情報: 6-76 ページの「[プロトコル・アドレスの構成](#)」で説明するように、各プロトコルには、アドレスに必要な独自の必須パラメータがあります。

例:

```
(address=
(protocol=tcp)
(host=sales-pc)
(port=1521)
)
```

ADDRESS_LIST

目的： 共通の特性を共有するアドレスのリストを定義します。
このパラメータは、DESCRIPTION または DESCRIPTION_LIST の下に埋め込まれる場合があります。DESCRIPTION または DESCRIPTION_LIST は、一般的に TNSNAMES.ORA または LISTENER.ORA ファイルのみで使用されます。

注意： TNSNAMES.ORA または LISTENER.ORA ファイルにあるリストが 1 つのみの場合、ADDRESS_LIST は必要ありません。

例：

```
(address_list=  
  (address=  
    (protocol=tcp)  
    (host=sales-pc)  
    (port=1521)  
  )  
  (address=  
    (protocol=tcp)  
    (host=hr-pc)  
    (port=1521)  
  )  
)  
(address_list=  
  (address=  
    (protocol=tcp)  
    (host=finance-pc)  
    (port=1521)  
  )  
  (address=  
    (protocol=tcp)  
    (host=sales-pc)  
    (port=1521)  
  )  
)  
)
```

プロファイル・パラメータ (SQLNET.ORA)

プロファイルでは、次のパラメータが使用できます。プロファイルは、構成ファイルの SQLNET.ORA に保存されます。SQLNET.ORA は UNIX の場合は \$ORACLE_HOME/network/admin、Windows の場合は ORACLE_HOME¥network¥admin に置かれます。

BEQUEATH_DETACH

目的: UNIX システムの Net8 内のシグナル処理をオン / オフに切り替えます。

デフォルト: NO で、シグナル処理をオンにします。

値:

- YES - シグナル処理をオフする
- NO - シグナル処理をオンにする

例: bequeath_detach=yes

DAEMON.TRACE_DIRECTORY

目的: Oracle Enterprise Manager のデーモン・トレース・ファイルの接続先ディレクトリを制御します。

デフォルト: UNIX プラットフォームでは \$ORACLE_HOME/network/trace、Windows プラットフォームでは ORACLE_HOME¥network¥trace

例: daemon.trace_directory=/oracle/traces

DAEMON.TRACE_LEVEL

目的: Oracle Enterprise Manager デーモンのトレースをオフにするか、特定のレベルでオンにします。

デフォルト: OFF

値:

- OFF - トレース出力しない
- USER - ユーザー・トレース情報
- ADMIN - 管理トレース情報
- SUPPORT - オラクル社サポート・サービス・トレース情報

例: daemon.trace_level=user

DAEMON.TRACE_MASK

目的:	Oracle Enterprise Manager デモンのトレース・エントリのみをトレース・ファイルに記録することを指定します。
デフォルト:	UNIX プラットフォームでは \$ORACLE_HOME/network/trace、Windows プラットフォームでは <i>ORACLE_HOME</i> \network\trace
例:	daemon.trace_mask=(106)

DISABLE_OOB

目的:	バンド外ブ레이크を使用禁止にします。
デフォルト:	OFF
使用上の注意:	基礎を形成するトランスポート・プロトコルが機能をサポートしない場合に、バンド外ブ레이크を使用禁止にします。
例:	disable_oob=off

LOG_DIRECTORY_CLIENT

目的:	ログ・ファイルが書き込まれるディレクトリを制御します。
デフォルト:	実行可能ファイルの起動元である現ディレクトリ
例:	log_directory_client=/oracle/network/log

LOG_DIRECTORY_SERVER

目的:	ログ・ファイルが書き込まれるディレクトリを制御します。
デフォルト:	実行可能ファイルの起動元である現ディレクトリ
例:	log_directory_server=/oracle/network/log

LOG_FILE_CLIENT

目的: Oracle クライアントのログ出力ファイル名を制御します。

デフォルト: SQLNET.LOG

例: log_file_client=client

LOG_FILE_SERVER

目的: Oracle サーバーのログ出力ファイル名を制御します。

デフォルト: SQLNET.LOG

例: log_file_server=svr.log

NAMES.DCE.PREFIX

目的: 名前の検索に使用する DCE セル名 (接頭辞) を指定します。

デフォルト: /./subsys/oracle/names

例: names.dce.prefix=/./subsys/oracle/names

NAMES.DEFAULT_DOMAIN

目的: クライアントが最も頻繁に名前を要求するドメインを指定します。このパラメータを設定すると、デフォルトのドメイン名が未修飾のサービス名に自動的に付けられます。エスケープされないドット (.) を含む名前には、デフォルトのドメインが付加されません。単純な名前は、後続するドットで修飾できます (たとえば、'rootserver.' など)。

デフォルト: NULL

例: names.default_domain=com

NAMES.DIRECTORY_PATH

目的: クライアント名の要求に使用されるネーミング・メソッド (TNSNAMES、Oracle Names など) の順序を指定します。

デフォルト: TNSNAMES, ONAMES, HOSTNAME

- 値:
- TNSNAMES
 - ONAMES
 - HOSTNAME
 - DCE
 - NIS
 - NOVELL

例: `names.directory_path=(tnsnames, onames)`

NAMES.INITIAL_RETRY_TIMEOUT

目的: クライアントが Preferred Servers List 内の次のサーバーに対する要求を送り直すまでに、Oracle Names Server からの応答を待つ時間を決定します。

デフォルト: 15

最小値: 1

最大値: 600

例: `names.initial_retry_timeout=20`

NAMES.MAX_OPEN_CONNECTIONS

目的:	Oracle Names クライアントが一度にオープンできる接続数を決定します。
デフォルト:	10
最小値:	3
最大値:	64
例:	<code>names.max_open_connections=3</code>

NAMES.MESSAGE_POOL_START_SIZE

目的:	転送されたメッセージ要求に対して使用するメッセージの初期数を、クライアントのメッセージ・プールに対して指定します。
デフォルト:	10
最小値:	3
最大値:	256
例:	<code>names.message_pool_start_size=10</code>

NAMES.NIS.META_MAP

目的:	NIS 属性を NIS マップ名にマップするときに使用するファイルを指定します。
デフォルト:	SQLNET.MAPS
例:	<code>names.nis.meta_map=sqlnet.maps</code>

NAMES.PREFERRED_SERVERS

目的:	クライアント名の要求に使用する Oracle Names Server の名前、アドレスおよび順序を指定します。
デフォルト:	なし
例:	<code>names.preferred_servers= (address_list= (address=(protocol=ipc)(key=n23)) (address=(protocol=tcp)(host=nineva)(key=1575)) (address=(protocol=tcp)(host=cicada)(key=1575)))</code>

NAMES.REQUEST_RETRIES

目的:	操作が失敗になる前に、クライアントが Preferred Server List 内の各サーバーを試行する回数を指定します。
デフォルト:	1
最小値:	1
最大値:	5
例:	<code>names.request_retries=5</code>

NAMESCTL.INTERNAL_ENCRYPT_PASSWORD

目的:	TRUE に設定すると、NAMESCTL はパスワードを暗号化して Oracle Names Server に送ります。 FALSE に設定すると、NAMESCTL はパスワードを暗号化せずに Oracle Names Server に送ります。これによって、非暗号化パスワードを NAMES.ORA ファイルの NAMES.PASSWORD パラメータで設定できます。
デフォルト:	FALSE
値:	TRUE FALSE
例:	<code>namesctl.internal_encrypt_password=true</code>

NAMESCTL.INTERNAL_USE

目的:	TRUE に設定すると、文書化されていない内部コマンドを NAMESCTL で使用できます。内部コマンドを識別できるように、すべての内部コマンドの先頭にアンダースコアが付いています。
-----	---

NAMESCTL.NO_INITIAL_SERVER

目的:	TRUE に設定すると、NAMESCTL はデフォルトの Oracle Names Server に接続できないときにエラー・メッセージを表示しません。
デフォルト:	FALSE
値:	TRUE FALSE
例:	<code>namesctl.no_initial_server=true</code>

NAMESCTL.NOCONFIRM

目的:	NAMESCTL コーティリティの実行中に確認をともなう重要なコマンド (STOP、RELOAD、RESTART) をプロンプトするかどうかを指定します。
デフォルト:	OFF
値:	ON OFF
例:	<code>namesctl.noconfirm=on</code>

NAMESCTL.SERVER_PASSWORD

目的: NAMES.ORA ファイルの **NAMES.PASSWORD** パラメータで設定した構成済みのパスワードに一致する値を指定します。これにより、NAMESCTL ユーティリティで STOP、RESTART、RELOAD などの保護オプションを実行するたびに、**SET PASSWORD** コマンドでパスワードを入力する必要がなくなります。

例: `namesctl.server_password=secret`

NAMESCTL.TRACE_LEVEL

目的: NAMESCTL プログラムのトレース・レベルを指定します。

デフォルト: OFF

値: OFF、USER、ADMIN、SUPPORT

例: `namesctl.trace_level=admin`

NAMESCTL.TRACE_FILE

目的: NAMESCTL のトレース出力を書き込むファイルを指定します。

デフォルト: `NAMESCTL_PID.TRC`

例: `namesctl.trace_file=nmsctl`

NAMESCTL.TRACE_DIRECTORY

目的: NAMESCTL ユーティリティのトレース出力を書き込むディレクトリを指定します。

デフォルト: UNIX では `$ORACLE_HOME/network/trace`、Windows NT では `ORACLE_HOME¥network¥trace`

例: `namesctl.trace_directory=/oracle/trace`

NAMESTL.TRACE_UNIQUE

目的:	作成したトレース・ファイルの名前にプロセス識別子を付加して、各トレース・ファイルが共存できるようにするかどうかを指定します。
デフォルト:	ON
値:	ON OFF
例:	<code>namesctl.trace_unique=on</code>

OSS.SOURCE.LOCATION

目的:	暗号化されたプライベート・キーを取り出す方法を定義します。
デフォルト:	なし
例:	<code>oss.source.location= (source= (method=oracle) (method_data= (sqlnet_address=oss)))</code>

OSS.SOURCE.MY_WALLET

目的:	Wallet の場所を指定します。Wallet は、SSL で処理される認証、キーおよびトラスト・ポイントです。
デフォルト:	なし
例:	<code>oss.source.my_wallet= (source= (method=file) (method_data= (directory=/home/smalladi/oss)))</code>

SQLNET.AUTHENTICATION_SERVICES

目的:	1 つ以上の認証サービスを使用可能にします。認証がインストールされている場合は、このパラメータを NONE または認証方法の 1 つに設定することをお勧めします。
デフォルト:	なし
値:	NONE - データベースはユーザー名とパスワードを使用してログインする ALL - すべての認証方法を使用可能にする BEQ - 必ず次の値と一緒に使用する NDS - NetWare Directory Services (NDS) を使用 NTS - Windows NT Native を使用 KERBEROS5 - Kerberos を使用 SECURID - SecurID を使用 CYBERSAFE - Cybersafe を使用 IDENTIX - Identix を使用 DCEGSSAPI - DCE GSSAPI を使用 RADIUS - RADIUS を使用
例:	<code>sqlnet.authentication_services =(beq, kerberos5, cybersafe)</code>

SQLNET.AUTHENTICATION_KERBEROS5_SERVICE

目的: Kerberos サービス・チケットを得るために使用するサービス名を定義します。

デフォルト: なし

例: `sqlnet.authentication_kerberos5_service= oracle`

SQLNET.AUTHENTICATION_GSSAPI_SERVICE

目的: CyberSAFE サービスのプリンシパルを定義します。

SQLNET.CLIENT_REGISTRATION

目的: クライアント・マシンの一意な識別子を設定します。この識別子は接続要求でリスナーに渡され、監査証跡に含められます。識別子として、最大 128 文字の任意の英数字を使用できます。

デフォルト: なし

例: `sqlnet.client_registration 1432`

SQLNET.CRYPTO_CHECKSUM_CLIENT

目的: このクライアントがサーバーに接続するときの希望のチェックサム動作を指定します。

デフォルト: ACCEPTED

値:

- ACCEPTED
- REJECTED
- REQUESTED
- REQUIRED

例: `sqlnet.crypto_checksum_client=accepted`

SQLNET.CRYPTO_CHECKSUM_SERVER

目的:	クライアントがこのサーバーに接続するときの希望のチェックサム動作を指定します。
デフォルト:	ACCEPTED
値:	<ul style="list-style-type: none">■ ACCEPTED■ REJECTED■ REQUESTED■ REQUIRED
例:	<code>sqlnet.crypto_checksum_server=accepted</code>

SQLNET.CRYPTO_CHECKSUM_TYPES_CLIENT

目的:	このクライアントが使用できる暗号チェックサム・アルゴリズムのリストを指定します。
デフォルト:	MD5
値:	MD5 - RSA データ・セキュリティの MD5 アルゴリズム
例:	<code>sqlnet.crypto_checksum_types_client=(md5)</code>

SQLNET.CRYPTO_CHECKSUM_TYPES_SERVER

目的:	このサーバーが使用できる暗号チェックサム・アルゴリズムのリストを指定します。
デフォルト:	MD5
値:	MD5 - RSA データ・セキュリティの MD5 アルゴリズム
例:	<code>sqlnet.crypto_checksum_types_server=(md5)</code>

SQLNET.CRYPTO_SEED

目的:	暗号キーを生成するときに使用する文字を指定します。文字がランダムなほど、キーの安全性が向上します。暗号化またはチェックサムをオンにするときは、必ずこのパラメータを指定する必要があります。10 ~ 70 文字のランダムな文字列を指定します。
デフォルト:	qwertyuiopasdfghjkl;xcvbnm,.s1
例:	sqlnet.crypto_seed="qwertyuiopasdfghjkl;xcvbnm,.s1"

SQLNET.ENCRYPTION_CLIENT

目的:	このクライアントがサーバーに接続するときの希望の動作を指定します。
デフォルト:	ACCEPTED
値:	<ul style="list-style-type: none">■ ACCEPTED■ REJECTED■ REQUESTED■ REQUIRED
例:	sqlnet.encryption_client=accepted

SQLNET.ENCRYPTION_SERVER

目的:	クライアントがこのサーバーに接続するときの希望の動作を指定します。
デフォルト:	ACCEPTED
値:	<ul style="list-style-type: none">■ ACCEPTED■ REJECTED■ REQUESTED■ REQUIRED
例:	sqlnet.encryption_server=accepted

SQLNET.ENCRYPTION_TYPES_CLIENT

目的:	このクライアントが使用できる暗号化アルゴリズムのリストを指定します。
デフォルト:	アルゴリズムが指定されていないときは、すべてのアルゴリズムが使用されます。
値:	次の値を 1 つ以上指定できます。 RC4_40 - RSA RC4 (40 ビット・キー・サイズ) 米国内および全世界 RC4_56 - RSA RC4 (56 ビット・キー・サイズ) 米国内のみ RC4_128 - RSA RC4 (128 ビット・キー・サイズ) 米国内のみ DES - 標準 DES (56 ビット・キー・サイズ) 米国内のみ DES40 - (40 ビット・キー・サイズ) 米国内および全世界
例:	<code>sqlnet.encryption_types_client=(rc4_40)</code>

SQLNET.ENCRYPTION_TYPES_SERVER

目的:	このクライアントがサーバーとして機能するときに使用できる暗号化アルゴリズムのリストを指定します。
デフォルト:	アルゴリズムが指定されていないときは、すべてのアルゴリズムが使用されます。
値:	RC4_40 - RSA RC4 (40 ビット・キー・サイズ) 米国内および全世界 RC4_56 - RSA RC4 (56 ビット・キー・サイズ) 米国内のみ RC4_128 - RSA RC4 (128 ビット・キー・サイズ) 米国内のみ DES - 標準 DES (56 ビット・キー・サイズ) 米国内のみ DES40 - (40 ビット・キー・サイズ) 米国内および全世界
例:	<code>sqlnet.encryption_types_server=(rc4_40, des, ...)</code>

SQLNET.EXPIRE_TIME

目的:	セッションが有効であるかどうかを確認するためのプローブを送る時間間隔を決定します。
デフォルト:	なし
最小値:	0 分
推奨値:	10 分
例:	<code>sqlnet.expire_time=10</code>

SQLNET.IDENTIX_FINGERPRINT_DATABASE

目的:	認証指紋データベースのサービス名または別名を指定します。
デフォルト:	なし
例:	<code>sqlnet.identix_fingerprint_database=fingrdb</code>

SQLNET.IDENTIX_FINGERPRINT_DATABASE_USER

目的:	指紋データベースに予約済みユーザー名を指定します。
デフォルト:	なし
例:	<code>sqlnet.identix_fingerprint_database_user=manager</code>

SQLNET.IDENTIX_FINGERPRINT_DATABASE_PASSWORD

目的:	指紋データベースに予約済みパスワードを指定します。
デフォルト:	なし
例:	<code>sqlnet.identix_fingerprint_database_password=password</code>

SQLNET.IDENTIX_FINGERPRINT_METHOD

目的:	指紋データベースのメソッド名を指定します。メソッド名は ORACLE でなければなりません。
デフォルト:	なし
例:	<code>sqlnet.identix_fingerprint_method=oracle</code>

SQLNET.KERBEROS5_CC_NAME

目的: Kerberos 資格証明キャッシュ・ファイルの完全パス名を指定します。

デフォルト: UNIX では /usr/tmp/krb5cache、
Windows では c:\¥tmp¥krb5cache

例: sqlnet.kerberos5_cc_name= /usr/tmp/krb5cache

SQLNET.KERBEROS5_CLOCKSKEW

目的: Kerberos 資格証明が無効になるまでの秒数を指定します。

デフォルト: 300

例: sqlnet.kerberos5_clockskew=1200

SQLNET.KERBEROS5_CONF

目的: Kerberos 構成ファイルの完全パス名を指定します。この構成ファイルには、デフォルト KDC の領域が入っていて、領域を KDC ホストにマップします。

デフォルト: UNIX では /krb5/krb.conf、
Windows では c:\¥krb5¥krb.conf

例: sqlnet.kerberos5_conf=/krb5/krb.conf

SQLNET.KERBEROS5_KEYTAB

目的:	Kerberos プリンシパル / シークレット・キー・マッピング・ファイルの完全パス名を指定します。このファイルを使用して、キーを抽出し、着信認証情報を解読します。
デフォルト:	UNIX では /etc/v5srvtab、Windows では c:\krb5\v5srvtab
例:	sqlnet.kerberos5_keytab=/etc/v5srvtab

SQLNET.KERBEROS5_REALMS

目的:	Kerberos 領域変換ファイルの完全パス名を指定します。このファイルを使用して、ホスト名またはドメイン名から領域にマップします。
デフォルト:	UNIX では /krb5/krb.realms、Windows では c:\krb5\krb.realms
例:	sqlnet.kerberos5_realms= /krb5/krb.realms

SQLNET.RADIUS_ALTERNATE

目的:	プライマリ・サーバーを利用できないときに使用する、代替用の RADIUS サーバーを指定します。サーバーの IP アドレスまたはホスト名を指定できます。
デフォルト:	なし
例:	sqlnet.radius_alternate=radius2

SQLNET.RADIUS_ALTERNATE_PORT

目的:	代替用の RADIUS サーバーのリスニング・ポートを指定します。
デフォルト:	1645
例:	sqlnet.radius_alternate_port= 1667

SQLNET.RADIUS_ALTERNATE_RETRIES

目的: 再送する回数を指定します。

デフォルト: 3

例: `sqlnet.radius_alternate_retries=4`

SQLNET.RADIUS_AUTHENTICATION_TIMEOUT

目的: 応答の待ち時間を指定します。

デフォルト: 5 秒

例: `sqlnet.radius_authentication_timeout= 10`

SQLNET.RADIUS_AUTHENTICATION

目的: プライマリ RADIUS サーバーの場所を、ホスト名または IP アドレスで指定します。

デフォルト: ローカル・ホスト

例: `sqlnet.radius_authentication= officeacct`

SQLNET.RADIUS_AUTHENTICATION_INTERFACE

目的: ユーザーとの対話に使用するユーザー・インタフェースを含むクラスを指定します。

デフォルト: DefaultRadiusInterface

例: `sqlnet.radius_authentication_interface=`

SQLNET.RADIUS_AUTHENTICATION_PORT

目的: プライマリ RADIUS サーバーのリスニング・ポートを指定します。

デフォルト: 1645

例: `sqlnet.radius_authentication_port= 1667`

SQLNET.RADIUS_AUTHENTICATION_RETRIES

目的: 再送する回数を指定します。

デフォルト: 3

例: `sqlnet.radius_authentication_retries=4`

SQLNET.RADIUS_AUTHENTICATION_TIMEOUT

目的: 応答の待ち時間を指定します。

デフォルト: 5 秒

例: `sqlnet.radius_authentication_timeout= 10`

SQLNET.RADIUS_CHALLENGE_RESPONSE

目的: 要求 - 応答のオンとオフを切り替えます。

デフォルト: OFF

値: ON | OFF

例: `sqlnet.radius_challenge_response=ON`

SQLNET.RADIUS_SECRET

目的: RADIUS 共有シークレットを含むファイルのフル・パスを指定します。

デフォルト: UNIX では `$ORACLE_HOME/network/security/radius.key`、
Windows NT では `ORACLE_`
`HOME¥network¥security¥radius.key`

例: `sqlnet.radius_secret= oracle/bin/admin/radiuskey`

SQLNET.RADIUS_SEND_ACCOUNTING

目的:	アカウントを ON または OFF にします。有効にした場合、パケットはリスニング・ポート +1 のアクティブな RADIUS サーバーに送信されます。デフォルトのポートは 1646 です。
デフォルト:	OFF
値:	ON OFF
例:	<code>sqlnet.radius_send_accounting=ON</code>

SSL_CLIENT_AUTHENTICATION

目的:	サーバーに加えて、クライアントを SSL を使用して認証するかどうかを制御します。
デフォルト:	TRUE
値:	TRUE FALSE
例:	<code>ssl_cipher_suite=(ssl_dh_dss_with_des_cdc_sha)</code>

SSL_CIPHER_SUITES

目的:	SSL で使用する暗号化とデータ整合性の組合せを制御します。
デフォルト:	なし
値:	Cipher Suite の詳細は、『Oracle8i Advanced Security 管理者ガイド』を参照してください。
例:	<code>ssl_cipher_suite=(ssl_rsa_with_rc4_138_md5)</code>

SSL_VERSION

目的:	SSL 接続のバージョンを合わせます。 クライアントとサーバーは、互換性のあるバージョンを使用する必要があります。
デフォルト:	UNDETERMINED
値:	UNDETERMINED 2.0 3.0
例:	<code>ssl_version=2.0</code>

TNSPING.TRACE_DIRECTORY

目的: トレース・ファイルの接続先ディレクトリを制御します。

デフォルト: \$ORACLE_HOME/network/trace

例: tnsping.trace_directory=/oracle/traces

TNSPING.TRACE_LEVEL

目的: 指定したレベルでトレースをオン / オフに切り替えます。

デフォルト: OFF

値:

- OFF - トレース出力しない
- USER - ユーザーによるエラー状況のトレース情報
- ADMIN - 管理トレース情報
- SUPPORT - オラクル社サポート・サービス・トレース情報

例: tnsping.trace_level=admin

TRACE_DIRECTORY_CLIENT

目的: トレース・ファイルの接続先ディレクトリを制御します。

デフォルト: \$ORACLE_HOME/network/trace

例: trace_directory_client=/oracle/traces

TRACE_DIRECTORY_SERVER

目的: トレース・ファイルの接続先ディレクトリを制御します。

デフォルト: \$ORACLE_HOME/network/trace

例: trace_directory_server=/oracle/traces

TRACE_FILE_CLIENT

目的: クライアントのトレース・ファイル名を制御します。

デフォルト: SQLNET.TRC

例: trace_file_client=clientsqlnet.trc

TRACE_FILE_SERVER

目的: サーバーのトレース・ファイル名を制御します。

デフォルト: SVR_PID.TRC

例: trace_file_server=svrsqlnet.trc

TRACE_LEVEL_CLIENT

目的: 指定したレベルでトレースをオン / オフに切り替えます。

デフォルト: OFF

値:

- OFF - トレース出力しない
- USER - ユーザー・トレース情報
- ADMIN - 管理トレース情報
- SUPPORT - オラクル社サポート・サービス・トレース情報

例: trace_level_client=user

TRACE_LEVEL_SERVER

目的:	指定したレベルでトレースをオン / オフに切り替えます。
デフォルト:	OFF
値:	<ul style="list-style-type: none">■ OFF - トレース出力しない■ USER - ユーザー・トレース情報■ ADMIN - 管理トレース情報■ SUPPORT - オラクル社サポート・サービス・トレース情報
例:	trace_level_server=admin

TRACE_UNIQUE_CLIENT

目的:	<p>各クライアント・トレース・ファイルに一意的な名前を割り当て、各トレース・ファイルがクライアントの次のオカレンスで上書きされないようにします。PID がファイル名の終わりに追加されず。</p> <p>各クライアントに一意的なトレース・ファイルを作成するかどうかを決定します。それぞれのクライアントでは、同じトレース・ファイル名が使用されます。値を OFF に設定すると、クライアントで新しいトレース・ファイルが作成されるときに、既存のファイルが上書きされます。この値を ON に設定すると、複数のトレース・ファイルが共存できるように、各トレース・ファイルの名前にスレッド識別子が追加されます。</p> <p>このパラメータにより、トレース・ファイルは <code>SQLNETTHREAD_ID.TRC</code> という一意的な名前になります。</p>
デフォルト:	ON
例:	trace_unique_client=on

USE_CMAN

目的:	このパラメータを TRUE に設定すると、すべてのセッションは Oracle Connection Manager を通じてサーバーに到達します。利用できる Oracle Connection Manager アドレスがない場合、接続は利用可能な任意のリスナー・アドレスを通して行われます。 注意: Oracle Names で Oracle Connection Manager を使用している場合は、クライアントと Oracle Names Server の両方で、このオプションを設定する必要があります。
デフォルト:	FALSE
値:	TRUE FALSE
例:	use_cman=true

USE_DEDICATED_SERVER

目的:	リスナーがこのクライアントからのセッションのための専用サーバー・プロセスを生成するようにします。
デフォルト:	OFF
値:	<ul style="list-style-type: none">■ ON - 専用サーバー・プロセスを事前に生成する■ OFF - 既存のサーバー・プロセスに渡す
例:	use_dedicated_server=on

ローカル・ネーミング・パラメータ (TNSNAMES.ORA)

ローカル・ネーミング・メソッドでは、TNSNAMES.ORA ファイルにネット・サービス名が追加されます。TNSNAMES.ORA は、UNIX では \$ORACLE_HOME/network/admin、Windows では ORACLE_HOME¥network¥admin に置かれます。

ネット・サービス名は、接続記述子に含まれているネットワーク・アドレスにマップされた別名です。接続記述子には、プロトコル・アドレスによるリスナーの場所や、接続するデータベースのサービス名が含まれます。他のサーバーのクライアントになるクライアントとサーバーは、このネット・サービス名を使用してアプリケーションに接続します。

