

Oracle9i Real Application Clusters

管理

リリース 2 (9.2)

2002 年 7 月

部品番号 : J06276-01

ORACLE®

Oracle9i Real Application Clusters 管理, リリース 2 (9.2)

部品番号 : J06276-01

原本名 : Oracle9i Real Application Clusters Administration, Release 2 (9.2)

原本部品番号 : A96596-01

原本著者 : Mark Bauer

原本協力者 : Lance Ashdown, Jonathan Creighton, Raj Kumar, Francisco Sanchez, David Austin, Jack Cai, Sohan Demel, Mitch Flatland, Carmen Frank, Vijay Lunawat, Dipak Saggi, Klaus Thielen, Steve Wertheimer, Valarie Moore

Copyright © 1998, 2002 Oracle Corporation. All rights reserved.

Printed in Japan.

制限付権利の説明

プログラム (ソフトウェアおよびドキュメントを含む) の使用、複製または開示は、オラクル社との契約に記載された制約条件に従うものとします。著作権、特許権およびその他の知的財産権に関する法律により保護されています。

当プログラムのリバース・エンジニアリング等は禁止されています。

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

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

危険な用途への使用について

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

Restricted Rights Notice

Programs delivered subject to the DOD FAR Supplement are "commercial computer software" and use, duplication, and disclosure of the Programs, including documentation, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement. Otherwise, Programs delivered subject to the Federal Acquisition Regulations are "restricted computer software" and use, duplication, and disclosure of the Programs shall be subject to the restrictions in FAR 52.227-19, Commercial Computer Software - Restricted Rights (June, 1987). Oracle Corporation, 500 Oracle Parkway, Redwood City, CA 94065.

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

目次

はじめに	xiii
Real Application Clusters の管理の新機能	xxiii
第 I 部 Real Application Clusters の管理の概要	
1 Real Application Clusters の管理の概要	
Real Application Clusters データベースの管理	1-2
Real Application Clusters でのパラメータ管理	1-2
Real Application Clusters での記憶域管理	1-2
管理に関する一般的な問題	1-2
Real Application Clusters 環境の Oracle Enterprise Manager	1-3
Real Application Clusters でのバックアップおよびリカバリ	1-3
Real Application Clusters の拡張性	1-4
トラブルシューティング	1-4
2 Real Application Clusters 環境でのパラメータの管理	
Real Application Clusters データベースのサーバー・パラメータ・ファイルの管理	2-2
サーバー・パラメータ・ファイルのバックアップ	2-2
Real Application Clusters のサーバー・パラメータ・ファイルのパラメータ値の設定	2-2
下位互換性のためのサーバー・パラメータ・ファイルのエクスポート	2-4
サーバー・パラメータ・ファイルのパラメータ設定	2-5

クライアント側のパラメータ・ファイルの使用	2-6
initsid.ora の用途	2-8
initdb_name.ora の用途	2-9
インスタンス固有のファイルにおける IFILE パラメータの配置および使用	2-10
クライアント側のパラメータ・ファイルにおける複数の IFILE エントリの使用	2-11
Real Application Clusters のパラメータの設定	2-12
サーバー・パラメータ・ファイルのパラメータによるインスタンスの一意識別	2-12
サーバー・パラメータ・ファイルに sid を使用したインスタンス固有のパラメータ設定	2-12
Real Application Clusters のパラメータ・タイプ	2-13
複数値のパラメータ	2-13
すべてのインスタンスで同じ値を設定する必要があるパラメータ	2-13
すべてのインスタンスで一意的な値を設定する必要があるパラメータ	2-14
クラスタのパラメータに関する考慮事項	2-15
Real Application Clusters の起動プロセスおよびパラメータ	2-19
従来のパラメータ・ファイルを使用して起動する場合の考慮事項	2-20
リモート・ノード上での 2 つのインスタンスの起動	2-20
インスタンス番号の設定	2-21

3 Real Application Clusters の記憶域コンポーネントの管理

Real Application Clusters におけるデータ・ファイルの追加	3-2
Real Application Clusters の自動 UNDO 管理	3-2
自動 UNDO 管理の使用	3-3
UNDO 表領域の切替え	3-4
システム・ロールバック・セグメント	3-5
パブリック・ロールバック・セグメントおよびプライベート・ロールバック・セグメント	3-5
手動 UNDO 管理を使用した自動 UNDO 管理のオーバーライド	3-6
Real Application Clusters での REDO ログ・ファイルの使用	3-7
Real Application Clusters でのトレース・ファイルとアラート・ファイルの管理	3-9
Real Application Clusters の順序番号ジェネレータ	3-9
CREATE SEQUENCE 文	3-10
CACHE オプション	3-10
ORDER オプション	3-11

4 サーバー制御ユーティリティ、SQL および SQL*Plus を使用した Real Application Clusters データベースの管理

SRVCTL を使用した Real Application Clusters 環境の管理	4-2
SRVCTL を使用した Real Application Clusters 管理の概要	4-2
グローバル・サービス・デーモン (GSD)	4-3
UNIX の GSD 実装	4-3
Windows の GSD 実装	4-4
SRVCTL 管理タスク	4-4
SRVCTL クラスタ・データベースのタスク	4-4
SRVCTL クラスタ・データベースの構成タスク	4-4
SRVCTL のコマンド構文	4-5
SRVCTL の構文の構成要素	4-5
SRVCTL の同時実行コマンド	4-5
SRVCTL 操作の停止	4-6
SRVCTL のエラー・メッセージ	4-6
SRVCTL の構文に共通する構成要素	4-6
コマンド構文	4-6
SRVCTL のコマンド	4-8
SRVCONFIG を使用した RAW デバイス構成のインポートとエクスポート	4-13
Oracle8i から Oracle9i への構成のアップグレード	4-13
GSD の管理	4-14
SQL および SQL*Plus を使用した Real Application Clusters データベースの管理	4-14
UNIX におけるクラスタ・モードでのデータベースの起動	4-14
Windows NT および Windows 2000 におけるクラスタ・モードでのデータベースの起動	4-15
RETRY を使用したクラスタ・モードでのデータベースのマウント	4-16
インスタンスの設定および接続	4-17
SET INSTANCE および SHOW INSTANCE コマンド	4-18
CONNECT コマンド	4-18
インスタンスの実行の確認	4-19
Real Application Clusters のインスタンスの停止	4-19
Real Application Clusters データベースの静止	4-20
インスタンスへの SQL および SQL*Plus コマンドの適用方法	4-21

第 II 部 Oracle Enterprise Manager を使用した Real Application Clusters の管理

5 Oracle Enterprise Manager を使用した Real Application Clusters データベースの管理

Oracle Enterprise Manager での管理の概要	5-2
コンソールの起動	5-2
ナビゲータ・ペインでのオブジェクトの表示	5-3
データベース固有のオブジェクト	5-4
インスタンス固有のオブジェクト	5-4
クラスタ・データベースのコンテキスト・メニューの使用	5-7
クラスタ・データベースの起動	5-8
クラスタ・データベースの停止	5-9
クラスタ・データベース操作結果の表示	5-11
「Status Details」 タブ	5-12
「Output」 タブ	5-14
クラスタ・データベースのステータス表示	5-15
「General」 タブ	5-16
「Status Details」 タブ	5-17
クラスタ・データベースまたはインスタンスに対するジョブの作成	5-18
ジョブの詳細の指定	5-19
「General」 タブ	5-19
「Tasks」 タブ	5-21
「Parameters」 タブ	5-21
クラスタ・データベースの停止タスクに関するパラメータ	5-23
クラスタ・データベース・イベントの登録	5-24

第 III 部 Real Application Clusters でのバックアップおよびリカバリ

6 Real Application Clusters 環境での Recovery Manager の使用

Real Application Clusters の Recovery Manager の構成 : 概要	6-2
Recovery Manager のスナップショット制御ファイルの位置の構成	6-2
Recovery Manager 制御ファイルの自動バックアップ機能の構成	6-3
Real Application Clusters 環境でのアーカイブ・ログの管理	6-4
Real Application Clusters 環境でのアーカイブ	6-4

Recovery Manager のアーカイブ構成スキーム	6-5
クラスタ・ファイル・システムのアーカイブ・スキーム	6-5
非 CFS のローカル・アーカイブ・スキーム	6-7
Real Application Clusters のアーカイブ・ログ・ファイルのフォーマットと接続先	6-11
Real Application Clusters でのアーカイブ・モードの変更	6-12
アーカイバ・プロセスの問合せ	6-13

7 Real Application Clusters 環境でのバックアップとリカバリの実行

Real Application Clusters でのバックアップの概要	7-2
Real Application Clusters 構成のターゲット・データベースへの接続	7-2
Real Application Clusters での Recovery Manager と Oracle Net	7-2
クラスタ・インスタンスへのチャンネル接続	7-3
ノード・アフィニティの認識	7-3
ファイルのアクセス可能性とバックアップ・メディア	7-4
Real Application Clusters の Recovery Manager バックアップ・スキーム	7-7
クラスタ・ファイル・システムのバックアップ・スキーム	7-7
非 CFS のバックアップ・スキーム	7-8
Real Application Clusters でのインスタンス・リカバリ	7-10
単一ノード障害	7-10
Real Application Clusters での複数ノード障害	7-11
Real Application Clusters でのインスタンス・リカバリのための制御ファイルと データ・ファイルへのアクセス	7-11
Real Application Clusters での Oracle インスタンス・リカバリの手順	7-12
Real Application Clusters でのメディア・リカバリの概要	7-13
Real Application Clusters の Recovery Manager リストア・スキーム	7-13
クラスタ・ファイル・システムのリストア・スキーム	7-13
非 CFS のリストア・スキーム	7-14
Real Application Clusters でのパラレル・リカバリ	7-16
Recovery Manager を使用したパラレル・リカバリ	7-16
SQL*Plus を使用したパラレル・リカバリ	7-17

第 IV 部 Real Application Clusters 環境の拡張性

8 Real Application Clusters でのノードとインスタンスの追加および インスタンスの削除

クラスタへのノードの追加	8-2
ノードを追加する手順の概要	8-2
クラスタウェア・レイヤーへのノードの追加	8-2
UNIX のクラスタウェア・レイヤーに対するノードの追加	8-3
Windows NT および Windows 2000 のクラスタウェア・レイヤーに対するノードの追加	8-4
Oracle レイヤーへのノードの追加	8-8
インスタンスの削除	8-19

第 V 部 リファレンス

A トラブルシューティング

トレース・ファイルの使用	A-2
バックグラウンド・スレッド・トレース・ファイル	A-2
ユーザー・スレッド・トレース・ファイル	A-3
アラート・ファイル	A-3
エラー・コール・トレース・スタック	A-4
オラクル社カスタマ・サポート・センターへの連絡方法	A-5
重大なエラー	A-5

B インスタンスとユーザーの、空きリストと空きリスト・グループ (オプション) への対応付け

インスタンスと空きリストの対応付け	B-2
既存の空きリスト・グループへの新しいインスタンスの割当て	B-2
FREELIST GROUPS および MAXINSTANCES	B-2
ユーザー・プロセスの、空きリストと空きリスト・グループへの対応付け	B-3
空きリストのある空き領域を管理するための SQL オプション	B-3
空きリスト・グループへのエクステンツの事前割当て	B-4

C Real Application Clusters 管理ツールのエラー・メッセージ

グローバル・サービス・デーモンのエラー・メッセージ (PRKA)	C-2
クラスタ・コマンドのエラー・メッセージ (PRKC)	C-4
Windows NT の Cluster Setup エラー・メッセージ (PRKI)	C-7
サーバー制御ユーティリティ (SRVCTL) のエラー・メッセージ (PRKO)	C-8
クラスタ・データベース管理のエラー・メッセージ (PRKP)	C-11
リポジトリのエラー・メッセージ (PRKR)	C-13

用語集

索引



2-1	従来のインスタンス固有の初期化ファイル	2-7
2-2	共通の初期化ファイルの例	2-9
3-1	REDO スレッド	3-7
5-1	データベースのサブフォルダ	5-5
5-2	「Cluster Database Instances」 フォルダ	5-6
5-3	「Status Details」 タブ	5-12
5-4	正常終了した停止の結果	5-13
5-5	正常終了した停止の結果	5-14
5-6	「General」 タブ	5-16
5-7	「Status Details」 タブ	5-17
5-8	「General」 タブ	5-20
5-9	「Parameters」 タブ	5-22
5-10	クラスタ・データベース停止パラメータ	5-23
5-11	Event Management の「Tests」 メニュー	5-25
6-1	クラスタ・ファイル・システムのアーカイブ・スキーム	6-6
6-2	非 CFS のローカル・アーカイブ・スキーム	6-8
7-1	Oracle インスタンスのリカバリ手順	7-12
8-1	Real Application Clusters の DBCA 「Welcome」 ページ	8-10
8-2	Database Configuration Assistant の「Operations」 ページ	8-11
8-3	Database Configuration Assistant の「Instance Management」 ページ	8-12
8-4	Database Configuration Assistant の「List of Cluster Databases」 ページ	8-13
8-5	Database Configuration Assistant の「List of Cluster Database Instances」 ページ	8-14
8-6	Database Configuration Assistant の「Adding an Instance」 ページ	8-15
8-7	Database Configuration Assistant の「Database Storage」 ページ	8-16
8-8	Database Configuration Assistant の「Summary」 ダイアログ・ボックス	8-18
8-9	DBCA の「Instance Management」 ページ	8-20
8-10	Database Configuration Assistant の「List of Cluster Databases」 ページ	8-21
8-11	Database Configuration Assistant の「List of Cluster Database Instances」 ページ	8-22
8-12	Database Configuration Assistant の「Error」 ダイアログ・ボックス	8-22
8-13	Database Configuration Assistant の「Summary」 ダイアログ・ボックス	8-23
8-14	Database Configuration Assistant の「Confirmation」 ダイアログ・ボックス	8-24

表

2-1	オプションのパラメータ・ファイルのネーミング規則および説明	2-7
2-2	sid およびインスタンス名の例	2-12
2-3	クラスタ内のインスタンスの初期化パラメータに関する注意	2-15
4-1	SRVCTL で共通の動詞	4-6
4-2	SRVCTL で共通の名詞	4-7
4-3	SRVCTL 構文のフラグ	4-7
4-4	SRVCTL で共通のコマンド・オプション	4-7
4-5	SRVCTL Add コマンド固有のオプション	4-8
4-6	SRVCTL Start コマンド固有のオプション	4-11
4-7	SRVCTL Stop コマンド固有のオプション	4-12
4-8	V\$ACTIVE_INSTANCES 列の説明	4-19
4-9	SQL*Plus コマンドのインスタンスへの適用方法	4-21
5-1	Real Application Clusters のコンテキスト・メニューの機能	5-7
5-2	クラスタ・データベースの起動タイプ	5-8
5-3	停止タイプ	5-9
5-4	コンポーネントの状態	5-13
5-5	「General」 タブのフィールド	5-16
5-6	「Create Job」 プロパティ・シートに含まれるタブ	5-19
5-7	「General」 タブのオプション	5-20
5-8	起動の「Parameters」 タブ	5-22
5-9	停止の「Parameters」 タブ	5-23
6-1	非 CFS のローカル・アーカイブ・スキームでのログの位置	6-10
6-2	共有読取りローカル・アーカイブ・スキームに関する NFS の構成	6-11
6-3	アーカイブ REDO ログ・ファイル名のフォーマット・パラメータ	6-11
A-1	バックグラウンド・スレッド・トレース・ファイル	A-2

はじめに

このマニュアルでは、シングル・**インスタンス**の管理タスクを支援する **Real Application Clusters** 固有の管理タスクについて説明します。このマニュアルの情報は、すべてのオペレーティング・システム上で動作する **Real Application Clusters** に適用されます。必要に応じて、プラットフォーム固有のマニュアルも参照してください。

内容は次のとおりです。

- [対象読者](#)
- [このマニュアルの構成](#)
- [関連文書](#)
- [表記規則](#)

対象読者

このマニュアルは、Real Application Clusters を管理するネットワーク管理者またはデータベース管理者を対象としています。

このマニュアルは、『Oracle9i データベース管理者ガイド』に記載されているシングル・インスタンスの Oracle データベース管理手順、および『Oracle9i Real Application Clusters 概要』に記載されている Real Application Clusters での処理の概念を理解していることを前提としています。また、『Oracle9i Real Application Clusters セットアップおよび構成』およびプラットフォーム固有のマニュアルを参照して、Real Application Clusters をインストールしておく必要もあります。

このマニュアルの構成

このマニュアルの構成は、次のとおりです。

第 I 部：「Real Application Clusters の管理の概要」

第 I 部では、Real Application Clusters の管理および初期管理タスクについて説明します。

第 1 章「Real Application Clusters の管理の概要」

この章では、Real Application Clusters ソフトウェアの管理タスクについて説明します。

第 2 章「Real Application Clusters 環境でのパラメータの管理」

この章では、パラメータ・ファイルおよび Real Application Clusters 固有のパラメータについて説明します。

第 3 章「Real Application Clusters の記憶域コンポーネントの管理」

この章では、Real Application Clusters の記憶域コンポーネントを管理する方法について説明します。

第 4 章「サーバー制御ユーティリティ、SQL および SQL*Plus を使用した Real Application Clusters データベースの管理」

この章では、サーバー制御ユーティリティ (SRVCTL)、SQL および SQL*Plus を使用して、Real Application Clusters データベースを管理する方法について説明します。

第 II 部：Oracle Enterprise Manager を使用した Real Application Clusters の管理

第 II 部では、Oracle Enterprise Manager を使用して、Real Application Clusters データベースを管理する方法について説明します。

第 5 章「Oracle Enterprise Manager を使用した Real Application Clusters データベースの管理」

この章では、Oracle Enterprise Manager を使用して、Real Application Clusters データベースを管理する方法について説明します。

第 III 部：Real Application Clusters でのバックアップおよびリカバリ

第 III 部では、Real Application Clusters のバックアップ手順およびリカバリ手順について説明します。

第 6 章「Real Application Clusters 環境での Recovery Manager の使用」

この章では、Real Application Clusters 用に Recovery Manager を構成する方法について説明します。

第 7 章「Real Application Clusters 環境でのバックアップとリカバリの実行」

この章では、Real Application Clusters データベースのバックアップ方法およびリカバリ方法について説明します。

第 IV 部：Real Application Clusters 環境の拡張性

第 IV 部では、Real Application Clusters 環境を拡張するためのノードおよびインスタンスの追加について説明します。

第 8 章「Real Application Clusters でのノードとインスタンスの追加およびインスタンスの削除」

この章では、Oracle Universal Installer および Database Configuration Assistant を使用して、Real Application Clusters でノードとインスタンスを追加する方法およびインスタンスを削除する方法について説明します。

第 V 部：リファレンス

第 V 部では、Real Application Clusters のリファレンス情報を示します。

付録 A「トラブルシューティング」

この付録では、Oracle のインストールに関する問題のトラブルシューティングにトレース・ファイルを使用する方法について説明します。また、オラクル社カスタマ・サポート・センターに連絡する方法についても説明します。

付録 B「インスタンスとユーザーの、空きリストと空きリスト・グループ（オプション）への対応付け」

この付録では、インスタンスとユーザーを空きリストと空きリスト・グループに対応付ける方法について説明します。また、SQL 固有の空きリスト・オプションおよび空きリスト・グループへのエクステンツの事前割当てについても説明します。

付録 C 「Real Application Clusters 管理ツールのエラー・メッセージ」

この付録では、Real Application Clusters 管理ツールのエラー・メッセージについて説明します。

用語集

用語集では、このマニュアルで使用する用語およびこのマニュアルに関連する用語の定義を示します。

関連文書

詳細は、次の Oracle マニュアルを参照してください。

- 『Oracle9i Real Application Clusters 概要』
- 『Oracle9i Real Application Clusters セットアップおよび構成』
- 『Oracle9i Real Application Clusters 配置およびパフォーマンス』
- 『Oracle9i Real Application Clusters Real Application Clusters Guard I - Concepts and Administration』
- 『Oracle9i Real Application Clusters Guard II Concepts, Installation, and Administration』
- Oracle9i Real Application Clusters Guard の、プラットフォーム固有のインストレーション・ガイド

インストレーション・ガイド

- 『Oracle9i for UNIX Systems インストレーション・ガイド』（AIX-Based Systems、Compaq Tru64、HP 9000 Series HP-UX、Linux Intel、Sun Solaris 共通）
- 『Oracle9i Database for Windows インストレーション・ガイド』
- Oracle Real Application Clusters Guard I のインストレーション・ガイド（プラットフォーム固有のインストレーション・ガイドが何種類かあります）
- 『Oracle9i Real Application Clusters Guard II Concepts, Installation, and Administration』

オペレーティング・システム固有の管理ガイド

- 『Oracle9i for UNIX Systems 管理者リファレンス』 (AIX-Based Systems、Compaq Tru64、HP 9000 Series HP-UX、Linux Intel、Sun Solaris 共通)
- 『Oracle9i Database for Windows 管理者ガイド』
- 『Oracle9i Real Application Clusters Real Application Clusters Guard I - Concepts and Administration』
- 『Oracle9i Real Application Clusters Guard II Concepts, Installation, and Administration』

Oracle9i Real Application Clusters の管理ガイド

- 『Oracle Enterprise Manager 管理者ガイド』
- 『Oracle Enterprise Manager Oracle Diagnostics Pack スタート・ガイド』

共通のマニュアル

- 『Oracle9i データベース概要』
- 『Oracle9i Net Services 管理者ガイド』
- 『Oracle9i データベース新機能』
- 『Oracle9i データベース・リファレンス』

このマニュアルの多くの例で、Oracle のインストール時にデフォルトとしてインストールされるシード・データベースのサンプル・スキーマを使用しています。これらのスキーマがどのように作成されているか、およびその使用方法については、『Oracle9i サンプル・スキーマ』を参照してください。

リリース・ノート、インストレーション・マニュアル、ホワイト・ペーパーまたはその他の関連文書は、OTN-J (Oracle Technology Network Japan) に接続すれば、無償でダウンロードできます。OTN-J を使用するには、オンラインでの登録が必要です。次の URL で登録できます。

<http://otn.oracle.co.jp/membership/>

OTN-J のユーザー名とパスワードを取得済みであれば、次の OTN-J Web サイトの文書セクションに直接接続できます。

<http://otn.oracle.co.jp/document/>

表記規則

このマニュアル・セットの本文とコード例に使用されている表記規則について説明します。

- 本文の表記規則
- コード例の表記規則
- Windows オペレーティング・システムの表記規則

本文の表記規則

本文には、特別な用語が一目でわかるように様々な表記規則が使用されています。次の表は、本文の表記規則と使用例を示しています。

表記規則	意味	例
太字	太字は、本文中に定義されている用語または用語集に含まれている用語、あるいはその両方を示します。	この句を指定する場合は、 索引構成表 を作成します。
固定幅フォントの大文字	固定幅フォントの大文字は、システムにより指定される要素を示します。この要素には、パラメータ、権限、データ型、Recovery Manager キーワード、SQL キーワード、SQL*Plus またはユーティリティ・コマンド、パッケージとメソッドの他、システム指定の列名、データベース・オブジェクトと構造体、ユーザー名、およびロールがあります。	この句は、NUMBER 列に対してのみ指定できます。 BACKUP コマンドを使用すると、データベースのバックアップを作成できます。 USER_TABLES データ・ディクショナリ・ビューの TABLE_NAME 列を問い合わせます。 DBMS_STATS.GENERATE_STATS プロシージャを使用します。
固定幅フォントの小文字	固定幅フォントの小文字は、実行可能ファイル、ファイル名、ディレクトリ名およびサンプルのユーザー指定要素を示します。この要素には、コンピュータ名とデータベース名、ネット・サービス名、接続識別子の他、ユーザー指定のデータベース・オブジェクトと構造体、列名、パッケージとクラス、ユーザー名とロール、プログラム・ユニット、およびパラメータ値があります。 注意: 一部のプログラム要素には、大文字と小文字の両方が使用されます。この場合は、記載されているとおりに入力してください。	「sqlplus」と入力して SQL*Plus を開きます。 パスワードは orapwd ファイルに指定されています。 データ・ファイルと制御ファイルのバックアップ /disk1/oracle/dbs ディレクトリに作成します。 department_id、department_name および location_id の各列は、hr.departments 表にあります。 初期化パラメータ QUERY_REWRITE_ENABLED を true に設定します。 oe ユーザーで接続します。 これらのメソッドは JRepUtil クラスに実装されます。

表記規則	意味	例
固定幅フォントの 小文字の イタリック	固定幅フォントの小文字のイタリックは、 プレースホルダまたは変数を示します。	<i>parallel_clause</i> を指定できます。 <code>Uold_release.SQL</code> を実行します。 <i>old_release</i> はアップグレード前にインストールしたリリースです。

コード例の表記規則

コード例は、SQL、PL/SQL、SQL*Plus または他のコマンドラインを示します。次のように、固定幅フォントで、通常の本文とは区別して記載されています。

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

次の表は、コード例の記載上の表記規則とその使用例を示しています。

表記規則	意味	例
[]	大カッコで囲まれている項目は、1つ以上のオプション項目を示します。大カッコ自体は入力しないでください。	<code>DECIMAL (digits [, precision])</code>
{ }	中カッコで囲まれている項目は、そのうちの1つのみが必要であることを示します。中カッコ自体は入力しないでください。	<code>{ENABLE DISABLE}</code>
	縦線は、大カッコまたは中カッコ内の複数の選択肢を区切るために使用します。オプションのうち1つを入力します。縦線自体は入力しないでください。	<code>{ENABLE DISABLE}</code> <code>[COMPRESS NOCOMPRESS]</code>
...	水平の省略記号は、次のどちらかを示します。 <ul style="list-style-type: none"> ■ 例に直接関係のないコード部分が省略されていること ■ コードの一部が繰り返し可能であること 	<code>CREATE TABLE ... AS subquery;</code> <code>SELECT col1, col2, ... , coln FROM employees;</code>

表記規則	意味	例
.	垂直の省略記号は、例に直接関係のない数行のコードが省略されていることを示します。	<pre>SQL> SELECT NAME FROM V\$DATAFILE; NAME ----- /fs1/dbs/tbs_01.dbf /fs1/dbs/tbs_02.dbf . . . /fs1/dbs/tbs_09.dbf 9 rows selected.</pre>
その他の表記	大カッコ、中カッコ、縦線および省略記号以外の記号は、示されているとおりに入力してください。	<pre>acctbal NUMBER(11,2); acct CONSTANT NUMBER(4) := 3;</pre>
イタリック	イタリックの文字は、特定の値を指定する必要があるプレースホルダまたは変数を示します。	<pre>CONNECT SYSTEM/system_password DB_NAME = database_name</pre>
大文字	大文字は、システムにより指定される要素を示します。これらの用語は、ユーザー定義用語と区別するために大文字で記載されています。大カッコで囲まれている場合を除き、記載されているとおりの順序とスペルで入力してください。ただし、この種の用語は大 / 小文字区別がないため、小文字でも入力できます。	<pre>SELECT last_name, employee_id FROM employees; SELECT * FROM USER_TABLES; DROP TABLE hr.employees;</pre>
小文字	小文字は、ユーザー指定のプログラム要素を示します。たとえば、表名、列名またはファイル名を示します。 注意： 一部のプログラム要素には、大文字と小文字の両方が使用されます。この場合は、記載されているとおりに入力してください。	<pre>SELECT last_name, employee_id FROM employees; sqlplus hr/hr CREATE USER mjones IDENTIFIED BY ty3MU9;</pre>

Windows オペレーティング・システムの表記規則

次の表は、Windows オペレーティング・システムの表記規則と使用例を示しています。

表記規則	意味	例
「スタート」→を選択	プログラムの起動方法。	Oracle Database Configuration Assistant を起動するには、「スタート」→「プログラム」→「Oracle - HOME_NAME」→「Configuration and Migration Tools」→「Database Configuration Assistant」を選択します。
ファイル名とディレクトリ名	ファイル名とディレクトリ名では、大/小文字は区別されません。特殊文字のうち、左山カッコ (<)、右山カッコ (>)、コロン (:)、二重引用符 (")、スラッシュ (/)、パイプ () およびハイフン (-) は使用できません。特殊文字のうち円記号 (¥) は、引用符で囲まれている場合にも要素のセパレータとして扱われます。ファイル名が¥¥で始まる場合、Windows では汎用命名規則を使用しているものとみなされます。	c:¥winnt"¥"system32 は C:¥WINNT¥SYSTEM32 と同じです。
C:¥>	現行のハード・ディスク・ドライブを示す Windows コマンド・プロンプトを表します。コマンド・プロンプトでのエスケープ文字はカレット (^) です。プロンプトには、作業中のサブディレクトリが反映されます。このマニュアルでは、コマンド・プロンプトと呼んでいます。	C:¥oracle¥oradata>
特殊文字	特殊文字のうち円記号 (¥) は、Windows のコマンド・プロンプトで二重引用符 (") のエスケープ文字として必要な場合があります。カッコと一重引用符 (') には、エスケープ文字は不要です。エスケープ文字と特殊文字の詳細は、Windows オペレーティング・システムのマニュアルを参照してください。	C:¥>exp scott/tiger TABLES=emp QUERY=¥"WHERE job='SALESMAN' and sal<1600¥" C:¥>imp SYSTEM/password FROMUSER=scott TABLES=(emp, dept)
HOME_NAME	Oracle ホーム名を表します。ホーム名は、英数字で 16 文字以内です。ホーム名に使用できる特殊文字は、アンダースコアのみです。	C:¥> net start OracleHOME_NAME_TNSListener

表記規則	意味	例
<code>ORACLE_HOME</code> と <code>ORACLE_BASE</code>	<p>Oracle8 リリース 8.0 以前では、Oracle コンポーネントをインストールすると、すべてのサブディレクトリはデフォルトで次のいずれかの名前のトップレベルの <code>ORACLE_HOME</code> ディレクトリに置かれていました。</p> <ul style="list-style-type: none"> ■ Windows NT の場合は <code>C:\orant</code> ■ Windows 98 の場合は <code>C:\orawin98</code> <p>このリリースは、Optimal Flexible Architecture (OFA) のガイドラインに準拠しています。すべてのサブディレクトリがトップレベルの <code>ORACLE_HOME</code> ディレクトリにあるとはかぎりません。<code>ORACLE_BASE</code> というトップレベル・ディレクトリがあり、デフォルトでは <code>C:\oracle</code> です。他の Oracle ソフトウェアがインストールされていないコンピュータに最新の Oracle リリースをインストールする場合、最初の Oracle ホーム・ディレクトリのデフォルト設定は <code>C:\oracle\orann</code> で、<code>nn</code> は最新のリリース番号です。Oracle ホーム・ディレクトリは、<code>ORACLE_BASE</code> の直下にあります。</p> <p>このマニュアルでは、すべてのディレクトリ・パスの例が、OFA の表記規則に従って示されています。</p>	<code>%ORACLE_HOME%\rdbms\admin</code> ディレクトリにアクセスします。

Real Application Clusters の管理の新機能

ここでは、**Real Application Clusters** の管理に関する Oracle9i リリース 2 (9.2) の新機能について説明します。

関連項目： Real Application Clusters の新機能の詳細は、『Oracle9i Real Application Clusters 概要』を参照してください。

内容は次のとおりです。

- **Real Application Clusters** データベースの管理に関する Oracle9i リリース 2 (9.2) の新機能

Real Application Clusters データベースの管理に関する Oracle9i リリース 2 (9.2) の新機能

■ Real Application Clusters の Recovery Manager の拡張

Real Application Clusters の Recovery Manager (RMAN) は、オートロケーション機能を使用して、Oracle Real Application Clusters 構成のノードの中から、バックアップまたはリストアの対象となるファイルにアクセスできるノードを、自動的に検出します。Recovery Manager のオートロケーションの対象になるのは、次のファイルです。

- バックアップ・ピースおよびバックアップ / リストア時のデータ・ファイルまたは制御ファイルのコピー
- バックアップ時のアーカイブ REDO ログ

Recovery Manager は、割り当てられているチャンネルに異なる PARMS または CONNECT 文字列がある場合は、常にオートロケーション機能を使用可能にします。

■ Real Application Clusters Guard II

Real Application Clusters Guard II は、包括的なワークロード管理をサポートし、Real Application Clusters データベースおよびそのアプリケーションの高可用性を維持します。標準のサーバー制御ユーティリティ (SRVCTL) および Database Configuration Assistant (DBCA) を使用して、Real Application Clusters Guard II を管理できます。

関連項目： SRVCTL および DBCA を使用した Real Application Clusters Guard II の管理方法については、『Oracle9i Real Application Clusters Guard II Concepts, Installation, and Administration』を参照してください。

■ Real Application Clusters 管理ツールのエラー・メッセージ

Real Application Clusters 管理ツールのエラー・メッセージの説明が、[付録 C 「Real Application Clusters 管理ツールのエラー・メッセージ」](#)に追加されました。この付録には、次のメッセージ・グループがあります。

- グローバル・サービス・デーモンのエラー・メッセージ (PRKA)
- クラスタ・コマンドのエラー・メッセージ (PRKC)
- Windows NT の Cluster Setup エラー (PRKI)
- サーバー制御ユーティリティ (SRVCTL) のエラー・メッセージ (PRKO)
- クラスタ・データベース管理メッセージ (PRKP)
- リポジトリ・エラー (PRKR)

第 I 部

Real Application Clusters の管理の概要

第 I 部では、Real Application Clusters でのパラメータの管理、記憶域コンポーネントの管理および一般的な管理手順について説明します。第 I 部に含まれる章は、次のとおりです。

- 第 1 章「Real Application Clusters の管理の概要」
- 第 2 章「Real Application Clusters 環境でのパラメータの管理」
- 第 3 章「Real Application Clusters の記憶域コンポーネントの管理」
- 第 4 章「サーバー制御ユーティリティ、SQL および SQL*Plus を使用した Real Application Clusters データベースの管理」

Real Application Clusters の管理の概要

この章では、**Real Application Clusters** の管理で使用するプロセスおよびコンポーネントの概要について説明します。内容は次のとおりです。

- **Real Application Clusters データベースの管理**
- **Real Application Clusters でのパラメータ管理**
- **Real Application Clusters での記憶域管理**
- **管理に関する一般的な問題**
- **Real Application Clusters 環境の Oracle Enterprise Manager**
- **Real Application Clusters でのバックアップおよびリカバリ**
- **Real Application Clusters の拡張性**
- **トラブルシューティング**

Real Application Clusters データベースの管理

このマニュアルでは、Real Application Clusters 固有の管理作業について説明します。これらの作業は、『Oracle9i データベース管理者ガイド』に記載されている作業を行ったうえでを行います。このマニュアルを使用して Real Application Clusters を管理する前に、『Oracle9i データベース管理者ガイド』をお読みください。次の項では、管理に関する Real Application Clusters 固有の問題の概要を示します。

Real Application Clusters でのパラメータ管理

クラスタ・データベース全体のパラメータ設定を確立し、**サーバー・パラメータ・ファイル**を使用してインスタンス固有の設定を指定できます。または、クライアント側のパラメータ・ファイルを使用して、パラメータ設定を管理できます。

Real Application Clusters には、3つのタイプのパラメータがあります。1つ以上のインスタンスに異なる値を指定できるパラメータがあります。すべてのインスタンスに対して同じ値または一意の値を指定する必要があるパラメータもあります。パラメータの設定は、Real Application Clusters データベースでの**インスタンス**の起動にも影響します。

関連項目： [第2章「Real Application Clusters 環境でのパラメータの管理」](#)

Real Application Clusters での記憶域管理

データ・ファイル管理や UNDO 表領域管理など、考慮する必要がある Real Application Clusters 固有の記憶域管理に関する問題があります。Real Application Clusters インスタンスでの REDO ログ・ファイルの管理および REDO スレッドの使用についても理解する必要があります。自動 UNDO 管理によって、UNDO 表領域の記憶域管理が簡素化されます。

関連項目： [第3章「Real Application Clusters の記憶域コンポーネントの管理」](#)

管理に関する一般的な問題

Real Application Clusters インスタンスおよびデータベースは、SQL、SQL*Plus およびコマンドライン・ユーティリティを使用して起動および停止できます。Oracle Net を経由してリモート・インスタンスを設定および接続し、管理操作を実行することもできます。ただし、SQL および SQL*Plus コマンドは、インスタンスに対して、ローカルおよびグローバルに影響する可能性があります。

SRVCTL ユーティリティを使用して、Real Application Clusters データベースで多くの管理作業を実行することもできます。SRVCTL を使用して、構成情報の管理などの他の作業を実行したり、インスタンスを削除、名前の変更および移動することもできます。

関連項目： 第4章「サーバー制御ユーティリティ、SQL および SQL*Plus を使用した Real Application Clusters データベースの管理」

Real Application Clusters 環境の Oracle Enterprise Manager

Real Application Clusters でのほとんどの管理作業は、**Oracle Enterprise Manager** を使用して実行できます。Oracle Enterprise Manager を使用すると、インスタンスの起動および停止、パフォーマンスの監視、ジョブのスケジューリング、バックアップおよびリカバリ操作を行えます。

関連項目： 第5章「Oracle Enterprise Manager を使用した Real Application Clusters データベースの管理」

Real Application Clusters でのバックアップおよびリカバリ

Real Application Clusters で使用するために、**Recovery Manager (RMAN)** を構成できます。Recovery Manager は、各制御ファイルのバックアップを自動的にを行い、チャンネル設定を自動的に保存するように構成することもできます。

Recovery Manager およびオペレーティング・システムのコマンドを使用して、Real Application Clusters のオープン・バックアップとクローズ・バックアップの両方を実行できます。Real Application Clusters のリカバリ・オプションには、完全リカバリ、不完全リカバリおよびブロック・メディア・リカバリがあります。パラレル・リカバリも使用できます。

関連項目：

- 第6章「Real Application Clusters 環境での Recovery Manager の使用」
- 第7章「Real Application Clusters 環境でのバックアップとリカバリの実行」

Real Application Clusters の拡張性

Real Application Clusters では、増加したシステム要件を満たしたり、障害の発生したハードウェアを交換するために、**ノード**およびインスタンスを追加したり、インスタンスを削除できます。ノードを動的に追加したり、インスタンスを追加および削除するには、**Oracle Universal Installer (OUI)** および **Database Configuration Assistant (DBCA)** を使用します。

関連項目： 第 8 章「Real Application Clusters でのノードとインスタンスの追加およびインスタンスの削除」

トラブルシューティング

Oracle は、処理情報を様々なトレース・ファイルおよびログ・ファイルに記録します。これらのファイルで、トラブルシューティングに必要な情報を参照できます。これらのファイルは、上書きされないように定期的にコピーする必要があります。

関連項目： トレース・ファイルおよびログ・ファイルの詳細については、第 A 章「トラブルシューティング」を参照してください。

Real Application Clusters 環境での パラメータの管理

この章では、サーバー・パラメータ・ファイルおよびクライアント側のパラメータ・ファイルの管理方法について説明します。また、**Real Application Clusters** でのパラメータの使用方法、およびパラメータの起動プロセスへの影響についても説明します。内容は次のとおりです。

- **Real Application Clusters** データベースのサーバー・パラメータ・ファイルの管理
- クライアント側のパラメータ・ファイルの使用
- **Real Application Clusters** のパラメータの設定
- **Real Application Clusters** のパラメータ・タイプ
- クラスタのパラメータに関する考慮事項
- **Real Application Clusters** の起動プロセスおよびパラメータ

関連項目： **Real Application Clusters** でのサーバー・パラメータ・ファイルの作成および構成については、『Oracle9i Real Application Clusters セットアップおよび構成』を参照してください。

Real Application Clusters データベースのサーバー・パラメータ・ファイルの管理

Oracle は、[サーバー・パラメータ・ファイル](#)のパラメータ設定を使用してデータベース・リソースを制御します。従来のクライアント側のパラメータ・ファイルも使用できますが、サーバー・パラメータ・ファイルを使用することをお勧めします。この項では、サーバー・パラメータ・ファイルの管理について説明します。内容は次のとおりです。

- [サーバー・パラメータ・ファイルのバックアップ](#)
- [Real Application Clusters のサーバー・パラメータ・ファイルのパラメータ値の設定](#)
- [下位互換性のためのサーバー・パラメータ・ファイルのエクスポート](#)
- [サーバー・パラメータ・ファイルのパラメータ設定](#)

サーバー・パラメータ・ファイルのバックアップ

リカバリのために、サーバー・パラメータ・ファイルのコピーを定期的に作成することをお勧めします。これには、CREATE PFILE FROM SPFILE 文を使用します。

データベースのサーバー・パラメータ・ファイルは、クライアント側の[初期化パラメータ・ファイル](#)を使用して[インスタンス](#)を起動することによってリカバリすることもできます。その後、CREATE SPFILE 文を使用してサーバー・パラメータ・ファイルを再作成します。

関連項目： CREATE PFILE 文および CREATE SPFILE 文の詳細については、『Oracle9i SQL リファレンス』を参照してください。

Real Application Clusters のサーバー・パラメータ・ファイルのパラメータ値の設定

Oracle は、サーバー・パラメータ・ファイルの値を、[Oracle Enterprise Manager](#)または ALTER SYSTEM SET 文を使用して変更するパラメータ設定へと自動的に更新します。さらに、ALTER SYSTEM RESET 構文を使用すると、サーバー・パラメータ・ファイルのパラメータ設定、および手動で設定したパラメータの影響を取り消すことができます。

たとえば、次のエントリを含むサーバー・パラメータ・ファイルでインスタンスを起動するとします。

```
*.OPEN_CURSORS=500
proddb1.OPEN_CURSORS=1000
```

注意： この例では、サンプル設定をテキストとして表示しますが、サーバー・パラメータ・ファイルはバイナリ・ファイルです。

先にデータベース全体のパラメータが 500 に設定されていますが、*sid* が *proddb1* のインスタンスでは、パラメータは 1000 に設定されます。パラメータ・ファイル内のインスタンス固有のパラメータ設定によって、データベース全体の設定の変更が防止されます。したがって、インスタンス *proddb1* のデータベース管理者 (DBA) は、そのインスタンスのパラメータ設定を完全に制御できます。この 2 種類の設定は、パラメータ・ファイル内でいずれの順序でも指定できます。

別の DBA が次のコマンドを実行するとします。

```
ALTER SYSTEM SET OPEN_CURSORS=1500 sid='*' SCOPE=MEMORY;
```

Oracle は、*sid* が *proddb1* 以外のすべてのインスタンスの設定を更新します。次のコマンドを実行して *sid* が *proddb1* のインスタンスのパラメータ設定を変更すると、パラメータは、それ以降、他のインスタンスによって設定された ALTER SYSTEM の値を受け入れるようになります。

```
ALTER SYSTEM RESET OPEN_CURSORS SCOPE=MEMORY sid='proddb1';
```

次に、他のインスタンスで次のコマンドを実行すると、*sid* が *proddb1* であるインスタンスの新しい設定も 2000 になります。

```
ALTER SYSTEM SET OPEN_CURSORS=2000 sid='*' SCOPE=MEMORY;
```

サーバー・パラメータ・ファイルに次のエントリが含まれるとします。

```
proddb1.OPEN_CURSORS=1000  
*.OPEN_CURSORS=500
```

次のコマンドを実行します。

```
ALTER SYSTEM RESET OPEN_CURSORS SCOPE=SPFILE sid='proddb1';
```

Oracle はサーバー・パラメータ・ファイルの最初のエントリを無視します。

クラスタ・データベースのデータベース全体でパラメータをデフォルト値にリセットするには、次の構文を入力します。

```
ALTER SYSTEM RESET OPEN_CURSORS SCOPE=SPFILE sid='*';
```

注意： ALTER SYSTEM 文のすべての変更がサーバー・パラメータ・ファイルに記録されるわけではありません。ALTER SYSTEM 文によっては、メモリー内のパラメータ設定のみが変更されます。

下位互換性のためのサーバー・パラメータ・ファイルのエクスポート

Oracle のクラスタ・ソフトウェアである Real Application Clusters の以前のリリースのパラメータ・ファイルに戻したり、サーバー・パラメータ・ファイルから従来のクライアント側のパラメータ・ファイルに切り替えることができます。これを行うには、CREATE PFILE 文の FROM オプションを使用します。次の構文を使用して、サーバー・パラメータ・ファイルの内容を Oracle9i リリース 1 (9.0.1) より前のリリースのパラメータ・ファイルにエクスポートします。

```
CREATE PFILE[= 'pfile-name'] FROM SPFILE [= 'raw_device_name'];
```

この文では、サーバー・パラメータ・ファイルの内容が、Oracle9i リリース 1 (9.0.1) より前のリリースの初期化パラメータ・ファイル pfile-name にエクスポートされます。PFILE または SPFILE ファイルの名前を指定しない場合、Oracle はプラットフォーム固有のデフォルトの PFILE 名および SPFILE 名を使用します。

Oracle は、PFILE をテキスト・ファイルとしてサーバー上に作成します。このファイルには、すべてのインスタンスのすべてのパラメータ設定が含まれます。オーバーライドされるエントリは、sid.parameter=value として表示されます。PFILE には、パラメータに対応付けられたすべてのコメントも含まれます。コメントは、パラメータ設定と同じ行に表示されます。すべての sid 固有のエントリは、インスタンス固有のパラメータ・ファイルに移動し、sid の指定を削除する必要があります。CREATE PFILE 文を実行するには、DBA 権限が必要です。

CREATE PFILE 文は、次の場合に実行できます。

- サーバー・パラメータ・ファイルのバックアップを作成する場合。**Recovery Manager (RMAN)** は、サーバー・パラメータ・ファイルのバックアップを取らないことに注意してください。
- 診断のために、現在、インスタンスに使用されているすべてのパラメータ値のリストを取得する場合。これには、SHOW PARAMETER コマンドを使用するか、または V\$PARAMETER 表を問い合わせます。
- サーバー・パラメータ・ファイルのエクスポートして出力ファイルを編集し、サーバー・パラメータ・ファイルを再作成して変更する場合。

サーバー・パラメータ・ファイルのパラメータ設定

サーバー・パラメータ・ファイルにインスタンス固有のパラメータ値を設定するには、*sid* 指定子を使用します。次に例を示します。

```
proddb1.OPEN_CURSORS = 1000
proddb2.OPEN_CURSORS = 1500
```

これによって、インスタンス *proddb1* の `OPEN_CURSORS` が 1000 に設定され、インスタンス *proddb2* の `OPEN_CURSORS` が 1500 に設定されます。Real Application Clusters データベースでは、これらのエントリが *sid* 固有のエントリとして認識されます。*sid* が *proddb1* のインスタンスを起動した場合は 1000 の値がパラメータに適用され、*sid* が *proddb2* のインスタンスを起動した場合は 1500 の値がパラメータに適用されます。