一般構文

TNSNAMES.ORA ファイルの基本的な構文を、[図 C-1](#) に示します。DESCRIPTION には接続記述子、ADDRESS にはリスナー・アドレス、および CONNECT_DATA にはデータベース・サービス名が含まれます。

図 C-1 TNSNAMES.ORA の一般構文

```
net_service_name=  
  (description=  
    (address= (protocol_address_information))  
    (connect_data=  
      (service_name=service_name)  
    )  
  )  
)
```

複数の記述

TNSNAMES.ORA ファイルには、複数の接続記述子を含めることもできます。各接続記述子には、1 つ以上のリスナー・アドレスが含まれます。図 C-2 は、複数のアドレスを持つ 2 つの接続記述子を示したものです。DESCRIPTION_LIST は、接続記述子のリストを定義します。

注意： Oracle Net8 Assistant は、複数の接続記述子をサポートしていません。

図 C-2 TNSNAMES.ORA の複数の記述

```
net_service_name=
(description_list=
(description=
(address= (protocol_address_information))
(address= (protocol_address_information))
(address= (protocol_address_information))
(connect_data=
(service_name=service_name)
)
)
)
(description=
(address= (protocol_address_information))
(address= (protocol_address_information))
(address= (protocol_address_information))
(connect_data=
(service_name=service_name)
)
)
)
```

複数のアドレス・リスト

TNSNAMES.ORA は、それぞれが独自の特性を持つ複数のアドレス・リストもサポートしています。図 C-3 は、2つのアドレス・リストを示したものです。最初のアドレス・リストでは、クライアント負荷のバランスを指定し、connect-time failover を使用しません。これらの設定は、ADDRESS_LIST の ADDRESS に指定されたリスナーにしか適用されません。次のアドレス・リストでは、connect-time failover を指定し、クライアント負荷のバランスを使用しません。これらの設定は、ADDRESS_LIST の ADDRESS に指定されたリスナーにしか適用されません。クライアントはまず、1番目または2番目のアドレスをランダムに選び、次に3番目と4番目のアドレスを順に試します。

注意： Oracle Net8 Assistant は、1つのアドレス・リストしかサポートしていません。

図 C-3 TNSNAMES.ORA の複数のアドレス・リスト

```
net_service_name=  
(description=  
  (address_list=  
    (load_balance=on)  
    (failover=off)  
    (address= (protocol_address_information))  
    (address= (protocol_address_information))  
  )  
  (address_list=  
    (load_balance=off)  
    (failover=on)  
    (address= (protocol_address_information))  
    (address= (protocol_address_information))  
  )  
  (connect_data=  
    (service_name=service_name)  
  )  
)
```

注意： リリース 8.1 以前のように、アドレス・リストが1つしかない場合は、リストを ADDRESS_LIST に埋め込む必要はありません。

クライアント負荷のバランスと DESCRIPTION_LIST の変更された機能

リリース 8.1 以前は、DESCRIPTION_LIST が複数のリスナーのクライアント負荷のバランスに使用されていました。クライアント負荷のバランスは、現在でも DESCRIPTION_LIST ではデフォルトで有効ですが、ADDRESS_LIST で明示的に指定することができ、一連の ADDRESS や DESCRIPTION に関連付けることもできます。図 C-4 は、これまでの TNSNAMES.ORA とリリース 8.1 の TNSNAMES.ORA を、Oracle Parallel Server 環境のクライアント負荷のバランスを使用して比較したものです。

図 C-4 リリース 8.1 とそれ以前のリリースのクライアント負荷のバランス

リリース 8.1 以前のクライアント負荷のバランス

次に示すのは、リリース 8.1 以前の TNSNAMES.ORA ファイルです。OP、OP1 および OP2 はネット・サービス名です。

OP により、クライアントは 2 つのインスタンスのいずれかをランダムに選択できます。OP1 と OP2 で、クライアントは特定の Oracle Parallel Server インスタンスに接続できます。

```
op=
(description_list =
  (description=
    (address=
      (protocol=tcp)
      (host=opsnt1)
      (port=1521))
    (connect_data=(sid=op1)))

  (description=
    (address=
      (protocol=tcp)
      (host=opsnt2)
      (port=1521))
    (connect_data=(sid=op2))))

op1=
(description=
  (address=
    (protocol=tcp)
    (host=opsnt1)
    (port=1521))
  (connect_data=(sid=op1)))

op2=
(description=
  (address=
    (protocol=tcp)
    (host=opsnt1)
    (port=1521))
  (connect_data=(sid=op2)))
```

リリース 8.1 でのクライアント負荷のバランス

次に示すのは、リリース 8.1 の TNSNAMES.ORA ファイルです。OP、OP1 および OP2 はネット・サービス名です。

OP により、クライアントはデータベース・サービス OP にランダムに接続し、DESCRIPTION ではなく複数の ADDRESS を使用して、2 つのインスタンスのいずれかを選択できます。OP1 と OP2 で、クライアントは特定の Oracle Parallel Server インスタンスに接続できます。

```
op=
(description=
  (load_balance=on)
  (address=
    (protocol=tcp)
    (host=opsnt1)
    (port=1521))

  (address=
    (protocol=tcp)
    (host=opsnt2)
    (port=1521))
  (connect_data=
    (service_name=op)))

op1=
(description=
  (address=
    (protocol=tcp)
    (host=opsnt1)
    (port=1521))
  (connect_data=
    (service_name=op)
    (instance_name=op1)))

op2=
(description=
  (address=
    (protocol=tcp)
    (host=opsnt2)
    (port=1521))
  (connect_data=
    (service_name=op)
    (instance_name=op2)))
```

例

クライアントが Oracle Connection Manager を通してサービスに接続する場合、最初のアドレス (またはアドレスのセット) は Oracle Connection Manager に接続するためのクライアント用のアドレスで、2 番目のアドレス (またはアドレスのセット) は、リスナーを使用してサービスに接続する Oracle Connection Manager 用のアドレスとなります。

Oracle Connection Manager の SOURCE_ROUTE パラメータを YES に設定すると、クライアントは接続する前にリストにあるすべてのアドレスを試みます。

図 C-5 は、複数の Oracle Connection Manager アドレスを使用する場合の例です。

図 C-5 TNSNAMES.ORA に指定する複数の Oracle Connection Manager アドレス

```
sample1=(description=
  (source_route=yes)
  (address=(protocol=tcp)(host=host1)(port=1630)) # hop 1
  (address_list=
    (failover=on)
    (load_balance=off) # hop 2
    (address=(protocol=tcp)(host=host2a)(port=1630))
    (address=(protocol=tcp)(host=host2b)(port=1630))
  )
  (address=(protocol=tcp)(host=host3)(port=1521)) # hop 3
  )
(connect_data=(service_name=sales.acme.com))
)
```

上の例では次の動作が行われます。

1. クライアントは、最初の Oracle Connection Manager のアドレスに接続するように指示されます。

```
(address=(protocol=tcp)(host=host1)(port=1630))
```

2. 最初の Oracle Connection Manager は、Oracle Connection Manager アドレスの 1 つを使用して、他の Oracle Connection Manager に接続します。1 つ目のアドレスに失敗したら、2 つ目のアドレスで接続を試みます。この動作は、次の部分で指示されます。

```
(address_list=
  (failover=on)
  (load_balance=off) # hop 2
  (address=(protocol=tcp)(host=host2a)(port=1630))
  (address=(protocol=tcp)(host=host2b)(port=1630))
)
```

3. 続いて Oracle Connection Manager は、次のリスナー・アドレスを使用してデータベース・サービスに接続します。

```
(address=(protocol=tcp)(host=host3)(port=1521)) # hop 3
```

図 C-6 は、2 つの Oracle Connection Manager と 2 つのリスナー・アドレス間のクライアント負荷のバランスの例を示しています。

図 C-6 TNSNAMES.ORA におけるクライアント負荷のバランス

```
sample2=(description=
  (load_balance=on)
  (failover=on)
  (address_list=
    (source_route=yes)
    (address=(protocol=tcp)(host=host1)(port=1630))
    (address=(protocol=tcp)(host=host2)(port=1521))
  )
  (address_list=
    (source_route=yes)
    (address=(protocol=tcp)(host=host3)(port=1630))
    (address=(protocol=tcp)(host=host4)(port=1521))
  )
  (connect_data=(service_name=sales.acme.com))
)
```

上の例では次の動作が行われます。

1. クライアントは、ADDRESS_LIST をランダムに選択し、選択した ADDRESS_LIST で失敗したら他にフェイルオーバーするように指示されます。この動作は、LOAD_BALANCE および FAILOVER パラメータを ON にすることで指示されます。
2. ADDRESS_LIST が選択されると、クライアントはまず ADDRESS_LIST に指定された Oracle Connection Manager のアドレスを使用して、Oracle Connection Manager に接続します。
3. 続いて Oracle Connection Manager は、ADDRESS_LIST に指定されたリスナー・アドレスを使用して、データベース・サービスに接続します。

DESCRIPTIONS および DESCRIPTION_LIST

DESCRIPTION

目的: 接続記述子の開始を表します。データベース・リスニング・アドレスの定義、接続するデータベースのサービス名が指定されます。このパラメータは、DESCRIPTION_LIST の下に埋め込まれる場合があります。

例:

```
net_service_name=
(description =
(address=...)
(connect_data=(service_name=sales.com))
```

DESCRIPTION_LIST

目的: 特定のネット・サービス名の接続記述子のリストを定義します。

例:

```
net_service_name=
(description_list=
(description=
(address=...)
(connect_data=(service_name=sales.com))
(description=
(address=...)
(connect_data=(service_name=sales2.com))
)
```

ADDRESS と ADDRESS_LIST

ADDRESS

目的: リスナー・プロトコル・アドレスを定義します。このパラメータは、ADDRESS_LIST または DESCRIPTION の下に埋め込まれる場合があります。

追加情報: 各プロトコルで使用する適切なプロトコルの説明は、6-76 ページの「[プロトコル・アドレスの構成](#)」を参照してください。

例:

```
net_service_name=  
(description=  
  (address =...)  
  (address =...)  
  )  
(connect_data=(service_name=sales.com))
```

ADDRESS_LIST

目的: アドレスのリストを定義します。リストが 1 つしかない場合は、ADDRESS_LIST は必要ありません。このパラメータは、DESCRIPTION または DESCRIPTION_LIST の下に埋め込まれる場合があります。

例:

```
net_service_name=  
(description=  
  (address_list=  
    (address =...)  
    (address =...)  
  )  
)  
(address_list=  
  (load_balance=off)  
  (address =...)  
  (address =...)  
  )  
)  
(connect_data=(service_name=sales.com))
```

リストのオプション・パラメータ

FAILOVER

目的:	<p>このパラメータを ON に設定すると、Net8 が最初のリスナーで接続に失敗した場合、別のリスナーにフェイルオーバーします。試してみるアドレスの数を決定します。OFF に設定すると、Net8 は 1 つのアドレスしか試みません。</p> <p>このパラメータは、DESCRIPTION_LIST、DESCRIPTION または ADDRESS_LIST の下に埋め込む必要があります。</p> <p>注意: LISTENER.ORA が静的なサービス構成パラメータで構成されている場合、この機能は動作しません。これらのパラメータの詳細は、C-57 ページの「SID_LIST_listener_name 静的サービス・セクション」を参照してください。Oracle8 リリース 8.0 または Oracle7 データベース、および Oracle Enterprise Manager では、静的な構成が必要です。</p>
デフォルト:	DESCRIPTION_LIST、DESCRIPTION、ADDRESS_LIST で ON
値:	ON OFF YES NO TRUE FALSE
例:	<pre>net_service_name= (description= (failover=on) (address =...) (address =...)) (connect_data=(service_name=sales.com)))</pre>

LOAD_BALANCE

目的:	<p>このパラメータを ON に設定すると、Net8 はリスナー・アドレスのリストをランダムな順で使用して、さまざまなリスナーの負荷のバランスをとります。OFF に設定すると、Net8 は成功するまでアドレスを順に試みます。</p> <p>このパラメータは、DESCRIPTION_LIST、DESCRIPTION または ADDRESS_LIST の下に埋め込む必要があります。</p>
デフォルト:	DESCRIPTION_LIST で ON
値:	ON OFF YES NO TRUE FALSE
例:	<pre>net_service_name= (description= (load_balance=on) (address =...) (address =...)) (connect_data=(service_name=sales.com))</pre>

SOURCE_ROUTE

目的:	<p>このパラメータを ON または YES に設定すると、Net8 は接続先に到達するまで、各アドレスを順に使用します。接続先に達するまでに複数のアドレスを経由する場合は、このパラメータが必要となります。</p> <p>このパラメータは Oracle Connection Manager で必要です。Oracle Connection Manager では、クライアントから Oracle Connection Manager への初期の接続が必要です。また、Oracle Connection Manager からリスナーへの第 2 の接続も必要です。</p> <p>このパラメータは、DESCRIPTION_LIST、DESCRIPTION または ADDRESS_LIST の下に埋め込む必要があります。</p>
デフォルト:	OFF
値:	YES、NO、ON、OFF
例:	<pre>net_service_name= (description = (source_route=yes) (address=...) (address=...)) (connect_data=(sid=sales))</pre>

TYPE_OF_SERVICE

目的 : Oracle RDB データベースで使用するサービスの種類を指定します。このサービスは、Rdb インタフェース・ツールで使用されます。この機能は、アプリケーションが Oracle Rdb および Oracle データベースの両方をサポートしている場合、およびアプリケーションをランダムに選択する（負荷のバランスをとる）場合にのみ使用してください。このパラメータは、DESCRIPTION の下に埋め込む必要があります。

例 :

```
net_service_name=
(description_list=
(description=
(address=...)
)
(connect_data=
(service_name=generic)
(rdb_database=[.mf]mf_personal.rdb)
(global_name=alpha5))
(type_of_service=rdb_database))
(description=
(address=...)
(connect_data=
(service_name=sales))
(type_of_service=oracle8_database)))
```

CONNECT_DATA セクション

CONNECT_DATA

目的: 接続するデータベースを定義します。このパラメータは、DESCRIPTION の下に埋め込む必要があります。

例:

```
net_service_name=
  (description=
    (address =...)
    (address =...)
  )
  (connect_data=(service_name=sales.com))
```

CONNECT_DATA には次のパラメータがあります。

- [FAILOVER_MODE](#)
- [GLOBAL_NAME](#)
- [HS](#)
- [INSTANCE_NAME](#)
- [RDB_DATABASE](#)
- [SDU](#)
- [SERVER](#)
- [SERVICE_NAME](#)

FAILOVER_MODE

目的: Net8 に、実行時に最初のリスナーで失敗した場合に、別のリスナーにフェイルオーバーするように指示します。構成に応じて、進行中のセッションや SELECT 文は自動的にフェイルオーバーされます。このパラメータは、CONNECT_DATA の下に埋め込む必要があります。

このタイプのフェイルオーバーは、透過的アプリケーション・フェイルオーバーと呼ばれます。「[Connect-Time Failover](#)」[FAILOVER](#) パラメータと混同しないようにしてください。透過的アプリケーション・フェイルオーバーの実装の詳細は、『Oracle8i チューニング』を参照してください。

FAILOVER_MODE

サブパラメータ:

FAILOVER_MODE は、次のサブパラメータをサポートしていません。

- BACKUP - フェイルオーバー・ノードを指定します。
- TYPE - **(必須)** フェイルオーバーの種類を指定します。デフォルトでは、Oracle Call Interface (OCI) アプリケーションに対して、3種類の Net8 フェイルオーバー機能を利用できません。

SESSION: セッションをフェイルオーバーします。すなわち、ユーザーの接続が失われた場合に、新しいセッションを自動的に作成して、ユーザーをバックアップします。このフェイルオーバーでは、選択は回復されません。

SELECT: カーソルをオープンしているユーザーが、障害後もカーソルをフェッチできるようにします。ただしこのモードでは、通常の実行操作を行うクライアント側にオーバーヘッドが生じます。

NONE: デフォルトです。フェイルオーバー機能を使用しません。フェイルオーバーが実行されないように、明示的に指定することもできます。

- METHOD: プライマリ・ノードからバックアップ・ノードにフェイルオーバーされる速さを決定します。

BASIC: フェイルオーバー時に接続を確立します。フェイルオーバー時まで、バックアップ・サーバーはほとんど何もする必要はありません。

PRECONNECT: あらかじめ接続を確立しておきます。フェイルオーバーは高速に行われますが、バックアップ・インスタンスはサポートされた各インスタンスからの接続すべてをサポートできなければなりません。

FAILOVER_MODE

例 :

次に、OPSHP1 で障害が発生した場合の、バックアップ・ノード OPSHP2 への透過的アプリケーション・フェイルオーバー (FAILOVER_MODE) の例を示します。処理中の SELECT 文がフェイルオーバーされます。

```
net_service_name=
(description=
(address=
(protocol=tcp)
(host=opshp1)
(port=1521))
(connect_data=
(service_name=op.us.oracle.com)
(failover_mode=
(backup=opshp2)
(type=select)
(method=basic)
)
)
)
```

次に、connect-time failover (FAILOVER) で使用されている、透過的アプリケーション・フェイルオーバーを示します。まず OPSHP1 で接続を試し、次に OPSHP2 で試してみます。接続後にノードに障害が発生した場合、アプリケーションは処理中の SELECT 文を保存して、まず OPSHP1、次に OPSHP2 にフェイルオーバーを試みます。

```
net_service_name=
(description=
(failover=on)
(address=
(protocol=tcp)
(host=opshp1)
(port=1521))
(address=
(protocol=tcp)
(host=opshp2)
(port=1521))
(connect_data=
(service_name=op.us.oracle.com)
(failover_mode=
(type=select)
)
)
)
```

GLOBAL_NAME

目的 : Oracle Rdb データベースを識別します。

例 :

```
net_service_name=
(description=
(address =...)
(address =...)
(connect_data=
(service_name=generic)
(rdb_database= [.mf]mf_personal.rdb)
(global_name= alpha5)))
```

HS

目的 : Net8 に、非 Oracle システムに接続するように指示します。このパラメータは、CONNECT_DATA の下に埋め込む必要があります。

デフォルト : なし

値 : OK

例 :

```
net_service_name=
(description =
(address=...)
(address=...)
(connect_data=
(sid=sales6)
(hs=ok))
```

INSTANCE_NAME

目的: アクセスするデータベース・インスタンス名を識別します。このパラメータは、CONNECT_DATA の下に埋め込まれる場合があります。インスタンス名は、INITSID.ORA ファイルの INSTANCE_NAME パラメータから取得できます。

一般的に INSTANCE_NAME は、インストール中またはデータベースの作成中に入力されたシステム識別子 (SID) です。

追加情報: インスタンス名の文字列設定は、6-76 ページの「[プロトコル・アドレスの構成](#)」を参照してください。

例:

```
net_service_name=
(description =
  (address=...)
  (address=...))
(connect_data=
  (service_name=sales.com)
  (instance_name=sales1))
```

RDB_DATABASE

目的: Oracle RDB データベースのファイル名を指定します。

例:

```
net_service_name=
(description=
  (address =...)
  (address =...))
(connect_data=
  (service_name=sales.com)
  (rdb_database= [.mf]mf_personal.rdb))
```

SDU

目的:	Net8 に、ネットワークを介して送信されるデータ・パケットの転送レートを、指定したセッション・データ・ユニット (SDU) サイズで最適化するように指示します。このパラメータは、CONNECT_DATA の下に埋め込まれる場合があります。
デフォルト:	2048
値:	SDU サイズは、通常の伝送フレーム・サイズの倍数に設定する必要があります。イーサネットの通常のフレーム・サイズは 1024 のため、イーサネット・プロトコルでの最も効率的な SDU サイズは 1024 の倍数です。ただし、1024 の 4 倍までに制限されます。
例:	<pre>net_service_name= (description = (address=...) (address=...)) (connect_data= (server_name=sales.com) (sdu=1024))</pre>

SERVER

目的:	Net8 に、MTS 環境で専用サーバーを使用するように指示します。このパラメータは、CONNECT_DATA の下に埋め込まれる場合があります。
デフォルト:	DEDICATED
値:	DEDICATED SHARED
例:	<pre>net_service_name= (description = (address=...) (address=...)) (connect_data= (server_name=sales.com) (server=dedicated))</pre>

SERVICE_NAME

目的: アクセスするリリース 8.1 サービスを識別します。このパラメータは、CONNECT_DATA の下に埋め込む必要があります。一般的に SERVICE_NAME は、インストール中またはデータベースの作成中に入力されたグローバル・データベース名、すなわちデータベース名とドメイン名からなる名前に設定されます。

追加情報: インスタンス名の設定は、6-2 ページの「サービス名とインスタンス名の設定」を参照してください。

例:

```
net_service_name=
  (description =
    (address=...)
    (address=...)
  (connect_data=
    (service_name=sales.com)
    (instance_name=sales1))
```

SID

目的: リリース 8.1 以前のデータベースの、データベース・インスタンスを識別します。このパラメータは、CONNECT_DATA の下に埋め込む必要があります。

例:

```
net_service_name=
  (description =
    (address=...)
    (address=...)
  (connect_data=
    (sid=sales))
```

リスナー・パラメータ (LISTENER.ORA)

ネットワーク・リスナー構成ファイル (LISTENER.ORA) では、次の項目を使用できます。

- リスナー・アドレス・セクション
- `SID_LIST_listener_name` 静的サービス・セクション
- 制御パラメータ:

LISTENER.ORA は、UNIX では `$ORACLE_HOME/network/admin`、Windows NT では `ORACLE_HOME¥network¥admin` に置かれます。

リスナー・アドレス・セクション

LISTENER.ORA のリスナー・アドレス・セクションは、リスナーのプロトコル・アドレスを定義します。

DESCRIPTION

目的:

リスナー・プロトコル・アドレスと、使用する [プレゼンテーション](#) および [セッション](#) 通信レイヤーを定義します。

これまでのリリースでは、Net8 はクライアント接続の確立に Two-Task Common (TTC) プレゼンテーション・レイヤー、および Transparent Network Substrate (TNS) Network Session (NS) レイヤーを使用する接続のみをサポートしていました。

クライアントをデータベースの Java オプションに接続するために、プレゼンテーション・レイヤーおよびセッション・レイヤー・サポートが拡張されています。Java オプションに接続するクライアントでは、セッション・レイヤーを使用しない General Inter-Orb Protocol (GIOP) が使用されます。

このパラメータは、DESCRIPTION_LIST の下に埋め込まれる場合があります。

指定されたプロトコル・アドレスのすべてが TTC および NS を使用する場合は、必ずしも DESCRIPTION を使用する必要はありません。

追加情報: 構成情報は、7-48 ページの「[Java オプション接続の構成](#)」を参照してください。

例:

```
listener_name =
  (description_list =
    (description =
      (address=
        (protocol=ipc)
        (key=extproc0))
      )
    (description =
      (protocol_stack =
        (presentation=giop)
        (session=raw))
      (address=
        (protocol=tcp)
        (host=sales-pc)
        (port=2481))
      )
  )
```

DESCRIPTION_LIST

目的: リスナー・プロトコル・アドレスとプロトコル・スタックの設定を定義します。

例:

```
listener_name =  
  (description_list =  
    (description =  
      (address=  
        (protocol=ipc)  
        (key=extproc0))  
      )  
    (description =  
      (protocol_stack =  
        (presentation=giop)  
        (session=raw))  
      (address=  
        (protocol=tcp)  
        (host=sales-pc)  
        (port=2481))  
      )  
    )  
  )
```

ADDRESS

目的: リスナー・プロトコル・アドレスを定義します。このパラメータは、ADDRESS_LIST または DESCRIPTION の下に埋め込まれる場合があります。

追加情報: 各プロトコルで使用する適切なプロトコルの説明は、6-76 ページの「[プロトコル・アドレスの構成](#)」を参照してください。

例:

```
listener_name =  
  (address=  
    (protocol=ipc)  
    (key=extproc0))  
  (address=  
    (protocol=tcp)  
    (host=sales-pc)  
    (port=2481))
```

ADDRESS_LIST

目的: リスナー・プロトコル・アドレスのリストを定義します。アドレスのリストが1つしかない場合は、必ずしもこのパラメータを使用する必要はありません。Oracle Net8 Assistant および Net8 Configuration Assistant はデフォルトで、リストが1つしかない場合でも、アドレスが指定されるたびに ADDRESS_LIST を作成します。このパラメータは DESCRIPTION または DESCRIPTION_LIST の下に埋め込まれる場合があります。

例:

```
listener_name =
  (description_list =
    (description =
      (address_list =
        (address=
          (protocol=ipc)
          (key=extproc0))
        )
      )
    (description =
      (protocol_stack =
        (presentation=giop)
        (session=raw)
      )
      (address_list =
        (address=
          (protocol=tcp)
          (host=sales-pc)
          (port=2481))
        )
      )
    (description =
      (address_list =
        (address=
          (protocol=tcp)
          (host=sales-pc)
          (port=1521))
        )
      )
    )
  )
)
```

PROTOCOL_STACK

目的： これまでのリリースでは、Net8 はクライアント接続の確立に Two-Task Common (TTC) プレゼンテーション・レイヤーと Network Session (NS) レイヤーを使用する接続のみをサポートしていました。

Java オプションに接続するクライアントでは、セッション・レイヤーを使用しない General Inter-Orb Protocol (GIOP) が使用されます。

このパラメータは、DESCRIPTION の下に埋め込む必要があります。

指定されたプロトコル・アドレスが TTC および NS を使用する場合は、必ずしも PROTOCOL_STACK を使用する必要はありません。

追加情報： 構成情報の詳細は、7-48 ページの「[Java オプション接続の構成](#)」を参照してください。

例：

```
listener_name =
  (description_list =
    (description =
      (address=
        (protocol=ipc)
        (key=extproc0))
      )
    (description =
      (protocol_stack =
        (presentation=giop)
        (session=raw))
      (address=
        (protocol=tcp)
        (host=sales-pc)
        (port=2481))
      )
    )
  )
```

PRESENTATION

目的: プレゼンテーション・レイヤーを識別します。

デフォルト: TTC

値: TTC | GIOP

標準的な Net8 クライアントは、**Two-Task Common (TTC)** の**プレゼンテーション・レイヤー**を使用します。

IOP クライアントは、GIOP の**プレゼンテーション・レイヤー**を必要とします。GIOP は、TCP/IP または SSL 付き TCP/IP を使用する oracle.aurora.server.SGiopServer または oracle.aurora.server.GiopServer の別名です。

追加情報: 構成情報の詳細は、7-48 ページの「**Java オプション接続の構成**」を参照してください。

例:

```
listener_name =
  (description_list =
    (description =
      (address=
        (protocol=ipc)
        (key=extproc0))
      )
    (description =
      (protocol_stack =
        (presentation=giop)
        (session=raw))
      (address=
        (protocol=tcp)
        (host=sales-pc)
        (port=2481))
      )
  )
```