次のパラメータ・ファイルのエントリについて考えます。

```
*.DB_FILE_MULTIBLOCK_READ_COUNT=16
```

これによって、すべてのインスタンスで `DB_FILE_MULTIBLOCK_READ_COUNT` パラメータの値が 16 に設定されます。*** (アスタリスク) の *sid* はグローバルであるとみなされ、パラメータ設定に指定された値はすべての *sid* に適用されるため、すべてのインスタンスの `DB_FILE_MULTIBLOCK_READ_COUNT` パラメータの値は 16 になります。

サーバー・パラメータ・ファイルは、リリース 1 (9.0.1) より前のリリースの、次の構文をサポートすることに注意してください。

```
parameter1 = value1
```

また、`parameter1` は、*sid* にかかわらず `value1` の値になります。特定の *sid* のパラメータは、次のとおりオーバーライドできます。

```
OPEN_CURSORS = 1000
proddb1.OPEN_CURSORS = 1500
```

この場合、*sid* が *proddb1* のインスタンスを除く、すべてのインスタンスの `OPEN_CURSORS` の値は 1000 になり、*sid* が *proddb1* のインスタンスの値は 1500 になります。

サーバー・パラメータ・ファイルのコメントの指定

パラメータ設定のコメントは、パラメータ設定と同じ行に指定します。たとえば、`init.ora` に次の行が含まれるとします。

```
# first comment
OPEN_CURSORS = value # second comment
```

文字列 `second comment` は、`OPEN_CURSORS` の設定に対応付けられています。このコメントは、`V$PARAMETER` ビューおよび `V$PARAMETER2` ビューに表示されます。また、前述の例のエントリ `#first comment` などのコメントも表示されます。

注意： サーバー・パラメータ・ファイルに対して値を追加または変更する場合は注意してください。不適切な操作を行うと、ファイルが破損してデータベースを起動できなくなる可能性があります。ファイルを修復するには PFILE を作成し、SPFILE を再生成する必要があります。

クライアント側のパラメータ・ファイルの使用

クライアント側の 1 つ以上のパラメータ・ファイルを使用して、Real Application Clusters でパラメータ設定を管理できます。STARTUP コマンドに PFILE を指定しない場合、デフォルトで、サーバー・パラメータ・ファイルが使用されます。

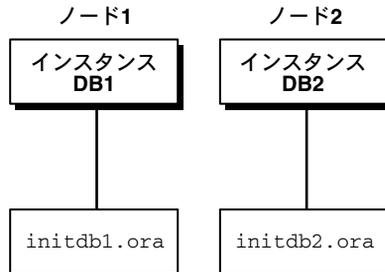
グローバル・パラメータをインスタンス固有のパラメータ・ファイル内に設定できます。これを行うには、インスタンス固有のすべてのパラメータ・ファイルのグローバル・パラメータを同じ値に設定する必要があります。グローバル・パラメータを 1 つのファイルに保持し、**IFILE** パラメータを使用してそのファイルを示すようにすることもできます。

注意： クライアント側のパラメータ・ファイルを使用する場合、Oracle が自動チューニングしたパラメータの変更は、停止後に失われます。また、クライアント側のパラメータ・ファイルを使用すると、パラメータ管理のオーバーヘッドが増加します。

クライアント側のパラメータ・ファイルのネーミング規則

クライアント側のパラメータ・ファイル名は、各インスタンスの *sid* および**グローバル・データベース名**に基づいて指定します。たとえば、各インスタンス固有のパラメータ・ファイルには **initsid.ora** という名前を付けます。*sid* は、インスタンスのシステム識別子です。共通のパラメータ・ファイルには、**initdbname.ora** という名前を付けます。*db_name* は、[図 2-1](#) に示すとおり、Real Application Clusters データベースのデータベース名です。

図 2-1 従来のインスタンス固有の初期化ファイル



パラメータ・ファイルには、インスタンス固有のパラメータ設定とグローバル・パラメータ設定の両方を含めることができます。グローバル・パラメータ設定を含める場合、これらのエントリは各インスタンスのファイルで同じである必要があります。

Oracle は、このファイルの先頭のエントリから読取りを開始します。パラメータの解析時に、このファイルに重複するパラメータがあると、Oracle はそのパラメータの最後の値を使用します。これは、インスタンス固有のパラメータ設定およびグローバル・パラメータ設定の両方で適用されます。表 2-1 に、オプションの初期化パラメータ・ファイルを示します。

表 2-1 オプションのパラメータ・ファイルのネーミング規則および説明

初期化パラメータ・ファイル/ ネーミング規則	説明
インスタンス固有のファイル: initsid.ora	各ノードのインスタンスには、独自の initsid.ora ファイルを使用できます。sid は、インスタンスの Oracle システム識別子 (sid) です。このファイルは、インスタンス固有のパラメータ設定を一意に定義します。このファイル内の IFILE パラメータを使用して、initdb_name.ora ファイルをコールします。
グローバル・ファイル: initdb_name.ora	各インスタンス固有のファイルにグローバル・パラメータ・ファイルの設定を含めない場合、共通パラメータを initdb_name.ora ファイルに格納する必要があります。db_name はデータベース名です。このファイルには、すべてのインスタンスで共有される共通データベース・パラメータが示されます。このファイルのコピーは、データベース・クラスタの各インスタンスに格納する必要があります。クラスタ・ファイル・システム (CFS) を使用する場合、同じ Oracle ホームから実行しているすべてのインスタンスがこのファイルを共有します。

initsid.ora の用途

initsid.ora ファイルは、IFILE パラメータによって、共通パラメータ設定用の共通ファイルを指します。initsid.ora ファイルは、インスタンスごとに次の項目を定義します。

- 一意のインスタンス名
- 一意のスレッド番号および**インスタンス番号**
- プライベート・ロールバック・セグメントまたは UNDO 表領域
- ローカル・**リスナー**（リスナーにデフォルトの TCP/IP ポート・アドレスを使用しない場合）

sid を導出するには、initdb_name.ora ファイルの DB_NAME のパラメータ値および**スレッド番号**を使用します。たとえば、DB_NAME が db で、1 つ目のインスタンスのスレッド ID が 1 の場合、sid は db1 です。同様に、2 つ目のインスタンスは、sid db2 を使用してそのインスタンスを識別します。これは、DBCA が sid を導出するときに使用する論理です。ただし、sid には、任意の値を指定することもできます。

例 2-1 および例 2-2 に、**ノード**番号 1 および 2 の各インスタンスに対する initsid.ora ファイルの内容を示します。

例 2-1 initdb1.ora

```
ifile='C:\OracleSW\admin\pdb\pfile\initdb.ora'  
thread=1  
instance_name=db1  
instance_number=1
```

例 2-2 initdb2.ora

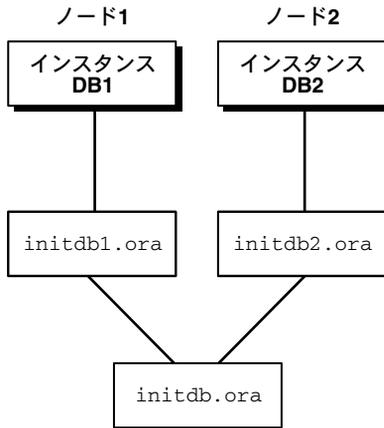
```
ifile='C:\OracleSW\admin\pdb\pfile\initdb.ora'  
thread=2  
instance_name=db2  
instance_number=2
```

関連項目： パラメータの詳細については、『Oracle9i データベース・リファレンス』を参照してください。

initdb_name.ora の用途

initdb_name.ora パラメータ・ファイルの名前は、[図 2-2](#) に示すとおり、各 initdb_name.ora ファイルの IFILE パラメータの設定を使用して指定します。

図 2-2 共通の初期化ファイルの例



従来のパラメータ・ファイルを使用する場合、すべてのインスタンスは同じ共通ファイルを使用する必要があります。

[例 2-3](#) に、**ハイブリッド**・データベースまたは汎用データベース用に作成された initdb_name.ora ファイル (initdb.ora) を示します。

例 2-3 initdb.ora

```

db_name="db"
db_domain=us.acme.com
cluster_database=true
service_names=db.us.acme.com
db_files=1024 # INITIAL
control_files=("${ORACLE_HOME}/db_control1", "${ORACLE_HOME}/db_control2")
open_cursors=100
db_file_multiblock_read_count=8 # INITIAL
db_block_buffers=13816 # INITIAL
shared_pool_size=19125248 # INITIAL
large_pool_size=18087936
java_pool_size=2097152
log_checkpoint_interval=10000
  
```

```
log_checkpoint_timeout=1800
processes=50 # INITIAL
parallel_max_servers=5 # SMALL
log_buffer=32768 # INITIAL
max_dump_file_size=10240 # limit trace file size to 5M each
global_names=true
oracle_trace_collection_name=""
background_dump_dest=C:\OracleSW\admin\%db%\bdump
user_dump_dest=C:\OracleSW\admin\%db%\udump
db_block_size=4096
remote_login_passwordfile=exclusive
os_authent_prefix=""
dispatchers="(protocol=TCP)(lis=listeners_db)"
compatible=9.2
sort_area_size=65536
sort_area_retained_size=65536
```

注意： DB_BLOCK_BUFFERS は将来のリリースで廃止されます。DB_BLOCK_BUFFERS を設定すると、Oracle はその値を使用しますが、警告をアラート・ログ・ファイルに記録します。

関連項目： パラメータの詳細は、『Oracle9i データベース・リファレンス』を参照してください。

インスタンス固有のファイルにおける IFILE パラメータの配置および使用

パラメータ・ファイル内でパラメータ・エントリが重複している場合、そのパラメータに対してファイル内で最後に指定された値が、それより前の値をオーバーライドします。Oracle が正しい共通パラメータ値を使用するには、IFILE パラメータを、すべてのインスタンス固有のパラメータ・ファイルの最後に配置します。逆に、インスタンス固有のパラメータ設定の前に IFILE パラメータを配置することによって、共通パラメータ値をオーバーライドできます。

注意： DBCA は、IFILE パラメータをパラメータ・ファイルの先頭に配置します。

クライアント側のパラメータ・ファイルにおける複数の IFILE エントリの使用

初期化パラメータ・ファイルに IFILE を 2 回以上指定して、複数のグローバル・パラメータ・ファイルを含めることができます。ただし、後続の共通パラメータ・ファイルのパラメータ値を誤ってリセットしないようにしてください。誤ってリセットしないかぎり、IFILE で指定されたファイル内の後続の各エントリは、それより前の値をオーバーライドします。たとえば、インスタンス固有のパラメータ・ファイルには、`init_dbname.ora` ファイルおよび他のパラメータ設定に対する個別のパラメータ・ファイルを含めることができます。次に例を示します。

```
IFILE=INIT_CLUSTER.ORA
IFILE=INIT_LOG.ORA
IFILE=INIT_GC.ORA
LOG_ARCHIVE_START=FALSE
THREAD=3
UNDO_MANAGEMENT=AUTO
```

この例では、`LOG_ARCHIVE_START=false` の値は、このパラメータに対して `INIT_LOG.ORA` パラメータ・ファイルに指定されたすべての値をオーバーライドします。これは、`LOG_ARCHIVE_START` パラメータが IFILE エントリの後にあるためです。

関連項目：

- 2-13 ページ「すべてのインスタンスで同じ値を設定する必要があるパラメータ」
- インスタンスの起動および停止の詳細については、第 4 章「サーバー制御ユーティリティ、SQL および SQL*Plus を使用した Real Application Clusters データベースの管理」を参照してください。

Real Application Clusters のパラメータの設定

前述のとおり、いくつかのパラメータは、Real Application Clusters のすべてのインスタンスで同じ値に設定する必要があります。その他のパラメータは、各インスタンスに一意の値を設定できます。

サーバー・パラメータ・ファイルのパラメータによるインスタンスの一意識別

各インスタンスには、パラメータ設定によってその特性が制御される要素またはコンポーネントがいくつかあります。これらは、データベース・オブジェクトや、インスタンス間の処理に役立つリソースのようなコンポーネントである場合があります。これらのインスタンスのコンポーネントを一意に識別するには、サーバー・パラメータ・ファイルの *sid* 指定子または *initsid.ora* のパラメータ設定を使用します。

関連項目： 初期化パラメータの詳細については、『Oracle9i データベース・リファレンス』を参照してください。

表 2-2 に、データベース名が *db* で、インスタンスのスレッド ID がそれぞれ 1、2 および 3 の場合の *sid* およびインスタンス名を示します。

表 2-2 *sid* およびインスタンス名の例

スレッド ID	<i>sid</i>	インスタンス名
1	db1	db1
2	db2	db2
3	db3	db3

サーバー・パラメータ・ファイルに *sid* を使用したインスタンス固有のパラメータ設定

サーバー・パラメータ・ファイルでは、*sid* 指定子を使用して、インスタンス固有の設定を識別します。また、次の値を指定するインスタンスを作成する場合も、サーバー・パラメータ・ファイルに *sid* 指定子を使用します。

- INSTANCE_NAME
- THREAD

Real Application Clusters のパラメータ・タイプ

この項では、Real Application Clusters 環境における次の3つのタイプの初期化パラメータについて説明します。

- 複数值のパラメータ
- すべてのインスタンスで同じ値を設定する必要があるパラメータ
- すべてのインスタンスで一意の値を設定する必要があるパラメータ

関連項目： その他の Oracle 初期化パラメータの詳細については、『Oracle9i データベース・リファレンス』を参照してください。

複数值のパラメータ

いくつかのパラメータは、1つ以上のインスタンスに対して異なる値を指定するように設定できます。複数值を設定できるパラメータには、デフォルト値、およびデフォルト設定を変更した各インスタンスの値が含まれます。

このようなパラメータに複数の値を設定するには、ALTER SYSTEM SET 文を使用します。ALTER SYSTEM SET 文を使用して、すべてのインスタンスに対して有効なグローバル値を定義することもできます。また、特定のインスタンスに対するこれらのグローバル値をオーバーライドすることもできます。

すべてのインスタンスで同じ値を設定する必要があるパラメータ

データベースの作成に重要な特定の初期化パラメータ、または特定のデータベース操作に影響する特定の初期化パラメータは、Real Application Clusters の各インスタンスで同じ値を設定する必要があります。これらのパラメータの値は、共通パラメータ・ファイル、または各インスタンスの `init_dbname.ora` ファイル内で指定します。次のリストは、すべてのインスタンスで同じ値を設定する必要があるパラメータを示しています。

- CONTROL_FILES
- DB_BLOCK_SIZE
- DB_FILES
- DB_NAME
- DB_DOMAIN
- ARCHIVE_LOG_TARGET
- ROW_LOCKING
- DML_LOCKS (0 (ゼロ) に設定されている場合のみ)
- LOG_ARCHIVE_DEST_n (オプション)

- MAX_COMMIT_PROPAGATION_DELAY
- SERVICE_NAMES
- ACTIVE_INSTANCE_COUNT
- TRACE_ENABLED
- GC_FILES_TO_LOCKS (オプション。Oracle ではリソースの割当てを自動的に制御するため、このパラメータの設定は不要です。)

関連項目： GC_FILES_TO_LOCKS の設定の詳細については、『Oracle9i Real Application Clusters 配置およびパフォーマンス』を参照してください。

すべてのインスタンスで一意的な値を設定する必要があるパラメータ

THREAD パラメータまたは ROLLBACK_SEGMENTS パラメータを使用する場合、サーバー・パラメータ・ファイルで sid 識別子を使用して、これらのパラメータに一意的な値を設定することをお勧めします。ただし、各インスタンスの INSTANCE_NUMBER に一意的な値を設定する必要があり、デフォルト値は使用できません。

- Oracle は、INSTANCE_NUMBER パラメータを使用して、起動時にインスタンスを識別します。また、INSTANCE_NUMBER を使用して、ALTER TABLE 文または ALTER CLUSTER 文にある ALLOCATE EXTENT 句の INSTANCE オプションによってインスタンスに空き領域を割り当てます。
- THREAD パラメータを指定します。これによって、インスタンスが起動時および停止時に異なるスレッド番号を取得するオーバーヘッドを回避できます。Oracle は、このスレッド番号を使用して、特定のインスタンスに REDO ログ・グループを割り当てます。管理を簡単にするために、THREAD パラメータと INSTANCE_NUMBER パラメータに同じ番号を使用します。
- データベース名とインスタンスに割り当てられたスレッド番号で構成される環境変数 ORACLE_SID を指定します。
- INSTANCE_NAME を指定してインスタンスを一意的に識別します。デフォルトは、インスタンスの sid です。INSTANCE_NAME に sid を使用することをお勧めします。
- Oracle は、インスタンスの起動時に、ROLLBACK_SEGMENTS 初期化パラメータで識別するロールバック・セグメント名に基づいて、プライベート・ロールバック・セグメントを取得します。このパラメータを使用してインスタンスにロールバック・セグメント名を宣言しない場合、Oracle はこのインスタンスに対してパブリック・ロールバック・セグメントを取得します。
- 自動 UNDO 管理を使用可能にして UNDO_TABLESPACE を指定する場合、各インスタンスでこのパラメータに一意的な値を設定します。

クラスタのパラメータに関する考慮事項

表 2-3 に、Real Application Clusters データベースでパラメータを使用する際の考慮事項を示します。表 2-3 では、パラメータがアルファベット順にリストされています。

表 2-3 クラスタ内のインスタンスの初期化パラメータに関する注意

パラメータ	説明
CLUSTER_DATABASE	Real Application Clusters モードでデータベースを起動するには、このパラメータを TRUE に設定します。
CLUSTER_DATABASE_INSTANCES	このパラメータを、Real Application Clusters 環境のインスタンスの数に設定します。このパラメータを適切に設定すると、メモリー使用を改善できます。

表 2-3 クラスタ内のインスタンスの初期化パラメータに関する注意 (続き)

パラメータ	説明
<p>CLUSTER_ INTERCONNECTS</p> <p>関連項目: 詳細は、プラットフォーム固有のマニュアルを参照してください。</p>	<p>CLUSTER_INTERCONNECTS パラメータは、一部のプラットフォームでサポートされます。このパラメータ固有の使用方法、パラメータの構文および動作は、プラットフォーム固有のマニュアルを参照してください。ただし、一般的には CLUSTER_INTERCONNECTS を設定する必要はありません。</p> <p>単一のクラスタ・インターコネクต์がある場合、このパラメータの設定は不要です。デフォルトのクラスタ・インターコネクต์が Real Application Clusters データベース (複数可) の帯域幅の要件を満たしている場合 (通常は満たしています) も、このパラメータの設定は不要です。</p> <p>複数のインターコネクต์をこのパラメータに指定すると、Oracle は、CLUSTER_INTERCONNECTS パラメータの情報を使用して、様々なネットワーク・インタフェース間でインターコネクต์通信を分散します。指定した構文は、リストされているインターコネクต์および関連付けられているオペレーティング・システムの IPC サービスの制限 (可用性など) を継承することに注意してください。</p> <p>単一のクラスタ・インターコネクต์で帯域幅の要件を満たすことができない場合はほとんどありませんが、その場合は CLUSTER_INTERCONNECTS の設定を考慮します。たとえば、1 つ以上のデータベースから非常に高いインターコネクต์帯域幅を要求されている一部のデータ・ウェアハウス環境では、このパラメータの設定が必要な場合があります。</p> <p>たとえば、高いインターコネクต์帯域幅の要件を持つ 2 つのデータベースがある場合は、オペレーティング・システムが提供するデフォルトのインターコネクต์をオーバーライドし、各サーバー・パラメータ・ファイルで次の構文を使用して、各データベースに異なるインターコネクต์を指定できます。</p> <p>データベース 1: CLUSTER_INTERCONNECTS = ip1</p> <p>データベース 2: CLUSTER_INTERCONNECTS = ip2</p> <p>ipn は、ドットで区切られた標準的な 10 進形式の IP アドレス (たとえば、144.25.16.214) です。</p> <p>ただし、非常に高い帯域幅を必要とするデータベースがある場合は、たとえば、次の構文を使用して複数のインターコネクต์を指定できます。</p> <p>CLUSTER_INTERCONNECTS = ip1:ip2:...:ipn</p> <p>前述の例のように、CLUSTER_INTERCONNECTS に複数の値を設定した場合、Oracle は、指定したすべてのインターコネクต์を使用します。リストされたすべてのインターコネクต์が操作可能であるかぎり、ロード・バランシングが提供されます。</p> <p>CLUSTER_INTERCONNECTS に指定したインターコネクต์への書込みでオペレーティング・システム・エラーが発生した場合、Oracle は、別のインタフェースが使用可能な場合でもエラーを戻します。これは、Oracle とインターコネクต์間の通信プロトコルが、使用しているプラットフォームに大きく依存する場合があるためです。</p>
DB_NAME	<p>インスタンス固有のパラメータ・ファイルで DB_NAME の値を設定する場合は、すべてのインスタンスに同じ値を設定する必要があります。</p>

表 2-3 クラスタ内のインスタンスの初期化パラメータに関する注意 (続き)

パラメータ	説明
DISPATCHERS	<p>共有サーバー構成を使用可能にするには、DISPATCHERS パラメータを設定します。DISPATCHERS パラメータには、多くの属性を含めることができます。</p> <p>少なくとも、PROTOCOL 属性および LISTENER 属性を構成することをお勧めします。PROTOCOL には、ディスパッチャがリスニングのエンド・ポイントを生成するネットワーク・プロトコルを指定します。LISTENER には、PMON プロセスがディスパッチャ情報を登録するリスナーの別名を指定します。別名には、tnsnames.ora ファイルなどのネーミング・メソッドを介して解決される名前を設定します。</p> <p>DISPATCHERS パラメータとその属性の構成、および共有サーバーの構成の詳細は、『Oracle9i Net Services 管理者ガイド』を参照してください。</p>
DML_LOCKS	<p>0 (ゼロ) に設定されている場合のみ、すべてのインスタンスで同じ値である必要があります。デフォルト値では、トランザクションごとに平均 4 つの表が参照されると想定されています。システムによっては、この値が十分ではない場合があります。DML_LOCKS の値を 0 (ゼロ) に設定すると、エンキューが使用不可になり、パフォーマンスがわずかに向上します。ただし、DROP TABLE、CREATE INDEX または明示的なロック文 (LOCK TABLE IN EXCLUSIVE MODE など) は使用できません。</p>
INSTANCE_NAME	<p>指定する場合、このパラメータの値はすべてのインスタンスに対して一意である必要があります。Real Application Clusters 環境では、すべてのインスタンスを単一データベース・サービスに対応付けることができます。クライアントは、データベースへ接続する特定のインスタンスを指定することによって、接続時ロード・バランシングをオーバーライドできます。INSTANCE_NAME には、このインスタンスの一意の名前を指定します。INSTANCE_NAME を sid と同じ値に設定することをお勧めします。</p>
LOG_ARCHIVE_FORMAT	<p>このパラメータは、ARCHIVELOG モードで REDO ログを使用している場合のみに適用できます。REDO ログ・ファイルをアーカイブするときは、テキスト文字列および変数を使用してデフォルトのファイル名フォーマットを指定します。このフォーマットから生成された文字列は、LOG_ARCHIVE_DEST_n パラメータに指定されている文字列に追加されます。スレッド番号を含める必要があります。</p> <p>次の変数をフォーマットに使用できます。</p> <ul style="list-style-type: none"> ■ %s: ログ順序番号 ■ %S:0 (ゼロ) が埋められたログ順序番号 ■ %t: スレッド番号 ■ %T:0 (ゼロ) が埋められたスレッド番号 <p>変数に大文字 (%S など) を使用すると、値が固定長になり、左詰に 0 (ゼロ) が埋められます。アーカイブ REDO ログ・ファイル名フォーマットを指定する例を次に示します。</p> <ul style="list-style-type: none"> ■ LOG_ARCHIVE_FORMAT = "LOG%s_%t.ARC"

表 2-3 クラスタ内のインスタンスの初期化パラメータに関する注意 (続き)

パラメータ	説明
MAX_COMMIT_PROPGATION_DELAY	<p>これは Real Application Clusters 固有のパラメータです。ただし、特に指定しないかぎり、このパラメータを変更しないでください。</p> <p>このパラメータは、インスタンスの SGA に保持されるシステム変更番号 (SCN) が、ログ・ライター (LGWR) ・プロセスによってリフレッシュされるまでの時間の最大値を指定します。また、問合せに対するスナップショット SCN の取得時に、ロック値からローカル SCN をリフレッシュするかどうかを決定します。単位は 100 分の 1 秒です。異なるインスタンスからの同一データの高速更新や問合せなどの異常な状況では、SCN は、すぐにはリフレッシュされない場合があります。パラメータを 0 (ゼロ) に設定すると、SCN はコミットの後すぐにリフレッシュされます。デフォルト値 (100 分の 700 秒、つまり 7 秒) は、推奨される既存の高パフォーマンス・メカニズムを維持できる上限値です。</p> <p>リモート・インスタンスがすぐにコミットを参照する必要がある場合は、このパラメータの値を変更する必要がある場合があります。</p>
NLS_* パラメータ (Oracle グローバリゼーション・サポート用)	『Oracle9i データベース・リファレンス』および『Oracle9i Database グローバリゼーション・サポート・ガイド』に記載されているとおり、数種類の グローバリゼーション・サポート パラメータがあります。インスタンスごとに異なる値を設定できます。
PROCESSES	SESSIONS パラメータおよび TRANSACTIONS パラメータのデフォルト値は、PROCESSES パラメータの値から直接的または間接的に導出されます。したがって、PROCESSES の値を変更する場合、これらの導出パラメータの値を調整するかどうかを評価する必要があります。デフォルト値を使用しない場合は、追加のバックグラウンド・プロセス用に、この表で前述されているいくつかのパラメータ値を増やす必要があります。
RECOVERY_PARALLELISM	ロールフォワードまたはキャッシュ・リカバリをスピードアップするために、このパラメータを設定して、インスタンスまたは障害リカバリに使用するプロセスの数を指定できます。0 (ゼロ) または 1 に設定すると、リカバリが 1 つのプロセスによってシリアルに実行されます。
ROLLBACK_SEGMENTS (ロールバック管理 UNDO モードのみで使用)	<p>このパラメータを手動のロールバック管理 UNDO モードのみで使用すると、1 つ以上のロールバック・セグメントを名前ごとにインスタンスに割り当てることによって、各インスタンスのプライベート・ロールバック・セグメントを指定できます。このパラメータを設定すると、ロールバック・セグメントの数がインスタンスに必要な最小数を超える場合でも、インスタンスは、このパラメータに指定されたすべてのロールバック・セグメントを取得します。この最小値は、次の割合から計算されます。</p> <p>TRANSACTIONS / TRANSACTIONS_PER_ROLLBACK_SEGMENT</p>
SESSIONS_PER_USER	SESSIONS_PER_USER 件数は、インスタンスごとに保存されます。あるユーザーの SESSIONS_PER_USER が 1 に設定された場合、各接続が異なるインスタンスからであるかぎり、そのユーザーはデータベースに 2 回以上ログインできます。

表 2-3 クラスタ内のインスタンスの初期化パラメータに関する注意 (続き)

パラメータ	説明
SPFILE	<p>SPFILE の値は、使用されている現行のサーバー・パラメータ・ファイルの名前です。クライアント側の PFILE の SPFILE パラメータを、使用するサーバー・パラメータ・ファイル名を示すように定義できます。</p> <p>サーバーは、デフォルトのサーバー・パラメータ・ファイルを使用する場合、SPFILE の値を内部的に設定します。</p>
THREAD	<p>指定する場合、このパラメータの値はすべてのインスタンスに対して一意である必要があります。THREAD は、Real Application Clusters パラメータであり、インスタンスに使用される REDO スレッド番号を指定します。</p> <p>Real Application Clusters では、スレッド番号が使用可能で、他のインスタンスが使用しないかぎり、使用可能なすべての REDO スレッド番号を指定できます。0 (ゼロ) の使用は、インスタンスでは、使用可能なすべてのパブリック・スレッドを使用できることを意味します。ただし、この方法はお勧めできません。</p>

関連項目： これらのパラメータの詳細については、『Oracle9i データベース・リファレンス』を参照してください。Real Application Clusters 環境における平行実行用の追加パラメータの詳細については、『Oracle9i Real Application Clusters 配置およびパフォーマンス』を参照してください。

Real Application Clusters の起動プロセスおよびパラメータ

Real Application Clusters では、最初のインスタンスによってデータベースが起動およびマウントされます。また、最初に起動するインスタンスの alert.log ファイルのエントリによって、そのインスタンスが最初に起動したことを確認できます。

注意： アラート・ログ・ファイルを検索するには、検索文字列 alert*.log を使用してください。Windows プラットフォームの場合、*alert.log のアスタリスクに SID が入ります。通常、alert.log は background_dump_dest ディレクトリにあります。

関連項目： インスタンスの起動の詳細については、第 4 章「サーバー制御ユーティリティ、SQL および SQL*Plus を使用した Real Application Clusters データベースの管理」を参照してください。

従来のパラメータ・ファイルを使用して起動する場合の考慮事項

従来のパラメータを使用しており、あるインスタンスのファイルにグローバル・パラメータが含まれる場合、そのグローバル・パラメータの値は、そのパラメータに対して他のインスタンスで設定されている値と一致する必要があります。一致しない場合、インスタンスはデータベースをマウントできません。

リモート・ノード上での2つのインスタンスの起動

サーバー制御ユーティリティ (SRVCTL) を使用してインスタンスを起動することをお勧めします。SRVCTL は、4-2 ページの「[SRVCTL を使用した Real Application Clusters 環境の管理](#)」に記載されているように、別の管理タスクに使用することもできます。この項では、SQL*Plus を使用したインスタンスの起動について説明します。

Oracle Net 経由で、1 つのノード上の SQL*Plus セッションから複数のノードを起動します。たとえば、ローカル・ノード上で SQL*Plus セッションを使用して、`init_db1.ora` および `init_db2.ora` という個別のパラメータ・ファイルを使用する 2 つのインスタンスをリモート・ノード上で起動できます。データベースに接続する前に、SQL*Plus で次の構文を入力し、最初のインスタンスに対してコマンドを発行します。

```
SET INSTANCE DB1;
```

次のコマンドを入力して、最初のインスタンスに接続し、そのインスタンスを起動した後、接続を切断します。

```
CONNECT / AS SYSDBA;  
STARTUP PFILE=$ORACLE_HOME/dbs/initsid.ora  
DISCONNECT;
```

`initsid.ora` ファイルには、RAW デバイス上の `spfile.ora` ファイルの位置を示すエントリが含まれます。

次の構文を使用して、2 つ目のインスタンスにコマンドをリダイレクトします。

```
SET INSTANCE DB2;
```

次のコマンドを入力して、2 つ目のインスタンスに接続し、起動します。

```
CONNECT / AS SYSDBA;  
STARTUP PFILE=full_pathINIT_DB2.ORA;
```

DB1 および DB2 は、2 つのインスタンスの `sid` です。これらの `sid` は、`tnsnames.ora` の `sid` エントリで定義されます。`initsid.ora` ファイルの `SPFILE` パラメータ・エントリは、リモート・インスタンス上の位置を指定します。

従来のパラメータ・ファイルの使用例における IFILE の使用

前述の例では、両方のパラメータ・ファイルで IFILE パラメータを使用して、init_dbname.ora ファイルの値を含めることができます。

インスタンス番号の設定

インスタンス番号は、起動時に INSTANCE_NUMBER パラメータを使用して明示的に指定する必要があります。これは、Real Application Clusters が使用可能または使用禁止でも行うことができます。INSTANCE_NUMBER は、インスタンスの THREAD を識別するために設定した値と同じ値にすることをお勧めします。次の SQL*Plus コマンドによって、各インスタンスの現在の番号を表示できます。

```
SHOW PARAMETER INSTANCE_NUMBER
```

注意： すべてのインスタンス番号は一意である必要があります。

管理操作を実行するためのみに、Real Application Clusters を使用禁止にしてインスタンスを起動する場合、パラメータ・ファイルの INSTANCE_NUMBER パラメータを省略できます。Real Application Clusters を使用禁止にして起動したインスタンスでは、1 以外のスレッドを指定して、そのスレッドに対応付けられたオンライン REDO ログ・ファイルを使用することもできます。

関連項目：

- Real Application Clusters の UNDO 領域の管理の詳細については、[第 3 章「Real Application Clusters の記憶域コンポーネントの管理」](#)を参照してください。
- 挿入および更新用の空き領域の割当ての詳細については、『Oracle9i Real Application Clusters 配置およびパフォーマンス』を参照してください。
- Oracle データベースの起動の詳細については、『Oracle9i データベース管理者ガイド』を参照してください。

Real Application Clusters の記憶域 コンポーネントの管理

この章では、**Real Application Clusters** の記憶域コンポーネントの管理方法を説明します。内容は次のとおりです。

- **Real Application Clusters** におけるデータ・ファイルの追加
- **Real Application Clusters** の自動 UNDO 管理
- **Real Application Clusters** での REDO ログ・ファイルの使用
- **Real Application Clusters** でのトレース・ファイルとアラート・ファイルの管理
- **Real Application Clusters** の順序番号ジェネレータ

関連項目： 記憶域コンポーネントの使用を最適化して、パフォーマンスを向上させる方法については、『Oracle9i Real Application Clusters 配置およびパフォーマンス』を参照してください。

Real Application Clusters におけるデータ・ファイルの追加

データ・ファイルは、Real Application Clusters データベースの実行中に追加できます。データ・ファイルを追加すると、Oracle は、既存のファイルへのリソース割当てを制御する場合と同様に、新しいファイルへのリソース割当てを自動的に制御します。

Real Application Clusters の自動 UNDO 管理

この項では、Real Application Clusters での自動 UNDO 管理および手動 UNDO 管理の使用方法について説明します。Oracle は、シングル・インスタンス環境で使用可能な UNDO 領域管理機能をさらに強化したものを Real Application Clusters に提供します。

UNDO 領域管理には、自動 UNDO 管理または手動 UNDO 管理のいずれかを使用できます。ただし、より透過的な自動 UNDO 管理方法を使用することをお勧めします。この項の内容は次のとおりです。

- [自動 UNDO 管理の使用](#)
- [UNDO 表領域の切替え](#)
- [システム・ロールバック・セグメント](#)
- [パブリック・ロールバック・セグメントおよびプライベート・ロールバック・セグメント](#)
- [手動 UNDO 管理を使用した自動 UNDO 管理のオーバーライド](#)

注意： 自動 UNDO 管理モードと手動 UNDO 管理モードを同時に使用することはできません。Real Application Cluster 環境のすべてのインスタンスは、同じ UNDO モードで操作してください。

自動 UNDO 管理の使用

Oracle は自動セグメント領域管理を使用して、空きリストとは対照的に、セグメント内の空き領域および使用済み領域をビットマップで管理します。自動セグメント領域管理は、空きリストよりも管理が容易で、領域の使用率を改善します。これは、行のサイズが頻繁に変化するオブジェクトの場合、特に明らかです。自動セグメント領域管理によって、同時アクセスによる変化に対する実行時調整も改善されます。さらに、パフォーマンス / 領域の比で表されるクラスタ・パフォーマンスも改善されます。

自動 UNDO 管理を使用するには、次のパラメータを設定します。

- **サーバー・パラメータ・ファイル**で、UNDO_MANAGEMENT グローバル・パラメータを auto に設定します。クライアント側のパラメータ・ファイルを使用する場合は、すべてのファイルで UNDO_MANAGEMENT に同じ値を設定する必要があります。
- UNDO_TABLESPACE パラメータを設定して、UNDO 表領域をインスタンスに割り当てます。

UNDO_TABLESPACE パラメータで使用する UNDO 表領域は、あらかじめ作成しておく必要があります。この表領域が作成されていないと、STARTUP コマンドは正常に実行されません。

UNDO_TABLESPACE パラメータを設定しない場合、各**インスタンス**では最初に使用可能な UNDO 表領域が使用されます。UNDO 表領域が使用不可の場合、インスタンスでは SYSTEM ロールバック・セグメントが使用されます。このため、特定のインスタンスに UNDO TABLESPACE を割り当てて、表領域の使用を制御することをお勧めします。

警告： UNDO には、SYSTEM ロールバック・セグメントを使用しないことをお勧めします。UNDO に SYSTEM ロールバック・セグメントを使用すると、Oracle はアラート・ファイルにメッセージを書き込み、UNDO 表領域を使用せずにデータベースが実行されていることを警告します。

自動 UNDO 管理を使用すると、Oracle は TRANSACTIONS パラメータの設定を無視します。これは、自動 UNDO 管理の場合、Oracle が SGA から自動 UNDO 管理用にトランザクション・オブジェクトを動的に割り当てるためです。

UNDO 表領域の切替え

ALTER SYSTEM SET UNDO_TABLESPACE 文を実行すると、UNDO 表領域の使用を動的にリダイレクトできます。たとえば、インスタンス db1 および db2 があり、それぞれ UNDO 表領域 undotbs01 および undotbs02 にアクセスするとします。アイドル状態の UNDO 表領域（たとえば、undotbs03）がある場合は、次の文をいずれかのインスタンスから実行して、UNDO 処理を undotbs03 にリダイレクトできます。

```
ALTER SYSTEM SET UNDO_TABLESPACE = undotbs3;
```

注意： 各インスタンスで 1 回に使用できる UNDO 表領域は 1 つのみです。また、インスタンスは UNDO 表領域を共有できません。

ユーザー・トランザクションは、Oracle がこの操作を実行している間も通常どおり継続されます。状況によっては、一時的に 1 つのインスタンスが 2 つの表領域に並行アクセスできます。このような状況は、表領域の切替え操作中に移行処理が行われる場合にのみ発生します。

また、この処理がすべてのユーザー・トランザクションがコミットするまで待つこともありません。かわりに、前の UNDO 表領域にアクティブ・トランザクションがある場合、その表領域はペンディング・オフライン状態になります。この場合、その表領域に対するすべてのトランザクションがコミットされるまで、ペンディング・オフライン状態の表領域を他のインスタンスで使用することはできません。

前述のサンプル・コードでは、前に使用された UNDO 表領域 undotbs01 または undotbs02 は、インスタンスの最後のアクティブ・トランザクションがコミットされるまで、そのインスタンスによって所有されたままになります。

1 つのインスタンスで 1 回に使用できる UNDO 表領域は 1 つのみですが、すべてのインスタンスは、読み込み一貫性のためにいつでも UNDO ブロックを読み取ることができます。また、UNDO 表領域が、別のインスタンスに UNDO 生成またはトランザクション・リカバリのために使用されていないかぎり、どのインスタンスもトランザクション・リカバリ中にその UNDO 表領域を更新できます。

関連項目： ALTER SYSTEM SET UNDO_TABLESPACE の詳細については、『Oracle9i データベース管理者ガイド』を参照してください。

システム・ロールバック・セグメント

自動 UNDO 管理を使用する場合、Oracle が使用する外部ロールバック・セグメントは SYSTEM ロールバック・セグメントのみです。SYSTEM ロールバック・セグメントは、各データベースに 1 つしかありません。SYSTEM ロールバック・セグメントは SYSTEM 表領域内に存在し、データベース作成時に Oracle によって自動的に作成されます。

Real Application Clusters データベースでは、すべてのインスタンスが同じ SYSTEM ロールバック・セグメントを使用します。通常の場合では、トランザクション表の作成などのシステム・トランザクションを実行する場合にのみ、SYSTEM ロールバック・セグメントが使用されます。通常、SYSTEM ロールバック・セグメントを管理するための操作を実行する必要はありません。

関連項目： UNDO 表領域で実行できる、その他の管理操作（UNDO 保存期間の設定や UNDO 表領域の削除など）の詳細については、『Oracle9i データベース管理者ガイド』を参照してください。

パブリック・ロールバック・セグメントおよびプライベート・ロールバック・セグメント

パブリック・ロールバック・セグメントとプライベート・ロールバック・セグメントは、同レベルのパフォーマンスを提供します。ただし、プライベート・ロールバック・セグメントの方が、インスタンスとロールバック・セグメントの一致に対して、より詳細な制御ができます。これによって、異なるインスタンスに対して異なるディスク上のロールバック・セグメントの場所を指定できるため、パフォーマンスが向上します。したがって、パフォーマンスを改善させるにはプライベート・ロールバック・セグメントを使用します。

パブリック・ロールバック・セグメントは、ロールバック・セグメントのプールを形成しません。追加のロールバック・セグメントを必要とするインスタンスは、このプールからロールバック・セグメントを取得できます。ただし、パブリック・ロールバック・セグメントを使用すると、インスタンスの停止と起動を同時に行った場合にデメリットがあります。

たとえば、インスタンス X を停止し、パブリック・ロールバック・セグメントを解放するとします。次に、インスタンス Y を起動します。このインスタンスは、インスタンス X から解放されたロールバック・セグメントを取得します。その後、インスタンスを起動しても元のロールバック・セグメントを取得することはできません。したがって、TRANSACTIONS パラメータおよび TRANSACTIONS_PER_ROLLBACK_SEGMENTS パラメータを正しく設定しないと、インスタンスの起動時に、強制的にパブリック・ロールバック・セグメントが割り当てられます。

パブリック・ロールバック・セグメントを使用すると、領域の使用を改善できます。たとえば、月ごとに異なるインスタンス上で実行する長時間のトランザクション用に 1 つの大きなパブリック・ロールバック・セグメントを作成する場合、ロールバック・セグメントをオフライン化して、再度オンライン化したり、より処理需要の高いインスタンスで使用するために、あるインスタンスから別のインスタンスに移動させることができます。

デフォルトでは、ロールバック・セグメントはプライベートで、パラメータ・ファイル内でそのロールバック・セグメントを指定したインスタンスによって使用されます。プライベート・ロールバック・セグメントは、ROLLBACK_SEGMENTS パラメータを使用して指定します。

また、次の規則が適用されます。

- パブリック・ロールバック・セグメントがインスタンスによって取得されると、そのインスタンスによって排他的に使用されます。
- ロールバック・セグメントが作成されると、パブリックとプライベートのいずれのロールバック・セグメントも ALTER ROLLBACK SEGMENT コマンドを使用してオンライン化できます。
- 1つのインスタンスで SYSTEM 以外のロールバック・セグメントがオンライン化されている場合、そのインスタンスの起動には、少なくとももう1つのロールバック・セグメントが必要です。

関連項目：

- インスタンスによるロールバック・セグメントの取得方法と監視方法の詳細については、『Oracle9i データベース管理者ガイド』を参照してください。
- DBA_ROLLBACK_SEGS および DBA_SEGMENTS の詳細については、『Oracle9i データベース・リファレンス』を参照してください。

手動 UNDO 管理を使用した自動 UNDO 管理のオーバーライド

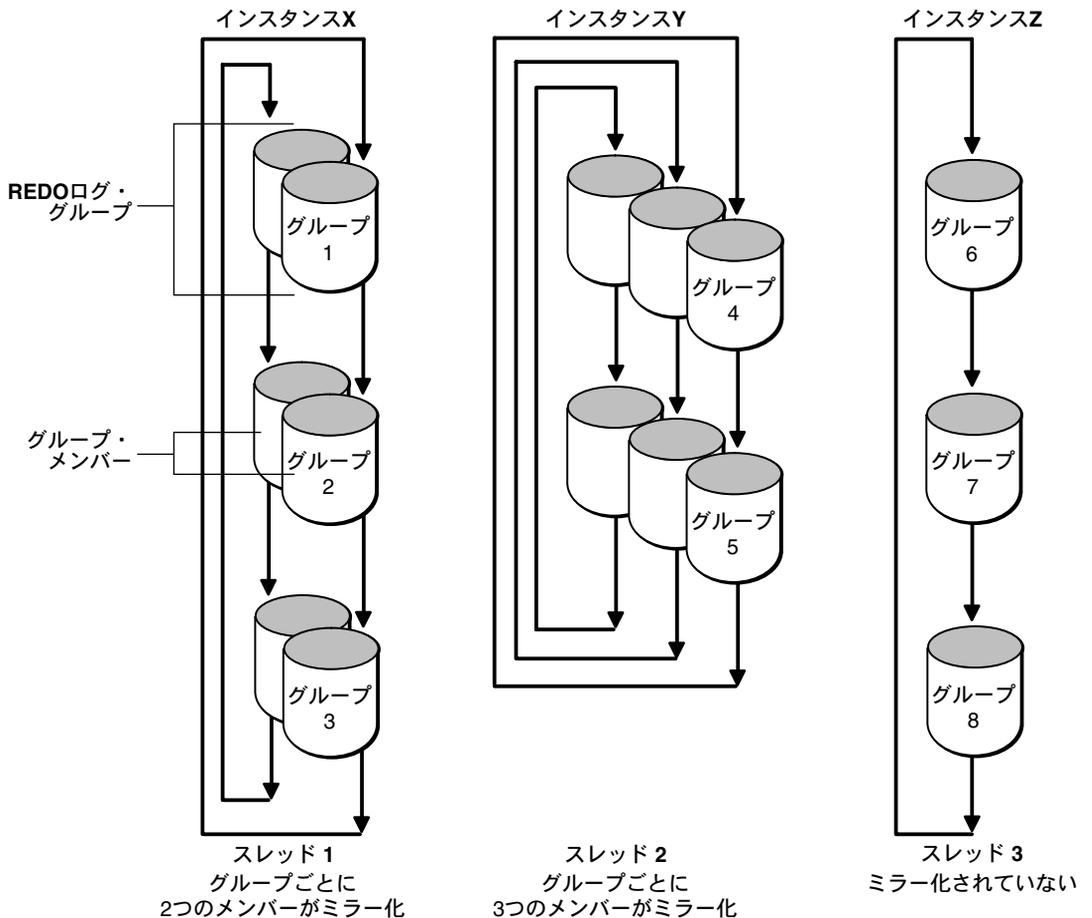
UNDO_MANAGEMENT パラメータを *manual* に設定することによって、デフォルトの自動 UNDO 管理をオーバーライドできます。ただし、この方法はお薦めできません。また、この項の後半で説明する推奨事項もお読みください。UNDO_MANAGEMENT パラメータを指定しない場合、Oracle はインスタンスを自動 UNDO 管理モードで起動します。

関連項目： 手動 UNDO 管理と空きリストの詳細については、[付録 B 「インスタンスとユーザーの、空きリストと空きリスト・グループ（オプション）への対応付け」](#) を参照してください。

Real Application Clusters での REDO ログ・ファイルの使用

各インスタンスには、そのインスタンスのオンライン REDO スレッドと呼ばれる独自のオンライン REDO ログ・グループがあります。『Oracle9i データベース管理者ガイド』を参照して、これらのオンライン REDO ログ・グループを作成し、グループ・メンバーを構築します。図 3-1 に、3 つの Real Application Clusters インスタンスの REDO スレッドを示します。