SESSION

目的:	セッション・レイヤーを識別します。
デフォルト:	NS
値:	NS RAW IOP クライアントは、 セッション・レイヤー を必要としません (RAW)。 標準的な Net8 クライアントは、 Transparent Network Substrate (TNS) Network Session (NS) のセッション・レイヤーを使用します。
例:	<pre>listener_name = (description_list = (description = (address= (protocol=ipc) (key=extproc0))) (description = (protocol_stack = (presentation=giop) (session=raw)) (address= (protocol=tcp) (host=sales-pc) (port=2481)))))</pre>

SID_LIST_listener_name 静的サービス・セクション

LISTENER.ORA の SID_LIST セクションは、リスナーでサービスされるデータベースのシステム識別子 (SID) を定義します。現在は、インスタンス情報が自動的にリスナーに登録されるため、このセクションはバージョン 7.x およびバージョン 8.0 の Oracle データベースのみで有効です。外部プロシージャや異種間サービスなどの他のサービス、および Oracle Enterprise Manager を含む管理ツールでは、静的な構成も必要です。

注意： connect-time failover をインプリメントしても、LISTENER.ORA ファイルの静的サービス構成パラメータは使用できません。ただし、Oracle8 リリース 8.0 または Oracle7 データベース、および Oracle Enterprise Manager では静的な構成が必要です。リスナーの静的な構成の詳細は、6-71 ページの「サービス情報によるリスナーの構成」を参照してください。

Oracle Enterprise 環境で、connect-time failover を設定する手順は、『Oracle8i Parallel Server セットアップおよび構成ガイド』を参照してください。

```
sid_list_listener_name=  
  (sid_list=  
    (sid_desc=  
      (global_dbname=global_database_name)  
      (sid_name=sid)  
      (oracle_home=oracle_home)  
      (prespawn_max=99)  
      (prespawn_list=  
        (prespawn_desc=  
          (protocol=tcp)  
          (pool_size=10)  
          (timeout=2))  
          (prespawn_desc=...))  
        (sid_desc=...)  
      )  
    )  
  )
```

SID_LIST には、次のパラメータが含まれます。

GLOBAL_DBNAME

目的: グローバル・データベース名は、データベース初期化パラメータ・ファイル (INITSID.ORA) の SERVICE_NAMES パラメータに指定されているデータベース名とそのドメイン名です。ネットワーク上でグローバル・データベース名を使用してデータベースを参照する場合は、リスナーにそのデータベースのグローバル・データベース名を指定する必要があります。このパラメータは、SID_DESC の下に埋め込む必要があり、SERVICE_NAMES パラメータの値と一致させてください。

例:

```
sid_list_listener_name=  
  (sid_list=  
    (sid_desc=  
      (global_dbname=oracle.com)  
      (sid_name=orcl)  
      (oracle_home=/usr/oracle)  
    )  
  )  
)
```

ORACLE_HOME

目的: 指定しているデータベースの Oracle ホームの位置を識別します。このパラメータは、SID_DESC の下に埋め込む必要があります。

例:

```
sid_list_listener_name=  
  (sid_list=  
    (sid_desc=  
      (sid_name=extproc)  
      (oracle_home=/usr/oracle)  
      (program=extproc)  
    )  
  )  
)
```

PROGRAM

目的: サービスの実行可能プログラム名を識別します。たとえば、`extproc.exe` などです。

例:

```
sid_list_listener_name=  
  (sid_list=  
    (sid_desc=  
      (sid_name=extproc)  
      (oracle_home=oracle)  
      (program=extproc)  
    )  
  )  
)
```

SID_NAME

目的: データベースの SID を識別します。その値は、オペレーティング・システム固有です。このパラメータは、`SID_DESC` の下に埋め込む必要があります。

このパラメータは、`INITSID.ORA` ファイルの `INSTANCE_NAME` パラメータの値に一致させます。

例:

```
sid_list_listener_name=  
  (sid_list=  
    (sid_desc=  
      (global_dbname=oracle.com)  
      (sid_name=orcl)  
      (oracle_home=/usr/oracle)  
    )  
  )  
)
```

SID_LIST

目的: SID のリストの設定を識別します。

例:

```
sid_list_listener_name=  
  (sid_list=  
    (sid_desc=...)  
    (sid_desc=...)  
  )
```

SID_DESC

目的: SID を定義します。このパラメータは、SID_LIST の下に埋め込まれる場合があります。

例:

```
sid_list_listener_name=  
  (sid_list=  
    (sid_desc=...)  
    (sid_desc=...)  
  )
```

PRESPAWN_MAX

目的: リスナーによって事前に生成される専用サーバー・プロセスの最大数を指定します。この数値は、最小でも各プロトコル用のプール・サイズの合計と同じでなければなりません。事前に生成される専用サーバー・プロセスが常に新しい接続に使用できるように、この値を大きな値に設定します。このパラメータは、SID_LIST の下に埋め込む必要があります。

事前に生成される専用サーバーは、Windows NT 上では構成できません。

例:

```
sid_list_listener_name=  
  (sid_list=  
    (sid_desc=  
      (global_dbname=oracle.com)  
      (sid_name=sid)  
      (oracle_home=/usr/oracle)  
      (prespawn_max=99))  
  )
```

PRESPAWN_LIST

目的: 事前に生成される専用サーバー・プロセスのリストの設定を指定します。このパラメータは、SID_DESC の下に埋め込む必要があります。

例:

```
sid_list_listener_name=  
  (sid_list=  
    (sid_desc=  
      (global_dbname=oracle.com)  
      (sid_name=sid)  
      (oracle_home=/usr/oracle)  
      (prespawn_max=99)  
      (prespawn_list=  
        (prespawn_desc=  
          (protocol=tcp)  
          (pool_size=10)  
          (timeout=2))  
          (prespawn_desc=...)))  
    )  
  )
```

PRESPAWN_DESC

目的: リスナーによって事前に生成される専用サーバー・プロセスのプロトコル、および事前に生成される専用サーバーの特性を定義します。Each protocol must have its own PRESPAWN_DESC description. このパラメータは、PRESPAWN_LIST の下に埋め込む必要があります。

追加情報: PROTOCOL、POOL_SIZE および TIMEOUT の説明は、6-73 ページの「[事前に起動または生成される専用サーバーの構成](#)」を参照してください。

例:

```
sid_list_listener_name=  
  (sid_list=  
    (sid_desc=  
      (global_dbname=oracle.com)  
      (sid_name=sid)  
      (oracle_home=/usr/oracle)  
      (prespawn_max=99)  
      (prespawn_list=  
        (prespawn_desc=  
          (protocol=tcp)  
          (pool_size=10)  
          (timeout=2))  
          (prespawn_desc=...)))  
    )  
  )
```

制御パラメータ：

次のパラメータはリスナーの動作を制御します。

CONNECT_TIMEOUT_*listener_name*

目的： 接続が開始されてからリスナーが有効な接続要求を待つ時間を決定します。

デフォルト： 10 秒

例： connect_timeout_listener=12

LOG_DIRECTORY_*listener_name*

目的： ログ・ファイルの配置ディレクトリを制御します。

デフォルト： UNIX では \$ORACLE_HOME/network/log、Windows NT では ORACLE_HOME¥network¥log

例： log_directory_listener=/oracle/network/admin/log

LOG_FILE_*listener_name*

目的： リスナーのログ・ファイル名を設定します。

デフォルト： LISTENER.LOG

例： log_file_listener=list.log

PASSWORDS_*listener_name*

目的： リスナー制御ユーティリティ (LSNRCTL) の認証に使用する非暗号化パスワードを設定します。1 つ以上のパスワードを設定できます。このオプション・パラメータに 1 つ以上のパスワードを指定すると、これらのパスワードのいずれかを使用するには、LSNRCTL を使用してネットワーク・リスナーの DBA 作業を実行する必要があります。

デフォルト： oracle

例： passwords_listener=(oracle8)

STARTUP_WAITTIME_listener_name

目的: ネットワーク・リスナーが、最初のリスナー・コントロール・ステータス・コマンドに回答するまでの待ち時間を秒数で設定します。

デフォルト: 0 秒

例: startup_waittime_listener=5

TRACE_DIRECTORY_listener_name

目的: ログ・ファイルの配置ディレクトリを制御します。

デフォルト: UNIX では \$ORACLE_HOME/network/trace、Windows NT では ORACLE_HOME¥network¥trace

例: trace_directory_listener=/oracle/network/admin/trace

TRACE_FILE_listener_name

目的: リスナーのトレース・ファイル名を設定します。

デフォルト: LISTENER.TRC

例: trace_file_listener=list.trc

TRACE_LEVEL_listener_name

目的:	指定したレベルでトレースをオン / オフに切り替えます。
デフォルト:	OFF
値:	<ul style="list-style-type: none">■ OFF - トレース出力しない■ USER - ユーザー・トレース情報■ ADMIN - 管理トレース情報■ SUPPORT - オラクル社サポート・サービス・トレース情報
例:	trace_level_listener=admin

USE_PLUG_AND_PLAY_listener_name

目的:	リスナーに、データベース情報を検索して、Oracle Names Server に登録するよう指示します。 クライアントがサーバーに接続を要求する場合、Oracle Names Server はサービス名や SID などの必要な情報をリスナーから取得します。
デフォルト:	OFF
値:	ON OFF
例:	use_plug_and_play=on

Oracle Names パラメータ (NAMES.ORA)

Oracle Names 構成ファイル (NAMES.ORA) では、次のパラメータが使用できます。NAMES.ORA は、UNIX では \$ORACLE_HOME/network/admin、Windows NT では *ORACLE_HOME*\network\admin に置かれます。

NAMES.ADDRESSES

目的:	Oracle Names Server がリスニングするアドレスを指定します。有効な ADDRESS または ADDRESS_LIST を指定します。 追加情報: 次を参照してください。 <ul style="list-style-type: none">■ C-5 ページの「ADDRESS と ADDRESS_LIST」(アドレスの構成について)■ 6-76 ページの「プロトコル・アドレスの構成」(各プロトコルで使用する適切なプロトコルについて)
デフォルト:	names.addresses= (address=(protocol=tcp)(host=oranamesrvr0)(port=1575)

NAMES.ADMIN_REGION

目的： 管理リージョンのデータ・ソースを記述します。このパラメータを設定すると、データベースが情報のリポジトリとして定義されます。このパラメータを設定しなければ、Oracle Names Server 間でデータのレプリケーションがキャッシュされます。

デフォルト： NULL

例：

```
names.admin_region=  
  (region=(name=local_region)  
  (type=rosdb)  
  (userid=names)  
  (password=names)  
  (description=  
    (address=(protocol=tcp)  
    (host=nineva)(port=1575)  
  )  
)  
(connect_data=  
  (service_name=em)  
)  
)  
(version=34619392)  
(refresh=14400)  
(retry=600)  
(expire=259200)
```

NAMES.AUTHORITY_REQUIRED

目的： システム問合せに権限応答が必要かどうかを指定します。

デフォルト： FALSE

例： names.authority_required=true

NAMES.CONNECT_TIMEOUT

目的:	Oracle Names Server が、クライアントからの接続が完了するまでに待機する時間を制限します。
デフォルト:	3 秒
最小値:	1 秒
最大値:	600 秒
例:	<code>names.connect_timeout=8</code>

NAMES.AUTO_REFRESH_EXPIRE

目的:	Oracle Names Server が、NAMES.DOMAIN.HINTS から取得した他のリージョンのデータベース・サーバー・アドレスを、キャッシュに格納しておく時間を秒単位で指定します。ここで指定した時間が経過すると、Oracle Names Server は他のリージョンのデータベース・サーバーに問合せを発行して、アドレスをリフレッシュします。
デフォルト:	600 秒
指定可能な値:	60 ~ 1209600 秒
例:	<code>names.auto_refresh_expire=1200000</code>

NAMES.AUTO_REFRESH_RETRY

目的:	自動リフレッシュ問合せが障害に陥ったときに、Oracle Names Server が再試行する間隔を秒単位で指定します。
デフォルト:	180
最小値:	60
最大値:	3600
例:	<code>names.auto_refresh_retry=180</code>

NAMES.CACHE_CHECKPOINT_FILE

目的:	Oracle Names Server がチェックポイント・ファイルを書き込むファイルの名前とパスを指定します。
デフォルト:	UNIX では \$ORACLE_HOME/network/names/ckpcch.ora、 Windows NT では ORACLE_HOME¥network¥names¥ckpcch.ora
例:	names.cache_checkpoint_ file=c:¥oracle¥network¥names¥cacheck.ora

NAMES.CACHE_CHECKPOINT_INTERVAL

目的:	Oracle Names Server が、格納済みデータのチェックポイントを チェックポイント・ファイルに書き込む間隔を指定します。各 Oracle Names Server は、キャッシュに格納したデータを定期的に ファイルに書き込んで、起動時の障害に備えることができます。
デフォルト:	0 (使用禁止)
最小値:	10
最大値:	259200 (3 日)
例:	names.cache_checkpoint_interval=24

NAMES.CONFIG_CHECKPOINT_FILE

目的:	Oracle Names Server の構成設定のチェックポイントに使用する ファイルの名前とパスを指定します。
デフォルト:	UNIX では \$ORACLE_HOME/network/names/ckcfg.ora、 Windows NT では ORACLE_HOME¥network¥names¥ckcfg.ora
例:	names.config_checkpoint_ file=c:¥oracle¥network¥names¥configck.ora

NAMES.DEFAULT_FORWARDERS

目的: 問合せを送るときに使用する、その他の Oracle Names Server のアドレス・リストを指定します。

例:

```
names.default_forwarders= (forwarder_list=
                           (forwarder=(name=rootserv1.com)(address=(protocol=tc
                           (port=4200)(host=roothost))))
```

NAMES.DEFAULT_FORWARDERS_ONLY

目的: TRUE の場合は、デフォルトの送り先としてリストされている Oracle Names Server のみを対象に、Oracle Names Server が問合せを送ります。

デフォルト: FALSE

使用上の注意: FALSE に設定されている場合、デフォルトの送り先としてリストされている Oracle Names Server は、キャッシュに見つかっている Oracle Names Server の前にコールされます。

NAMES.DOMAIN_HINTS

目的: 1 つ以上の外部リージョンのサーバー名およびアドレス、ドメインをすべてリストします。Oracle Names Server がその他のリージョンにあるサーバーを認識できるようになります。この中には、少なくともルート・リージョンにない全サーバーのルート・リージョンが含まれます。最適化が必要になれば、その他のリージョンを指定できます。

例:

```
names.domain_hints=
(hint_desc=(hint_list=
            (hint=(name=rootserv1.com)
              (address=(protocol=tc
              (port=4200)
              )
            )
          )
        )
```

NAMES.DOMAINS

目的 : サーバーのローカル・リージョン内のドメインのリスト、およびこれらのドメイン内のデータに対するデフォルト生存時間 (TTL) を指定します。

例 : `names.domains=
(domain_list=
(domain=(name=)(min_ttl=86400))
(domain=(name=com)(min_ttl=8640))
)`

NAMES.FORWARDING_AVAILABLE

目的 : このパラメータを ON に設定すると、Oracle Names Server は操作を外部リージョンに転送します。

デフォルト : ON

値 : ON | OFF

例 : `names.forwarding_available=off`

NAMES.FORWARDING_DESIRED

目的 : このパラメータを TRUE に設定すると、Oracle Names Server は問合せを送ります。

デフォルト : TRUE

値 : TRUE | FALSE

例 : `names.forwarding_desired=true`

NAMES.KEEP_DB_OPEN

目的 : リージョン・データベースへの TNS 接続を、操作間でオープンしたままにするかどうかを指定します。このパラメータを FALSE に設定すると、ロード、再ロードまたは再ロードチェックが終わるたびに接続がクローズされます。

デフォルト : TRUE

値 : TRUE | FALSE

例 : `names.keep_db_open=FALSE`

NAMES.LOG_DIRECTORY

目的:	Oracle Names Server 操作イベントのログ・ファイルを書き込むディレクトリの名前を指定します。
デフォルト:	UNIX では \$ORACLE_HOME/network/log、Windows NT では ORACLE_HOME¥network¥log
例:	names.log_directory=c:¥oracle¥network¥names

NAMES.LOG_FILE

目的:	Oracle Names Server 操作イベントを書き込む出力ファイルの名前を指定します。ファイル名の拡張子として必ず .LOG が付くので、このパラメータで拡張子を入力しないでください。
デフォルト:	NAMES
例:	names.log_file=onames

NAMES.LOG_STATS_INTERVAL

目的:	ログ・ファイルに完全な統計をダンプする間隔を秒数で指定します。
デフォルト:	0 (0=OFF)
最小値:	10 (統計の入力を ON にする最小値)
最大値:	なし
例:	names.log_stats_interval=12

NAMES.LOG_UNIQUE

目的:	このパラメータを TRUE に設定すると、ログ・ファイル名が一意的な名前になるので、既存のログ・ファイルは上書きされません。
デフォルト:	FALSE
値:	TRUE FALSE
例:	names.log_unique=true

NAMES.MAX_OPEN_CONNECTIONS

目的:	Oracle Names Server が一度にオープンできる接続数を指定します。この値は 10 と指定するか、リスニング用に 1 接続、クライアント用に 5 接続、ローカル管理リージョンで定義されている各外部ドメイン用に 1 接続の合計値を指定します (10 または合計値の大きい方を指定します)。 計算で求めた値を、ほとんどのインスタレーションに適用できません。
デフォルト:	入力データに基づいて計算されます。
最小値:	2
最大値:	64
例:	<code>names.max_open_connections=52</code>

NAMES.MAX_REFORWARDS

目的:	サーバーが操作を転送する最大回数を指定します。
デフォルト:	2
最小値:	1
最大値:	15
例:	<code>names.max_reforwards=2</code>

NAMES.MESSAGE_POOL_START_SIZE

目的:	転送されたメッセージの着信または発信に使用するメッセージの初期数を、サーバーのメッセージ・プールに対して指定します。
デフォルト:	10
最小値:	3
最大値:	256
例:	<code>names.message_pool_start_size=10</code>

NAMES.NO_MODIFY_REQUESTS

目的:	TRUE に設定すると、サーバーはリージョン内のデータを変更する操作を拒否します。
デフォルト:	FALSE
値:	TRUE FALSE
例:	<code>names.no_modify_requests=true</code>

NAMES.NO_REGION_DATABASE

目的:	TRUE に設定すると、サーバーはリージョン・データベースを検索しません。
デフォルト:	FALSE
例:	<code>names.no_region_database=true</code>

NAMES.PASSWORD

目的:	STOP、RESTART および RELOAD など、NAMESCTL における特定の重要な操作を安全に行うために、Oracle Names Server にパスワードを設定します。 このパラメータを Oracle Net8 Assistant で設定すると、パスワードが暗号化されます。クリアテキスト・パスワードは手動で設定します。パスワードがクリアテキストの場合、SQLNET.ORA ファイルで設定される NAMESCTL.INTERNAL_ENCRYPT_PASSWORD パラメータは FALSE に設定されます。
デフォルト:	なし
例:	<code>names.password=625926683431aa55</code>

NAMES.REGION_CHECKPOINT_FILE

目的:	リージョンのデータ (ローカル・リージョン内の Oracle Names Server のドメイン・アドレス、データベース・アドレスなど) にチェックポイントを使用するためのファイルの名前とパスを指定します。
デフォルト:	UNIX では \$ORACLE_HOME/network/names/ckpreg.ora、Windows NT では ORACLE_HOME¥network¥names¥ckpreg.ora
例:	names.region_checkpoint_ file=c:¥oracle¥network¥names¥regionck.ora

NAMES.RESET_STATS_INTERVAL

目的:	Oracle Names Server が収集した統計を蓄積する時間を秒数で指定します。指定した頻度でゼロにリセットされます。デフォルト値の 0 では、統計はリセットされません。
デフォルト:	0 (リセットしない)
最小値:	10
最大値:	なし
例:	names.reset_stats_interval=15

NAMES.SAVE_CONFIG_ON_STOP

目的:	このパラメータを TRUE に設定すると、実行時の構成設定が NAMES.ORA に保存されます。NAMESCTL SET 操作で変更されたパラメータは、NAMES.ORA で設定する前に置き換えられます。
デフォルト:	FALSE
例:	names.save_config_on_stop=FALSE

NAMES.SERVER_NAME

目的:	各 Oracle Names Server を名前で一意に識別します。特定の Oracle Names Server を参照するときは、必ずこの名前を使用します。
デフォルト:	ONAMES_onames_server
例:	names.server_name=namesrv1.us.oracle.com

NAMES.TRACE_DIRECTORY

目的: Oracle Names Server トレース・セッションのトレース・ファイルを書き込むディレクトリの名前を指定します。

デフォルト: UNIX では \$ORACLE_HOME/network/trace、Windows NT では ORACLE_HOME¥network¥trace

例: names.trace_directory=/oracle/network/admin/trace

NAMES.TRACE_FILE

目的: Oracle Names Server トレース・セッションの出力ファイルの名前を指定します。ファイルの拡張子は常に .TRC です。

デフォルト: NAMES

例: names.trace_file=onames

NAMES.TRACE_FUNC

目的: 内部のメカニズムが関数名でトレースを制御できるようにします。

デフォルト: FALSE

例: names.trace_func=false

NAMES.TRACE_LEVEL

目的: Oracle Names Server のトレース・レベルを指定します。

デフォルト: OFF

値:

- OFF - トレース出力しない
- USER - ユーザー・トレース情報
- ADMIN - 管理トレース情報
- SUPPORT - オラクル社サポート・サービス・トレース情報

例: names.trace_level=admin

NAMES.TRACE_UNIQUE

目的:	各トレース・ファイルに一意的な名前を割り当て、複数のトレース・ファイルが存在できるかどうかを指定します。この値を ON に設定すると、プロセス識別子が各トレース・ファイルの名前に追加されます。
デフォルト:	ON
値:	ON OFF
例:	<code>names.trace_unique=on</code>

Oracle Connection Manager パラメータ (CMAN.ORA)

Oracle Connection Manager 構成ファイル (CMAN.ORA) では、次のパラメータが使用できません。CMAN.ORA は、UNIX では \$ORACLE_HOME/network/admin、Windows NT では `ORACLE_HOME\network\admin` に置かれます。

CMAN

目的: Oracle Connection Manager ゲートウェイ・プロセス **CMGW** のリスニング・アドレスを指定します。

デフォルト: `cmn=(address=(protocol=tcp)(host=local_host)(port=1630))`

構文: `cmn=[(address_list=]
(address= ...)
[(address= ...)]`

CMAN_ADMIN

目的: Oracle Connection Manager 管理プロセス **CMADMIN** のリスニング・アドレスを指定します。

デフォルト: `cmn_admin=(address=(protocol=tcp)(host=anyhost)(port=1830))`

構文: `[(address_list=]
(address= ...)
[(address= ...)]`

CMAN_PROFILE

目的 : Oracle Connection Manager の実行に関連するパラメータを設定パス名します。

デフォルト :

- ANSWER_TIMEOUT=0
- AUTHENTICATION_LEVEL=0
- LOG_LEVEL=0
- MAX_FREELIST_BUFFERS=2048
- MAXIMUM_CONNECT_DATA=1024
- MAXIMUM_RELAYS=128
- RELAY_STATISTICS=NO
- SHOW_TNS_INFO=NO
- UNIX では TRACE_DIRECTORY=\$ORACLE_HOME/network/trace、
Windows NT では TRACE_DIRECTORY=ORACLE_HOME¥network¥trace
- TRACING=NO
- USE_ASYNC_CALL=YES

値 :

- ANSWER_TIMEOUT=[0 ~ n]
- AUTHENTICATION_LEVEL=[0, 1]
- LOG_LEVEL=[0 ~ 4]
- MAXIMUM_CONNECT_DATA=[257 ~ 4096]
- MAX_FREELIST_BUFFERS=[0 ~ 10240]
- MAXIMUM_RELAYS=[0 ~ 2048]
- RELAY_STATISTICS=[YES, NO]
- SHOW_TNS_INFO=[YES, NO]
- TRACE_DIRECTORY=*pathname*
- TRACING=[YES, NO]
- USE_ASYNC_CALL=[YES, NO]

CMAN_PROFILE

```

例 :
      cman_profile =
      (parameter_list=
        (maximum_relays=512)
        (log_level=1)
        (tracing=yes)
        (relay_statistics=yes)
        (show_tns_info=yes)
        (use_async_call=yes)
        (authentication_level=0)
      )

```

CMAN_PROFILE パラメータ	説明
ANSWER_TIMEOUT	Oracle Connection Manager が、着信接続要求に関連するプロトコル・ハンドシェークをタイムアウトする際の秒数です。設定範囲は 0 ~ n です。
AUTHENTICATION_LEVEL	1 を指定すると、CMAN は安全性の高いネットワーク・サービスを使用しない接続要求を拒否するように指示されます。安全性の高いネットワーク・サービスは、Oracle Advanced Security の一部です。 デフォルトは 0 で、Secure Network Services を必要とさせません。
LOG_LEVEL	Oracle Connection Manager で実行されるロギングのレベルを決定します。 <ul style="list-style-type: none"> ■ デフォルトは 0 で、ロギングを実行しません。 ■ 設定値の範囲は 0 から 4 です。 <ul style="list-style-type: none"> レベル 1 - 基本的なレポート レベル 2 - RULE_LIST の検索レポート レベル 3 - リレー・ブロック・レポート レベル 4 - リレー I/O カウント・レポート CMGW ゲートウェイ・プロセスはログ・ファイル CMAN_PID.LOG を作成し、CMADMIN 管理プロセスはログ・ファイル CMADM_PID.LOG を作成します。ログ・ファイルは、UNIX では \$ORACLE_HOME/network/log、Windows NT では ORACLE_HOME¥network¥log に配置されます。
MAX_FREELIST_BUFFERS	リレーがクローズした後に、TNS がオペレーティング・システムに返さずにフリーリストに維持するバッファの最大値を決定します。

CMAN_PROFILE パラメータ	説明
MAXIMUM_CONNECT_DATA	着信接続要求の接続データ文字列の長さを制限します。 設定範囲は 257 ~ 4096 です。
MAXIMUM_RELAYS	Oracle Connection Manager がサポートする、同時接続の最大数を決定します。設定範囲は 0 ~ 2048 です。
RELAY_STATISTICS	YES を設定すると、CMGW プロセスはリレー I/O アクティビティに関する次の統計を維持します。 <ul style="list-style-type: none">■ IN バイト数■ OUT バイト数■ IN パケット数■ OUT パケット数 この情報は CMAN_PID.LOG ファイルに保存されます。
SHOW_TNS_INFO	YES を指定すると、CMGW プロセスは TNS イベントを CMAN_PID.LOG ファイルに記録します。
TRACING	YES を設定すると、Oracle Connection Manager でのトレースを使用可能にします。CMGW ゲートウェイ・プロセスはトレース・ファイル CMAN_PID.TRC を、CMADMIN 管理プロセスはトレース・ファイル CMADM_PID.TRC を、それぞれ作成します。
TRACE_DIRECTORY	トレース・ディレクトリを指定します。
USE_ASYNC_CALL	YES を設定すると、CMGW プロセスは Net8 接続の確立フェーズの応答またはコールに、すべて非同期ファンクションを使用します。 注意: CMGW はバンド外ブレークをサポートするので、サーバーに対してバンド外ブレークを送ります。

CMAN_RULES

目的:	Oracle Connection Manager のネットワーク・アクセス制御の一部の規則を設定します。
構文:	<pre> cman_rules= (rules_list= (rule=(src=hostname)(dst=hostname)(srv= service_name sid) (act=accept)) [(rule= ...)]) </pre>
値:	<ul style="list-style-type: none"> ■ SRC - セッション要求のソース・ホスト名または IP アドレス (ドット表記法) ■ DST - 接続先サーバーのホスト名または IP アドレス (ドット表記法) ■ SRV - データベース・サーバー SID ■ ACT - 前の特性を持つ着信要求を受け入れるか拒否する
使用上の注意:	<p>ホスト名のワイルド・カード文字は、'x' です。IP アドレス (d.d.d.d) の場合は、この d にワイルド・カード 'x' を付けることができます。</p> <p>注意: クライアントの接続文字列に SID と SERVICE_NAME の両方が含まれている場合、クライアントがアクセスを行うには、要求された名前の両方がそれぞれ規則で許可される必要があります。</p>
例:	<pre> cman_rules= (rule_list= (rule=(src=client1-pc)(dst=sales-pc)(srv=sales.com)(act=reject)) (rule=(src=144.25.23.45)(dst=144.25.187.200)(srv=db1)(act=accept)) (rule=(src=spcstn1)(dst=db2-pc)(srv=db2)(act=accept))) </pre>

プロトコル固有のパラメータ (PROTOCOL.ORA)

PROTOCOL.ORA 構成ファイル内の次のパラメータは、Net8 に使用できます。

protocol.EXCLUDED_NODES

目的:	どのノードに有効ノード・チェックがないかを定義します。
例:	<code>protocol.excluded_nodes= (hostname ip_address, hostname ip_address, ...)</code>
例:	<code>tcp.excluded_nodes= (hr.com, 144.25.5.25)</code>

protocol.INVITED_NODES

目的:	どのノードに有効ノード・チェックがあるかを定義します。 <i>protocol.EXCLUDED_NODES</i> とこのリストの両方がある場合は、このリストが優先されます。
構文:	<code>protocol.invited_nodes= (hostname ip_address, hostname ip_address, ...)</code>
例:	<code>tcp.invited_nodes= (sales.com, 144.185.5.73)</code>

protocol.VALIDNODE_CHECKING

目的:	ホスト権限を有効にした接続先へのクライアントの接続アクセスを制限します。
デフォルト:	NO
値:	YES NO
例:	<code>tcp.validnode_checking=yes</code>

TCP.NODELAY

目的:	パッファ・フラッシングで遅延なしを指定します。
デフォルト:	NO
値:	YES NO
例:	<code>tcp.nodelay=yes</code>

A

API

「[Net8 Open](#)」を参照。

ASCII キャラクタ・セット (ASCII character set)

American Standard Code for Information Interchange キャラクタ・セットの略。デジタル・データを使用して英数字情報を表現するための規則。IBM および IBM 互換コンピュータを除く大半のコンピュータで使用している照合順番。EBCDIC 文字セットと対比。

B

bequeath

リスナーとサーバーが同じノード上に存在する場合、リスナーは、接続要求の受信時に専用サーバーを作成または起動できる。専用サーバーは1つのネットワーク・セッションのみにコミットされ、そのネットワーク・セッションを実行している間存在する。リスナーが専用サーバー・プロセスを作成し、そのプロセスにネットワーク・セッションの制御を渡す、または bequeath するときに発生するイベントの順序。

C

CDS

Cell ディレクトリ・サービス。

CMAN.ORA ファイル

Oracle Connection Manager のリスナー・アドレス、Oracle Connection Manager パラメータ、および [Net8 アクセス制御 \(Net8 access control\)](#) 規則を指定する構成ファイル。

CONNECT DATA

接続記述子の一部。接続記述子は CONNECT DATA キーワードで始まり、接続先のサービスを指定する。

connect-time failover

最初のリスナーが応答しない場合、クライアントの接続要求は他のリスナーに転送される。リスナーは接続の前にインスタンスが起動されているかどうかを確認するため、connect-time failover はデータベース・インスタンス登録 (database instance registration) によって有効になる。

D

DNS

Domain Name Service。TNSNAMES.ORA ファイル (TNSNAMES.ORA file) に指定されたホスト名を、ホスト・インターネット・アドレス (IP) に変換するマシン。

G

General Inter-Orb Protocol (GIOP)

メッセージを提供する Java オプション接続で使用される、プレゼンテーション・レイヤー (presentation layer) の 1 つ。

I

IIOB クライアント (IIOB Clients)

General Inter-Orb Protocol (GIOP) を使用して Java オプションへアクセスするクライアント。次のようなクライアントを指す。

- Enterprise JavaBeans (EJB)
- CORBA Servers
- Java ストアード・プロシージャ

Inter-Orb Protocol (IIOB)

Java オプションへの接続用の、TCP/IP または SSL 付き TCP/IP 上の General Inter-Orb Protocol (GIOP) の実装。

J

Java Database Connectivity (JDBC) ドライバ (Java DataBase Connectivity Drivers)

Java プログラマに Oracle データベースへのアクセスを提供する JDBC ドライバ。

L

LISTENER.LOG ファイル (LISTENER.ORA file)

サーバー上の 1 つ以上のリスナーを記述する構成ファイル。リリース 8.1 までのデータベースでは、LISTENER.ORA ファイルで、リスナーによってサービスされるデータベースの **Oracle システム識別子 (Oracle System Identifier: SID)** を定義することもできる。リリース 8.1 のデータベースでは、**データベース・インスタンス登録 (database instance registration)** により、データベースの識別子を必要としない。

N

NDS

NetWare ディレクトリ・サービス (NDS)。

Net8

ネットワーク上でクライアント / サーバー通信とサーバー / サーバー通信の両方を可能にする Oracle のリモート・データ・アクセス・ソフトウェア。Net8 は分散処理と分散データベース機能をサポートする。Net8 は多数の通信プロトコルで動作し、それらを相互に接続する。Net8 は SQL*Net バージョン 2 との後方互換性を有する。

Net8 Configuration Assistant

特定のクライアントおよびサーバーの Net8 コンポーネントを、インストール後に構成するための Java ベースのツール。

Net8 Open

Net8 へのアプリケーション・プログラム・インタフェース (API)。これによって、プログラマは開発環境にすでに展開されている Net8 ネットワークを使用しながら、データベース・アプリケーションとデータベース以外のアプリケーションの両方を開発できる。Net8 Open は、すべての業界標準ネットワーク・プロトコルに対して単一の共通インタフェースをアプリケーションに提供する。

Net8 アクセス制御 (Net8 access control)

指定したサーバーに対するアクセスを、クライアントに許可または禁止するためのルールを設定する Oracle Connection Manager の機能。ファイアウォール・サポートとも呼ばれる。

NetWare

Novell が開発したネットワーク・オペレーティング・システム。

NI

ネットワーク・インタフェース。

NIS

NL

ネットワーク・ライブラリ。

NN

ネットワーク・ネーミング (Oracle Names)

NPI

ネットワーク・プログラム・インタフェース。

NR

ネットワーク・ルーティング。

NS

ネットワーク・セッション。

NT

ネットワーク・トランスポート。

O

OCI JDBC ドライバ (OCI JDBC Driver)

Oracle の Level 2 JDBC ドライバ。クライアント / サーバー型の Java アプリケーション・プログラム、および Java ベースの中間層開発者を対象とする。JDBC/OCI ドライバは、JDBC の呼び出しを Oracle Call Interface (OCI) の呼び出しに変換する。変換された呼び出しは、Net8 を通じて Oracle データベース・サーバーに送信される。

OPI

オープン・プログラム・インタフェース。

Oracle Connection Manager

接続集中化 (connection concentration)、複数プロトコル・サポート (multi-protocol support) および Net8 アクセス制御 (Net8 access control) を提供するソフトウェア・コンポーネント。

Oracle Connection Manager 制御ユーティリティ (Oracle Connection Manager Control Utility: CMCTL)

Net8 に含まれているユーティリティ。Oracle Connection Manager の起動および停止、状態の取得などのさまざまな機能を制御する。

ORACLE_HOME

ディレクトリベースのオペレーティング・システムで、Oracle ディレクトリ階層内の最高位ディレクトリに付けられた代替名。

Oracle Names

Oracle Names Server の 1 システムで構成されるディレクトリ・サービスで、ネットワーク上の各 Net8 サービスに対して名前からアドレスへの変換処理を提供する。

Oracle Names Server

クライアントが長々しいアドレスではなく単純な名前で接続を要求できるように、Oracle Names を使用してサービスのネットワーク・アドレスを単純な名前と一緒に格納するサーバー。

Oracle Names の基盤構造 (Oracle Names infrastructure)

名前の割り当て方および Oracle Names の動作を制御する最初の決定事項とポリシー。基盤構造は、ユーザーと管理者が Oracle Names システムと対話する方法を定義する。

Oracle Net8 Assistant

構成および管理を支援する、Java ベースのツール。

Profile - ローカル・マシンのローカル構成ファイル (SQLNET.ORA) を、ユーザーが作成または編集するためのツール。

Net Service Names - ローカル名ファイル (TNSNAMES.ORA) を、ユーザーが作成または編集するためのツール。

Listeners - ローカル・マシンのローカル構成ファイル (LISTENER.ORA) を、ユーザーが作成または編集するためのツール。

Oracle Names Servers - ユーザーがローカル・マシンから Oracle Names Server を構成および管理したり、リモート・マシンの Oracle Names Server を管理するためのツール。

Oracle Rdb

Digital 社の 64 ビット・プラットフォーム用データベース。Net8 を利用すると、Rdb サーバーは Oracle データベースへクライアントがアクセスすると同様に扱われる。Oracle Rdb は独自のリスナーを使用するため、クライアントは Oracle7 の場合と同じ方法で Rdb と対話できる。

Oracle システム識別子 (Oracle System Identifier: SID)

実行中の Oracle8i ソフトウェアの特定のインスタンスを識別する名前。どんなデータベースでも、データベースを参照しているインスタンスが少なくとも 1 つは存在する。

8.1 より古いリリースでは、SID はデータベースの識別に使用されている。SID は、[TNSNAMES.ORA ファイル \(TNSNAMES.ORA file\)](#) ファイルの接続記述子の CONNECT DATA 部、および [LISTENER.LOG ファイル \(LISTENER.ORA file\)](#) ファイルのネットワーク・リスナーの定義に含まれる。デフォルト構成を選択すると、SID はデフォルトで "*" となる。

Oracle プロトコル (Oracle protocol)

ネットワーク上でのデータ転送方法を定義する一連の規則。TCP/IP や SPX など、いくつかの業界標準トランスポート・プロトコルがある。

R

RDBMS

リレーショナル・データベース管理システム。

RPC

リモート・プロシージャ・コール。

S

SID

「[Oracle システム識別子 \(Oracle System Identifier: SID\)](#)」を参照。

SID_LIST_listener_name

リスナーでサービスされるデータベースの [Oracle システム識別子 \(Oracle System Identifier: SID\)](#) を定義する、[LISTENER.LOG ファイル \(LISTENER.ORA file\)](#) のセクション。現在は、リリース 8.1 インスタンス情報が自動的にリスナーに登録されるため、このセクションはバージョン 7.x およびバージョン 8.0 の Oracle データベースのみで有効。外部プロシージャや異種間サービスなどの他のサービス、および Oracle Enterprise Manager バージョン 2 を含む管理ツールでは、静的な構成も必要となる。

SPX

Sequenced Packet Exchange。多くの主要ネットワーク管理システム（特に Novell Advanced NetWare）に適合する高性能ネットワーク・プロトコル。

SQL*Net

Net8 の疑似カーソル。Oracle サーバーと協力して、Oracle RDBMS または Oracle Tools (SQL*Forms など) を走らせる複数のコンピュータが、ネットワークを介してデータを交換できるようにする Oracle 製品。SQL*Net は分散処理と分散データベース機能をサポートする。SQL*Net は多くの通信プロトコルで動作し、それらを相互に接続する。

T

Thin JDBC ドライバ (Thin JDBC Driver)

Thin JDBC ドライバは、Oracle の Type 4 ドライバで、Java アプレットおよび Java アプリケーションの開発者用に設計されている。JDBC ドライバは、Java ソケットを通して Oracle データベース・サーバーに直接接続を確立する。データベースへのアクセスは、Net8 と [Two-Task Common \(TTC \)](#) の実装に支援される。

TNS

「[Transparent Network Substrate \(TNS \)](#)」を参照。

TNSNAMES.ORA ファイル (TNSNAMES.ORA file)

サービス名にマップされた接続記述子が入っているファイル。このファイルは、すべてのクライアントまたは個々のクライアントが使用できるように、集中的またはローカルに管理される。

Transparent Network Substrate (TNS)

Net8、Oracle Connection Manager および Oracle Names に組み込まれた、標準のネットワーク・トランスポート・プロトコルで機能する基本的なテクノロジー。

Two-Task Common (TTC)

TTC は、クライアントとサーバーの異なるキャラクタ・セットや形式に対して、キャラクタ・セットとデータ型の変換を提供するために、標準的な Net8 接続で使用される [プレゼンテーション・レイヤー \(presentation layer \)](#)。

U

UPI

ユーザー・プログラム・インタフェース。

あ

アクセス制御 (access control)

「[Net8 アクセス制御 \(Net8 access control \)](#)」を参照。

アドレス (address)

リスナー・アドレス、Oracle Connection Manager、Oracle Names Server などのネットワーク・オブジェクトを、一意に識別するためのネットワーク上の場所。アドレスには特定のフォーマットがあり、一意でなければならない。「[予約済み Oracle Names Server \(well-known Oracle Names Server \)](#)」も参照。

い

異種間サービス (heterogeneous services)

Oracle サーバーから非 Oracle サーバーをアクセスするための基本的な技術を提供する、統合されたコンポーネント。異種間サービスでは次のことが可能となる。

- Oracle 以外のシステムに格納されたデータに対して、Oracle サーバーに存在するデータと同じように、Oracle SQL を使用して透過的にアクセスできる。
- Oracle 分散環境から、Oracle プロシージャ・コールを使用して、Oracle 以外のシステム、サービスまたはアプリケーション・プログラムインタフェース (API) に透過的にアクセスできる。

位置の透過性 (location transparency)

データ表の置かれている位置がわからなくても、アプリケーションがそのデータ表にアクセスできるようにする分散データベースの特性。すべてのデータ・テーブルは1台のデータベース内にあるものと見なされ、システムは表名に基づいてデータの実際の位置を判別する。ユーザーは1つの文で複数のノード上のデータを参照できる。システムは、必要に応じて、SQL 文 (の一部) を自動的かつ透過的にリモート・ノードに経路指定して実行する。ユーザーまたはアプリケーションに影響を与えないで、データをノード間で移動できる。

委任管理 (delegated administration)

ルート管理リージョン (root administrative region) の下にある1つ以上の管理リージョンにネットワーク管理を委任する Net8 ネットワーク。分散管理または非集中管理とも呼ばれる。集中管理 (central administration) と対比。

委任管理リージョン (delegated administrative region)

階層的にルート管理リージョン (root administrative region) の下にあるリージョン。ルート管理リージョン以外のすべてのリージョン。

インスタンス登録 (instance registration)

「データベース・インスタンス登録 (database instance registration)」を参照。

え

エラー・メッセージ (error message)

プログラムまたはコマンドの実行を妨げる問題または状態を知らせるコンピュータ・プログラムのメッセージ。

か

階層ネーミング・モデル (hierarchical naming model)

階層的に関連付けられた複数のドメインに名前を分割する Oracle Names の基盤構造。集中管理または委任管理のどちらでも、階層ネーミング・モデルを使用できる。

外部ドメイン (foreign domains)

所定の管理リージョン内で管理されないドメイン群。外部ドメインとリージョンの関係は相対的なもので、絶対的な意味の外部ドメインは存在しない。ネットワーク管理者は、一般的にネーム・サーバーのキャッシュ・パフォーマンスを最適化するために特定のリージョンに対する外部ドメインを定義する。

外部ネーミング (external naming)

NIS や NDS などの、サポートされているサードパーティ製のネーミング・サービスを使用する、[ネット・サービス名 \(net service name\)](#) 変換。

外部プロシージャ (external procedure)

PL/SQL コードから呼び出せる、第三代言語 (3GL) で記述された関数またはプロシージャ。外部プロシージャとして、C のみがサポートされている。

管理リージョン (administrative region)

Net8 ネットワーク・コンポーネントを管理するための構造エンティティ。各管理リージョンには次の要素が含まれる。

- 1つ以上のドメイン
- 1つ以上の Oracle Names Server
- 1つ以上のデータベースおよびリスナー

き

キーワード値ペア (keyword-value pair)

接続記述子および多くの構成ファイルで標準的な情報単位として使用する、キーワードと値の組み合わせ。キーワードと値の対はネストすることができる。つまり、キーワードは別のキーワードと値の対を値としてもつことができる。

キャッシュ (cache)

前にアクセスされたデータが再び要求されたときに、その要求を素早く処理できるようにデータを格納しておくメモリー。

共有サーバー (shared server)

「[マルチスレッド・サーバー \(multi-threaded server: MTS\)](#)」を参照。

均一ネーミング・モデル (flat naming model)

1つのドメインのみが存在する Oracle Names 基盤構造。このドメイン内では、すべての名前が一意でなければならない。

クライアント (client)

別のアプリケーションまたはコンピュータのサービス、データ、処理を要求するユーザー、ソフトウェア・アプリケーションまたはコンピュータ。2タスク環境では、クライアントはユーザー・プロセスとなる。ネットワーク環境では、クライアントはローカル・ユーザー・プロセスで、サーバーはローカルまたはリモートとなる。

クライアント/サーバー・アーキテクチャ (client-server architecture)

2台のCPU間で処理を分割するソフトウェア・アーキテクチャ。1台のCPUは、トランザクションでサービスを要求して受け取るクライアントとして機能し、もう1台はトランザクションでサービスを提供するサーバーとして機能する。

クライアント負荷のバランス (client load balancing)

ロード・バランシング。複数のリスナーが1つのデータベースをサービスする場合、クライアントは接続要求を送るリスナーをランダムに選択できる。このランダム化によって、すべてのリスナーが着信接続要求をサービスする負荷を共有できる。

クライアント・プロファイル (client profile)

クライアントのプロパティ。ネーミング・メソッド (naming methods) の優先順位、クライアントとサーバーのロギング (logging) とトレース (tracing) 名前を要求するドメイン、および Oracle Names および Oracle Advanced Networking Option に対するその他のクライアント・オプションが含まれる。

グローバル・データベース・リンク (global database link)

ネットワーク内の各データベースを他のすべてのデータベースとリンクするデータベース・リンク。このリンクによって、ネットワーク内のどのデータベースを使用するユーザーも、SQL文またはオブジェクト定義でグローバル・オブジェクト名を指定することができる。(Debunkのグローバル・オブジェクト名は、データベース・サービス名と同じでなければならない。)

「データベース・リンク (database link)」¹、「プライベート・データベース・リンク (private database link)」および「パブリック・データベース・リンク (public database link)」も参照。

グローバル・データベース名 (global database name)

他のデータベースから、データベースを一意に識別するための完全な名前。グローバル・データベース名の形式は、*database_name.database_domain*。たとえば、次のように指定する。

SALES.US.ORACLE.COM

データベース名の SALES は、データベースを呼ぶ際の単純な名前を表す。データベース・ドメインの .US.ORACLE.COM は、データベースが配置されているデータベース・ドメインを表し、グローバル・データベース名を一意に識別する。可能であれば、データベース・ドメインにネットワーク・ドメインを反映させることをお勧めする。

グローバル・データベース名は、データベースのデフォルト・サービス名で、[初期化ファイル \(initialization file: INITSID.ORA\)](#) ファイルの SERVICE_NAMES パラメータで指定される。

リ

構成ファイル (configuration files)

ネットワークのコンポーネントを識別および特徴付けるためのファイル。構成のほとんどは、ネットワーク・コンポーネントをネーミングし、コンポーネント間の関係を識別するプロセス。

れ

サービス名 (service name)

データベースの論理表現。データベースをクライアントに提供する方法。データベースは複数のサービスとして提供できる。また、サービスは複数のデータベース・インスタンスとしてインプリメントできる。サービス名は、インストール中またはデータベースの作成中に入力された[グローバル・データベース名 \(global database name\)](#) すなわちデータベース名とドメイン名からなる名前を表す文字列となる。グローバル・データベース名が確かでない場合は、データベース初期化ファイルの INITSID.ORA の SERVICE_NAMES パラメータの値から取得できる。

サービス名は、[TNSNAMES.ORA ファイル \(TNSNAMES.ORA file\)](#) 中の接続記述子の CONNECT DATA 部分に含まれる。

サービス・レプリケーション (service replication)

ネットワーク上でディレクトリ・システムを完全にレプリケートするプロセス。新しいサービスは1つのネーム・サーバーのみに登録する必要がある。新しく登録されたサービスは、サービス・レプリケーション・プロセスによって、ネットワーク上のすべてのアクティブ・ネーム・サーバーに自動的にレプリケートされる。

し

システムまたはトポロジー・データ (system or topology data)

Oracle Names Server が通常の機能を制御したり、他の Oracle Names Server と通信する際に使用するデータ。交換およびルート・リージョンの Oracle Names Server、委任リージョンの Oracle Names Server などを含む。

事前に開始または事前に生成される専用サーバー・プロセス (prestarted or prespawnded dedicated server process)

事前に生成される専用サーバー・プロセスは、着信接続要求前に Net8 リスナーで事前に開始される。所定のマシンでマルチスレッド・サーバーを使用していないかサポートしていない場合は、事前に生成された専用サーバー・プロセスによってサーバー上での接続確立時間を短縮できる。また、他の接続でサーバーの切断および再作成するときにサーバー・プロセスを再利用することによって、割当て済みのメモリー・システム・リソースを有効に利用する。

集中管理 (central administration)

ネットワーク管理が、ネットワーク全体に対して 1 つの管理リージョン (administrative region) で構成されている Oracle Names ネットワーク。集中管理では、すべてのネーム・サーバーがネットワーク内の他のネーム・サーバーおよびすべてのサービスを認識する。委任管理 (delegated administration) と対比。

初期化ファイル (initialization file: INITSID.ORA)

データベースとインスタンスを初期化する情報を含むファイル。

せ

セッション・レイヤー (session layer)

セッション・レイヤーは、プレゼンテーション・レイヤー (presentation layer) のエンティティに必要なサービスを提供する。エンティティでは、対話を構成および同期することができ、データ交換の管理が有効となる。このレイヤーは、クライアントとサーバー間でネットワーク・セッションを確立、管理および終了する。セッション・レイヤーの例には、ネットワーク・セッション (Network Session: NS) がある。

接続 (connection)

ネットワーク上の 2 つのプロセス間で行われる対話。接続は、宛先 (サーバー) との接続を要求する起動側 (クライアント) によって開始される。

接続記述子 (connect descriptor)

TNSNAMES.ORA ファイル (TNSNAMES.ORA file) で指定する、ネットワーク接続の宛先を示す特殊なフォーマットの記述子。接続記述子には次のものが含まれる。

- プロトコル・アドレスによるリスナーの場所を含む、サービスへのネットワーク・ルート
- リリース 8.1 サービスを識別するためのサービス名 (service name) またはリリース 8.1 より古いデータベースを識別するための Oracle システム識別子 (Oracle System Identifier: SID)
- 接続するリリース 8.1 サービスのオプション・インスタンス名

クライアントとサーバーは、このネット・サービス名を使用してアプリケーションに接続する。

接続先 (destination)

接続の終点であるクライアント。接続の起動側は、データまたはサービスを接続先に要求する。

接続集中化 (connection concentration)

Oracle Connection Manager の機能。サーバーのリソースを節約するために、クライアントから送られる複数の接続要求を統合して、サーバーと単一接続を確立すること。

接続プーリング (connection pooling)

マルチスレッド・サーバーに対する物理ネットワーク接続数を最大にすることのできる、リソース利用およびユーザー拡張機能。Oracle Connection Manager の機能で、サーバーのリソースを節約するために、クライアントから送られる複数の接続要求を統合して、サーバーと単一接続を確立する。

接続負荷のバランス (connection load balancing)

ロード・バランシング。同じサービスに対するさまざまなインスタンスやディスパッチャ間のアクティブな接続を、均等に分散する。これにより、リスナーはディスパッチャの接続数およびインスタンスが起動しているノードの負荷に基づいて、ルーティングを決定できる。

接続文字列 (connect string)

ユーザー名、パスワードおよびネット・サービス名 ([net service name](#)) など、ユーザーが接続するサービスに渡す情報。たとえば、次のように指定する。

```
connect username/password@net_service_name
```

接続要求 (connection request)

起動側が接続の開始を要求する際にリスナーに送信する通知。

専用サーバー (dedicated server)

各ユーザー・プロセスに対して専用のサーバー・プロセスを必要とするサーバー。[マルチスレッド・サーバー \(multi-threaded server: MTS \)](#) と対比。Net8 は、既存のサーバー・プロセスのアドレスをクライアントに送り返す。次に、クライアントは指定されたサーバー・アドレスに接続要求を再送する。

た

多重化 (multiplexing)

オペレーティング・システムのリソースを節約するために、複数のセッションを結合して 1 つのトランスポート接続で転送すること。「[接続集中化 \(connection concentration \)](#)」も参照。

ディスパッチャ (dispatcher)

ディスパッチャは、クライアントごとに専用サーバーを必要とせずに、多数のクライアントが同じサーバーに接続できるようにする。ディスパッチャは、複数の着信ネットワーク・セッション要求を処理して共有サーバー (shared server) に送る。

データ・パケット (data packet)

「パケット (packet)」を参照。

データベース・インスタンス登録 (database instance registration)

データベース・インスタンスは、起動時に自身をリスナーに登録する。データベース・インスタンス登録は、次の2つの要素からなる。

- サービス登録。リスナーにデータベース・サービス名やインスタンス名などのインスタンス情報を提供する。
- MTS ディスパッチャの登録。リスナーにディスパッチャ情報を提供する。

データベース管理者 (database administrator: DBA)

(1) Oracle Server またはデータベース・アプリケーションを操作および管理する人。(2) DBA 権限を所有し、データベース管理操作を実行できる Oracle ユーザー名。通常、この2つは同じ人物を表す。多くのサイトでは複数の DBA が配置される。

データベース初期化ファイル (database initialization file)

「初期化ファイル (initialization file: INITSID.ORA)」を参照。

データベース・リンク (database link)