図 3-1 REDO スレッド



- インスタンス X は、3つのグループのオンライン REDO ログ・グループが含まれているスレッド 1 を使用します。スレッド 1 は多重化され、各グループには、2つの REDO ログ・ファイルのコピー（またはメンバー）が含まれています。
- インスタンス Y は、2つのグループのオンライン REDO ログ・ファイルが含まれているスレッド 2 を使用します。スレッド 2 は多重化され、各 REDO ログ・グループには、3つのメンバーが含まれています。
- インスタンス Z は、3つのグループのオンライン REDO ログ・ファイルが含まれているスレッド 3 を使用します。ただし、これらの REDO ログ・ファイルは多重化されていません。各グループに含まれているメンバーは 1 つのみです。

注意： Database Configuration Assistant (DBCA) では、2つのログ・ファイル（各グループのメンバーが 1 つ、各インスタンスのスレッドが 1 つ）が作成されます。

グループ番号は、データベース内で一意である必要があります。ただし、スレッドへのグループの割当て順序とインスタンスへのスレッドの割当て順序は任意です。

たとえば、[図 3-1](#) では、スレッド 1 にグループ 1、2 および 3 が含まれ、スレッド 2 にはグループ 4 および 5 が含まれていますが、かわりにスレッド 1 にグループ 2、4 および 5 を割り当て、スレッド 2 にグループ 1 および 3 を割り当てることもできます。V\$LOGFILE ビューには、各 REDO ログ・ファイルに対応付けられたグループ番号が表示されます。

各スレッドには、異なる数のグループおよびメンバーを構成できますが、管理を簡単にするために、すべてのスレッドを標準構成にすることをお勧めします。できるだけ、すべてのスレッドをスレッド X または Y に示したとおりに構成します。スレッド Z に示すミラー化されていない REDO ログ・グループを使用することはお勧めしません。ただし、標準以外の REDO ログ構成はパフォーマンスの上では有効です。

ミラー化のオーバーヘッドが少ないときにパフォーマンスが向上するインスタンスでは、ミラー化の程度を変える必要がある場合があります。たとえば、1 つ目のインスタンスにそれぞれ 2 つのメンバーを含む 3 つのグループを含み、2 つ目のインスタンスに 4 つの非多重化ログ・ファイルを含み、3 つ目のインスタンスにそれぞれ 4 つのメンバーを含む 2 つのグループを含むことができます。

Real Application Clusters では、各インスタンスに 2 つ以上のグループのオンライン REDO ログ・ファイルが必要です。現行グループが一杯になると、インスタンスは次のログ・ファイル・グループへの書込みを開始します。Oracle は、REDO ログ・スイッチ時に、一杯になったグループとそのスレッド番号をアーカイブ後に識別する情報を制御ファイルに書き込みます。

注意： 可用性の要件が高いサイトでは、MAXLOGHISTORY が有効です。このオプションは、特にインスタンスとログ・ファイルの数が多いたまのりカバリ管理に有効です。

関連項目： 多重 REDO ログ・ファイルの詳細は、『Oracle9i データベース管理者ガイド』を参照してください。

Real Application Clusters でのトレース・ファイルとアラート・ファイルの管理

Oracle は、Real Application Clusters 環境で発生する重要なイベントに関する情報をトレース・ファイルとアラート・ファイルに記録します。Real Application Clusters のトレース・ファイルとアラート・ファイルは、シングル・インスタンス Oracle データベースの場合と同じです。

これらのファイルを頻繁に監視し、すべてのインスタンスについてファイルのコピーを定期的に作成してください。これによって、これらのファイルの内容が保存され、誤ってファイルが上書きされることを回避できます。

関連項目： トレース・ファイルとアラート・ログの詳細については、A-2 ページの「[トレース・ファイルの使用](#)」を参照してください。

Real Application Clusters の順序番号ジェネレータ

Real Application Clusters では、複数インスタンスのユーザーが、最小限の同期で一意的な順序番号を生成できます。順序番号ジェネレータを使用すると、複数インスタンスは、インスタンス間での順序番号の競合なしに、またトランザクションがコミットされるのを待たずに、順序にアクセスして増分できます。

各インスタンスには、順序番号に高速アクセスするための順序キャッシュを独自に構成できます。Oracle は [グローバル・キャッシュ・サービス \(GCS\) ・リソース](#) を使用して、Real Application Clusters 内にあるインスタンスの順序を調整します。

この項では、CREATE SEQUENCE 文とそのオプションについて説明します。

- CREATE SEQUENCE 文
- CACHE オプション
- ORDER オプション

CREATE SEQUENCE 文

SQL 文 CREATE SEQUENCE を使用すると、1つのデータベース・オブジェクトを設定できます。このオブジェクトから、複数のユーザーが一意的な整数を生成し、他のユーザーがトランザクションをコミットするまで待つことなく、同じ順序番号ジェネレータにアクセスできます。

Real Application Clusters では、複数インスタンスのユーザーが、インスタンス間での連携または競合が最小限の状態、一意の順序番号を生成できます。

順序番号は、CYCLE オプションを使用しないかぎり、常に一意の番号です。ただし、次の項で説明するとおり、ORDER オプションを指定せずに CACHE オプションを使用すると、正規の順序でない順序番号を割り当てることができます。

関連項目： CREATE SEQUENCE および CYCLE オプションの詳細については、『Oracle9i SQL リファレンス』を参照してください。

CACHE オプション

CREATE SEQUENCE の CACHE オプションを使用すると、順序番号を事前に割り当て、それらの番号をインスタンスの SGA に保持して、高速アクセスを実現できます。CACHE オプションには、キャッシュする順序番号の数を引数として指定できます。デフォルト値は 20 です。

順序番号をキャッシュするとパフォーマンスは大幅に改善されますが、順序内の一部の番号が失われる可能性があります。この場合、順序番号は古い順になりません。順序番号が失われても、主キー用の一意の番号を生成するために順序が使用される場合など、一部のアプリケーションでは問題ありません。

任意の順序のキャッシュは、その順序から番号が最初に要求された時点で移入されます。キャッシュされた番号セットの最後の番号が割り当てられると、そのキャッシュに別の番号セットが再移入されます。

各インスタンスは、メモリー内に独自の順序番号キャッシュを保持します。インスタンスが停止すると、キャッシュされた順序値のうち、コミットされた DML 文に使用されなかったものが失われる場合があります。失われる値の予想数は、最高で、CACHE オプションの値に停止するインスタンスの数を掛けた値になります。キャッシュされた順序番号は、インスタンスが正常に停止されたときでも失われる可能性があります。

ORDER オプション

CREATE SEQUENCE の ORDER オプションを使用すると、要求された順で確実に順序番号を生成できます。ORDER オプションは、複数のプロセスおよびインスタンスの要求順序を示す必要がある、タイムスタンプ番号およびその他の順序に使用できます。

Oracle が順序どおりに順序番号を発行する必要がない場合は、CREATE SEQUENCE の NOORDER オプションを使用すると、Real Application Clusters 環境のオーバーヘッドを大幅に削減できます。

注意： Real Application Clusters データベースでは、データベースがクラスタ・モードでマウントされているときに、CREATE SEQUENCE の ORDER オプションと CACHE オプションを同時に指定することはできません。各インスタンスにキャッシュされた順序値がある場合、Oracle では順序付けを保証しません。したがって、CACHE オプションと ORDER オプションの両方を指定して順序を作成する必要がある場合、順序付けは行われませんが、キャッシュは行われません。

サーバー制御ユーティリティ、SQL および SQL*Plus を使用した Real Application Clusters データベースの管理

この章では、**サーバー制御ユーティリティ (SRVCTL)**、SQL および SQL*Plus を使用して、**Real Application Clusters** 環境のインスタンスとデータベースを管理する方法について説明します。内容は次のとおりです。

- SRVCTL を使用した Real Application Clusters 環境の管理
- SRVCTL を使用した Real Application Clusters 管理の概要
- グローバル・サービス・デーモン (GSD)
- SRVCTL 管理タスク
- SRVCTL のコマンド構文
- SRVCTL の構文に共通する構成要素
- SRVCONFIG を使用した RAW デバイス構成のインポートとエクスポート
- Oracle8i から Oracle9i への構成のアップグレード
- SQL および SQL*Plus を使用した Real Application Clusters データベースの管理
- インスタンスの設定および接続
- Real Application Clusters データベースの静止
- インスタンスへの SQL および SQL*Plus コマンドの適用方法

関連項目： **Oracle Enterprise Manager** を使用して Real Application Clusters を管理する方法については、第 5 章「Oracle Enterprise Manager を使用した Real Application Clusters データベースの管理」を参照してください。

SRVCTL を使用した Real Application Clusters 環境の管理

この章では、Oracle [サーバー制御ユーティリティ \(SRVCTL\)](#) とともに Real Application Clusters を使用して Real Application Clusters 環境を管理する方法について説明します。内容は次のとおりです。

- [SRVCTL を使用した Real Application Clusters 管理の概要](#)
- [グローバル・サービス・デーモン \(GSD\)](#)
- [SRVCTL 管理タスク](#)
- [SRVCTL のコマンド構文](#)
- [SRVCTL の構文に共通する構成要素](#)

関連項目： Oracle Enterprise Manager を使用した Real Application Clusters の管理の詳細については、[第 5 章「Oracle Enterprise Manager を使用した Real Application Clusters データベースの管理」](#)を参照してください。

SRVCTL を使用した Real Application Clusters 管理の概要

オラクル社では、Real Application Clusters 環境を管理するための主要ツールとして SRVCTL を使用することをお勧めします。SRVCTL は、いくつかの Oracle のツール製品で使用される構成情報を管理します。たとえば、Oracle Enterprise Manager および [Oracle Intelligent Agent](#) は、SRVCTL で生成された構成情報を使用して、クラスタ内のノードを検出および監視します。

SRVCTL を使用してクラスタの構成操作を実行すると、SRVCTL は構成データを Server Management (SRVM) 構成リポジトリに格納します。SRVCTL は、各ノードで SQL*Plus をコールし、その他の操作（インスタンスの起動や停止など）を実行します。

SRVCTL は、その他の Oracle 管理インタフェースで使用されるリポジトリと同じ SRVM 構成リポジトリを使用します。

関連項目： SRVM 構成リポジトリの詳細については、『[Oracle9i Real Application Clusters セットアップおよび構成](#)』を参照してください。

注意： 構成の変更前および変更後に、SRVM 構成リポジトリをバックアップしてください。

SRVCTL を使用する前に、**グローバル・サービス・デーモン (GSD)** が実行されていることを確認してください。確認するには、使用しているプラットフォームに基づいて、適切なコマンドを実行して GSD を実行します。Oracle は、GSD がすでに実行されていることを示すメッセージを表示します。GSD の詳細については、4-3 ページの「**グローバル・サービス・デーモン (GSD)**」を参照してください。

注意： SRVCTL を使用するには、管理するデータベースの構成情報をあらかじめ作成しておく必要があります。構成情報を作成するには、**Real Application Cluster** データベースの手動による作成方法を示している章の説明に従って `srvctl add` コマンドを使用するか、または **Database Configuration Assistant (DBCA)** を使用します。詳細については、『Oracle9i Real Application Clusters セットアップおよび構成』を参照してください。

関連項目： Oracle Intelligent Agent の詳細については、『Oracle Intelligent Agent ユーザーズ・ガイド』を参照してください。

グローバル・サービス・デーモン (GSD)

グローバル・サービス・デーモン (GSD) は SRVCTL からの接続要求などの情報を記録し、これらのレコードを `$ORACLE_HOME/srvm/log` ディレクトリ内の `gsdaemon_node_name.log` に格納します。GSD のクライアント (SRVCTL、DBCA、Oracle Enterprise Manager など) は GSD と対話して、クラスタ内のノードに対して様々な管理操作を実行します。

GSD は、管理機能および Oracle のツール製品が適切に動作するように、Real Applications Clusters データベース内のすべてのノードで起動する必要があります。

たとえば、Oracle Enterprise Manager を使用してインスタンスを起動すると、Intelligent Agent によって SRVCTL コマンドが起動されます。要求された操作に対応するこれらのコマンドは、GSD によって実行されます。

UNIX の GSD 実装

UNIX プラットフォーム上の GSD の名前は `gsd` で、`$ORACLE_HOME/bin` ディレクトリにあります。GSD は SRVCTL からの接続要求などの情報を記録し、これらのレコードを `$ORACLE_HOME/srvm/log` ディレクトリの `gsdaemon_node_name.log` ファイルに格納します。ここで、`node_name` はクラスタ・ファイル・システム環境をサポートするためのインスタンス番号を示します。

Windows の GSD 実装

Windows NT と Windows 2000 のプラットフォーム上の GSD の名前は *OracleGSDService* で、`%ORACLE_HOME%\bin` ディレクトリにあります。GSD サービスは SRVCTL からの接続要求などの情報を記録し、これらのレコードを `%ORACLE_HOME%\srvml\log` ディレクトリの `gsdaemon_node_name.log` ファイルに格納します。ここで、`node_name` はクラスタ・ファイル・システム環境をサポートするノードの名前を示します。

SRVCTL 管理タスク

この項では、SRVCTL ユーティリティを使用して Real Application Clusters を管理するタスクについて説明します。SRVCTL を使用すると、次の 2 つの管理タスクを実行できます。

- [SRVCTL クラスタ・データベースのタスク](#)
- [SRVCTL クラスタ・データベースの構成タスク](#)

SRVCTL クラスタ・データベースのタスク

- クラスタ・データベースの起動および停止
- クラスタ・データベース・インスタンスの起動および停止
- クラスタ・データベース、クラスタ・データベース・インスタンスのステータスの取得

SRVCTL クラスタ・データベースの構成タスク

- クラスタ・データベース構成情報の追加および削除
- クラスタ・データベース構成に対するインスタンスの追加および削除
- クラスタ・データベース構成内でのインスタンスの移動
- クラスタ・データベース構成内のインスタンス環境設定および解除
- クラスタ・データベース構成内のクラスタ・データベース全体の環境設定および解除

SRVCTL のコマンド構文

この項では、SRVCTL のコマンド構文について説明します。

SRVCTL の構文の構成要素

SRVCTL の構文には、次の構成要素があります。

```
srvctl verb noun options
```

説明：

- `srvctl` は、SRVCTL コマンドです。
- `verb` (動詞) は、起動、停止または削除などの動作を示すワードです。
- `noun` (名詞) は、SRVCTL が実行する動作の対象となるオブジェクト (`database` や `instance` など) です。略称を使用することもできます (この場合はそれぞれ `db` と `inst`)。
- `options` (オプション) は、先行する `verb` (動詞) と `noun` (名詞) の組合せの使用方法を拡張して、コマンドに情報を追加します。つまり、`-i` オプションは、カンマで区切られたインスタンス名のリストが続くことを示し (名前のリストではなく、1つの値のみを指定する場合があります)、`-n` オプションは、ノード名またはカンマで区切られたノード名のリストが続くことを示します。

各 SRVCTL コマンドのコマンド構文およびオプションをオンライン表示するには、次の構文を入力します。

```
srvctl verb noun -h
```

コマンドの動詞と名詞のリストを表示するには、次の構文を入力します。

```
srvctl
```

SRVCTL のバージョン・ナンバーを表示するには、次の構文を入力します。

```
srvctl -V
```

SRVCTL の同時実行コマンド

1つの SRVCTL アクションは、一度に1つのオブジェクトで実行されます。SRVCTL では、同じオブジェクトに対する複数コマンドの同時実行はサポートしていません。

SRVCTL 操作の停止

SRVCTL を停止するために [Ctrl] キーを押しながら [C] を押すと、この操作はトラップされます。SRVCTL が実行中のコマンドの進捗に関するメッセージを表示している場合は、これらのメッセージ表示が停止し、制御が戻ります。デフォルトでは、実行中または実行のキューに入っているコマンドは停止しません。

SRVCTL のエラー・メッセージ

SRVCTL のエラー・メッセージは、このマニュアルの付録 C 「Real Application Clusters 管理ツールのエラー・メッセージ」に記載されています。

SRVCTL の構文に共通する構成要素

SRVCTL コマンド構文で共通の動詞とオプションは、次のとおりです。SRVCTL で実行されるすべての操作では、大 / 小文字の区別はなく、そのまま保持されます。

コマンド構文

次の各項では、SRVCTL のコマンド構文について説明します。この項では、次の書式の構文について説明します。

```
srvctl verb noun [options]
```

共通の動詞

表 4-1 に、共通の動詞 (verb) をアルファベット順にリストします。これらの動詞は、任意の SRVCTL コマンドで使用できます。

表 4-1 SRVCTL で共通の動詞

動詞	意味
add	データベースまたはインスタンスを追加します。
config	データベースまたはインスタンスの構成をリストします。
getenv	SRVM 構成の環境変数をリストします。
modify	インスタンスの構成を変更します。
remove	データベースまたはインスタンスを削除します。
setenv	SRVM 構成の環境変数を設定します。
start	データベースまたはインスタンスを起動します。
status	データベースまたはインスタンスのステータスを表示します。

表 4-1 SRVCTL で共通の動詞（続き）

動詞	意味
stop	データベースまたはインスタンスを停止します。
unsetenv	SRVM 構成の環境変数を未指定の状態にします。

共通の名詞

表 4-2 に、共通の名詞 (noun) をアルファベット順にリストします。これらの名詞は、任意の SRVCTL コマンドで使用できます。

表 4-2 SRVCTL で共通の名詞

名詞 (略称)	意味
database (db)	操作対象はデータベースのオブジェクトです。
instance (inst)	操作対象はインスタンスのオブジェクトです。

構文のフラグ

表 4-3 に、SRVCTL コマンド構文の各種フラグとその意味を示します。

表 4-3 SRVCTL 構文のフラグ

フラグ	意味
-h	ヘルプ
-i	インスタンス
-n	ノード
-f	強制

共通のオプション

表 4-4 に、共通のコマンド・オプションをアルファベット順にリストします。これらのオプションは、任意の SRVCTL コマンドで使用できます。その他のオプションは、実行する動詞によって異なります。

表 4-4 SRVCTL で共通のコマンド・オプション

オプション	意味
-d	データベース名
-h	使用方法を表示

表 4-4 SRVCTL で共通のコマンド・オプション (続き)

オプション	意味
-i	操作対象を示すカンマで区切られたインスタンス名のリスト
-n	ノード名またはカンマ区切りのノードのリスト

SRVCTL のコマンド

次の各項では、SRVCTL のコマンドをアルファベット順に説明します。

SRVCTL Add

データベースまたは指定したインスタンスの構成情報を追加します。インスタンスを追加する場合、-i で指定する名前は、INSTANCE_NAME パラメータおよび ORACLE_SID パラメータに一致している必要があります。

```
srvctl add database -d database_name [-m domain_name] -o oracle_home [-s spfile]
srvctl add instance -d database_name -i instance_name -n node_name
```

表 4-5 SRVCTL Add コマンド固有のオプション

コマンド	オプション
-m	「us.mydomain.com」形式のデータベース・ドメイン名
-m オプションのデータベース・ドメイン名は、INIT.ORA または SPFILE の DB_DOMAIN パラメータおよび DB_NAME パラメータと一致している必要があります。データベースを追加する場合、-d で指定する名前は DB_NAME パラメータと一致している必要があります。	
-n	インスタンスをサポートするノード名
-o	lsnrctl (ノード・オプション) および Oracle バイナリ (その他のオプション) を配置する \$ORACLE_HOME
-s	SPFILE 名

SRVCTL Add の例

次に、add コマンドの使用例を示します。

新規データベースを追加するには、次の構文を入力します。

```
srvctl add database -d mydb -o /ora/ora9
```

指定したインスタンスをデータベースに追加するには、次の構文を入力します。

```
srvctl add instance -d mydb -i mydb01 -n gm01
srvctl add instance -d mydb -i mydb02 -n gm02
srvctl add instance -d mydb -i mydb03 -n gm03
```

SRVCTL Config

SRVM 構成ファイルに格納されている構成を表示します。

```
srvctl config database
```

構成されているデータベースのリストを表示します。

```
srvctl config database -d database_name
```

指定したデータベース構成は、次のフォーマットで表示されます。

```
nodename1 instancename1 oraclehome
nodename2 instancename2 oraclehome
```

SRVCTL Config の例

次に、config コマンドの使用例を示します。

データベース構成を表示するには、次の構文を入力します。

```
srvctl config database -d mydb
```

SRVCTL Getenv

getenv 操作は、環境に関する値を SRVM 構成ファイルから取得および表示します。

```
srvctl getenv database -d database_name [-t name[,name,...]]
srvctl getenv instance -d database_name -i instance_name [-t name[,name,...]]
```

SRVCTL Getenv の例

データベースの環境変数をすべてリストするには、次の構文を入力します。

```
srvctl getenv database -d mydb
```

SRVCTL Modify

インスタンスとノードの構成を変更可能にします。modify を使用して、(使用しない場合は再入力が必要となる) SRVM 構成の環境を保持します。構成の説明は、SRVM 構成リポジトリで変更されます。変更は、次のアプリケーション再起動時に有効になります。

インスタンス・オプションを指定して変更すると、指定したインスタンスが新規ノードに移動します。この新規ノードは、同じデータベースの別のインスタンスをサポートできません。この移動は永続的です。

```
srvctl modify instance -d database_name -i instance_name -n node_name
```

SRVCTL Modify の例

次に、`modify` コマンドの使用例を示します。

指定したインスタンスを別のノードで実行するように変更するには、次の構文を入力します。

```
srvctl modify instance -d mydb -n my_new_node
```

SRVCTL Remove

SRVM リポジトリから構成情報を削除します。オブジェクトの環境設定も削除されます。強制フラグ (`-f`) を使用しない場合は、削除を確認するプロンプトが表示されます。

次に、続行するかどうかの質問が表示されます。強制 (`-f`) オプションを指定すると、プロンプトの表示をせずに削除します。

```
srvctl remove database -d database_name [-f]
srvctl remove instance -d database_name -i instance_name [-f]
```

SRVCTL Remove コマンド固有のオプション

`-f` プロンプトの表示なしにアプリケーションの削除を強制実行します。

SRVCTL Remove の例

次に、`remove` コマンドの使用例を示します。

データベースのアプリケーションを削除するには、次の構文を入力します。

```
srvctl remove database -d mydb
```

データベースの指定したインスタンスのアプリケーションを削除するには、次の構文を入力します。

```
srvctl remove instance -d mydb -i mydb01
srvctl remove instance -d mydb -i mydb02
srvctl remove instance -d mydb -i mydb03
```

SRVCTL Setenv

setenv 操作は、SRVM 構成ファイル内の環境に関する値を設定します。

```
srvctl setenv database -d database_name -t name=value [,name=value,...]
srvctl setenv instance -d database_name [-i instance_name] -t name=value
[,name=value,...]
```

SRVCTL Setenv の例

次に、setenv コマンドの使用例を示します。

データベースの環境を設定するには、次の構文を入力します。

```
srvctl setenv database -d mydb -t LANG=en
```

SRVCTL Start

データベース、すべてのインスタンス、または指定したインスタンス、およびデータベースに関連付けられているすべてのリスナーを起動します（それぞれ起動されていない場合）。