リモート・データベース、およびそのデータベースに至る通信パス、オプションでユーザー名とパスワードを識別するために、ローカル・データベースまたはネットワーク定義に格納されているネットワーク・オブジェクト。定義されると、リモート・データベースへのアクセスにはデータベース・リンクが使用される。

あるデータベース・リンクから別のデータベースへのパブリックまたはプライベート・データベース・リンクは、DBA またはユーザーによってローカル・データベース上に作成される。

グローバル・データベース・リンクは、Oracle Names でネットワーク内の各データベースから他のすべてのデータベースに自動的に作成される。グローバル・データベース・リンクはネットワーク定義に格納される。

「グローバル・データベース・リンク (global database link)」、 「プライベート・データベース・リンク (private database link)」 および 「パブリック・データベース・リンク (public database link)」を参照。

デフォルト・ドメイン (default domain)

大半のクライアント要求が行なわれる **ドメイン (domain)**。クライアントが常駐しているドメイン、またはクライアントがネットワーク・サービスを頻繁に要求するドメイン。デフォルト・ドメインは、未修飾ネットワーク名の要求に追加すべきドメインを決定するクライアント構成パラメータでもある。名前の要求に文字 "." が入っていない場合は、未修飾の要求。

と

ドメイン (domain)

ネットワーク・サービスのネーミングを簡略化するために、データベースなどのネットワーク・オブジェクトをグループ化したもの。ドメイン内では、すべての名前が一意でなければならない。

ドメスティック・ドメイン (domestic domains)

所定の管理リージョン内で管理されるドメイン群。ドメスティック・ドメインとリージョンの関係は相対的なもので、絶対的な意味のドメスティック・ドメインはない。ローカル・ドメインとも呼ばれる。

トレース (tracing)

操作に関する詳細情報を出力ファイルに書き込む機能。トレース機能によって、操作が実行されるときイベントを示す詳しい文が生成される。管理者はトレース機能を使用して、異常な状態を診断する。通常、トレース機能はオンになっていない。

「[ロギング \(logging \)](#)」も参照。

ね

ネーミング・メソッド (naming methods)

データベース・サービスに接続しようとするときに、クライアント・アプリケーションがネット・サービス名をネットワーク・アドレスに変換するために使用する方法。Net8 で提供されるネーミング・メソッドは次の4種類。

- [ホスト・ネーミング \(host naming \)](#)
- [ローカル・ネーミング \(local naming \)](#)
- Oracle Names
- [外部ネーミング \(external naming \)](#)

ネーミング・モデル (naming model)

名前を割り当てることができる一連のドメインと構造。

均一ネーミング・モデルでは単一のドメイン。

階層ネーミング・モデルでは、最高位のドメインがルート・ドメインで、その他のドメインはすべて階層的に関連付けられる。

ネット・サービス名 (net service name)

使いやすく覚えやすい接続記述子の名前。エンドユーザーは、適切なサービス名を知っているのみで接続を確立できる。それぞれの接続記述子に、ネットワーク定義内のサービス名が割り当てられる。次の識別を行う。

- プロトコル・アドレスによるリスナーの場所を含む、サービスへのネットワーク・ルート
- リリース 8.1 サービスを識別するための**サービス名 (service name)**、またはリリース 8.1 より前のデータベースを識別するための**Oracle システム識別子 (Oracle System Identifier: SID)**
- 接続するリリース 8.1 サービスのオプション・インスタンス名

ネットワーク (network)

ハードウェアとソフトウェアによってリンクされ、データまたは周辺装置（あるいは、その両方）を共有できる複数のコンピュータで構成されているグループ。

ネットワーク・オブジェクト (network object)

ネットワーク上でアドレスを直接指定できるサービス。リスナーや Oracle Names Server など。

ネットワーク管理者 (network administrator)

ネットワーク・コンポーネントのインストール、構成、テストといったネットワーク管理作業を実行する人。ネットワーク管理者は、構成ファイル、接続記述子、サービス名、別名、パブリック・データベース・リンク、グローバル・データベース・リンクなどを管理する。

ネットワーク・キャラクタ・セット (network character set)

Oracle の定義では、キーワードと値の対（つまり、接続記述子と構成ファイル）で値として使用できる文字のセット。文字セットには、大文字と小文字の英数字およびいくつかの特殊文字が含まれる。

ネットワーク・サービス (network service)

Oracle Application Network では、サービス利用者に対していくつかのタスクが実行される。たとえば、ネーム・サーバーは名前解決サービスをクライアントに提供する。

ネットワーク・サービス名 (network service name)

サービス名のネットワーク設定。ネットワーク設定は、リスナーなどの任意の数の Net8 コンポーネントを含む、サービスへの 1 つ以上のネットワーク・ルートを指定する。ネット・サービス名は、ネットワーク接続が複数のリスナー間で均一にロードされるかどうか、ネッ

トワーク接続を確立できないときに代替リスナーにフェイルオーバーされるかどうかも指定する。

ネットワーク・セッション (Network Session: NS)

クライアント・アプリケーションとサーバー間で接続を確立および維持するために、標準的な Net8 接続で使用される **セッション・レイヤー (session layer)**。

ネットワーク・プロトコル (network protocol)

「**Oracle プロトコル (Oracle protocol)**」を参照。

ネットワーク・リスナー (network listener)

「**リスナー (listener)**」を参照。

の

ノード (node)

ネットワークの一部を構成するコンピュータまたは端末。

は

パケット (packet)

接続またはデータ転送が要求されるたびに、ネットワーク上で送信される情報のブロック。パケットに含まれる情報は、パケットの種類（接続、受入、リダイレクト、データなど）によって異なる。パケット情報は障害追跡時に役立つ。

パスワード (password)

データ・セキュリティに使用し、所有者のみが認識できる文字列（単語または句）。オペレーティング・システムまたはソフトウェア・アプリケーション（Oracle データベースなど）と接続する場合に、オペレーティング・システム・ログイン ID または Oracle ユーザー名、アカウント名などと一緒にパスワードを入力する。ユーザー名または ID は公開されているが、秘密のパスワードによって、ユーザー名の所有者のみがその名前を使用したり、そのデータにアクセスすることができる。

パブリック・データベース・リンク (public database link)

すべてのユーザーがアクセスできるローカル・データベース上で、DBA が作成するデータベース・リンク。

「**データベース・リンク (database link)**」、**「グローバル・データベース・リンク (global database link)**」および「**プライベート・データベース・リンク (private database link)**」も参照。

パラメータ (parameter)

プログラムまたはコマンド、関数に渡す情報。ファイル指定、キーワード、定数値などがある。

ひ

非集中管理 (decentral administration)

「[委任管理 \(delegated administration\)](#)」を参照。

ふ

ファイアウォール・サポート (firewall support)

「[Net8 アクセス制御 \(Net8 access control\)](#)」を参照。

フェイルオーバー (failover)

「[connect-time failover](#)」を参照。

複数プロトコル・サポート (multi-protocol support)

異なるネットワーク・プロトコルをもつクライアントとサーバーが相互に通信できるようにする、Oracle Connection Manager の機能。SQL*Net バージョン 2 Oracle MultiProtocol Interchange ですでに実現している機能を置換したもの。

プライベート・データベース・リンク (private database link)

あるユーザーが専用に使用するデータベース・リンク。

「[データベース・リンク \(database link\)](#)」、[「グローバル・データベース・リンク \(global database link\)」](#) および [「パブリック・データベース・リンク \(public database link\)」](#) も参照。

プレゼンテーション・レイヤー (presentation layer)

通信中にアプリケーション・レイヤー・エンティティが通信または参照する情報の表現を管理する。セッション・レイヤーの例は、[Two-Task Common \(TTC\)](#) と [General Inter-Orb Protocol \(GIOP\)](#)。

プロトコル・スタック (protocol stack)

特定の [プレゼンテーション・レイヤー \(presentation layer\)](#) と [セッション・レイヤー \(session layer\)](#) の組合せを示す。

分散管理 (distributed administration)

「[委任管理 \(delegated administration\)](#)」を参照。

分散処理 (distributed processing)

フロントエンド処理とバックエンド処理を別々のコンピュータで行うこと。Net8 は、リモート・データベースに透過的に接続することにより分散処理をサポートする。

へ

別名 (alias)

既存のネットワーク・オブジェクトに対する代替名。別名は最初のネットワーク・オブジェクトと同じ名前に変換される。Oracle Names Server には、定義済みのネット・サービス名、データベース・サーバーまたはデータベース・リンクの別名が格納される。

ほ

ホスト・ネーミング (host naming)

ネット・サービス名 (net service name) 解決により、TCP/IP 環境内のユーザーは既存のネーム変換サービスを通して、ネット・サービス名を変換できる。このネーム変換サービスには、Domain Name Service (DNS) や NIS、中央でメンテナンスされる /etc/hosts ファイルのセットの場合などがある。ホスト・ネーミングでは、サーバーのホスト名またはホスト名の別名を使用するのみで、ユーザーは Oracle サーバーに接続できる。クライアント構成で、この機能を利用する必要はない。この方法は単純な TCP/IP 環境にお薦めする。

ま

マルチスレッド・サーバー (multi-threaded server: MTS)

より多くのユーザーをサポートするために、多数のユーザー・プロセスがごくわずかなサーバー・プロセスを共有できるように構成されたサーバー。MTS を使用しなければ、各ユーザー・プロセスはそれぞれ専用サーバー (dedicated server) プロセスを必要とする。接続を要求する各クライアントに対して、新しいサーバー・プロセスが作成される。MTS (または共有サーバー (shared server)) の構成により、多数のユーザー・プロセスがディスパッチャ (dispatcher) に接続できる。

ゆ

ユーザー名 (username)

ユーザーを Oracle サーバーおよび他のユーザーが識別するための名前。すべてのユーザー名にパスワードが関連付けられており、Oracle データベースに接続するときはユーザー名とパスワードを両方入力する必要がある。

優先 Oracle Names Server (preferred Oracle Names Server)

名前を解決するときにクライアントが優先的に使用する Oracle Names Server。通常は、クライアントと物理的に最も近い Oracle Names Server、または最も安価なネットワーク・リンクで利用できる Oracle Names Server。

よ

予約済み Oracle Names Server (well-known Oracle Names Server)

1 つ以上の Oracle Names Server のアドレスが、Oracle Names Server とそのクライアントの両方にハードコード化される。予約済みアドレスをもつ Oracle Names Server にはそのアドレスでアクセスできるので、クライアントに構成ファイルでサーバーを探すように指示する必要がない。

り

リージョン・データベース (region database)

Oracle Names 情報を格納する Oracle データベースの表。

リージョン (region)

「[管理リージョン \(administrative region \)](#)」を参照。

リスナー (listener)

着信クライアント接続要求をリスニングし、サーバーへのトラフィックを管理する、サーバー上の個別のプロセス。

リスナーは、クライアント要求を受け取ってサーバーに渡す。クライアント（またはクライアントとして動作するサーバー）はネットワーク・セッションをサーバーに要求し、リスナーが実際の要求を受け取る。クライアントの情報がリスナーの情報と一致すると、リスナーがサーバーへの接続を許可する。

リスナー制御ユーティリティ (Listener Control Utility: LSNRCTL)

Net8 に含まれているユーティリティ。リスナーの起動および停止、状態の取得などのさまざまな機能を制御する。

る

ルート・ドメイン (root domain)

階層ネーミング・モデル内で最高位のドメイン。

ろ

ルート管理リージョン (root administrative region)

分散環境内で最高位の管理リージョン。ルート管理リージョンはルート・ドメインを含む。

ローカル・ネーミング (local naming)

ネット・サービス名 (net service name) 変換。各クライアントの `TNSNAMES.ORA` ファイル (`TNSNAMES.ORA file`) に構成および格納されている情報を使用して、ネットワーク・アドレスを検索する。ローカル・ネーミングは、ほとんど変更がなく、サービス数の少ない単純な分散ネットワークに最も適する。

ロード・バランシング (load balancing)

クライアント接続を複数のリスナー、ディスパッチャ、インスタンスおよびノードに均等に分散し、1つのコンポーネントに負荷がかかりすぎないようにする機能。

ロード・バランシングは、次のレベルで行われる。

- クライアント負荷のバランス (client load balancing)
- 接続負荷のバランス (connection load balancing)

ロギング (logging)

エラーおよびサービス・アクティビティ、統計をログ・ファイルに書き込む機能。ログ・ファイルは、画面に表示されるエラー・メッセージで障害を識別できない場合に、管理者用の追加情報として使用できる。ログ・ファイルでは、さまざまなレイヤーのソフトウェアの状態がエラー・スタックを通して示される。

「[トレース \(tracing \)](#)」も参照。

索引

記号

- " 記号
 予約済み、構成ファイル, C-4
- # 記号
 予約済み、構成ファイル, C-4
- ' 記号
 予約済み、構成ファイル, C-4
- () 記号
 予約済み、構成ファイル, C-4
- = 記号
 予約済み、構成ファイル, C-4
- | (スラッシュ) 記号
 予約済み、構成ファイル, C-4

数字

- 1521 ポート, 4-24, 6-16, 6-27, 6-66, 6-80
- 1521 ポート、デフォルト以外のリスナー・ポートを使用する場合, 4-24, 6-67
- 1575 ポート, 2-45, 6-32, 6-80, C-65
 Oracle Names Server, 2-45
- 1630 ポート, 2-46, 4-15, 6-80, 7-30, 7-34, 7-37, B-9, C-77
- 1646 ポート, C-26
- 1830 ポート, 4-15, 4-20, 5-21, 6-80, 7-30, 7-34
- 2481 ポート, 6-80, 7-56, 7-57
- 2481 ポート、デフォルト以外の ~ を使用する場合, 4-24, 7-57
- 2482、デフォルト以外の ~ を使用する場合, 4-24, 7-57
- 2482 ポート, 6-80, 7-56, 7-57
- 2483 ポート, 6-80

A

- ACTIVE_DELAYS, A-90
- ACT パラメータ, 7-38, C-81
- ADDRESS_LIST パラメータ, 4-11, 6-77, C-6, C-39, C-53
- ADDRESS (ADD または ADDR) 属性, 7-61, 9-4
- ADDRESS パラメータ, 6-12, 6-76, 7-53, C-5, C-39, C-52
- Advanced Program-to-Program Communication (APPC), 4-29
- 「Always Use Dedicated Server」オプション, 7-25
- ANSWER_TIMEOUT パラメータ, C-79
- APPC アーキテクチャ, 4-29
- ARGS パラメータ, 6-78
- ARGV0 パラメータ, 6-78
- A.SMD レコード, 8-18, A-44
- 「Attempts Per Names Server」オプション, 6-39
- AUTHENTICATION_LEVEL パラメータ, C-79
- AUTOMATIC_IPC パラメータ, 4-21, 5-19, 7-25

B

- BACKUP パラメータ, C-44
- bequeath
 - アドレスのパラメータ, 6-78
 - イベント・プログラミング, 10-14
 - 接続, 8-8
 - 説明, 4-30
- BEQUEATH_DETACH パラメータ, 7-23, C-7, 10-15
- bequeathed セッション, 2-25

C

- CDS 値, 7-19

Cell ディレクトリ・サービス (CDS), 3-6
CHANGE_PASSWORD コマンド, A-4
CKCFG.ORA ファイル, 6-25
CKPCCH.ORA ファイル, 6-25
CKPREG.ORA ファイル, 6-25
「Client Registration ID」オプション, 7-22
CMADM_PID.LOG ファイルの内容, 11-18, 11-27
CMADM_PID.TRC ファイルの内容, 11-32
CMADMIN プロセス, 4-15, 5-21
 目的, 2-47
CMAN_ADMIN パラメータ, 4-15, 4-20, 5-21
CMAN_PID.LOG ファイルの内容, 11-18, 11-27
CMAN_PID.TRC ファイルの内容, 11-32
CMAN_PROFILE パラメータ, C-78
CMAN_RULES パラメータ, 7-38, C-81
CMAN.LOG ファイル, 11-18
CMAN.LOG ファイルの内容, 11-18
CMAN.ORA
 「Oracle Connection Manager、構成ファイル」を参照
CMAN パラメータ, C-77
CMCTL
 「Oracle Connection Manager 制御ユーティリティ」を参照
 目的, 2-47
CMGW プロセス
 目的, 2-46
CNAME.SMD レコード, 8-18, A-44
COMMUNITY パラメータ, 5-4
CONNECT_DATA パラメータ, 6-12, 7-7, C-43
CONNECT_TIMEOUT パラメータ, 11-13, C-62
CONNECTIONS (CON または CONN) 属性, 7-47, 9-5
connect-time failover, 2-29, C-40
connect-time failover、構成, 7-11
CORBA Servers
 構成, 7-49, 7-62
 説明, 7-48
「Custom」インストール・タイプ, 4-22
CyberSAFE 認証方法
 「Oracle Advanced Security」を参照

D

DAEMON.TRACE_DIRECTORY パラメータ, C-7
DAEMON.TRACE_LEVEL パラメータ, C-7
DAEMON.TRACE_MASK パラメータ, C-8
DB_DOMAIN パラメータ, 4-5, 5-17
DB_NAME パラメータ, 5-17

DBAccess Interface, 2-18
DBSNMP_START コマンド, A-5
DBSNMP_STATUS コマンド, A-5
DBSNMP_STOP コマンド, A-6
DCE Integration、説明, 4-35
DDO
 「動的検出オプション」を参照
「Default Domain」オプション, 6-39
DELEGATE_DOMAIN コマンド, 6-50, A-33
DESCRIPTION_LIST パラメータ, C-38, C-52
DESCRIPTION パラメータ, 6-12, C-38, C-51
「Disable Out-of-Band Break」オプション, 7-23
DISABLE_OOB パラメータ, 7-23
DISPATCHERS (DIS または DISP) 属性, 7-47, 7-61, 9-5
Distributed Computing Environment (DCE) Integration
 説明, 7-28
DLCR.RDBMS.OMD レコード, A-44
DL.RDBMS.OMD レコード, 8-18, A-44
DLCR.RDBMS.OMD レコード, 8-18
DNS、説明、用語集 -2
Domain Name Service、説明、用語集 -2
DOMAIN_HINT コマンド, A-34
DST パラメータ, 7-38, C-81
DUMP_TNSNAMES コマンド, 5-24, A-35

E

Enterprise JavaBeans (EJB)
 構成, 7-49, 7-62
 説明, 7-48
EXIT コマンド
 CMCTL、リファレンス, A-89
 LSNRCTL、リファレンス, A-6
 NAMESCTL、リファレンス, A-36

F

failover
 connect-time, 2-29, 4-12, 7-11, C-40
FAILOVER_MODE パラメータ, C-43
FAILOVER パラメータ, 2-29, 4-12, 4-16, 5-11, 7-11, 7-12, 7-14, C-40
finger ユーティリティ, 10-11
FLUSH_NAME コマンド, A-38
FLUSH コマンド, A-37

G

General Inter-Orb Protocol (GIOP)

「GIOP」を参照

GIOP

- oracle.aurora.server.GiopServer, 7-50
- oracle.aurora.server.SGiopServer, 7-50
- 説明, 2-14
- ディスパッチャの構成, 7-61
- プレゼンテーション, 7-49
- プレゼンテーション・レイヤー, 4-14, C-51, C-55
- リスナーの構成, 7-54

GIOP プレゼンテーション・レイヤー, 4-14

「Global Database Name」オプション, 6-71, 7-8

「Global Database Name」フィールド, 7-69

GLOBAL_DBNAME パラメータ, 6-71, C-58

GLOBAL_NAME パラメータ, 7-8, 7-69, C-46

H

「Handler Name」オプション, 7-7

HANDLER_NAME パラメータ, 7-7

HELP コマンド

LSNRCTL、リファレンス, A-7

NAMESCTL、リファレンス, A-39

HOSTNAME 値, 7-19

HOSTS ファイル, 2-43

HOST パラメータ, 6-79

HS パラメータ, 7-7, C-46

I

Identifix, 7-27

Identix 認証方法

「Oracle Advanced Security」を参照

IIOP

GIOP プレゼンテーション・レイヤー, 2-14, 4-14

MTS_DISPATCHER パラメータの構成, 7-52

SSL の構成, 7-62

セッション・レイヤーを使用しない, 2-14

説明, 7-48, 7-49

IIOP クライアント

CORBA Servers, 7-48

Enterprise JavaBeans (EJB), 7-48

GIOP プレゼンテーション・レイヤー, 2-14, 4-14, 9-7, C-55

Java スタアド・プロシージャ, 7-48

oracle.aurora.server.GiopServer プレゼンテーション・レイヤー, 7-50

oracle.aurora.server.SGiopServer プレゼンテーション・レイヤー, 7-50

SSL サポート

IIOP クライアントの SSL サポート, 7-51

SSL の構成, 7-62

構成, 7-48, 7-49, 7-62

セッションベースの接続, 7-50

セッション・レイヤーを使用しない, 2-14

通常の IIOP 接続, 7-50

ディスパッチャへの直接接続, 7-51, 7-61

デフォルト以外のリスナー・ポートの構成, 4-24

独自のリスナー・ポートの構成, 7-57

リスナーからの接続, 7-50

リスナーのポート 2481, 4-24

リスナーのポート 2482, 4-24

リスナー用 2481 ポート, 7-57

リスナー用 2482 ポート, 7-57

IIOP スタック通信

説明, 2-14

INITDB_NAME.ORA ファイル

LOCAL_LISTENER パラメータ, 6-67

「Initial Preallocated Requests」オプション, 6-39

INITSID.ORA ファイル

DB_DOMAIN パラメータ, 5-17

DB_NAME パラメータ, 5-17

INSTANCE_NAME パラメータ, 5-17, 6-4

LOCAL_LISTENER パラメータ, 4-24, 6-16, 6-66, 6-67

MTS_DISPATCHERS パラメータ, 9-4

SERVICE_NAMES パラメータ, 6-4

SERVICE_NAME パラメータ, 5-17

「Instance Name」オプション, 7-7

INSTANCE_NAME パラメータ, 4-3, 4-9, 4-17, 5-17, 6-3, 6-4, 7-7, C-47

INTCHG.ORA ファイル、移行上の注意点, 5-22

Internet Inter-Orb Protocol (IIOP)

「IIOP」を参照

Inter-Orb Protocol (IIOP)

構成, 7-48

説明, 7-48

IPC、アドレスのパラメータ, 6-78

J

Java Database Connectivity (JDBC) Drivers

「 JDBC 」を参照

Java DBAccess Interface レイヤー、説明、2-18

Java NS、説明、2-18

Java Sockets、説明、2-18

Java VM、説明、7-49

JavaNet、説明、2-18

JavaTTC、2-18

JavaTTC、説明、2-18

Java オプション

構成、7-48

接続、7-48

ディスパッチャの構成、7-61

リスナーの構成、7-52、7-54

Java ストアド・プロシージャ

構成、7-49

説明、7-48

JDBC

Java DBAccess Interface、2-18

Java NS、2-18

Java Sockets、2-18

JavaNet、2-18

JavaTTC、2-18

OCI、2-16

TCP/IP プロトコル、2-18

Thin、2-16

ドライバ、2-16

K

Kerberos 認証方法

「 Oracle Advanced Security 」を参照

KEY パラメータ、6-78

L

LISTENER.LOG ファイルの内容、11-18

LISTENER.ORA

「 リスナー、構成ファイル 」を参照

LISTENER.TRC ファイルの内容、11-32

LISTENER (LIS または LIST) 属性、7-58、9-5

LISTENER 属性、4-24、6-66、7-57、7-58

LLU_NAME パラメータ、6-78

LLU パラメータ、6-78

LOAD_BALANCE パラメータ、2-30、4-10、4-18、

5-11、7-11、C-41

LOAD_TNSNAMES コマンド、5-25、5-28、6-43、A-40

LOCAL_LISTENER 属性、7-57

LOCAL_LISTENER パラメータ、4-24、6-16、6-66、
6-67、7-58、9-5

LOCAL_LU_NAME パラメータ、6-78

LOCAL_LU パラメータ、6-78

LOG_DIRECTORY_CLIENT パラメータ、11-20、C-8

LOG_DIRECTORY_SERVER パラメータ、11-20、C-8

LOG_DIRECTORY パラメータ、11-21、C-62

LOG_FILE_CLIENT パラメータ、11-20、C-9

LOG_FILE_SERVER パラメータ、11-20、C-9

LOG_FILE パラメータ、11-21、C-62

LOG_LEVEL パラメータ、11-22、C-79

LOG_STATS コマンド、A-40、A-41

LSNRCTL

「 リスナー制御ユーティリティ (LSNRCTL) 」を参
照

LU_NAME パラメータ、6-78

LU6.2

アドレスのパラメータ、6-78

サポートするトポロジー、4-29

サポートするネットワーク、4-29

説明、4-29

M

MAX_FREELIST_BUFFERS パラメータ、C-79

「 Maximum Open Connections 」オプション、6-39

「 Maximum Prespawnd Servers 」オプション、6-73

「 Maximum Wait Each Attempt 」オプション、6-39

MAXIMUM_CONNECT_DATA パラメータ、C-80

MAXIMUM_RELAYS パラメータ、C-80

MDN パラメータ、6-78

METHOD パラメータ、C-44

「 Minimal 」インストール・タイプ、4-22

MODE パラメータ、6-78

MOST_RELAYS、A-90

MTS_DISPATCHERS パラメータ、7-31、7-47、9-4

ADDRESS 属性、7-61、9-4

CONNECTIONS 属性、7-47、9-5

DISPATCHERS 属性、7-47、7-61、9-5

LISTENER、7-58

LISTENER 属性、4-24、6-66、7-57、7-58、9-5

MULTIPLEX 属性、7-31、9-6

POOL 属性、7-47、9-6

PRESENTATION 属性, 7-50, 7-52, 7-55, 7-61, 9-7
PROTOCOL 属性, 7-31, 7-47, 7-52, 7-55, 9-4
SERVICE 属性, 9-7
SESSIONS 属性, 9-7
TICKS 属性, 7-47, 9-7
初期設定, 9-8
MTS_SERVICE パラメータ, 9-7
MULTIPLEX (MUL または MULT) 属性, 7-31, 9-6
MultiProtocol Interchange, 5-2

N

Named Pipes

アドレスのパラメータ, 6-78
説明, 4-28
NAMES.ADDRESSES パラメータ, 6-28, 6-33, C-65
NAMES.ADMIN_REGION パラメータ, 5-25, 6-28, C-66
NAMES.AUTHORITY_REQUIRED パラメータ, C-66
NAMES.AUTO_REFRESH_EXPIRE パラメータ, C-67
NAMES.AUTO_REFRESH_RETRY パラメータ, C-67
NAMES.CACHE_CHECKPOINT_FILE パラメータ, C-68
NAMES.CACHE_CHECKPOINT_INTERVAL パラメータ, C-68
NAMES.CONFIG_CHECKPOINT_FILE パラメータ, C-68
NAMES.CONNECT_TIMEOUT パラメータ, C-67
NAMESCTL
「Oracle Names 制御ユーティリティ」を参照
NAMESCTL.INTERNAL_ENCRYPT_PASSWORD パラメータ, C-13
NAMESCTL.INTERNAL_USE パラメータ, C-13
NAMESCTL.NO_INITIAL_SERVER パラメータ, C-13
NAMESCTL.NOCONFIRM パラメータ, C-13
NAMESCTL.SERVER_PASSWORD パラメータ, C-14
NAMESCTL.TRACE_DIRECTORY パラメータ, C-14
NAMESCTL.TRACE_FILE パラメータ, C-14
NAMESCTL.TRACE_LEVEL パラメータ, C-14
NAMESCTL.TRACE_UNIQUE パラメータ, C-15
NAMES.DCE.PREFIX パラメータ, C-9, C-12
NAMES.DEFAULT_DOMAIN パラメータ, 6-15, 6-39, 7-16
NAMES.DEFAULT_FORWARDERS_ONLY パラメータ, C-69
NAMES.DEFAULT_FORWARDERS パラメータ, C-69