注意： start コマンドの場合および接続文字列を使用するその他の操作の場合に、接続文字列を指定しないと、Oracle は「/ as sysdba」を使用してインスタンスでの操作を実行します。つまり、このような操作を実行するには、OSDBA グループのメンバーであることが必要です。OSDBA グループ・メンバーシップの詳細については、『Oracle9i のインストール・ガイド』を参照してください。

```
srvctl start database -d database_name [-o start_options] [-c connect_string]
srvctl start instance -d database_name -i instance_name [,instance_name-list] [-o
start_options] [-c connect_string]
```

表 4-6 SRVCTL Start コマンド固有のオプション

コマンド	オプション
-o	PFILE を含む、SQL*Plus の startup コマンドに直接渡されるオプション
-c	SQL*Plus を使用して Oracle インスタンスに接続するための接続文字列

SRVCTL Start の例

次に、start コマンドの使用例を示します。

データベースと使用可能なすべてのインスタンスを起動するには、次の構文を入力します。

```
srvctl start database -d mydb
```

インスタンスを起動するには、次の構文を入力します。

```
srvctl start instance -d mydb -i mydb1,mydb4
```

SRVCTL Status

指定したデータベースの現在の状態を表示します。

```
srvctl status database -d database_name  
srvctl status instance -d database_name -i instance_name [,instance_name-list]
```

SRVCTL Status の例

次に、status コマンドの使用例を示します。

データベースと全インスタンスのステータスを表示するには、次の構文を入力します。

```
srvctl status database -d mydb
```

指定したインスタンスのステータスを表示するには、次の構文を入力します。

```
srvctl status instance -d mydb -i mydb1,mydb2
```

SRVCTL Stop

データベースとすべてのインスタンス、または指定したインスタンスを停止します。このコマンドでリスナーが停止することはありません。

```
srvctl stop database -d database_name [-o stop_options] [-c connect_string]  
srvctl stop instance -d database_name -i instance_name [,instance_name_list] [-o  
stop_options] [-c connect_string]
```

表 4-7 SRVCTL Stop コマンド固有のオプション

コマンド	オプション
-c	SQL*Plus を使用して Oracle インスタンスに接続するための接続文字列
-o	SQL*Plus の shutdown コマンドに直接渡されるオプション

SRVCTL Stop の例

次に、stop コマンドの使用例を示します。

データベースとすべてのインスタンスを停止するには、次の構文を入力します。

```
srvctl stop database -d mydb
```

指定したインスタンスを停止するには、次の構文を入力します。

```
srvctl stop instance -d mydb -i mydb1
```

SRVCTL Unsetenv

unsetenv 操作は、SRVM 構成ファイルの環境に関する値の設定を解除します。

```
srvctl unsetenv database -d database_name -t name[,name,...]
srvctl unsetenv instance -d database_name [-i instance_name] -t name[,name,...]
```

SRVCTL Unsetenv の例

次に、unsetenv コマンドの使用例を示します。

環境変数を未指定の状態にするには、次の構文を入力します。

```
srvctl unsetenv database -d mydb -t CLASSPATH
```

SRVCONFIG を使用した RAW デバイス構成のインポートとエクスポート

SRVCONFIG を使用すると、構成ファイルがクラスタ・ファイル・システムのファイルまたは RAW デバイスにあるかどうかに関係なく、RAW デバイス構成情報をインポートおよびエクスポートできます。このインポートとエクスポートによって、SRVM 構成情報をバックアップまたはリストアできます。たとえば、次の構文を使用すると、指定のテキスト・ファイルに構成情報の内容がエクスポートされます。

```
srvconfig -exp file_name
```

また、次の例では、指定したテキスト・ファイルから、コマンドを実行する Real Application Clusters 環境の構成リポジトリに構成情報がインポートされます。

```
srvconfig -imp file_name
```

Oracle8i から Oracle9i への構成のアップグレード

Oracle8i から Oracle9i にアップグレードする場合は、次に示すインストール後の手順を実行して構成情報をアップグレードします。この手順は、Real Application Clusters データベースごとに実行します。

1. `gsdctl stop` コマンドを実行して、すべてのグローバル・サービス・デーモン (GSD) を停止します。
2. `db_name.conf` ファイルがある UNIX クラスタ内のノードから、次のコマンドを実行します。

```
srvconfig -conv $Oracle_Home/ops/db_name.conf
```

GSD の管理

gsdctl コマンドを使用すると、プラットフォーム上の GSD サービスの起動、停止およびステータスの取得を実行できます。gsdctl のオプションは、次のとおりです。

- gsdctl start: GSD サービスを起動します。
- gsdctl stop: GSD サービスを停止します。
- gsdctl stat: GSD サービスのステータスを取得します。

SQL および SQL*Plus を使用した Real Application Clusters データベースの管理

Real Application Clusters データベースの管理に SRVCTL を使用することをお勧めしますが、SQL および SQL*Plus を使用することもできます。この項を参照して SQL および SQL*Plus を使用してタスクを実行する前に、各ノードで **Cluster Manager (CM)** コンポーネントが起動されていることを確認してください。

クラスタ・データベースを起動するために使用する SQL および SQL*Plus のプロシージャは、プラットフォームによって異なります。この項の内容は次のとおりです。

- [UNIX におけるクラスタ・モードでのデータベースの起動](#)
- [Windows NT および Windows 2000 におけるクラスタ・モードでのデータベースの起動](#)
- [RETRY を使用したクラスタ・モードでのデータベースのマウント](#)

UNIX におけるクラスタ・モードでのデータベースの起動

Real Application Clusters データベースをクラスタ・モードで起動するには、次の手順を実行します。

1. **Cluster Manager** ソフトウェアが実行されていることを確認します。**Cluster Manager** ソフトウェアの管理方法については、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。**Cluster Manager** が使用できない場合、または Oracle がこのコンポーネントと通信できない場合、「ORA-29701: Cluster Manager に接続できません。」というエラー・メッセージが表示されます。
2. オペレーティング・システム固有の必須プロセスを起動します。これらのプロセスの詳細は、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。
3. **リスナー** が起動されていない場合は、各ノードで起動します。次のコマンドを入力します。

```
LSNRCTL  
LSNRCTL> start [listener_name]
```

ここで、*listener_name* は **listener.ora** ファイルで定義されているリスナー名です。デフォルトのリスナー名 LISTENER を使用する場合、リスナーを識別する必要はありません。

LSNRCTL を使用すると、リスナーが正常に起動されたことを示すステータス・メッセージが表示されます。そのリスナーに使用可能であるはずのすべてのサービスが、ステータス・メッセージのサービス・サマリーに示されているかどうかを確認できます。リスナーの状態は、LSNRCTL STATUS コマンドを使用しても確認できます。

4. SQL*Plus を起動して、いずれかのノードでデータベースを起動します。その後、次のコマンドを入力します。

```
CONNECT SYS/password as SYSDBA
STARTUP PFILE=init$ORACLE_sid.ora
```

クラスタ・モードで起動する最初のインスタンスによって、他のインスタンスのすべてのグローバル・パラメータ値が決まります。別のインスタンスをクラスタ・モードで起動すると、Real Application Clusters データベースは、パラメータ・ファイルのすべてのグローバル・パラメータの値をすでに使用されている値と比較して、いずれかの値が適切でない場合はメッセージを発行します。インスタンスのグローバル・パラメータ値が正しくない場合、そのインスタンスはデータベースをマウントできません。

5. 残りのノードで、データベースを起動します。

```
CONNECT SYS/password as SYSDBA
STARTUP PFILE=$ORACLE_sid.ora;
```

Windows NT および Windows 2000 におけるクラスタ・モードでのデータベースの起動

Windows プラットフォームで Real Application Clusters データベースをクラスタ・モードで起動するには、次の手順を実行します。

1. 各ノードで OracleServicesid インスタンスを起動します。
 - MS-DOS コマンドラインから次のコマンドを入力します。


```
C:¥> net start OracleServicesid
```
 - コントロールパネルの「サービス」ウィンドウから「OracleServicesid」を選択し、「Start」をクリックします。
2. リスナーが起動されていない場合、各ノードで起動します。次のコマンドを入力します。

```
LSNRCTL
LSNRCTL> start [listener_name]
```

ここで、*listener_name* は *listener.ora* ファイルで定義されているリスナー名です。デフォルトのリスナー名 *LISTENER* を使用する場合、リスナーを識別する必要はありません。

LSNRCTL を使用すると、リスナーが正常に起動されたことを示すステータス・メッセージが表示されます。そのリスナーに使用可能であるはずのすべてのサービスが、ステータス・メッセージのサービス・サマリーにリストされているかどうかを確認できます。リスナーの状態は、*LSNRCTL STATUS* コマンドを使用しても確認できます。

3. **SQL*Plus** を起動して、いずれかのノードでデータベースを起動します。その後、次のコマンドを入力します。

```
CONNECT SYS/password
STARTUP PFILE=%ORACLE_HOME%\%database%\init%instance_name.ora;
```

クラスタ・モードで起動する最初のインスタンスによって、他のインスタンスのすべてのグローバル・パラメータ値が決まります。別のインスタンスをクラスタ・モードで起動すると、**Real Application Clusters** データベースは、パラメータ・ファイルのすべてのグローバル・パラメータの値をすでに使用されている値と比較して、いずれかの値が適切でない場合はメッセージを発行します。インスタンスのグローバル・パラメータ値が正しくない場合、そのインスタンスはデータベースをマウントできません。

4. 残りのノードで、データベースを起動します。

```
CONNECT SYS/password
STARTUP PFILE=%ORACLE_HOME%\%database%\init%instance_name.ora;
```

RETRY を使用したクラスタ・モードでのデータベースのマウント

他のインスタンスが同じデータベースをリカバリ中にインスタンスを起動し、クラスタ・モードでデータベースをマウントしようとしても、リカバリが完了するまでは、現行のインスタンスはデータベースをマウントできません。インスタンスの起動を繰り返し実行するか代わりに、*STARTUP RETRY* 文を使用します。これによって、再試行が成功するか、再試行制限に達するまで、新しいインスタンスは 5 秒ごとにデータベースをマウントしようとしません。次の構文を使用します。

```
STARTUP OPEN database_name RETRY
```

インスタンスがデータベースをマウントしようとする最大回数を設定するには、**SQL*Plus** の *SET* コマンドに *RETRY* オプションを指定して使用します。正数 (10 など)、またはキーワード *INFINITE* を指定できます。

他のインスタンスによるリカバリでしかデータベースがオープンできない場合、*RETRY* を使用しても接続の試行は繰り返されません。たとえば、あるインスタンスでデータベースが排他モードでマウントされている場合は、他のインスタンスがクラスタ・モードで *STARTUP RETRY* コマンドを実行しても機能しません。

注意： インスタンスを起動してもデータ・ファイルに影響しないため、データ・ファイルをマウントせずにインスタンスを起動できます。

インスタンスの設定および接続

インスタンスを設定して、接続する前に、Real Application Clusters のノードおよびこれらのノードにアクセスするクライアントに、Oracle Net をインストールおよび構成する必要があります。これによって、クライアントはノードにリモート接続を確立できます。

関連項目：

- 『Oracle9i Real Application Clusters セットアップおよび構成』
- 『Oracle9i Net Services 管理者ガイド』

SQL*Plus コマンドは、4-18 ページの「[SET INSTANCE および SHOW INSTANCE コマンド](#)」に示すいくつかの場合を除いて、現行のインスタンスで処理されます。

現行のインスタンスは、SQL*Plus セッションを開始するローカル・インスタンスまたはリモート・インスタンスです。SQL*Plus のプロンプトでは、どのインスタンスが現行のインスタンスであるかは表示されないため、正しいインスタンスにコマンドを発行する必要があります。

SQL*Plus セッションを開始して、インスタンスを指定せずにデータベースに接続すると、すべての SQL*Plus コマンドはローカル・インスタンスで処理されます。この場合も、デフォルト・インスタンスが現行のインスタンスです。

現行のインスタンスをローカル・インスタンスからリモート・インスタンスに切り替えるには、次のいずれかを実行します。

- リモート・インスタンス・[ネット・サービス名](#)を指定して、次の CONNECT コマンドを再実行します。

```
CONNECT SYSTEM/MANAGER@net_service_name
```

- データベースからの接続を切断し、次の SET INSTANCE コマンドを実行します。

```
SET INSTANCE net_service_name
```

ユーザー ID およびパスワードのみを指定して、CONNECT コマンドを再発行します。インスタンス経由でデータベースに接続した状態で、CONNECT コマンドを使用してリモート・インスタンスを指定すると、切断せずに 1 つのインスタンスから別のインスタンスに切り替えることができます。

関連項目：

- ネット・サービス名の構成の詳細は、『Oracle9i Net Services 管理者ガイド』を参照してください。
- SET INSTANCE コマンドおよび CONNECT コマンドで使用する接続文字列の完全な書式については、オペレーティング・システム固有の Oracle マニュアルを参照してください。

SET INSTANCE および SHOW INSTANCE コマンド

SET INSTANCE を使用して、リモート・ノード上のインスタンスに対して STARTUP コマンドを実行する場合、リモート・インスタンスのパラメータ・ファイルは、ローカル・ノードからアクセス可能である必要があります。

SHOW INSTANCE コマンドは、現行のインスタンスのネット・サービス名を表示します。SQL*Plus セッション中に SET INSTANCE を使用しなかった場合、SHOW INSTANCE は、値 local を戻します。

デフォルト・インスタンスにリセットするには、ネット・サービス名を指定せずに SET INSTANCE を使用するか、local を指定します。SET INSTANCE の後に default という単語は指定しないでください。これは、default という名前のインスタンスの接続文字列を示します。

CONNECT コマンド

SYSOPER または **SYSDBA** で接続すると、インスタンスの起動や停止などの権限を必要とする操作を実行できます。複数の SQL*Plus セッションが、同時に同じインスタンスに接続できます。他のインスタンスに接続すると、SQL*Plus によって最初のインスタンスとの接続が自動的に切断されます。

関連項目：

- ネット・サービス名の適切な指定については、『Oracle9i Net Services 管理者ガイド』を参照してください。
- SYSDBA 権限または SYSOPER 権限でデータベースに接続する方法については、『Oracle9i データベース管理者ガイド』を参照してください。

インスタンスの実行の確認

インスタンスの実行を確認するには、次のとおり実行します。

1. 任意のノードで、次のコマンドを入力します。

```
CONNECT SYS/password
SELECT * FROM V$ACTIVE_INSTANCES;
```

次のような出力が表示されます。

```
INST_NUMBER INST_NAME
-----
1 db1-sun:db1
2 db2-sun:db2
3 db3-sun:db3
```

表 4-8 に、この SELECT 文で出力される列について説明します。

表 4-8 V\$ACTIVE_INSTANCES 列の説明

列	説明
INST_NUMBER	インスタンス番号を識別します。
INST_NAME	ホスト名およびインスタンス名を識別します。

Real Application Clusters のインスタンスの停止

Real Application Clusters インスタンスの停止方法は、シングル・インスタンス環境でインスタンスを停止する場合と同じです。ただし、次の相違点があります。

- Real Application Clusters では、1つのインスタンスを停止しても、実行中の他のインスタンスの操作を妨げることはありません。
- 共有モードでマウントした Real Application Clusters データベースを停止するには、Real Application Clusters 環境のすべてのインスタンスを停止します。
- インスタンスが異常終了した場合、Oracle はそのインスタンスで実行中のすべてのユーザー・プロセスを、データベースから強制的にログオフします。現在、データベースにアクセス中のユーザー・プロセスがある場合、Oracle はそのアクセスを終了して、「ORA-01092: Oracle インスタンスが終了しました。強制的に切断されます」というエラー・メッセージを表示します。インスタンスが停止したときにユーザー・プロセスがデータベースにアクセスしていない場合、Oracle は、次のコール時または Oracle への要求実行時に、「ORA-01012: ログオンされていません。」というエラー・メッセージを表示します。
- NORMAL または IMMEDIATE での停止後は、インスタンスのリカバリは不要です。ただし、SHUTDOWN ABORT コマンドを発行した後、またはインスタンスが異常終了した後は、リカバリが必要です。まだ実行中のインスタンスの SMON プロセスが、停止した

インスタンスに対してインスタンスのリカバリを実行します。他に実行中のインスタンスがない場合は、次にデータベースをオープンするインスタンスが、リカバリが必要なすべてのインスタンスのリカバリを実行します。

- LOCAL オプションを指定した SHUTDOWN TRANSACTIONAL コマンドは、インスタンス上のすべてのアクティブ・トランザクションがコミットまたはロールバックされた後に、そのインスタンスを停止する場合に有効です。これは、このコマンドが SHUTDOWN IMMEDIATE の場合に実行する機能とは別の機能です。他のインスタンス上のトランザクションがこの操作を妨げることはありません。LOCAL オプションを省略した場合、この操作は、停止前に起動された他のすべてのインスタンス上のトランザクションがコミットまたはロールバックされるまで待機します。
- 複数の SQL*Plus セッションが同じインスタンスに同時に接続されている場合、インスタンスを正常に停止するには、1つを除くすべてのセッションを切断しておく必要があります。複数の SQL*Plus セッション（または他のすべてのセッション）がインスタンスに接続されている場合は、SHUTDOWN コマンドの IMMEDIATE オプションまたは ABORT オプションを使用して、そのインスタンスを停止できます。

関連項目： Oracle データベースを停止する方法の詳細については、『Oracle9i データベース管理者ガイド』を参照してください。

Real Application Clusters データベースの静止

Real Application Clusters データベースの静止方法は、シングル・インスタンス・データベースを静止する場合と同じです。ただし、この項で説明する例外があります。たとえば、データベースが静止中の場合は、1つのインスタンスでそのデータベースをオープンできません。ALTER SYSTEM QUIESCE RESTRICTED 文を発行しても、Oracle がその処理を完了していない場合、そのデータベースをオープンできません。データベースがすでに静止している場合も、そのデータベースをオープンできません。さらに、ALTER SYSTEM QUIESCE RESTRICTED 文および ALTER SYSTEM UNQUIESCE 文を実行すると、コマンドを発行したインスタンスのみでなく、Real Application Clusters 環境のすべてのインスタンスが影響を受けます。

関連項目： データベース静止機能の詳細については、『Oracle9i データベース管理者ガイド』を参照してください。ALTER SYSTEM QUIESCE RESTRICTED 構文の詳細については、『Oracle9i SQL リファレンス』を参照してください。

インスタンスへの SQL および SQL*Plus コマンドの適用方法

ほとんどの SQL 文は、現行のインスタンスに適用されます。たとえば、ALTER SYSTEM SET CHECKPOINT LOCAL 文は、デフォルト・インスタンスまたはすべてのインスタンスではなく、現在接続しているインスタンスにのみ適用されます。

ALTER SYSTEM CHECKPOINT LOCAL も、現行のインスタンスに適用されます。対照的に、ALTER SYSTEM CHECKPOINT または ALTER SYSTEM CHECKPOINT GLOBAL は、すべてのインスタンスに適用されます。

ALTER SYSTEM SWITCH LOGFILE は、現行のインスタンスにのみ適用されます。グローバル・ログ・スイッチを強制するには、ALTER SYSTEM ARCHIVE LOG CURRENT 文を使用します。ALTER SYSTEM ARCHIVE LOG の THREAD オプションを使用すると、特定のインスタンスの各オンライン REDO ログ・ファイルをアーカイブできます。

表 4-9 に、共通の SQL*Plus コマンドのインスタンスへの適用方法を示します。

表 4-9 SQL*Plus コマンドのインスタンスへの適用方法

SQL*Plus コマンド	SQL*Plus コマンドが適用されるインスタンス
ARCHIVE LOG	常に現行のインスタンスに適用されます。
CONNECT	CONNECT コマンドにインスタンスが指定されていない場合は、デフォルト・インスタンスに適用されます。
HOST	現行のインスタンスおよびデフォルト・インスタンスの位置に関係なく、SQL*Plus セッションを実行しているノードに適用されます。
RECOVER	特定のインスタンスではなく、データベースに適用されます。
SHOW INSTANCE	現行のインスタンスに関する情報を表示します。リモート・インスタンスにコマンドをリダイレクトしている場合、現行のインスタンスはデフォルトのローカル・インスタンスではない場合があります。
SHOW PARAMETER および SHOW SGA	現行のインスタンスからパラメータおよび SGA 情報を表示します。
STARTUP および SHUTDOWN	常に現行のインスタンスに対して適用されます。権限付きの SQL*Plus コマンドです。

注意： 権限付きの SQL*Plus コマンドの実行時に Oracle が使用するセキュリティ・メカニズムは、オペレーティング・システムによって異なります。ほとんどのオペレーティング・システムは、オペレーティング・システムへのログイン用に安全性の高い認証メカニズムを備えています。これらのシステムでは、通常、STARTUP および SHUTDOWN が使用できるかどうかは、オペレーティング・システムのデフォルトの権限によって決まります。詳細については、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。

第 II 部

Oracle Enterprise Manager を使用した Real Application Clusters の管理

第 II 部では、Oracle Enterprise Manager の Server Management (SRVM) コンポーネントの使用について説明します。第 II 部に含まれる章は、次のとおりです。

- 第 5 章「Oracle Enterprise Manager を使用した Real Application Clusters データベースの管理」

Oracle Enterprise Manager を使用した Real Application Clusters データベースの管理

この章では、**Oracle Enterprise Manager** を使用して、Real Application Clusters データベースを管理する方法について説明します。ここでは、Oracle Enterprise Manager による Real Application Clusters の管理についてのみ説明します。『Oracle Enterprise Manager 管理者ガイド』および Oracle Enterprise Manager の他のマニュアルの補足としてお読みください。内容は次のとおりです。

- **Oracle Enterprise Manager** での管理の概要
- コンソールの起動
- ナビゲータ・ペインでのオブジェクトの表示
- クラスタ・データベースのコンテキスト・メニューの使用
- クラスタ・データベースの起動
- クラスタ・データベースの停止
- クラスタ・データベース操作結果の表示
- クラスタ・データベースのステータス表示
- クラスタ・データベースまたはインスタンスに対するジョブの作成
- ジョブの詳細の指定
- クラスタ・データベース・イベントの登録

Oracle Enterprise Manager での管理の概要

Oracle Enterprise Manager の **コンソール** では、Graphical User Interface (GUI) によって、Oracle 環境を集中的に制御できます。Oracle Enterprise Manager のコンソールを使用すると、様々なクラスタ・データベース管理タスクを開始できます。

クラスタ・データベースの構成を作成または変更した場合は、Oracle Enterprise Manager を使用して、そのクラスタ・データベースを構成したノードを検出する必要があります。検出が完了すると、Oracle Enterprise Manager によって検出されたノードに対応付けられているデータベースとそのインスタンスおよびリスナーをすべて管理できます。

ジョブのスケジュールやイベントの登録に加え、データベース、インスタンスおよびそのリスナーを起動、停止および監視できます。これらのタスクは、複数のクラスタ・データベースで同時に実行できます。コンソールを使用すると、スキーマ、セキュリティおよびクラスタ・データベースの格納機能を管理することもできます。

関連項目： 構成情報については『Oracle9i Real Application Clusters セットアップおよび構成』を、FAQ に対する回答については Oracle Enterprise Manager のオンライン・ヘルプ・システムを参照してください。

コンソールの起動

Oracle Enterprise Manager のコンソールを使用するには、次のコンポーネントを起動します。

1. 各ノードの **Oracle Intelligent Agent**
2. 各ノードの **グローバル・サービス・デーモン**
3. **Management Server**
4. Oracle Enterprise Manager のコンソール

関連項目： Oracle Enterprise Manager をインストールする際の Real Application Clusters 固有の情報については『Oracle9i Real Application Clusters セットアップおよび構成』を、Oracle Intelligent Agent の詳細については『Oracle Intelligent Agent ユーザーズ・ガイド』を参照してください。

ナビゲータ・ペインでのオブジェクトの表示

ナビゲータ・ペインでは、シングル・インスタンス・データベースを表示および管理する場合と同じ方法で、クラスタ・データベース・インスタンスを表示および管理できます。

Oracle Enterprise Manager によってノードが検出されると、「Navigator」ウィンドウに、クラスタ・データベース、そのインスタンスおよびそのノードで検出されたリスナーがすべて表示されます。

クラスタ・データベースで有効な情報は、基本的に、シングル・インスタンス・データベースの場合と同じです。シングル・インスタンス・データベースの場合と同様に、クラスタ・データベースとそのすべての関連要素は、Oracle Enterprise Manager のコンソールのマスター・ビューとディテール・ビューを使用して管理できます。

ナビゲータ・ツリーでは、クラスタ・データベースは、「Databases」フォルダの下の、シングル・インスタンス・データベースと同じレベルに位置しています。5-5 ページの図 5-1 に示すように、シングル・インスタンス・データベースと同様、各クラスタ・データベースのフォルダには、インスタンスと「Instance」、「Schema」、「Security」、「Storage」および「Distributed」というサブフォルダが含まれています。

注意：『Oracle9i Real Application Clusters セットアップおよび構成』に従ってデータベースに優先接続情報を設定する前に、クラスタ・データベースを展開しようとする、「Database Connect Information」ダイアログ・ボックスに、データベースの接続情報の入力を求めるプロンプトが表示されます。

「Cluster Database」サブフォルダ内のオブジェクトを選択することで、プロパティ・シートにアクセスし、シングル・インスタンス・データベースの場合と同様に、これらのオブジェクトのプロパティを確認および変更できます。たとえば、新規に REDO ログ・グループを追加するには、「Storage」フォルダを展開し、「Redo Log Groups」サブフォルダを右クリックして、「Create」を選択します。「Cluster Database Instances」フォルダの下に、検出されたすべてのクラスタ・データベース・インスタンスが表示されます。

クラスタ・データベースにアクセスすると、「Instance」フォルダには、シングル・インスタンス・データベースの場合と異なるサブフォルダの内容のみが表示されます。「Instance」フォルダ内で、インスタンス・データベースのサブフォルダは次の 2 つの部分に分割されています。

- データベース固有のオブジェクト
- インスタンス固有のオブジェクト

データベース固有のオブジェクト

データベース固有のすべての機能は、クラスタ・データベースの「Instance」サブフォルダの直下に表示されます。使用可能なデータベース固有の機能によって、リソース・コンシューマ・グループを表示、作成および変更できます。

インスタンス固有のオブジェクト

インスタンス固有のすべての機能は、「Cluster Database Instances」サブフォルダ内の個々のインスタンス・アイコンの下に表示されます。インスタンス固有の機能は次のとおりです。

Configuration

インスタンスの状態の表示、初期化パラメータの表示と編集、アーカイブ・モードのオン / オフの切替え、およびアクティブなリソース・プランのパフォーマンス統計情報の表示が可能です。また、インスタンスに割り当てられている現在の UNDO 表領域とその保存期間を表示または変更できます。

Stored Configurations

インスタンスの複数の起動構成を作成、編集および格納できます。これによって、`initsid.ora` パラメータ・ファイルを追跡する必要がなくなります。

Sessions

接続ユーザーのステータスの表示、セッションに対する最新の SQL の表示およびセッションの終了ができます。

Locks

現在保持されているユーザー・タイプ・ロックおよびシステム・タイプ・ロックに関する詳細を表示できます。

Resource Plans

クラスタ・データベースのリソース・プランを定義および変更できます。個々のインスタンスに対して特定のリソース・プランをアクティブにすることもできます。

Resource Plan Schedule

リソース・プランの実行をスケジュールリングできます。

図 5-1 に、一般的なデータベースのフォルダの内容を示します。

図 5-1 データベースのサブフォルダ

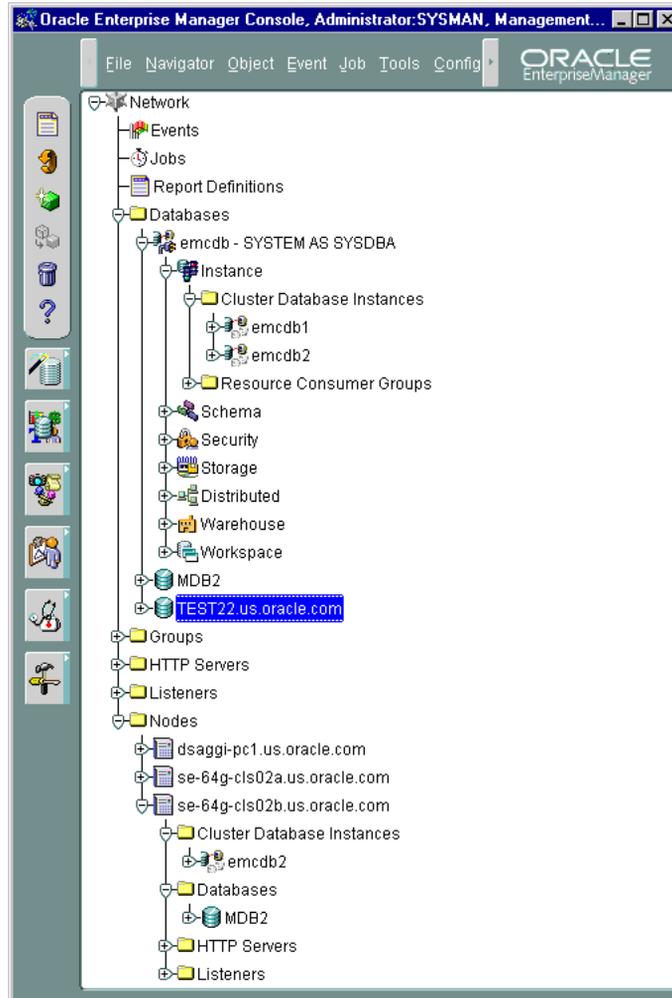
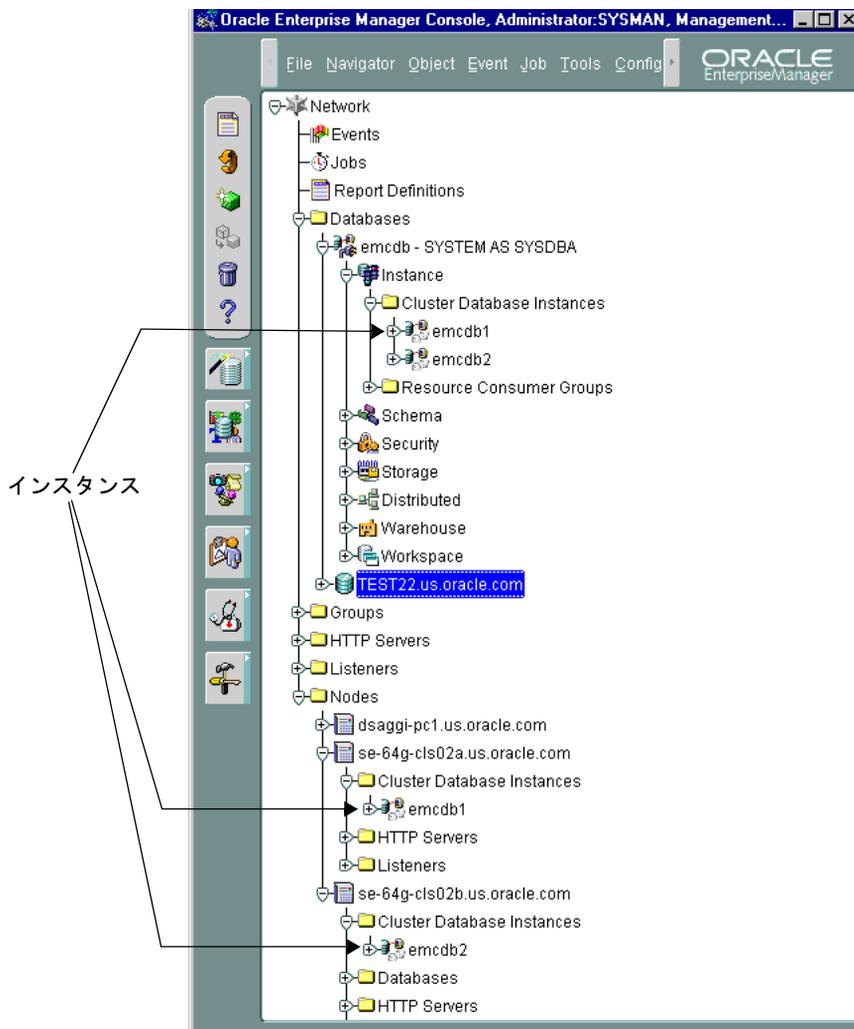


図 5-2 に示すように、「Cluster Database Instances」フォルダを展開すると、検出された各データベースまたはノードに属するインスタンスが表示されます。

図 5-2 「Cluster Database Instances」 フォルダ



注意： インスタンスには、シングル・インスタンス・データベースと同様のコンテキスト・メニューがあります。

クラスタ・データベースのコンテキスト・メニューの使用

クラスタ・データベース上で右クリックすると、クラスタ・データベース固有のコンテキスト・メニューが表示されます。コンテキスト・メニューには、表 5-1 に示す機能のエントリが含まれています。その他すべてのメニューは、シングル・インスタンスの Oracle Enterprise Manager に表示されるメニューと同じです。選択すると、シングル・インスタンス・データベースとは異なるダイアログ・ボックスが表示されます。

表 5-1 Real Application Clusters のコンテキスト・メニューの機能

オプション	説明
Startup	データベースを起動します。 関連項目 : 5-8 ページ「 クラスタ・データベースの起動 」
Shutdown	データベースを停止します。 関連項目 : 5-9 ページ「 クラスタ・データベースの停止 」
Results	起動および停止の結果を表示します。 関連項目 : 5-11 ページ「 クラスタ・データベース操作結果の表示 」
View Edit Details	アクティブなインスタンスを含む、クラスタ・データベースの状態を確認できます。 関連項目 : 5-15 ページ「 クラスタ・データベースのステータス表示 」
Related Tools	クラスタ・データベースで使用可能な他のツールへアクセスできます。

クラスタ・データベースの起動

コンソールを使用して、クラスタ・データベース全体を起動するか、またはクラスタ・データベース内で選択したインスタンスを起動できます。クラスタ・データベースまたはそのインスタンスを起動するには、次の手順を実行します。

1. ナビゲータ・ペインで、「Databases」フォルダを展開します。
2. クラスタ・データベースを右クリックします。
データベースのオプション・メニューが表示されます。
3. メニューから「Startup」を選択します。
「Cluster Database Startup」ダイアログ・ボックスが表示されます。
4. 起動タイプを選択します。表 5-2 に、選択できるタイプを示します。

表 5-2 クラスタ・データベースの起動タイプ

オプション	説明
No Mount	インスタンスの起動時にデータベースをマウントしません。
Mount	データベースをマウントしますが、オープンしません。
Open	(デフォルト) データベースをオープンします。
Force the instance(s) to start	現在実行中の Oracle インスタンスを SHUTDOWN モードの ABORT で停止してから再起動します。インスタンスが実行中で FORCE が指定されない場合、エラーが発生します。 注意: 通常の状態下では FORCE モードを使用しないでください。FORCE モードは、デバッグ時または異常な状況のみで使用してください。
Restrict access to the database	起動されたインスタンスに、RESTRICTED SESSION システム権限を持つユーザーのみがアクセスできるようにします。常時接続しているユーザーは影響を受けません。

5. すべてのインスタンスおよび関連するリスナーを起動するには、「Startup」をクリックします。選択したインスタンスのみを起動するには、次の手順を実行します。
 - a. 「Select Instances」をクリックします。
「Select Instances to Start」ダイアログ・ボックスが表示されます。
 - b. 起動するインスタンスを「Available」リストで選択し、「Add」をクリックします。
 - c. 「OK」をクリックして、「Select Instances to Start」ダイアログ・ボックスを閉じます。

d. 「Cluster Database Startup」ダイアログ・ボックスで「Startup」をクリックします。

5-11 ページの「クラスタ・データベース操作結果の表示」に示すとおり、「Cluster Database Startup Results」ダイアログ・ボックスに起動操作の進捗が表示されます。インスタンスが正常に起動されると、「Cluster Database Started」メッセージ・ボックスに正常終了時メッセージが表示されます。「Cluster Database Started」メッセージ・ボックスで「OK」をクリックしてメッセージを受け入れ、「Cluster Database Startup Results」ダイアログ・ボックスで「Close」をクリックします。

起動に失敗すると、「Cluster Database Started」メッセージ・ボックスに失敗時メッセージが表示されます。障害に関する情報を表示するには、「Cluster Database Startup Results」ダイアログ・ボックスの「View Details」をクリックします。詳細を確認した後、「Close」をクリックします。

クラスタ・データベースの停止

コンソールを使用して、クラスタ・データベース全体またはクラスタ・データベース内で選択したインスタンスを停止できます。いったんすべてのインスタンスが停止すると、データベースも停止しているとみなされます。

注意： クラスタ・データベースは完全に停止されますが、データベース・リスナーなどの一部のサービスは引き続き実行されています。

クラスタ・データベースまたはそのインスタンスを停止するには、次の手順を実行します。

1. ナビゲータ・ペインで、「Databases」フォルダを展開します。

2. クラスタ・データベースを右クリックします。

データベースのオプション・メニューが表示されます。

3. メニューから「Shutdown」を選択します。

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

4. 停止タイプを選択します。表 5-3 に、選択できるタイプを示します。

表 5-3 停止タイプ

オプション	説明
Normal	現在接続しているユーザーがデータベースへの接続を切断するまで待機し、それ以降の接続を禁止し、データベースをクローズおよびディスマウントしてからインスタンスを停止します。次の起動時に、インスタンスのリカバリは必要ありません。

表 5-3 停止タイプ (続き)

オプション	説明
Immediate	(デフォルト) 現在のコールが完了するまで待機せず、それ以降の接続を禁止し、データベースをクローズおよびディスマウントします。インスタンスがすぐに停止します。接続ユーザーが切断する必要はなく、次の起動時に、インスタンスのリカバリは必要ありません。
Abort	できるだけ速く停止します。接続ユーザーが切断する必要はありません。データベースはクローズまたはディスマウントされませんが、インスタンスは停止します。次の起動時には、インスタンスのリカバリが必要です。 注意: バックグラウンド・プロセスが異常終了した場合、このオプションを使用する必要があります。
Transactional	トランザクションが完了するまで停止を待機します。
Shutdown Database Only	(デフォルト) データベースのみを停止します。インスタンスに必要なリスナーなどのサービスは引き続き実行され、使用可能です。
Shutdown Database And Other Services	データベース、およびリスナーなどの関連サービスを停止します。

5. すべてのインスタンスを停止するには、「Shutdown」をクリックします。

選択したインスタンスのみを停止するには、次の手順を実行します。

- a. 「Select Instances」をクリックします。
「Select Instances to Stop」ダイアログ・ボックスが表示されます。
- b. 停止するインスタンスを「Available」リストで選択し、「Add」をクリックします。
- c. 「OK」をクリックして、「Select Instances to Stop」ダイアログ・ボックスをクローズします。
- d. 「Cluster Database Shutdown」ダイアログ・ボックスで「Shutdown」をクリックします。

「Cluster Database Shutdown Progress」ダイアログ・ボックスに、停止操作の処理経過が表示されます。

インスタンスが正常に停止されると、「Cluster Database Stopped」メッセージ・ボックスに正常終了時メッセージが表示されます。「Cluster Database Stopped」メッセージ・ボックスで「OK」をクリックしてメッセージを受け入れ、「Cluster Database Shutdown Results」ウィンドウで「Close」をクリックします。

停止が失敗すると、「Cluster Database Stopped」メッセージ・ボックスに失敗時メッセージが表示されます。「Cluster Database Shutdown Progress」ダイアログ・ボックスに停止の失敗に関する情報を表示するには、「View Details」をクリックします。詳細を確認した後、「Close」をクリックします。

クラスタ・データベース操作結果の表示

「Cluster Database Startup/Shutdown Results」ダイアログ・ボックスには、選択したインスタンスの起動操作および停止操作の処理経過に関する情報が表示されます。処理結果は、次の2つのビューに表示されます。

- 「Status Details」タブ
- 「Output」タブ

「Cluster Database Startup/Shutdown Results」ダイアログ・ボックスは、起動操作または停止操作中に自動的に表示されます。次の手順で表示することもできます。

1. ナビゲータ・ペインで、「Databases」フォルダを展開します。
2. クラスタ・データベースを右クリックします。
データベースのオプション・メニューが表示されます。
3. メニューから「Results」を選択します。

「Status Details」タブ

「Status Details」タブは、クラスタに対する起動または停止操作の実行中に表示され、操作の進捗に応じて動的に更新されます。

図 5-3 に、2 ノードのクラスタに対する起動操作が正常終了した場合の「Status Details」タブを示します。

図 5-3 「Status Details」タブ

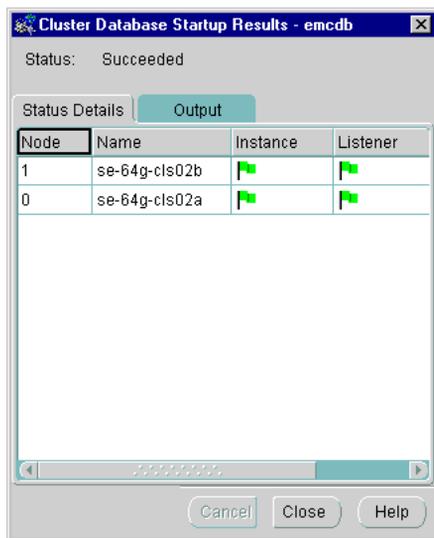


図 5-4 に、2 ノードのクラスタに対する停止操作が正常終了した場合の「Status Details」タブを示します。

図 5-4 正常終了した停止の結果

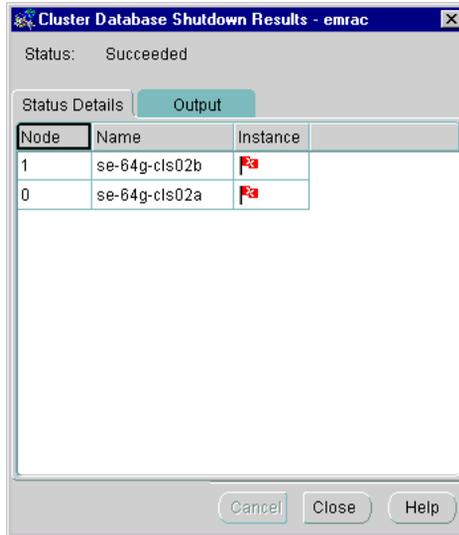


表 5-4 に、各コンポーネントの状態について示します。

表 5-4 コンポーネントの状態

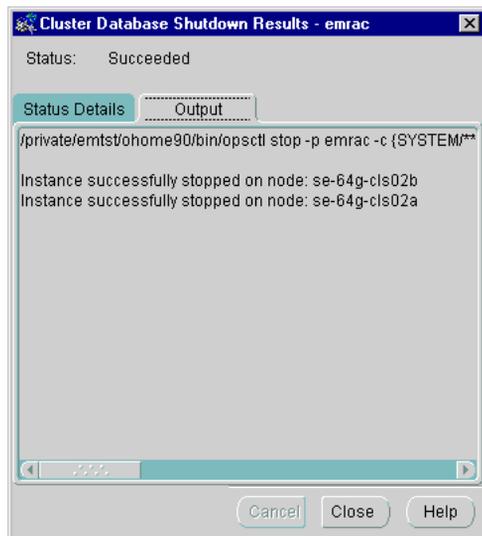
状態	説明
Up (緑色のフラグ)	コンポーネントが実行中です。
Down (赤色のフラグ)	コンポーネントが実行中ではありません。
In Progress (タイマー)	Oracle Enterprise Manager がコンポーネントの状態を判断できません。この状態は、通常、コンポーネントの起動または停止操作が完了していない場合に発生します。
Component does not exist on this node (空白のバックグラウンド)	コンポーネントがノード上に構成されていません。 各ノードにすべてのコンポーネント (リスナー、インスタンス) が必要なわけではありません。

「Output」タブ

「Output」タブには、ノードによって実行されたコマンド、および対応付けられたすべてのエラー・メッセージがテキスト形式で表示されます。

図 5-5 に、停止が正常終了した場合の「Output」タブを示します。

図 5-5 正常終了した停止の結果



Oracle Enterprise Manager では、図 5-5 に示す「Shutdown」タブに類似した「Results」タブが表示されます。

クラスタ・データベースのステータス表示

「Edit Cluster Database」ダイアログ・ボックスには、クラスタで使用可能なインスタンスおよびクラスタ・コンポーネントの状態などの、データベースに関するステータス情報が表示されます。

注意： このダイアログ・ボックスを表示するには、データベースに接続している必要があります。そのためクラスタ・データベースが停止している場合は表示されません。

データベースに関するステータス情報を表示するには、次の手順を実行します。

1. ナビゲータ・ペインで、「Databases」フォルダから *database_name* を展開します。
2. ナビゲータ・ペインで、「Databases」フォルダの下のクラスタ・データベースを右クリックします。

データベースのオプション・メニューが表示されます。

3. メニューから「View/Edit Details」を選択します。

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

処理結果は、次の2つのビューに表示されます。

- 「General」タブ
- 「Status Details」タブ

「General」タブ

「General」タブには、V\$ACTIVE_INSTANCES 表に問い合わせることによって、図 5-6 のように現在実行中のインスタンスに関する情報が表示されます。Oracle Enterprise Manager は、クラスタ・データベースに接続します。そのため、データベースが停止している場合、このタブは表示されません。

図 5-6 「General」タブ



表 5-5 に、「General」タブのフィールドについて示します。

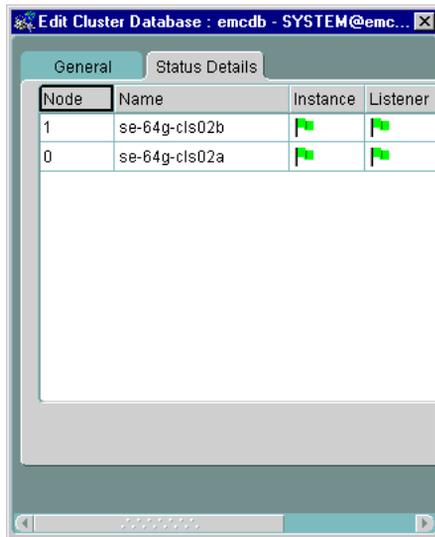
表 5-5 「General」タブのフィールド

列名	説明
Instance Number	インスタンス番号を示します。
Instance Name	インスタンスおよびそのインスタンスを実行するノードに対して指定された名前です。node:instance_name という書式の名前です。
Secondary	ノードが、プライマリ / セカンダリ・インスタンス構成のセカンダリ・インスタンスである場合は、チェックマークが表示されます。

「Status Details」タブ

「Status Details」タブには、[図 5-7](#)に示すとおり、クラスタおよび関連コンポーネントの全体的な状態が表示されます。このタブには、リスナーやインスタンスなど、様々なコンポーネントの状態が、クラスタ・データベースを構成したすべてのノードについて表示されます。