NAMES.DEFAULT_ZONE パラメータ, 5-4
NAMES.DEFAULT.DOMAIN パラメータ, C-9
NAMES.DIRECTORY_PATH パラメータ, 6-9, 6-19, 6-54, 6-57, 6-60, C-10
CDS, 7-19
HOSTNAME, 7-19
NDS, 7-19
NIS, 7-19
ONAMES, 7-19
TNSNAMES, 7-19
NAMES.DOMAIN_CHECKPOINT_FILE パラメータ, 4-21
NAMES.DOMAIN_HINTS パラメータ, C-69
NAMES.DOMAINS パラメータ, 6-29, 6-30, 6-33, C-70
NAMES.FORWARDING_AVAILABLE パラメータ, C-70
NAMES.FORWARDING_DESIRED パラメータ, C-70
NAMESINI.SQL スクリプト, 5-24, 6-26
NAMES.INITIAL_RETRY_TIMEOUT パラメータ, 6-39
NAMES.INITIAL_RETRY_TIMEOUT パラメータ, C-10
NAMES.KEEP_DB_OPEN パラメータ, C-70
NAMES.LOG_DIRECTORY パラメータ, 11-21, C-71
NAMES.LOG_FILE パラメータ, 11-21, C-71
NAMES.LOG_STATS_INTERVAL パラメータ, C-71
NAMES.LOG_UNIQUE パラメータ, C-71
NAMES.LOG ファイルの内容, 11-18
NAMES.MAX_OPEN_CONNECTIONS パラメータ, 6-39
NAMES.MAX_OPEN_CONNECTIONS パラメータ, C-72
NAMES.MAX_REFORWARDS パラメータ, C-72
NAMES.MESSAGE_POOL_START_SIZE パラメータ, 6-39
NAMES.MESSAGE_POOL_START_SIZE パラメータ, C-11, C-72
NAMES.NO_MODIFY_RESPONSE パラメータ, C-73
NAMES.NO_REGION_DATABASE パラメータ, C-73
NAMES.ORA
「Oracle Names Server、構成ファイル」を参照
NAMES.PASSWORDS パラメータ, C-73
NAMES.PREFERRED_SERVERS パラメータ, 7-20, C-12
NAMES.REGION_CHECKPOINT_FILE パラメータ, C-74
NAMES.REQUEST_RETRIES パラメータ, 6-39, C-12
NAMES.RESET_STATS_INTERVAL パラメータ, C-74

- NAMES.SAVE_CONFIG_ON_STOP パラメータ, C-74
- NAMES.SERVER_NAMES パラメータ, C-74, 6-33
- NAMES.SERVER_NAME パラメータ, 6-28
- NAMES.TOPOLOGY_CHECKPOINT_FILE パラメータ, 4-21
- NAMES.TRACE_DIRECTORY パラメータ, 11-37, C-75
- NAMES.TRACE_FILE パラメータ, 11-37, C-75
- NAMES.TRACE_FUNC パラメータ, C-75
- NAMES.TRACE_LEVEL パラメータ, C-75
- NAMES.TRACE_UNIQUE パラメータ, 11-37, C-76
- NAMES.TRC ファイルの内容, 11-32
- NAMESUPG.SQL スクリプト, 5-23, 6-26
- NAMES.USE_PLUG_AND_PLAY パラメータ, 4-21
- NDS
 - 「NetWare ディレクトリ・サービス」を参照
- Net8, 11-1
 - JDBC ドライバ, 2-16
 - Oracle DCE Integration との互換性, 7-28
 - SQL*Net との違い, 5-2
 - インストール, 4-22
 - 概念の説明, 1-2
 - 概要, 1-2
 - 主要機能, 2-4
 - スタック通信, 2-12
 - 制御ユーティリティ, 8-15
 - 接続操作, 2-2
 - 操作, 2-4
 - データ操作, 2-4
 - 例外操作, 2-4
- Net8 Server
 - インストール, 4-23
- Net8 Client
 - インストール, 4-26
 - 説明, 4-26, 4-27
- Net8 Configuration Assistant, 4-31
 - インストール後の使用, 4-23, 4-26
 - 起動, 4-31
 - 説明, 4-5, 4-31, 5-2
 - ネーミング・メソッドの構成, 4-24
 - ネット・サービス名の改名, 7-3
 - ネット・サービス名の構成, 6-18
 - ネット・サービス名の削除, 7-6
 - ネット・サービス名の属性の変更, 7-2
 - リスナーの構成, 4-23
 - ローカル・ネーミング・メソッドの構成, 6-13, 6-18
- Net8 Easy Config
 - 説明, 4-31, 5-2
- Net8 Firewall Proxy, 2-49
- Net8 OPEN
 - API 関数コール, 10-3
 - API の検索, 10-8
 - C 言語との互換性, 10-3
 - アプリケーションを使用するためのシステム構成, 10-9
 - エラー・メッセージ, 10-12
 - 構成要件, 10-8
 - サンプル・アプリケーション, 10-11
 - 非 SQL 情報との統合, 10-2
 - 分散アプリケーション用, 10-2
 - ライブラリ, 10-8
- Net8 OPEN による独自アプリケーションの開発, 10-8
- Net8 OPEN の ftp サンプル, 10-11
- Net8 Server
 - 説明, 4-23, 4-27
- Net8 アクセス制御, 2-49, 7-37, 4-33
- Net8 スタック通信
 - Oracle プロトコル, 2-12
 - TTC, 2-11, 9-7
 - Two-Task Common (TTC) プレゼンテーション, 2-11
 - クライアント・アプリケーション, 2-10
 - セッション, 2-12
- Net8 と SQL*Net の違い, 5-2
- NetWare ディレクトリ・サービス (NDS), 3-6, 6-51, 7-19
 - 説明, 4-34
- Network Security
 - DES_40, 4-34
 - DES_56, 4-34
 - MD5, 4-34
 - RC4_128, 4-34
 - RC4_40, 4-34
 - RC4_56, 4-34
 - 説明, 4-34
- NI
 - 「ネットワーク・インタフェース」を参照
- NIS
 - 「ネットワーク情報サービス」を参照
- NIS マップ, 6-52
- NMCBILD.SQL スクリプト, 5-27
- NMCGRNT.SQL スクリプト, 5-27
- NNL-00018 エラー・メッセージ, 6-31

NNL-00024 エラー・メッセージ, 6-31

NPI

「ネットワーク・プログラム・インタフェース」を
参照

NS.SMD レコード, 8-18, A-44

NS セッション, 4-14, C-51

O

ONAMES 値, 7-19

Open System Interconnection (OSI) モデル

クライアント・アプリケーション・レイヤー、説
明, 2-7

セッション・レイヤー, 2-8

説明, 2-6

トランスポート・レイヤー, 2-8

物理, 2-8

プレゼンテーション・レイヤー, 2-7

リンク・レイヤー, 2-8

OPI

「Oracle プログラム・インタフェース」を参照

ORA-00508 エラー・メッセージ, 11-9

ORA-1017 エラー・メッセージ, 11-3

ORA-1034 エラー・メッセージ, 8-13, 11-3

ORA-12154 エラー・メッセージ, 8-13, 11-8

ORA-12198 エラー・メッセージ, 11-8

ORA-12203 エラー・メッセージ, 8-12, 11-8, 11-9,
11-13

サンプルのエラー・スタック, 11-17

トラブルシューティング, 8-12

ORA-12224 エラー・メッセージ, 11-11

ORA-12500 エラー・メッセージ, 11-11

ORA-12533 エラー・メッセージ, 11-11

ORA-12535 エラー・メッセージ, 11-13

ORA-12538 エラー・メッセージ, 11-9

ORA-12541 エラー・メッセージ, 8-13

ORA-12545 エラー・メッセージ, 11-12

ORA-12547 エラー・メッセージ, 11-13

ORA-12560 エラー・メッセージ, 11-12

ORA-3113 エラー・メッセージ, 11-12

ORA-3121 エラー・メッセージ, 11-12

Oracle Advanced Security

CyberSAFE 認証方法, 4-35

DCE Integration, 4-35, 7-28

DES_40, 4-34

DES_56, 4-34

Identifix 認証方法, 7-27

Identix 認証方法, 4-35

Kerberos 認証方法, 4-35, 7-27

MD5, 4-34

Network Security, 4-34

Oracle Advanced Security の使用, 7-27

Oracle Connection Manager での使用, 7-27

RADIUS 認証方法, 4-35, 7-27

RC4_128, 4-34

RC4_40, 4-34

RC4_56, 4-34

Secure Sockets Layer (SSL) 認証方法, 7-27

SecurID 認証方法, 4-35

Single Sign-On, 4-35

SSL 認証方法, 4-35, 7-27

暗号化, 7-27

機能, 4-34

構成, 7-27, 7-28

生物学的認証方法, 7-27

セキュリティ機能, 7-27

説明, 4-34

RSA データ・セキュリティ RC4, 7-27

データ暗号化規格, 7-27

認証, 7-27

Oracle Connection Manager

CMADMIN プロセス, 2-47

CMCTL プロセス, 2-47

CMGW プロセス, 2-46

Net8 Firewall Proxy, 2-49

Net8 アクセス制御の構成, 7-37

Net8 アクセス制御を使用する場合の CMAN.ORA の
構成, 7-37

Net8 アクセス制御, 2-49, 4-33

Oracle Connection Manager アドレスの構成, 7-41

Oracle MultiProtocol Interchange のかわりに使用,
5-22

SOURCE_ROUTE パラメータ, C-41

TNSNAMES.ORA ファイル, 7-33

アップグレード, 5-21

アドレスの構成, 6-76

移行, 5-21

移行上の注意点, 5-21

起動, 8-7

構成パラメータのリファレンス, C-77 ~ C-81

構成ファイル

リリース 8.1 の新しいパラメータ, 4-20

構成ファイルのサンプル, B-9

接続集中化, 2-48, 3-11, 4-33

- 接続集中化の構成, 7-30
- 接続集中化を使用する場合の CMAN.ORA の構成, 7-30
- 接続のルーティング, 1-6, 7-25
- 説明, 4-33
- テスト, 8-19
- トレース・ファイル, 11-37
- 複数プロトコル環境での役割、図, 2-50
- 複数プロトコル・サポート, 2-49, 4-33, 5-22
- 複数プロトコル・サポートの構成, 7-34
- 複数プロトコル・サポートを使用する場合の CMAN.ORA の構成, 7-34
- プロセス, 2-46
- マルチスレッド・サーバーの使用, 7-30, 2-48
- リスナー・アドレスの構成, 7-44
- リリース 8.1 での構成ファイルの変更点, 4-15
- ログ・ファイル, 11-22
- ログ・ファイル、理解, 11-27
- Oracle Connection Manager 制御ユーティリティ (CMCTL)
 - 使用, 8-16
 - ネットワークのテスト, 8-16
 - 起動, 8-7
- Oracle Database Configuration Assistant
 - Oracle Enterprise Manager のリスナーの構成, 4-24
- Oracle Database Configuration Assistant、リスナーの構成, 4-24
- Oracle DCE Integration
 - Net8 との互換性, 7-28
 - 説明, 7-28
- Oracle Enterprise Manager
 - connect-time failover, 7-12, 2-29, 6-71
 - LISTENER.ORA の SID_LIST パラメータ, 2-29, 4-13, 5-20, 6-71, 7-12, C-40
 - LISTENER.ORA の静的サービス情報, 2-29, 5-20, 6-71, 7-12, C-40
 - リスナーの静的情報の構成, 4-24
- 「Oracle Home Directory」オプション, 6-71
- 「Oracle Home」オプション, 7-67
- Oracle Names
 - NAMESINI.SQL スクリプト, 5-24, 5-29, 6-26
 - NAMESUPG.SQL スクリプト, 5-23, 5-27, 5-29, 6-26
 - Oracle Connection Manager 情報の格納, 2-36, 6-23
 - Oracle Names Server の検出, 6-35, 6-36
 - Oracle Names Server の作成, 6-25
 - .SDNS.ORA ファイル, 2-45
 - SDNS.ORA ファイル, 2-45
 - TNSNAMES.ORA ファイルの Oracle Names へのロード, 5-25
 - アーキテクチャ, 2-31
 - アップグレード, 5-23, 5-30
 - 移行, 5-10, 5-23 ~ 5-30
 - 委任管理リージョン, 2-42
 - インストール, 6-24
 - オブジェクトの問合せ, 8-18
 - オブジェクトの登録, 6-41
 - 階層ネーミング・モデル, 2-38
 - 外部ネーミングでの使用, 3-7
 - 管理リージョン, 2-33
 - 共存, 5-10
 - クライアントとサーバーの構成, 6-36
 - クライアント・キャッシュ, 8-4
 - グローバル・データベースの名前とアドレスの格納, 2-35
 - グローバル・データベース・リンク修飾子の登録, 6-44, 6-46
 - 検出, 2-45
 - 構成, 6-21 ~ 6-50
 - コマンド行からの起動, 8-18
 - コントロールパネルからの起動, 8-7
 - 集中化されたネーミング, 3-5
 - 使用, 6-21
 - 情報のリポジトリとしてデータベースを使用, 2-34
 - シングル・ドメイン・ネーミング・モード, 2-37
 - 推奨環境, 3-9
 - 接続, 8-8
 - 接続の確立, 3-5
 - 説明, 2-31, 4-33, 6-5
 - その他の Oracle Names Server の名前とアドレスの格納, 2-35
 - データのレプリケーション, 6-32
 - データのレプリケート, 2-34
 - データベースのグローバル・データベースの名前とアドレスの格納, 6-21
 - データベース・リンクの格納, 2-36, 6-22
 - グローバル・データベース・リンクの登録, 6-44
 - デフォルト・ドメイン, 2-38
 - 動的検出オプション, 2-43
 - ドメイン, 2-33
 - ドメインの委任, 6-50
 - トレース・ファイル, 11-37
 - ネットワーク・コンポーネントのネーミング, 2-37
 - ネット・サービス名の格納, 2-35, 6-22

- ネット・サービス名の登録, 6-41
- バージョン 1, 2-42
- バージョン 2, 2-43
- バージョン 8, 2-44
- バージョン間の相違, 2-42
- 複数ドメインの構成, 6-29, 6-33
- 複数のドメイン, 2-39
- 複数のリージョンの使用, 2-40
- 別名の格納, 2-36, 6-23
- リージョン, 2-33
- リージョン・データベース, 2-34
- リージョン・データベースの構成, 6-26, 6-31
- リージョン・データベースへの情報の格納, 6-26
- リスナーの構成, 6-40
- 利点と欠点, 3-9
- ルート管理リージョン, 2-40
- ログ・ファイル, 11-21
- その他の Oracle Names Server の名前とアドレスの格納, 6-22
- Oracle Names Server
 - CKCFG.ORA ファイル, 6-25
 - CKPCCH.ORA ファイル, 6-25
 - CKPREG.ORA ファイル, 6-25
 - Oracle Connection Manager 情報の格納, 2-36, 6-23
 - Oracle Names Server の検出, 6-35
 - TNSNAMES.ORA ファイルの Oracle Names へのロード, 5-25
 - TNSPING を使用するテスト, 8-23
 - アドレスの構成, 6-76
 - 委任管理リージョン, 2-40
 - オブジェクトの登録, 6-41
 - 格納されるデータ, 2-35
 - 格納できるデータ, 2-35
 - 起動, 6-31, 8-3
 - クライアント・キャッシュの起動, 8-4
 - グローバル・データベースの名前とアドレスの格納, 2-35
 - グローバル・データベース・リンク修飾子の登録, 6-44
 - グローバル・データベース・リンクの登録, 6-44
 - 構成パラメータのリファレンス, C-65 ~ C-76
 - 構成ファイル、説明, 6-6
 - 構成ファイルのサンプル, B-7
 - 作成, 6-25
 - 説明, 1-5, 2-31
 - その他の Oracle Names Server の名前とアドレスの格納, 2-35, 6-22
- データのレプリケーション, 6-32
- データベースのグローバル・データベースの名前とアドレスの格納, 6-21
- データベース・リンクの格納, 2-36, 6-22
- テスト, 8-16
- デフォルト, 7-20
- 登録
 - グローバル・データベース・リンク修飾子, 6-46
 - ドメインの委任, 6-50
 - トレース・ファイル, 11-37
 - ネット・サービス名の格納, 2-35, 6-22
 - ネット・サービス名の登録, 6-41
 - 複数ドメインの構成, 6-29, 6-33
 - プロトコル・アドレスの構成, 6-64
 - 別名の格納, 2-36, 6-23
 - 優先, 2-42
 - 予約済み, 2-43
 - リージョン・データベースの作成, 6-27
 - ルート管理リージョン内, 2-40
 - レプリケーション用の作成, 6-32
 - ログ・ファイル, 11-21
- Oracle Names Server の検出, 2-45, 6-35
- Oracle Names 制御ユーティリティ (NAMESCTL)
 - Oracle Names Server の起動, 6-31, 6-35, 8-3
 - Oracle Names Server の検出, 6-35, 6-37
 - TNSNAMES.ORA の Oracle Names へのロード, 5-25, 5-28, 6-43
 - オブジェクトの間合せ, 8-18
 - グローバル・データベース・リンクの登録, 6-45
 - ネットワークのテスト, 8-15
 - ネットワーク・オブジェクトのテスト, 8-17
 - ネット・サービス名の登録, 6-42
 - ドメインの委任, 6-50
 - 別名の登録, 6-49
- Oracle Names でのデータのレプリケーション, 2-34, 6-32
- Oracle Names によるオブジェクトの登録, 6-41
- Oracle Names によるグローバル・データベース・リンク修飾子の登録, 6-46
- Oracle Names の NAMESUPG.SQL スクリプト, 5-27
- Oracle Net8 Assistant, 5-28, 6-43, 7-69
 - 「Always Use Dedicated Server」オプション, 7-25
 - 「Attempts Per Names Server」オプション, 6-39
 - 「Client Registration ID」オプション, 7-22
 - CONNECT_DATA の構成, 7-8
 - 「Default Domain」オプション, 6-39
 - 「Disable Out-of-Band Break」オプション, 7-23

「Global Database Name」オプション, 6-71, 7-8, 7-69
 「Handler Name」オプション, 7-7
 IOP クライアントを使用する場合のリスナーの構成, 7-56
 「Initial Preallocated Requests」オプション, 6-39
 「Instance Name」オプション, 7-7
 「Maximum Open Connections」オプション, 6-39
 「Maximum Prespawned Servers」オプション, 6-73
 「Maximum Wait Each Attempt」オプション, 6-39
 Oracle Advanced Security の構成, 7-27, 7-28
 Oracle Connection Manager を使用する場合のクライアントの構成, 7-41, 7-44
 「Oracle Home Directory」, 6-71
 「Oracle Home」オプション, 7-67
 Oracle Names Server に対する複数ドメインの構成, 6-29, 6-34
 Oracle Names Server の起動, 6-31, 6-35, 8-3
 Oracle Names Server の検出, 6-35, 6-37
 Oracle Names でのデフォルト以外のリスナーの構成, 7-60
 Oracle Names にデフォルト以外のリスナーを構成, 6-68
 Oracle Names によるグローバル・データベース・リンクの登録, 6-45
 Oracle Names によるネット・サービス名の登録, 6-42
 Oracle Names による別名の登録, 6-49
 「Oracle RDB Database」オプション, 7-8
 Oracle RDB データベースの構成, 7-70
 「Prefer Connection Manager Routing」オプション, 7-25
 「Program Name」オプション, 7-67
 「RDB Database」オプション, 7-69
 「Register Service with Oracle Names」オプション, 6-40
 「Session Data Unit (SDU)」オプション, 7-7
 「SID」オプション, 7-67
 「System Identifier (SID)」オプション, 6-71
 「Timeout」オプション, 6-74
 「TNS Time Out Value」オプション, 7-22
 TNSNAMES.ORA データの Oracle Names Server へのロード, 5-25
 TNSNAMES.ORA ファイルの Oracle Names へのロード, 5-25
 TNSNAMES.ORA ファイルのアップグレード, 5-11
 TNSNAMES.ORA ファイルの移行, 5-11

「Turn Off UNIX Signal Handling」オプション, 7-23
 「Type of Service」オプション, 7-8
 「Use Dedicated Server」オプション Oracle, 7-7
 「Use for Heterogeneous Services」オプション, 7-7, 7-66
 「Use Options Compatible with Net8 8.0 Clients」オプション, 5-11
 「Use Oracle8i Release 8.0 Compatible Identification」オプション, 5-11
 アドレスの追加, 7-3
 異機種間サービスの構成, 7-65
 異機種間サービスを使用する場合の Oracle8 データベースの構成, 7-65, 7-70
 外部プロシージャの構成, 7-63
 管理リージョンへのドメインの委任, 6-50
 起動, 4-32
 クライアント負荷のバランスの構成, 7-11
 クライアント・プロファイルの構成, 7-15
 サービス情報によるリスナーの構成, 6-72
 事前に生成される専用サーバーの構成, 6-74
 静的なリスナー情報の構成, 6-71
 接続オプションの詳細設定, 7-8
 接続性の検証, 8-26
 接続要求のルーティング, 7-25
 説明, 4-32, 5-2
 デフォルトの Oracle Names Server の構成, 7-20
 ドメインの構成, 7-16
 トレース・ファイルの構成, 11-38
 ネーミング・メソッドの優先順位, 7-18
 ネットワーク・オブジェクトのテスト, 8-17
 ネット・サービス名の構成, 6-14
 フェイルオーバーの構成, 7-11
 複数アドレス・オプション, 7-11
 プロトコル・アドレスの構成, 6-64
 プロトコル・アドレスの削除, 7-5
 プロファイルの詳細オプションの構成, 7-22
 リージョン・データベース環境での Oracle Names の構成, 6-27
 リスナー名の構成, 6-69
 リリース 8.0 サービスの構成, 5-14
 リリース 8.1 サービスの構成, 5-12
 ローカル・ネーミング・メソッドの構成, 6-14
 ログ・ファイルの構成, 11-23
 Oracle Network Manager、説明, 5-2
 Oracle Parallel Server
 connect-time failover, 2-29, 4-16
 connect-time failover の構成, 7-12

- connect-time failover の有効化, 4-12
- FAILOVER_MODE パラメータ, C-43
- FAILOVER パラメータ, 4-16, C-40
- INSTANCE_NAME パラメータ, 4-17, C-47
- LOAD_BALANCE パラメータ, 4-18, C-41
- クライアント負荷のバランス, 4-10
- クライアント負荷のバランスの構成, 7-12
- クライアント負荷のバランスの有効化, 4-10
- 「Oracle RDB Database」オプション, 7-8
- Oracle RDB データベース
 - 接続の構成, 7-69
 - 説明, 7-69
- ORACLE_HOME、定義, xviii
- ORACLE_HOME パラメータ, 6-71, 7-64, 7-67, C-58
- 『Oracle8i Advanced Security 管理者ガイド』
 - RADIUS 認証方法, 4-6
 - SSL 認証方法, 4-7
- oracle.aurora.server.GiopServer, 9-7
- oracle.aurora.server.SGiopServer, 7-50
- Oracle コール・インタフェース (OCI) レイヤー、説明, 2-11
- Oracle システム識別子 (SID) リスナーの構成, 6-71
- Oracle プログラム・インタフェース (OPI), 2-13
- Oracle プログラム・インタフェース (OPI) 説明, 2-13
- Oracle プロトコル
 - Bequeath, 4-30
 - LU6.2, 4-29
 - Named Pipes, 4-28
 - SPX, 4-27
 - SSL 付き TCP/IP, 4-6, 4-27
 - TCP/IP, 4-27
 - アドレスの構成, 6-78
 - 説明, 2-12, 4-27
- Oracle ホーム・ディレクトリ
 - 説明, 6-71
 - リスナーの構成, 6-71
- Oracle ホーム・ディレクトリ、定義, xviii
- OSS.MY.WALLET パラメータ, C-15
- OSS.SOURCE.CERTIFICATES パラメータ, C-15
- OSS.SOURCE.MY_WALLET パラメータ, 7-62
- OUT_OF_RELAY, A-90

P

- PARTNER_LU_LOCAL_ALIAS パラメータ, 6-78
- PARTNER_LU_NAME パラメータ, 6-78

- PASSWORDS パラメータ, C-62
- PASSWORD コマンド, A-42
- PING コーティリティ, 8-16, A-43
 - Oracle Names Server のテスト, 8-16
- PIPE パラメータ, 6-78
- PLU_LA パラメータ, 6-78
- PLU パラメータ, 6-78
- POOL_SIZE パラメータ, 6-73
- POOL_SIZE パラメータ、リスナー作成のサーバー・プロセスにおける役割, 2-23
- POOL 属性, 7-47, 9-6
- PORT パラメータ, 6-80
- 「Prefer Connection Manager Routing」オプション, 7-25
- PRESENTATION (PRE または PRES) 属性, 7-50, 7-52, 7-55, 7-61, 9-7
- PRESENTATION パラメータ, 4-19, 7-53, C-55
- PRESPAWN_DESC パラメータ, C-61
- PRESPAWN_LIST パラメータ, C-61
- PRESPAWN_MAX パラメータ, 2-24, 6-73, C-60
 - リスナー作成のサーバーにおける役割, 2-24
- 「Program Name」オプション, 7-67
- PROGRAM パラメータ, 6-78, 7-64, 7-67, C-59
- PROTOCOL_STACK パラメータ, 4-19, 7-53, C-54
- protocol.EXCLUDED_NODES パラメータ, C-82
- protocol.INVITED_NODES パラメータ, C-82
- PROTOCOL.ORA, C-82
- protocol.VALIDNODE_CHECKING パラメータ, C-82
- PROTOCOL (PRO または PROT) 属性, 7-31, 7-47, 7-52, 7-55, 9-4
- PROTOCOL パラメータ, 6-73, 6-78

Q

- QUERY コマンド, 6-35, 8-17, 8-18
 - A.SMD レコード, 8-18, A-44
 - CNAME.SMD, 8-18, A-44
 - DLCR.TDBMS.OMD レコード, 8-18, A-44
 - DL.RDBMS.OMD, 8-18, A-44
 - NAMESCTL、リファレンス, A-44
 - NS.SMD, 8-18, A-44
 - SMD レコード, 6-68
 - VIADD.NPO.OMD, 8-18, A-44
- QUEUESIZE パラメータ, 6-70, 8-13
 - リスナー・キュー・サイズの調整, 3-14, 6-70, 8-13
- QUIT コマンド
 - CMCTL の STOP NOW コマンドを参照

CMCTL の STOP コマンドを参照
LSNRCTL、リファレンス、A-8
NAMESCTL、リファレンス、A-46

R

RADIUS 認証方法

「Oracle Advanced Security」を参照
『Oracle8i Advanced Security 管理者ガイド』を参照

RAW セッション・レイヤー、4-14, 7-53
「RDB Database」オプション、7-69
RDB_DATABASE パラメータ、7-8, 7-69, C-47
「Register Service with Oracle Names」オプション、6-40
REGISTER コマンド、6-42, 6-45, 6-49, A-46
RELAY_STATISTICS パラメータ、C-80
RELOAD コマンド

LSNRCTL、リファレンス、A-8
NAMESCTL、リファレンス、A-47, A-48

REORDER_NS コマンド、5-25, 6-35, 6-37, 7-20
REPEAT コマンド、A-49
RESET_STATS コマンド、A-50
RESTART コマンド、A-51
ROSBILD.SQL スクリプト、5-27
ROSFILES

NMCBILD.SQL スクリプト、5-27
NMCGRNT.SQL スクリプト、5-27
ROSBILD.SQL スクリプト、5-27
ROSGRNT.SQL スクリプト、5-27

ROSFILES、Oracle Names への移行、5-26

ROSGRNT.SQL スクリプト、5-27

RSA データ・セキュリティ

「Oracle Advanced Security」を参照

S

SAVE_CONFIG コマンド、A-9

.SDNS.ORA ファイル、2-45

SDNS.ORA ファイル、2-45

SDU

構成、7-7

「セッション・データ・ユニット」を参照

SDU パラメータ、7-7, C-48

Secure Sockets Layer (SSL) 認証方法

「Oracle Advanced Security」を参照

SecurID 認証方法

「Oracle Advanced Security」を参照

SERVER パラメータ、6-78, 7-7, C-48

SERVICE_NAMES パラメータ、6-4, 9-7

SERVICE_NAME パラメータ、4-3, 4-9, 4-18, 5-7, 5-17, 5-19, 6-3, C-49

SERVICES コマンド、2-21, A-10

SERVICE (SER または SERV) 属性、9-7

SERVICE パラメータ、6-79

「Session Data Unit (SDU) Size」オプション、7-7

SESSIONS (SES または SESS) 属性、9-7

SESSION パラメータ、4-19, 7-53, C-56

SET CACHE_CHECKPOINT_INTERVAL コマンド、A-53

SET CONNECT_TIMEOUT コマンド、A-13

SET CURRENT_LISTENER コマンド、A-13

SET DEFAULT_DOMAIN コマンド、A-54

SET DISPLAYMODE コマンド

LSNRCTL、リファレンス、A-15

SET FORWARDING_AVAILABLE コマンド、A-55

SET LOG_DIRECTORY コマンド、A-16

SET LOG_FILE_NAME コマンド、A-56

SET LOG_FILE コマンド、A-16

SET LOG_STATS_INTERVAL コマンド、A-57

SET LOG_STATUS コマンド、A-17

SET NAMESCTL_TRACE_LEVEL コマンド、A-58

SET PASSWORD コマンド

LSNRCTL、8-5

LSNRCTL、リファレンス、A-17

NAMESCTL、リファレンス、A-59

SET RAWMODE コマンド

LSNRCTL、リファレンス、A-18

SET REQUESTS_ENABLED コマンド、A-60

SET RESET_STATS_INTERVAL コマンド、A-61

SET SAVE_CONFIG_INTERVAL コマンド、A-62

SET SAVE_CONFIG_ON_STOP コマンド、A-19

NAMESCTL、リファレンス、A-62

SET SERVER コマンド、A-63

SET STARTUP_WAITTIME コマンド、A-19

SET TRACE_FILE_NAME コマンド、A-64

SET TRACE_LEVEL コマンド、A-65

SET TRC_DIRECTORY コマンド、A-20

SET TRC_FILE コマンド、A-20

SET TRC_LEVEL コマンド、A-21

SET コマンド、A-12, A-52

SHOW CACHE_CHECKPOINT_INTERVAL コマンド、A-66

SHOW CONNECT_TIMEOUT コマンド、A-23

SHOW CURRENT_LISTENER コマンド、A-23

SHOW DEFAULT_DOMAIN コマンド、A-68

SHOW DISPLAYMODE コマンド, A-23
 SHOW FORWARDING_AVAILABLE コマンド, A-67
 SHOW LOG_DIRECTORY コマンド, A-23
 SHOW LOG_FILE_NAME コマンド, A-69
 SHOW LOG_FILE コマンド, A-23
 SHOW LOG_STATS_INTERVAL コマンド, A-70
 SHOW LOG_STATUS コマンド, A-23
 SHOW NAMESCTL_TRACE_LEVEL コマンド, A-71
 SHOW PASSWORD コマンド, A-23
 SHOW RAW_MODE コマンド, A-23
 SHOW REQUESTS_ENABLED コマンド, A-72
 SHOW RESETS_STATS_INTERVAL コマンド, A-73
 SHOW SERVER コマンド, A-73
 SHOW STARTUP_WAITTIME コマンド, A-23
 SHOW STATUS コマンド, A-74
 SHOW SYSTEM_QUERIES コマンド, A-75
 SHOW TRACE_FILE_NAME コマンド, A-76
 SHOW TRACE_LEVEL コマンド, A-77
 SHOW TRC_DIRECTORY コマンド, A-23
 SHOW TRC_FILE コマンド, A-23
 SHOW TRC_LEVEL コマンド, A-23
 SHOW USE_PLUGANDPLAY コマンド, A-23
 SHOW VERSION コマンド, A-78
 SHOW_TNS_INFO パラメータ, C-80
 SHOWSAVE_CONFIG_ON_STOP コマンド, A-23
 SHOW コマンド, A-23
 SHUTDOWN コマンド, A-79
 SID、リスナーの構成, 6-71
 SID_DESC パラメータ, C-60
 SID_LIST パラメータ, 4-13, 6-8, C-57, C-60
 SID_NAME パラメータ, 6-71, 7-64, 7-67, C-59
 SID パラメータ, 4-9, 5-7, C-49
 「SID」フィールド, 7-67
 Single Sign-On、説明, 4-35
 SMD レコード, 6-68
 SNA, 4-29
 SOURCE_ROUTE パラメータ, 4-13, 5-11, 7-11, 7-12, 7-33, 7-36, 7-40, C-41
 SPAWN コマンド, A-24
 SPX
 アドレスのパラメータ, 6-79
 説明, 4-27
 SQL*Net Easy Configuration、説明, 5-2
 SQLNET.ALTERNATE_PORT パラメータ, C-23
 SQLNET.AUTHENTICATION_KERBEROS5_SERVICE
 パラメータ, C-17
 SQLNET.AUTHENTICATION_SERVICES パラメータ,
 C-16
 SQLNET.AUTHENTICATION_TIMEOUT パラメータ,
 C-24
 SQLNET.CLIENT_AUTHENTICATION パラメータ,
 C-26
 SQLNET.CLIENT_REGISTRATION パラメータ, 7-22,
 C-17
 SQLNET.CRYPTO_CHECKSUM_CLIENT パラメータ,
 C-17
 SQLNET.CRYPTO_CHECKSUM_SERVER パラメータ,
 C-18
 SQLNET.CRYPTO_CHECKSUM_TYPES_CLIENT パラ
 メータ, C-18
 SQLNET.CRYPTO_CHECKSUM_TYPES_SERVER パラ
 メータ, C-18
 SQLNET.CRYPTO_SEED パラメータ, C-19
 SQLNET.ENCRYPTION_CLIENT パラメータ, C-19
 SQLNET.ENCRYPTION_SERVER パラメータ, C-19
 SQLNET.ENCRYPTION_TYPES_CLIENT パラメータ,
 C-20
 SQLNET.ENCRYPTION_TYPES_SERVER パラメータ,
 C-20
 SQLNET.EXPIRE_TIME パラメータ, 7-22, C-21
 SQLNET.IDENTIX_FINGERPRINT_DATABASE_
 PASSWORD パラメータ, C-21
 SQLNET.IDENTIX_FINGERPRINT_DATABASE_USER
 パラメータ, C-21
 SQLNET.IDENTIX_FINGERPRINT_DATABASE パラ
 メータ, C-21
 SQLNET.IDENTIX_FINGERPRINT_METHOD パラメー
 タ, C-21
 SQLNET.KERBEROS5_CC_NAME パラメータ, C-22
 SQLNET.KERBEROS5_CLOCKSKEW パラメータ,
 C-22
 SQLNET.KERBEROS5_CONF パラメータ, C-22
 SQLNET.KERBEROS5_KEYTAB パラメータ, C-23
 SQLNET.KERBEROS5_REALMS パラメータ, C-23,
 C-25
 SQLNET.LOG ファイル
 サンプルのクライアント側エラー・スタック,
 11-17
 SQLNET.LOG ファイルの内容, 11-18
 SQLNET.ORA ファイル
 「プロファイル (SQLNET.ORA)」を参照
 SQLNET.RADIUS_ALTERNATE_PORT パラメータ,
 C-23

SQLNET.RADIUS_ALTERNATE_RETRIES パラメータ,
C-24
SQLNET.RADIUS_AUTHENTICATION_INTERFACE パ
ラメータ, C-24
SQLNET.RADIUS_AUTHENTICATION_PORT パラメー
タ, C-24
SQLNET.RADIUS_AUTHENTICATION_TIMEOUT パラ
メータ, C-25
SQLNET.RADIUS_AUTHENTICATION パラメータ,
C-24
SQLNET.RADIUS_CHALLENGE_RESPONSE パラメー
タ, C-25
SQLNET.RADIUS_SEND_ACCOUNTING パラメータ,
C-26
SQLNETthread_id.TRC ファイル, 11-37, C-29
SQLNET.TRC ファイル, 11-37, C-29
SQLNET.TRC ファイルの内容, 11-32
SRC パラメータ, 7-38, C-81
SRV パラメータ, 7-38, C-81
SSL
 IIOP の構成, 7-62
SSL_CIPHER_SUITES パラメータ, C-26
SSL_CLIENT_AUTHENTICATION, 7-62
SSL_VERSION, 7-62
SSL_VERSION パラメータ, C-26
SSL 付き TCP/IP, 4-7
 アドレスのパラメータ, 6-79, 6-80
 説明, 4-27
SSL 認証方法
 「Oracle Advanced Security」を参照
 『Oracle8i Advanced Security 管理者ガイド』を参照
SSL プロトコル付き TCP/IP
 説明, 4-7
START CMAN コマンド, 8-7
START_CLIENT_CACHE, 8-4
START_CLIENT_CACHE コマンド, A-82
STARTUP_WAITTIME パラメータ, C-63
STARTUP コマンド
 NAMESCTL、制限, A-3, A-30
 NAMESCTL、リファレンス, A-82
START コマンド
 CMCTL、リファレンス, A-89
 LSNRCTL, 6-64, 8-5
 LSNRCTL、リファレンス, A-24
 NAMESCTL, 6-31, 6-35, 8-3
 NAMESCTL、リファレンス, A-80
STATS コマンド, A-90

STATUS コマンド
 CMCTL、リファレンス, A-91
 LSNRCTL, 8-5
 LSNRCTL、リファレンス, A-26
 NAMESCTL、リファレンス, A-83
STOP NOW コマンド
 CMCTL、リファレンス, A-92
STOP コマンド
 CMCTL、リファレンス, A-91
 LSNRCTL, 6-64, 8-5
 LSNRCTL、リファレンス, A-28
 NAMESCTL、リファレンス, A-84
SVR_PID.TRC ファイルの内容, 11-32
「System Identifier (SID)」オプション, 6-71
System Network Architecture (SNA), 4-29

T

TCP/IP
 アドレスのパラメータ, 6-79
 説明, 4-27
TCP.NODELAY パラメータ, 3-14, C-82
TICKS 属性, 7-47, 9-7
TIMED_QUERY コマンド, A-85
「Timeout」オプション, 6-74
TIMEOUT パラメータ, 6-74
TIMEOUT パラメータ、リスナー作成のサーバー・プ
ロセスにおける役割, 2-24
TNS
 「Transparent Network Substrate」を参照
 「TNS Time Out Value」オプション, 7-22
TNS_ADMIN 環境変数, 11-5
TNS_ADMIN パラメータ, 11-5
TNS-01169 エラー・メッセージ, 8-14
TNSAPI.A ファイル, 10-8
TNSAPI.DLL ファイル, 10-8
TNSAPI.H ファイル, 10-8
TNSAPI.LIB ファイル, 10-8
TNSNAMES.ORA
 「ローカル・ネーミング、構成ファイル」を参照
TNSNAMES.ORA の Oracle Names へのロード, 5-28,
6-43
TNSNAMES 値, 7-19
TNSNAV.ORA ファイル、移行上の注意点, 5-22
TNSNET.ORA ファイル、移行上の注意点, 5-22
TNSPING.TRACE_DIRECTORY パラメータ, 11-35,
C-27

TNSPING.TRACE_LEVEL パラメータ, 11-35, C-27
TNSPING ユーティリティ, 8-21
TRCROUTE ユーティリティとの比較, 8-24
TOTAL_REFUSED, A-90
TOTAL_RELAYS, A-90
TP_NAME パラメータ, 6-79
TPN パラメータ, 6-79
TRACE_DIRECTORY_CLIENT パラメータ, 11-33, C-27
TRACE_DIRECTORY_SERVER パラメータ, 11-34, C-27
TRACE_DIRECTORY パラメータ, 11-36, 11-37, C-63, C-80
TRACE_FILE_CLIENT パラメータ, 11-33, C-27
TRACE_FILE_SERVER パラメータ, 11-34, C-28
TRACE_FILE パラメータ, 11-36, C-63
TRACE_LEVEL_CLIENT パラメータ, 11-33, C-28
TRACE_LEVEL_SERVER パラメータ, 11-34, C-29
TRACE_LEVEL パラメータ, 11-36, C-64
TRACE_UNIQUE_CLIENT パラメータ, C-29
TRACE コマンド, A-29
TRACING パラメータ, 11-37, C-80
Transparent Network Substrate (TNS)
 主なコンポーネント, 2-12
 スタック通信におけるレイヤー, 2-12
 利点, 2-6
TRCROUTE ユーティリティ
 説明, 8-24
 パフォーマンスへの影響, 8-24
 要件, 8-24
 リスナー負荷のバランスでの使用, 8-25
TTC プレゼンテーション・レイヤー, 2-11, 9-7, C-55
「Turn Off UNIX Signal Handling」オプション, 7-23
Two-Task Common (TTC) プレゼンテーション・レイヤー, C-55
 TTC プレゼンテーション・レイヤー, C-51
 説明, 2-11, 9-7
「Type of Service」, 7-8
「Type of Service」オプション, 7-69
TYPE_OF_SERVICE パラメータ, 7-8, 7-69, C-42
TYPE パラメータ, C-44
「Typical」インストール・タイプ, 4-22

U

UNIX
 シグナル処理と Net8 の併用, 10-14

シグナル・ハンドラでの Oracle のオペレーティング・システム依存コール, 10-14
シグナル・ハンドラの欠点, 10-14
シグナル・ハンドラの使用, 10-13
UNIX シグナル・ハンドラの使用禁止, 10-15
UNREGISTER コマンド, A-86
「Use Dedicated Server」オプション, 7-7
「Use each address in order until destination reached」オプション, 5-14
「Use for Heterogeneous Services」オプション, 7-7, 7-66
「Use Options Compatible with Net8 8.0 Clients」オプション, 5-11
「Use Oracle8i Release 8.0 Compatible Identification」オプション, 5-11
USE_ASYNC_CALL パラメータ, C-80
USE_CMAN パラメータ, 7-25, 7-32, 7-35, 7-39, 7-40, C-30
USE_DEDICATED_SERVER パラメータ, 7-25, C-30
USE_PLUG_AND_PLAY パラメータ, 5-24, 6-40, C-64
USE_PLUGANDPLAY パラメータ, A-22

V

VIADD.NPO.OMD レコード, 8-18, A-44
VERSION コマンド
 LSNRCTL、リファレンス, A-29
 NAMESCTL、リファレンス, A-87

Y

ypserv, 6-51

あ

アーキテクチャ
 Net8, 2-5
 Oracle Connection Manager, 2-46
 Oracle Names, 2-31
アップグレード, 5-1, 5-30
 LISTENER.ORA の変更点, 5-20
 Oracle Connection Manager, 5-21
 Oracle Names の NAMESINI.SQL スクリプト, 5-24
 Oracle Net8 Assistant による TNSNAMES.ORA, 5-11
 SQLNET.ORA の変更点, 5-19
 TNSNAMES.ORA の変更点, 5-19
 構成ファイルの変更, 5-19
 サービス名とインスタンス名の確認, 5-17

- ソフトウェア, 5-18
 - 定義, 5-16
- アドレス、構成, 6-64
- アプリケーション
 - Net8 OPEN による開発, 10-8
 - Net8 OPEN のサンプル・プログラム, 10-11
 - 独自の ~ を使用するためのシステム構成, 10-9
- アプリケーション・プログラム・インタフェース (API) Oracle 以外のデータ・ソース用, 10-2

い

異機種間サービス

- 接続の構成, 7-65
- 説明, 7-65
- 移行, 5-1 ~ 5-30
 - INTCHG.ORA ファイル, 5-22
 - LISTENER.ORA の変更点, 5-20
 - NAMESUPG.SQL スクリプト, 5-27
 - Oracle Connection Manager, 5-21
 - Oracle Names, 5-23, 5-30
 - Oracle Names、リージョン・データベースの使用, 5-23
 - Oracle Names の NAMESINI.SQL スクリプト, 5-29
 - Oracle Names の NAMESUPG.SQL スクリプト, 5-23, 5-29
 - Oracle Net8 Assistant による TNSNAMES.ORA, 5-11
 - ROSFILERS から Oracle Names, 5-26
 - SQLNET.ORA の変更点, 5-19
 - TNSNAMES.ORA の変更点, 5-19
 - TNSNAV.ORA ファイル, 5-22
 - TNSNET.ORA ファイル, 5-22
 - 構成ファイルの変更, 5-19
 - サービス名とインスタンス名の確認, 5-17
 - ソフトウェア, 5-18
 - 定義, 5-16
 - 動的検出オプションによる Oracle Names から Oracle Names バージョン 8 への移行, 5-24
- 異機種間サービスの構成, 6-71
- イタリック体、定義, xviii
- 委任管理リージョン
 - Oracle Names, 2-40
 - 図, 2-41
 - ドメイン, 6-50
 - ルートの下, 2-42
- インスタンス名
 - 確認, 5-17

- 説明, 6-3
- インストール
 - 「Custom」, 4-22
 - 「Minimal」, 4-22
 - Net8 Client, 4-26
 - Net8 Server, 4-23
 - Oracle Names, 6-24
 - 「Typical」, 4-22
 - 概要, 4-22
 - デフォルトの構成, 4-23
 - ローカル・ネーミングのデフォルトの構成, 6-13

え

エラー・スタック

- サンプル, 11-17
 - 説明, 11-15
 - 代表的なレイヤー, 11-17
 - ログ・ファイルのエントリ, 11-25
- エラー・メッセージ
 - 12203, 11-13
 - 12535, 11-13
 - 12547, 11-13
 - Net8 OPEN の 20002-20021, 10-12
 - NNL-00018, 6-31
 - NNL-00024, 6-31
 - ORA-00508, 11-9
 - ORA-1017, 11-3
 - ORA-1034, 8-13, 11-3
 - ORA-12154, 8-13, 11-8
 - ORA-12198, 11-8
 - ORA-12203, 8-12, 11-8, 11-9
 - ORA-12203、サンプルのエラー・スタック, 11-17
 - ORA-12203、トラブルシューティング, 8-12
 - ORA-12224, 11-11
 - ORA-12500, 11-11
 - ORA-12533, 11-11
 - ORA-12538, 11-9
 - ORA-12541, 8-13
 - ORA-12545, 11-12
 - ORA-12560, 11-12
 - ORA-3113, 11-12
 - ORA-3121, 11-12
 - TNS-0116, 8-14
 - TNS-1169, 8-14
 - オラクル社サポート・サービスへの連絡, 11-58
 - 解決, 11-1, 11-58

トレース・データの例, 11-42
トレース・ファイル, 11-42
ログ・ファイルによる追跡, 11-25
円記号、定義, xviii

お

大文字、定義, xviii
オラクル社サポート・サービス、連絡, 11-58

か

解決

エラー、「トラブルシューティング」も参照
階層ネーミング・モデル

図, 2-38

説明, 2-38

外部ネーミング

Cell ディレクトリ・サービス, 3-6

NetWare ディレクトリ・サービスの使用, 3-6

NetWare ディレクトリ・サービス (NDS) の使用,
4-34

Oracle Names での使用, 3-7

機能, 4-34

構成, 3-7, 6-51

推奨環境, 3-9

接続の確立, 3-6

説明, 1-4, 3-6, 6-5

ネットワーク情報サービスの使用, 3-6, 6-51

ネットワーク情報サービス (NIS) の使用, 4-34

利点と欠点, 3-9

外部プロシージャ

構成, 7-63

説明, 7-63

監査証跡

情報を使用するためのスクリプト, 11-26

説明, 11-25

管理リージョン

Oracle Names, 2-33

委任, 2-40

委任、ルートの下, 2-42

説明, 2-33

定義, 6-23

ドメインの委任, 6-50

ルート, 2-40

関連する Oracle 製品

Oracle Advanced Security, 7-27

き

キーワード構文規則、構成ファイル, C-3

キーワードの値、ネットワーク・キャラクタ・セット,
C-4

規則、構成ファイルの構文, C-2

起動

Net8 コンポーネント, 8-2

Oracle Connection Manager, 8-7

Oracle Names Server, 6-31, 8-3

Oracle Names クライアント・キャッシュ, 8-4

Oracle Net8 Assistant, 4-32

インスタンス

マルチスレッド・サーバー, 9-4

ディスパッチャとインスタンス, 9-8

データベース, 8-6

リスナー, 8-5

起動のトラブルシューティング, 8-11

キャラクタ・セット

ネットワーク、キーワードの値, C-4

ネット・サービス名, C-5

キュー・サイズ, 3-14, 6-70

キュー・サイズ、接続要求の調整, 8-13

共存

Oracle Names, 5-10

Oracle8 Client リリース 8.0 または SQL*Net Client から Oracle8i データベース・リリース 8.1 への接続, 5-6

Oracle8i Client リリース 8.1 から Oracle8i データベース・リリース 8.1 への接続, 5-5

Oracle8i Client リリース 8.1 から Oracle8 リリース 8.0 または Oracle7 データベースへの接続, 5-8

TNSNAMES.ORA の SERVICE_NAME パラメータ,
5-7

TNSNAMES.ORA の SID パラメータ, 5-7

サードパーティ製 Oracle7 アプリケーション, 5-7

共存の問題, 5-5 ~ 5-10

共有サーバー

推奨環境, 3-10

説明, 2-20

「マルチスレッド・サーバー」も参照

利点と欠点, 3-10

く

クライアント

3GL を使用するテスト, 8-9

Net8 アクセス制御を使用する場合の構成, 7-39
Oracle Connection Manager アドレスの構成, 7-41
Oracle Connection Manager 機能を使用する場合の構成, 7-40
Oracle Names Server の検出, 6-36
Oracle RDB データベースの構成, 7-69
Oracle RDB の構成, 7-70
OS を使用するテスト, 8-9
SQLNET.LOG, 11-20
アップグレード, 5-19
アプリケーションを使用するテスト, 8-9
移行, 5-19
インストール後に Net8 Configuration Assistant を使用して構成, 4-26
接続集中化を使用する場合の構成, 7-32
接続性のテスト, 8-21
接続の方法, 8-8
特殊コマンドを使用するテスト, 8-10
ドメインの構成, 7-16, 7-20
トレース・ファイル, 11-33
複数プロトコルでのセッション、図, 2-50
複数プロトコル・サポートを使用する場合の構成, 7-35
プロファイルの構成, 7-15
マルチスレッド・サーバー・モードの変更, 9-11
ランダム化, 2-30, C-41
ローカル・ネーミングによる構成, 6-20
ロード・バランシング, 2-30, 4-10, 7-11, C-41
ログ・ファイル, 11-20
クライアントのトラブルシューティング, 11-4
クライアントのトレース, 11-37, C-29
クライアント負荷のバランス, 2-30, 4-10, C-41
構成, 7-11
ネットワーク・パフォーマンスを向上するために使用, 3-12
クライアント・アプリケーション・レイヤー、説明, 2-10
グローバル・データベース名
説明, 1-3, 4-9, 5-17, 6-71, C-58
定義, 6-3
リスナーの構成, 6-71, C-58
グローバル・データベース・リンク, 2-36, 6-22, 6-44
Oracle Names による登録, 6-45
リンク修飾子の追加, 6-46
グローバル・データベース・リンク修飾子, 6-44

け

計画

概要, 3-2
サマリー, 3-15
計画結果のサマリー, 3-15

こ

構成

CONNECT_DATA 属性, 7-7
connect-time failover, 7-11
CORBA Servers, 7-49, 7-62
Enterprise JavaBeans (EJB), 7-49 ~ 7-62
IIOP クライアント, 7-48, 7-49, 7-62
Inter-Orb Protocol (IIOP) クライアント, 7-48
Java オプション, 7-48
Java スタアド・プロシージャ, 7-49
Net8 Configuration Assistant, 4-23, 4-26
Net8 OPEN, 10-8
Net8 アクセス制御, 7-37
Net8 アクセス制御のための CMAN.ORA, 7-37
Net8 アクセス制御のためのクライアント, 7-39
Oracle Advanced Security, 7-28
Oracle Database Configuration Assistant とリスナー, 4-24
Oracle Names, 6-4, 6-5, 6-21, 6-50
Oracle Names Server, 6-27
Oracle Names Server 間でのデータのレプリケーション, 6-32
Oracle Names なし, 6-4
Oracle Names のリージョン・データベース, 6-26, 6-31
Oracle Names を使用するためのクライアントとサーバー, 6-36
Oracle RDB データベースとの接続, 7-69
異機種間サービスへの接続, 7-65
外部ネーミング, 3-7, 6-5
外部ネーミングのネット・サービス名, 6-51
外部プロシージャ, 7-63
クライアントの Oracle Connection Manager, 7-40
クライアント負荷のバランス, 7-11
事前に生成される専用サーバー, 6-73
接続集中化, 7-30
接続集中化のための CMAN.ORA, 7-30
接続集中化のためのクライアント, 7-32
接続集中化のためのサーバー, 7-31

- 接続プーリング, 7-47
- 独自のアプリケーションを使用するためのシステム, 10-9
- ドメイン, 7-16, 7-20
- 複数アドレス, 7-3
- 複数ドメイン, 6-29, 6-33
- 複数プロトコル・サポート, 7-34
- 複数プロトコル・サポートのための CMAN.ORA, 7-34
- 複数プロトコル・サポートのためのクライアント, 7-35
- プロトコル・アドレス, 6-64, 6-76
- プロファイル, 7-15
- プロファイルでの接続要求のルーティング, 7-25
- ホスト・ネーミング, 6-5, 6-7, 6-51
- リスナー, 6-62 ~ 6-75
- リスナー名, 6-69
- ローカル・ネーミング, 6-5, 6-12
- ローカル・ネーミングのネット・サービス名, 6-14
- ローカル・ネーミング用のクライアント, 6-20
- ローカル・ネーミング用のサーバー, 6-14
- ローカル・ネーミング用のリスナー, 6-21, 6-40
- 構成作業
 - 概要, 6-2
- 構成ファイル
 - Oracle Connection Manager (CMAN.ORA), B-9
 - Oracle Names Server (NAMES.ORA), B-7
 - 構文規則, C-2
 - プロトコル・アドレス, 6-76
 - プロファイル (SQLNET.ORA), B-2
 - リスナー (LISTENER.ORA), B-6
 - ローカル・ネーミング (TNSNAMES.ORA), B-5
- 構成ファイルのコメント, C-3
- 構文
 - CMCTL, 8-16, A-88 ~ A-92
 - LSNRCTL, 8-15, A-4 ~ A-29
 - NAMESCTL, 8-15, A-33 ~ A-87
 - Oracle Connection Manager 構成パラメータ, C-77 ~ C-81
 - Oracle Names Server 構成パラメータ, C-65 ~ C-76
 - 構成ファイルの規則, C-2
 - プロトコル構成パラメータ, C-82
 - プロファイル構成パラメータ, C-7 ~ C-30
 - リスナー構成パラメータ, C-50 ~ C-64
 - ローカル・ネーミング構成のパラメータ, C-31 ~ C-49

- 互換性
 - Oracle DCE Integration と Net8, 7-28
 - TRCROUTE ユーティリティと旧バージョンの SQL*Net, 8-24
- ゴシック体の小文字、定義, xviii
- 子プロセスの終了, 10-15
- コマンド
 - CMCTL のリファレンス, A-89 ~ A-92
 - LSNRCTL のリファレンス, A-4 ~ A-29
 - NAMESCTL のリファレンス, A-33 ~ A-87
- コマンド行
 - CMCTL コマンド, A-88
 - LSNRCTL コマンド, A-3
 - NAMESCTL コマンド, A-30

さ

- サードパーティ・アプリケーション, 5-7
- サーバー
 - IIOP 接続の構成, 7-49
 - Java オプションの構成, 7-49
 - Oracle Names Server の検出, 6-36
 - Oracle Names のリージョン・データベースの構成, 6-26
 - アップグレード, 5-18
 - 移行, 5-18
 - インストール後に Net8 Configuration Assistant を使用して構成, 4-23
 - 外部プロシージャの構成, 7-64
 - サーバー間の対話、説明, 2-13
 - スタック通信, 2-13
 - 接続集中化を使用する場合の構成, 7-31
 - 接続の切離し, 2-2
 - 接続の方法, 8-8
 - 専用
 - マルチスレッド~との比較, 9-2
 - 専用サーバーを事前に生成するためのリスナーの構成, 6-73
 - データベースの起動, 8-6
 - テスト, 8-19
 - トラブルシューティング, 11-3
 - トレース・ファイル, 11-33
 - プロファイルの構成, 7-15
- マルチスレッド
 - 専用~との比較, 9-2
 - マルチスレッド・サーバーと専用サーバーの構成, 9-10

- ループバック・テスト, 11-4
- ローカル・ネーミングによる構成, 6-14
- ログ・ファイル, 11-20
- サービス
 - グローバル・データベース名, 1-3
 - サービス名, 2-2
 - 接続, 1-3
 - 説明, 1-3, 6-2
- サービス名
 - アドレスの格納、NDS, 6-56
 - 確認, 5-17
 - キャラクタ・セットのキーワード値, C-5
 - 説明, 2-2, 6-2
- サービス・ネーミング、説明, 4-3
- 作成
 - リージョン・データベースのある Oracle Names Server, 6-27
 - リージョン・データベースのない Oracle Names Server, 6-32

し

- シグナル処理
 - Net8 との併用, 10-14
 - Oracle のオペレーティング・システム依存コールの要件, 10-14
 - UNIX の考慮事項, 10-13
- 事前に開始される専用サーバー
 - 推奨環境, 3-10
 - 利点と欠点, 3-10
- 事前に起動される専用サーバー・プロセス
 - 「事前に生成される専用サーバー・プロセス」を参照
- 事前に生成される専用サーバー, 2-23
 - 数, 6-73
 - 構成, 6-73
- 集中化されたネーミング
 - 推奨環境, 3-9
 - 接続の確立, 3-5
 - 説明, 3-5
 - 利点と欠点, 3-9
- シングル・ドメイン・ネーミング・モデル, 2-37
- 診断
 - エラー、「トラブルシューティング」も参照
 - クライアント, 11-4
 - サーバー, 11-3

す

- スキーマの拡張、NDS, 6-55
- 「スタート」ボタン -> を選択、定義, xviii
- スタック通信
 - Net8, 2-12
 - サーバー側の対話, 2-13
 - 説明, 2-6
- スラッシュ、定義, xviii

せ

- 制御ユーティリティ
 - 説明, 8-15, A-1
 - リスナー制御ユーティリティ (LSNRCTL), 8-15
- 生物学的
 - 「Oracle Advanced Security」を参照
- セキュリティ、NAMESTL ユーティリティ, A-3, A-31
- セッションの確立
 - bequeath の使用, 8-8
 - Oracle Names の使用, 3-5, 8-8
 - 外部ネーミングの使用, 3-6
 - 集中化されたネーミングの使用, 3-5
 - ホスト・ネーミングの使用, 3-4, 8-8
 - マルチスレッド・サーバー, 2-20
 - ローカル・ネーミングの使用, 3-5, 8-8
- セッションベースの IIOP, 7-50
- セッション・データ・ユニット
 - 構成, 7-7
 - ネットワーク・パフォーマンスを向上するために調整, 3-13
- セッション・レイヤー
 - NS, 2-12, 4-14, C-51
 - RAW, 4-14, 7-53
 - Transparent Network Substrate (TNS), 2-12
 - Two-Task Common (TTC), 2-12
 - 説明, 2-8, 2-12
 - ネットワーク・セッション (NS), 4-14, C-51
 - ネットワーク・ネーミング (NN), 2-12
 - ネットワーク・ルーティング (NR), 2-12
- セッション・レイヤーを使用しない, 4-14, 7-53
- 接続, 7-50
 - bequeath, 8-8
 - Oracle Connection Manager, 1-6, 7-25
 - Oracle Names, 8-8
 - Oracle7 から Oracle8 データベースへの接続, 5-6

- Oracle8 Client リリース 8.0 または SQL*Net Client から Oracle8i データベース・リリース 8.1, 5-6
- Oracle8i Client リリース 8.1 から Oracle8i データベース・リリース 8.1, 5-5
- Oracle8i Client リリース 8.1 から Oracle8 リリース 8.0 または Oracle7 データベースへの接続, 5-8
- 一般的な問題, 8-11
- エラー回避のためのリスナー・キュー・サイズの調整, 6-70, 8-13, 3-14
- 検証, 8-26
- コンポーネントの正常起動, 8-2
- サービス, 1-3
- セッションベースの IOP, 7-50
- 接続記述子, 2-2
- 接続の方法, 8-8
- 接続文字列, 1-3, 6-2, 8-2
- 接続文字列の入力, 8-8
- 専用サーバー, 7-25
- 専用サーバー、説明, 2-23
- 同時、接続数の増加, 6-70
- ネット・サービス名の利用, 1-3
- ホスト・ネーミング, 8-8
- マルチスレッド・サーバー、説明, 2-20
- 要求、既存のバックログ・サイズ, 8-13
- リスナー, 2-28
- リリース 8.0 または 7.x のサービス, 5-14
- リリース 8.1 サービス, 5-12
- ルーティング, 1-6
- ローカル・ネーミング, 8-8
- 接続記述子, 2-2, C-31, 6-12
- 接続集中化, 2-48, 4-33, 7-30
 - 接続プーリングに対する相対的な利点, 3-12
 - ネットワーク・パフォーマンスを向上するために使用, 3-11
- 接続操作, 2-2
- 接続のルーティング, 1-6, 7-25
- 接続プーリング
 - 接続集中化に対する相対的な利点, 3-12
 - ネットワーク・パフォーマンスを向上するために使用, 2-50, 3-11
 - マルチスレッド・サーバーでの使用, 7-47
- 接続負荷のバランス, 2-30, 4-2
- 接続文字列, 6-2, 8-2
- 切断
 - 異常終了, 2-3
 - サーバー, 2-2
 - 追加接続要求, 2-3

- ユーザーによる実行, 2-3
- 専用サーバー
 - bequeathed セッション, 2-25
 - 事前に生成される専用サーバー, 2-23
 - 接続のルーティング, 7-25
 - 説明, 2-23
 - マルチスレッド・サーバーとの比較, 9-2
 - マルチスレッド・サーバーとの併用, 9-10

そ

操作

- Net8, 2-4
- 接続, 2-2
- データ, 2-4
- 例外, 2-4

た

- 大カッコ、定義, xviii
- タイマーによる切断
 - 「デッド接続検出」を参照
- 多重化, 4-33
 - 接続集中化で使用, 3-11

つ

- 通常の IOP 接続, 7-50
- 通信、スタック, 2-6

て

- ディスパッチャ
 - IOP クライアントの接続, 7-51, 7-61
 - 数の設定, 9-8
 - 起動する数, 9-8
 - 説明, 2-20
- データ暗号化規格
 - 「Oracle Advanced Security」を参照
- データ操作
 - 同期式, 2-4
 - 非同期式, 2-4
- データ転送、最大化, 3-13
- データ転送の最大化、SDU サイズの調整, 3-13
- データの受信
 - 同期式, 2-4
 - 非同期式, 2-4

- データの送信
 - 同期式, 2-4
 - 非同期式, 2-4
- データベース
 - Oracle Names のリポジトリ, 2-34
 - 分散, 2-5
- データベース・リンク, 2-36
- データベース・インスタンス登録
 - connect-time failover, 2-29
 - MTS ディスパッチャの登録, 2-27
 - クライアント負荷のバランス, 2-30
 - サービス登録, 2-27, 6-2
 - 接続負荷のバランス, 2-30
 - 説明, 2-27, 4-2
 - 利点, 2-27
 - ロード・バランシング, 2-30
- データベース・リンク, 6-22, 6-44
 - グローバル・データベース・リンク, 2-36, 6-22
 - パブリック・データベース, 2-36, 6-22
 - プライベート・データベース, 2-36, 6-22
- テスト
 - 3GL を使用するクライアント, 8-9
 - Oracle Connection Manager, 8-19
 - Oracle Names Server, 8-16
 - OS を使用するクライアント, 8-9
 - アプリケーションを使用するクライアント, 8-9
 - 制御ユーティリティ, 8-15
 - 特殊コマンドを使用するクライアント, 8-10
 - ネットワーク接続性, 8-21
 - ネットワーク・オブジェクト, 8-17
 - リスナー, 8-19
- デッド接続検出
 - SQLNET.EXPIRE_TIME, C-21
 - 制約, 7-22
 - 説明, 2-3
- デフォルト・ドメイン
 - NULL, 7-16
 - Oracle Names, 2-38

と

- 透過的アプリケーション・フェイルオーバー, C-43
- 同期式データ操作, 2-4
- 動的検出オプション, 2-43
 - 移行上の注意点, 5-24

登録

- Oracle Names によるグローバル・データベース・リンク, 6-44
- Oracle Names によるグローバル・データベース・リンク修飾子, 6-44
- Oracle Names によるネット・サービス名, 6-41

ドメイン, 4-5

- 委任, 6-50
- 関連する複数の階層, 2-39, 6-29, 6-33
- シングル・ドメイン・ネーミング構造、図, 2-37
- 説明, 2-33, 6-23
- デフォルト, 2-38
- ネーミングの考慮事項, 2-33
- ルート管理リージョンでの要件, 2-40

トラブルシューティング, 11-1 ~ 11-58

- エラー・メッセージORA-12203, 8-12
- エラー・メッセージTNS-01169, 8-14
- オラクル社サポート・サービスへの連絡, 11-58
- 起動と接続の一般的な問題, 8-11
- クライアント, 11-4
- サーバー, 11-3
- 質問, 11-14
- トレース・ファイル, 11-15
- ループバック・テスト, 11-4
- ログ・ファイル, 11-15
- ログ・ファイルの使用, 11-25

トランスポート・レイヤー、説明, 2-8

トレース・ファイル

- CMADM_PID.TRC, 11-37
- LISTENER.TRC, 11-36
- NAMES.TRC, 11-37
- SQLNETthread_id.TRC, 11-37, C-29
- SQLNET.TRC, 11-37, C-29
- エラー・メッセージ情報, 11-42
- エラー・メッセージの例, 11-42
- クライアントの SQLNET.TRC, 11-33
- サーバーの SVR_PID, 11-33
- デフォルト名, 11-32
- パケット・データの例, 11-41
- 例, 11-53

に

- 認証、Oracle Advanced Security の使用, 7-27

ね

ネーミング・メソッド

Net8 Configuration Assistant による構成, 4-24

Oracle Names, 1-4, 3-5, 6-5, 6-21, 7-19

外部ネーミング, 1-4, 3-6, 4-34, 6-5, 6-51, 7-19

集中化されたネーミング, 3-5

説明, 1-4, 3-3

選択, 3-8

ホスト・ネーミング, 1-4, 3-3, 6-5, 6-7, 7-19

優先順位, 7-18

ローカル・ネーミング, 1-4, 3-5, 6-5, 6-12, 7-19

ネーミング・モデル

階層, 2-38

シングル・ドメイン, 2-37

ネストされたアドレス, 4-11

ネットワーク暗号化 (NAE) エラー・スタックの~レイヤー, 11-17

ネットワーク情報サービス (NIS), 4-34, 7-19, 3-6, 6-51

ネットワーク認証 (NA)

エラー・スタックの~レイヤー, 11-17

スタック通信におけるレイヤー, 2-12

ネットワークの構成

ローカル管理, 6-4

集中管理, 6-4

ネットワーク・インタフェース (NI)

エラー・スタックの~レイヤー, 11-17

説明, 2-12

ネットワーク・キャラクタ・セット、キーワードの値、C-4

ネットワーク・サービス (NA) エラー・スタックの~レイヤー, 11-17

ネットワーク・セッション, 4-14

ネットワーク・セッション (NS), 4-14, C-51

ネットワーク・セッション (NS) エラー・スタックの~レイヤー, 11-17

ネットワーク・トランスポート (NT) エラー・スタックの~レイヤー, 11-17

ネットワーク・ネーミング (NN)

エラー・スタックの~レイヤー, 11-17

スタック通信におけるレイヤー, 2-12

ネットワーク・パフォーマンスの向上

SDU サイズの調整, 3-13

クライアント負荷のバランス, 3-12

クライアント要求のランダム化, 2-30, C-41

接続集中化の使用, 3-11

接続プーリング, 2-50, 3-11

リスナー・キュー・サイズ, 3-14

ネットワーク・プログラム・インタフェース (NPI), 2-13

ネットワーク・ルーティング (NR)

エラー・スタックの~レイヤー, 11-17

スタック通信におけるレイヤー, 2-12

ネット・サービス名

CONNECT_DATA の構成, 7-7

Oracle Connection Manager アドレスの構成, 7-41

Oracle Names, 1-4, 3-5

Oracle Names による登録, 6-42

TNSPING を使用するテスト, 8-23

アドレス、変換, 3-3

アドレスの追加, 7-3

外部ネーミング, 1-4, 3-6

改名, 7-3

サービスの識別, 6-2

削除, 7-6

説明, 4-4, 6-2

定義, 1-3, 2-2

ネーミング・メソッドの優先順位, 7-18

複数アドレス, 7-3, 7-11

プロトコル・アドレスの構成, 6-64

プロトコル・アドレスの削除, 7-5

変更, 7-2

ホスト・ネーミング, 3-3

ローカル・ネーミング, 1-4, 3-5

の

ノード, 2-6

は

パケット

種類, 11-41

トレース・データの例, 11-41

パスワード

NAMESCTL アクセス, A-3, A-31

リスナーを停止するために必要, 8-14

バッファ・フラッシング、説明, 3-14

パブリック・データベース・リンク, 2-36, 6-22

パラメータ

Net8 で廃止, 5-4

Oracle Connection Manager 構成のリファレンス, C-77 ~ C-81

Oracle Names Server 構成のリファレンス,
C-65 ~ C-76
Oracle プロトコル, 6-78
プロトコル構成のリファレンス, C-82
プロファイル構成のリファレンス, C-7 ~ C-30
リスナー構成のリファレンス, C-50 ~ C-64
リリース 8.1 で CMAN.ORA ファイルに追加された
新しいパラメータ, 4-20
リリース 8.1 で LISTENER.ORA ファイルに追加さ
れた新しいパラメータ, 4-19
リリース 8.1 で TNSNAMES.ORA ファイルに追加さ
れた新しいパラメータ, 4-16, 4-18
リリース 8.1 で廃止, 4-21
ローカル・ネーミング構成のリファレンス,
C-31 ~ C-49

ひ

非同期式データ操作, 2-4

ふ

ファイアウォール・サポート, 2-49, 4-33
Net8 Firewall Proxy, 2-49
Oracle Connection Manager, 2-49
フェイルオーバー, C-43
Oracle Enterprise Manager, 2-29, 6-71, 7-12
透過的アプリケーション・フェイルオーバー,
2-29, C-43
複数アドレス, 7-3, 7-11
connect-time failover の構成, 7-12
クライアント負荷のバランスの構成, 7-12
複数プロトコル・サポート, 2-49, 4-33, 5-2, 5-22
構成, 7-34
物理レイヤー、説明, 2-8
プライベート・データベース・リンク, 2-36, 6-22
プレゼンテーション・レイヤー
GIOP, 2-14, 4-14, 4-19, 7-49, 7-50, 9-7, C-51,
C-55
oracle.aurora.server.GiopServer, 9-7
oracle.aurora.server.SGiopServer, 7-50, 9-7
TTC, 2-11, 4-19, 9-7, C-51, C-55
Two-Task Common (TTC), 2-11, 4-19, 9-7, C-51,
C-55
説明, 2-7
プロトコル
アドレスの構成, 6-76, 6-78

構成パラメータのリファレンス, C-82
ネットワーク・レイアウトの選択, 3-2
ネットワーク・レイアウトのための複数の～の選
択, 3-2

プロファイル
構成, 7-15
プロファイル (SQLNET.ORA)
記述, 6-6
構成, 7-15
構成パラメータのリファレンス, C-7 ~ C-30
構成ファイルのサンプル, B-2
詳細オプションの構成, 7-22
接続要求のルーティング, 7-25
デフォルトの Oracle Names Server の構成, 7-20
ドメインの構成, 7-16
ネーミング・メソッドの優先順位, 7-18
分散処理, 2-5
分散データベース, 2-5

へ

変換
ネット・サービス名のアドレス, 3-3

ほ

ポート番号の範囲, 6-80
ホスト・ネーミング
説明, 1-4
構成, 6-7
推奨環境, 3-8
接続, 8-8
接続の確立, 3-4, 8-8
説明, 6-5
必要性, 6-7
利点と欠点, 3-8

ま

マルチスレッド・サーバー
Oracle Connection Manager での使用, 2-48
Oracle Connection Manager の使用, 7-30
起動, 9-4
初期化パラメータ, 9-4
セッション要求のルーティング, 2-20
接続プーリング, 2-50
接続プーリングによる使用, 7-47

接続負荷のバランス, 2-30
説明, 2-4, 2-20
専用サーバーとの比較, 9-2
専用サーバーとの併用, 9-10
ディスパッチャ, 2-20
ディスパッチャの構成, 9-8

や

山カッコ、定義, xviii

ゆ

ユーザーによる切断, 2-3
優先 Oracle Names Server, 2-42
ユーティリティ
finger, 10-11
制御, 8-15

よ

予約済み Oracle Names Server, 2-43

り

リージョン

Oracle Names, 2-33, 2-40
委任管理, 2-40
委任管理、図, 2-41
複数の管理ネットワークの構成, 2-40
ルート管理, 2-40
ルートの下の委任管理, 2-42

リージョン・データベース

Oracle Names Server の作成, 6-27
Oracle Names 情報の格納, 6-26
定義, 2-34
データベース・サーバーの構成, 6-26, 6-31

リスナー, 6-71

connect-time failover, 2-29, 4-12
IIOP クライアントの接続, 7-50, 7-52
IIOP クライアントのためにデフォルト以外のポート
番号を構成, 4-24
IIOP クライアントの独自ポートの構成, 7-57
Net8 Client のためにデフォルト以外のポート番号を
構成, 4-24
Net8 Configuration Assistant による構成, 4-23
Net8 クライアントのためにデフォルト以外のポート

番号を構成, 6-67

Oracle Connection Manager アドレスの構成, 7-44
Oracle Database Configuration Assistant による構成,
4-24
Oracle Enterprise Manager の構成, 2-29, 4-13, 5-20,
6-71, 7-12, C-40, 4-24
Oracle Names を使用する構成, 6-40
Oracle システム識別子 (SID) の構成, 6-71
Oracle ホーム・ディレクトリの構成, 6-71
アドレスの追加, 7-3
インスタンス登録, 2-27
外部プロシージャの構成, 6-71, 7-63, 7-64
監査証跡とログ・ファイル, 11-25
起動, 8-5
キュー・サイズの拡大, 3-14, 6-70
キュー・サイズの調整, 3-14, 6-70, 8-13
クライアント接続、説明, 2-28
グローバル・データベース名の構成, 6-71, C-58
構成, 6-62, 6-75
構成パラメータのリファレンス, C-50 ~ C-64
構成ファイル, 1-3
リリース 8.1 の新しいパラメータ, 4-19
構成ファイル、説明, 6-6
構成ファイルのサンプル, B-6
サービス登録, 2-27
制御ユーティリティ、「リスナー制御ユーティリ
ティ (LSNRCTL)」を参照
専用サーバーを事前に生成するための構成, 6-73
代表的な Net8 接続、図, 2-28
停止、エラー TNS-01169, 8-14
テスト, 8-19
デフォルトの構成, 6-62
透過的アプリケーション・フェイルオーバー, 2-29
同時接続の処理, 6-70
トレース・ファイル, 11-36
名前の構成, 6-69
部分アドレスのリスニング, 2-23
プロトコル・アドレスの構成, 6-64
プロトコル・アドレスの削除, 7-5
ホスト・ネーミングの構成, 6-8
リスナー・アドレスの構成, 6-76
リリース 8.1 での構成ファイルの変更点, 4-13
ローカル・ネーミングによる構成, 6-21
ロード・バランシング, 2-30
ログ・ファイル, 11-15, 11-21
リスナーからの IIOP クライアント接続, 7-52
リスナー間でクライアント要求をランダム化, 2-30,

C-41
リスナー制御ユーティリティ (LSNRCTL)
SET PASSWORD コマンド, 8-14
説明, 8-15
ネットワークのテスト, 8-15
リスナーの起動, 6-64, 8-5
リスナーの停止, 6-64
リスナー負荷のバランス、TRCROUTE の使用, 8-25
リスナー・アドレス, 6-76
利点
TNS, 2-6
リファレンス
CMCTL コマンド, A-89 ~ A-92
LSNRCTL コマンド, A-4 ~ A-29
NAMECTL コマンド, A-33 ~ A-87
Oracle Connection Manager 構成, C-77 ~ C-81
Oracle Names Server 構成, C-65 ~ C-76
プロトコルの構成, C-82
プロファイル構成, C-7 ~ C-30
リスナー構成, C-50 ~ C-64
ローカル・ネーミング構成, C-31 ~ C-49
リリース 8.1 で廃止されたパラメータ, 4-21, 5-4
リリース 8.1 の CMAN.ORA パラメータ, 4-20
リリース 8.1 の LISTENER.ORA パラメータ, 4-19
リリース 8.1 の TNSNAMES.ORA パラメータ,
4-16 ~ 4-18
リリース 8.1 の機能, 4-2
CMAN.ORA ファイルでの変更点, 4-15
CMAN.ORA ファイルの新しいパラメータ, 4-20
DB_DOMAIN パラメータのデフォルト, 4-5
Java オプションのサポート, 4-5
LISTENER.ORA ファイルでの変更点, 4-13
LISTENER.ORA ファイルの新しいパラメータ, 4-19
Net8 Configuration Assistant, 4-5
RADIUS 認証方法, 4-6
SSL 認証方法, 4-7
SSL プロトコル付き TCP/IP, 4-6
TNSNAMES.ORA ファイルでの変更点, 4-9, 4-13
TNSNAMES.ORA ファイルの新しいパラメータ,
4-16, 4-18
サービス・ネーミング, 4-3
接続負荷のバランス, 4-2
データベース・インスタンス登録, 4-2
リンク・レイヤー、説明, 2-8

る

ルート

Oracle Names の管理リージョン, 2-40
管理リージョンの Oracle Names Server の要件, 2-40
管理リージョンのドメインの要件, 2-40
リージョン、データ定義の要件, 2-40
ルート管理リージョン、説明, 2-40
ルート・ドメイン、説明, 2-40
ループバック・テスト, 11-4

れ

例外操作、説明, 2-4

ろ

ローカル・ネーミング

connect-time failover, 4-12
Net8 Configuration Assistant による構成, 6-13
Oracle Net8 Assistant による構成, 6-14
クライアントの構成, 6-20
クライアント負荷のバランス, 4-10
構成, 6-12
構成パラメータのリファレンス, C-31 ~ C-49
構成ファイル
リリース 8.1 の新しいパラメータ, 4-16 ~ 4-18
構成ファイル、説明, 6-6
構成ファイルのサンプル, B-5
サーバーの構成, 6-14
推奨環境, 3-8
接続, 8-8
接続の確立, 3-5, 8-8
説明, 1-4, 6-5
デフォルトの構成, 6-13
プロトコル・アドレスの構成, 6-64
リスナーの構成, 6-21
利点と欠点, 3-8
リリース 8.1 での構成ファイルの変更点,
4-9 ~ 4-13

ロード・バランシング

クライアント負荷のバランス, 2-30, C-41
接続負荷のバランス, 2-30

ロード・バランシング、説明, 2-30

ログ・ファイル

CMADM_PID.LOG, 11-22
CMAN_PID.LOG, 11-22

CMAN_PID.TRC , 11-37
LISTENER.LOG , 11-21
NAMES.LOG , 11-21
エラーの追跡に使用 , 11-25
クライアントとサーバーの SQLNET.LOG , 11-20
デフォルト名 , 11-18
リスナー , 11-25