図 5-7 「Status Details」タブ



Node	Name	Instance	Listener
1	se-64g-clis02b		
0	se-64g-clis02a		

[表 5-4](#)に、各コンポーネントの状態について示します。

クラスタ・データベースまたはインスタンスに対するジョブの作成

ジョブ・スケジューリング・システムは、クラスタ・データベースとインスタンスの両方において繰り返し行われるジョブをスケジューリングおよび自動化するための、非常に確実で柔軟性のあるメカニズムを提供します。

コンソールには、カスタマイズされたスケジュールを開発するための完全な機能を搭載したスケジューリング・メカニズムが含まれています。このツールを使用すると、管理を完全に自動化できるため、ユーザーは他の作業に集中できます。

クラスタ・データベースまたはインスタンスをターゲットとするジョブを作成できます。新しいジョブを作成するには、次の手順を実行します。

1. 「Job」 → 「Create Job」 を選択します。
2. 「Create Job」 プロパティ・シートのタブに必要な項目を入力します。
3. 「Submit」 をクリックして、Oracle Intelligent Agent にジョブを送信します。「Active Jobs」 ウィンドウにジョブが表示されます。
4. ジョブを選択する場合は「Submit」を、ジョブをライブラリに追加する場合は「Add to Library」を、選択と追加の両方を行う場合は「Submit and Add」を選択します。次に、右側下部で実行操作に関するボタンをクリックします。「Job Library」 ウィンドウにジョブが表示されます。保存したジョブは後で変更または送信できます。

ジョブの詳細の指定

新しいジョブの詳細は、「Create Job」プロパティ・シートで指定できます。「Create Job」プロパティ・シートには、表 5-6 に示すタブが含まれています。

表 5-6 「Create Job」プロパティ・シートに含まれるタブ

タブ	説明
General	ジョブ名、説明、ターゲット・タイプおよび宛先を指定します。
Tasks	ジョブのタスクを選択します。
Parameters	タスクの実行時パラメータを設定します。このタブに表示されるパラメータは、「Task」リスト・ボックスで選択したタスクによって異なります。
Schedule	Oracle Enterprise Manager がジョブを実行する時間および頻度をスケジューリングします。
Access	ジョブを実行する管理者を指定します。

次のタブには、クラスタ・データベース固有のオプションが含まれています。

- 「General」タブ
- 「Tasks」タブ
- 「Parameters」タブ

関連項目： ジョブのスケジューリングの一般的な情報は、『Oracle Enterprise Manager 管理者ガイド』を参照してください。

「General」タブ

図 5-8 に示すとおり、「General」タブでは次の項目を指定できます。

- Job Name
- Target Type (Cluster Database または Cluster Database Instance)
- Description
- Targets

図 5-8 「General」 タブ

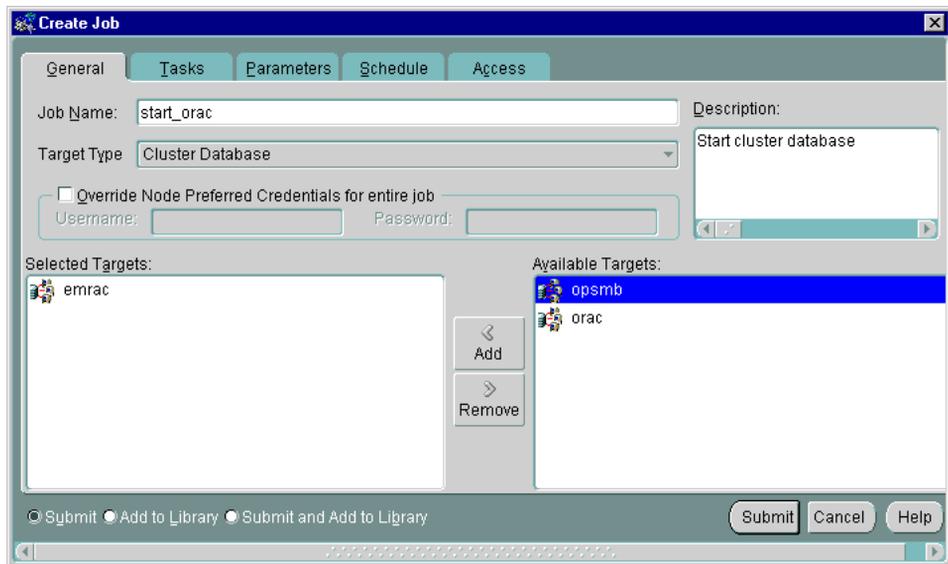


表 5-7 に、「General」 タブに表示されるオプションを示します。

表 5-7 「General」 タブのオプション

オプション	説明
Job Name	新しいジョブの名前を入力します。
Description	ジョブの説明を入力します。
Target Type	ドロップダウン・リスト・ボックスからターゲット・タイプを選択します。「Cluster Database」、「Cluster Database Instance」、「Database」、「Listener」または「Node」の中から選択できます。
Available Targets	ターゲットは「Target Type」の選択によって決まります。ターゲットには、「Cluster Database」、「Cluster Database Instance」、「Database」、「Listener」および「Node」があります。 「Available Targets」リストでジョブのターゲットをクリックします。「Add」をクリックして、ターゲットを「Selected Targets」リストに移動します。ジョブからターゲットを削除するには、「Selected Targets」リストでターゲットをクリックしてから、「Remove」をクリックします。

「Tasks」タブ

「Tasks」タブでは、ジョブに対して実行するタスクを選択します。表示されるタスクのリストは、「General」タブの「Target Type」でクラスタ・データベースを選択したか、クラスタ・データベース・インスタンスを選択したかによって異なります。「Add」ボタンと「Remove」ボタンを使用して、「Available Tasks」リストと「Job Tasks」リストの間でタスクを移動します。

クラスタ・データベースのターゲットに関するタスク

「Target Type」で「Cluster Database」を選択している場合は、クラスタ・データベース固有の次のタスクを選択できます。

- Shutdown Cluster Database
- Startup Cluster Database

関連項目： これらのタスクおよび設定するパラメータの詳細は、『Oracle Enterprise Manager 管理者ガイド』を参照してください。

「Parameters」タブ

「Parameters」タブでは、「Tasks」タブで選択したジョブ・タスクのパラメータ設定を指定できます。表示されるパラメータは、ジョブ・タスクによって異なります。クラスタ・データベースの起動タスクと停止タスクに固有のパラメータを次の項に示します。

関連項目： インスタンス・タスクに関して設定するパラメータの詳細については、『Oracle Enterprise Manager 管理者ガイド』を参照してください。

クラスタ・データベースの起動タスク

「Tasks」タブで「Startup Cluster Database」タスクを選択すると、[図 5-9](#)の画面が表示されます。

図 5-9 「Parameters」 タブ

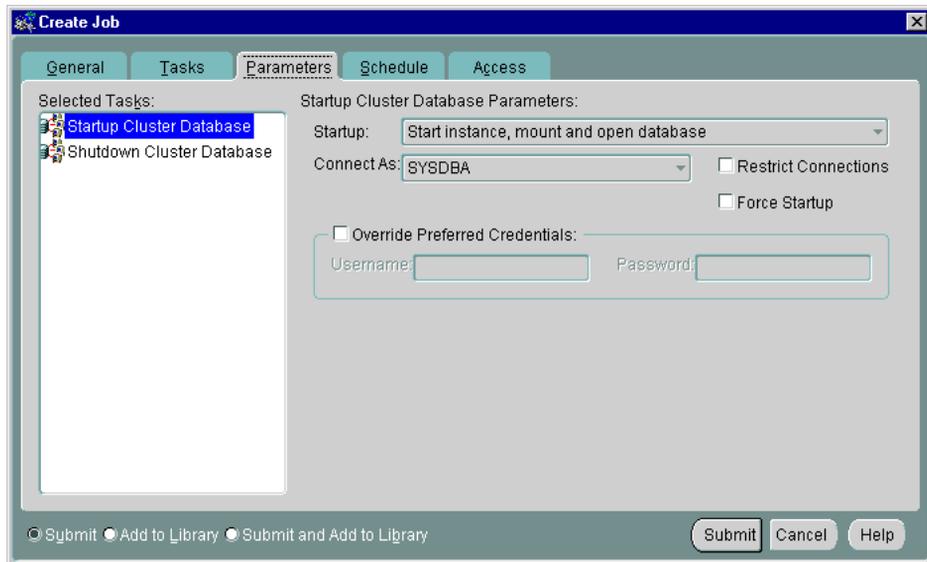


表 5-8 に、起動の「Parameters」タブに含まれるオプションを示します。

表 5-8 起動の「Parameters」タブ

パラメータ	説明
Startup	ドロップダウン・リスト・ボックスで選択したジョブに起動モードを選択します。
Connect As	<p>ロールを指定します。</p> <p>Oracle7 では「Normal」のみが使用可能です。Oracle8i 以上のリリースでは、SYSOPER ロールおよび SYSDBA ロールを使用して、最大限のデータベース管理権限を取得できます。データベースの停止や起動などのジョブ・タスクを実行するには、SYSDBA 権限または SYSOPER 権限が必要です。</p> <p>関連項目 : SYSOPER ロールおよび SYSDBA ロールの詳細については、『Oracle9i データベース管理者ガイド』を参照してください。</p> <p>SYSDBA 権限を持たないユーザーが SYSDBA 権限を使用して接続しようとする、無効なユーザー名またはパスワードが入力されたことを示すエラー・メッセージが表示されます。</p>

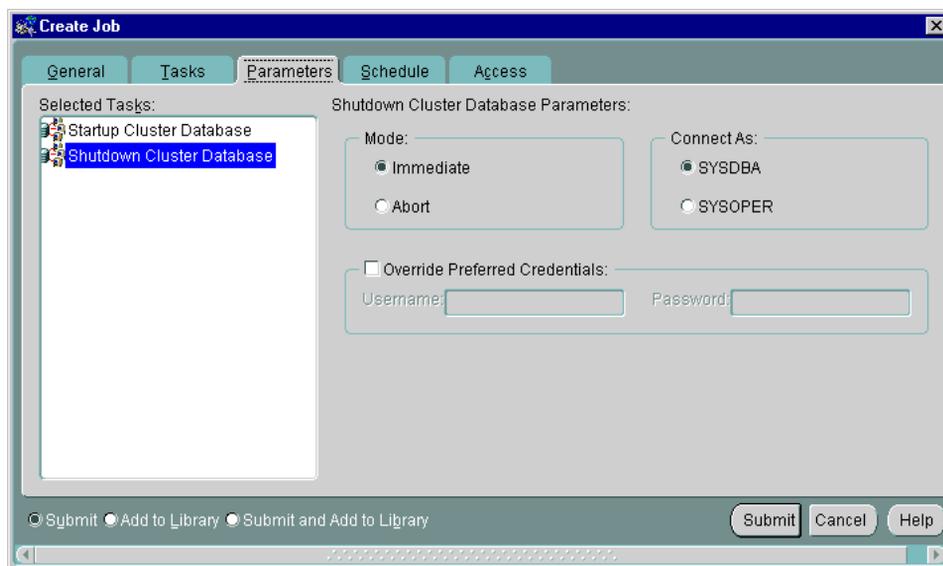
表 5-8 起動の「Parameters」タブ (続き)

パラメータ	説明
Override Preferred Credentials	データベースに設定されている優先接続情報を使用するか、または他のデータベース・ユーザー名およびパスワードを入力できます。

クラスタ・データベースの停止タスクに関するパラメータ

「Tasks」タブで「Shutdown Cluster Database」を選択すると、[図 5-10](#) に示す画面が表示されます。

図 5-10 クラスタ・データベース停止パラメータ



タブにパラメータ・エントリを入力し、「Submit」をクリックして、クラスタ・データベースの停止タスクを実行します。表 5-9 に、停止の「Parameters」タブに含まれるオプションを示します。

表 5-9 停止の「Parameters」タブ

オプション	説明
Mode	「Immediate」(デフォルト) または 「Abort」 を選択します。
Connect As	「SYSDBA」(デフォルト) または 「SYSOPER」 を選択します。

表 5-9 停止の「Parameters」タブ (続き)

オプション	説明
Override Preferred Credentials	データベースに設定されている優先接続情報または他のデータベース・ユーザー名およびパスワードを使用できます。

クラスタ・データベース・イベントの登録

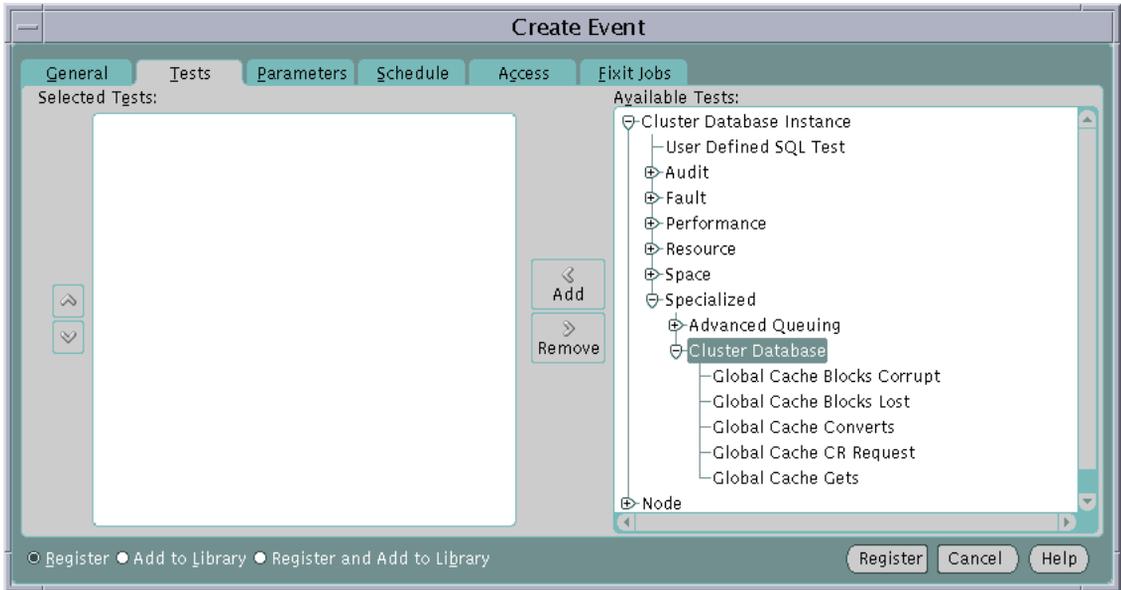
Oracle Enterprise Manager には、クラスタ・データベース・インスタンスのテストを監視できる Event Management 機能が含まれています。クラスタ・データベース・インスタンスでは、シングル・インスタンス・データベースで使用可能なすべてのテストを実行できます。これらの標準のイベント・テストに加え、クラスタ・データベース・インスタンスのみで使用可能な特別なイベント・テストがあります。これらのテストには、グローバル・キャッシュ変換、読み込み一貫性要求などがあります。クラスタ・ノードで実行する Intelligent Agent は、イベント・テストに関するデータを収集し、特定のイベント条件の発生を検出します。

Event Management 機能にアクセスするには、「Event」メニューで「Create Event」を選択します。「Create Event」ウィンドウが表示されます。「Create Event」ウィンドウで次のサブページを使用し、ターゲットの選択、パラメータの設定および通知プリファレンスの設定などを実行できます。

- 「General」 ページを使用して、クラスタ・データベースおよびクラスタ・データベースのインスタンスのターゲット・タイプを選択します。
- 「Test」 ページを使用して、スケジューリングするクラスタ・データベース固有のテストのタイプを選択します。
- 「Parameters」 ページを使用して、テスト・イベントのパラメータを選択します。
- 「Schedule」 ページを使用して、テスト・イベントの実行をスケジューリングします。
- 「Access」 ページを使用して、通知オプションを設定します。
- 「Fixit Jobs」 ページを使用して、イベント発生直後に実行するジョブを指定します。

図 5-11 に、クラスタ・データベース固有のイベント・テストの選択リストを示します。

図 5-11 Event Management の「Tests」メニュー



第 III 部

Real Application Clusters でのバックアップ およびリカバリ

第 III 部では、Real Application Clusters 環境での Recovery Manager (RMAN) の使用方法について説明します。Real Application Clusters 環境でのバックアップとリカバリの実行方法についても説明します。第 III 部に含まれる章は、次のとおりです。

- [第 6 章「Real Application Clusters 環境での Recovery Manager の使用」](#)
- [第 7 章「Real Application Clusters 環境でのバックアップとリカバリの実行」](#)

Real Application Clusters 環境での Recovery Manager の使用

この章では、Real Application Clusters 環境での **Recovery Manager (RMAN)** の使用方法について説明します。内容は次のとおりです。

- Real Application Clusters の Recovery Manager の構成 : 概要
- Recovery Manager のスナップショット制御ファイルの位置の構成
- Recovery Manager 制御ファイルの自動バックアップ機能の構成
- Real Application Clusters 環境でのアーカイブ・ログの管理
- Recovery Manager のアーカイブ構成スキーム
- Real Application Clusters でのアーカイブ・モードの変更
- アーカイバ・プロセスの問合せ

Real Application Clusters の Recovery Manager の構成 : 概要

この章に記載されている手順は、**Oracle Enterprise Edition** および **Oracle Real Application Clusters** をインストールし、次のマニュアルを参照した後で、実行してください。

- 『Oracle9i Recovery Manager ユーザーズ・ガイド』
- 『Oracle9i Recovery Manager リファレンス』

関連項目： バックアップとリカバリ用に **Oracle Enterprise Manager** を使用する方法については、『Oracle Enterprise Manager 管理者ガイド』を参照してください。

この章では、次の構成手順について説明します。

1. スナップショット制御ファイルの位置の構成
2. 制御ファイルの自動バックアップ機能の構成
3. アーカイブ・スキームの構成
4. データベースのアーカイブ・モードの変更（オプション）
5. アーカイバ・プロセスの監視

Recovery Manager のスナップショット制御ファイルの位置の構成

Real Application Clusters では、**スナップショット制御ファイル**と同一のコピーがバックアップを実行するすべてのノードに存在する必要があります。そのため、バックアップに使用するすべてのノード上に、スナップショット制御ファイルの同じ接続先ディレクトリがあることを確認してください。

Recovery Manager は、スナップショット制御ファイルを作成し、ターゲット制御ファイルの内容に関する一貫したレコードを取得した後、バックアップを実行します。Recovery Manager は、バックアップを実行するノード上にスナップショット制御ファイルを作成します。次の Recovery Manager のコマンドを実行して、スナップショット制御ファイルのデフォルトの位置を決定します。

```
SHOW SNAPSHOT CONTROLFILE NAME;
```

必要な場合は、デフォルトの位置を変更できます。たとえば、UNIX の場合は、Recovery Manager のプロンプトで次のコマンドを入力し、スナップショット制御ファイルの位置を \$ORACLE_HOME/dbs/scf/snap_prod.cf に指定できます。

```
CONFIGURE SNAPSHOT CONTROLFILE NAME TO '?/dbs/scf/snap_prod.cf';
```

このコマンドは、クラスタ・データベース全体にわたるスナップショット制御ファイルの位置構成をグローバルに設定します。したがって、バックアップを実行するすべてのノードに \$ORACLE_HOME/dbs/scf ディレクトリがあることを確認してください。

CONFIGURE コマンドを使用すると、複数の Recovery Manager セッションに影響を与える永続的な設定を作成できます。したがって、スナップショット制御ファイルの位置を変更しないかぎり、再度このコマンドを実行する必要はありません。

注意： クラスタ・ファイル・システム (CFS) または RAW デバイス接続先をスナップショット制御ファイルの位置に指定することもできます。このファイルは、Real Application Clusters の他のデータ・ファイルと同じように、クラスタ内のすべてのノードで共有されます。

関連項目： スナップショット制御ファイルの構成方法については、『Oracle9i Recovery Manager ユーザーズ・ガイド』を参照してください。

Recovery Manager 制御ファイルの自動バックアップ機能の構成

Recovery Manager 制御ファイルの自動バックアップ機能によって、BACKUP コマンドと COPY コマンドの実行後に、制御ファイルのバックアップが自動的に作成されます。この機能は、リカバリ・カタログがない場合でも Recovery Manager で制御ファイルをリストアできるため、障害時リカバリでは重要です。

CONFIGURE CONTROLFILE AUTOBACKUP を ON に設定すると、BACKUP コマンドまたは COPY コマンドの実行後に、制御ファイルとサーバー・パラメータ・ファイルのバックアップが Recovery Manager によって自動的に作成されます。リカバリ・カタログと現行の制御ファイルの両方が失われた場合でも、Recovery Manager によって、自動バックアップされた制御ファイルをリストアできます。

CONFIGURE CONTROLFILE AUTOBACKUP FORMAT コマンドを使用すると、Recovery Manager によって指定されたこのファイルのデフォルト名を変更できます。このコマンドで絶対パス名を指定する場合、そのパスはバックアップを実行するすべてのノードに存在している必要があります。

Recovery Manager は、最初に割り当てられたチャンネルで制御ファイルの自動バックアップを実行します。したがって、異なるパラメータを使用して複数のチャンネルを割り当てる場合、特に CONNECT コマンドを使用してチャンネルを割り当てる場合は、どのチャンネルで制御ファイルの自動バックアップを実行するかを決定する必要があります。そのノードに、常に最初にチャンネルを割り当ててください。

関連項目： 制御ファイルの自動バックアップ機能の使用方法については、『Oracle9i Recovery Manager ユーザーズ・ガイド』を参照してください。

Real Application Clusters 環境でのアーカイブ・ログの管理

スナップショット制御ファイルの位置を構成し、制御ファイルの自動バックアップ機能を使用可能にした後は、各ノードが生成するアーカイブ・ログを管理する環境の構成方法を決定します。

クラスタ・ノードでアーカイブ・ログが生成されると、そのログのファイル名は、ターゲット・データベースの制御ファイルに必ず記録されます。リカバリ・カタログを使用している場合、アーカイブ・ログのファイル名は、再同期化の実行時にリカバリ・カタログにも記録されます。

使用するアーカイブ・ログのネーミング計画は重要です。これは、あるノードが、特定のファイル名でログをファイル・システムに書き込んだ場合、そのファイルは、このログにアクセスするすべてのノードで読取り可能である必要があるためです。たとえば、ノード1がログを `/oracle/arc_dest/log_1_100.arc` にアーカイブした場合は、ノード2のファイル・システムで `/oracle/arc_dest/log_1_100.arc` の読取りが可能な場合のみ、ノード2はこのログをバックアップできます。

選択するバックアップとリカバリの計画は、各ノードのアーカイブ先を構成する方法によって異なります。アーカイブ・ログのバックアップを1つのノードのみで実行するか、全ノードで実行するかに関係なく、すべてのログがバックアップされることを確認する必要があります。リカバリの実行は1つのインスタンスのみでも可能であるため、リカバリを行うノードにはクラスタ内のすべてのログに対する読取りアクセス権が必要です。

Real Application Clusters 環境でのアーカイブ

ここでの主要な構成タスクは、リカバリ時に（可能な場合はバックアップ時にも）、各ノードからすべてのアーカイブ・ログが読み取れるようにすることです。次の使用例は、クラスタ・データベースにアーカイブを構成する際のアーカイブ・ログのネーミングに関する注意点を示しています。

この使用例では、次の条件が満たされていることが前提です。

- 各ノードが、各ノードの同じ名前前のローカル・アーカイブ・ディレクトリに書き込むように構成されている。
- クラスタ・ファイル・システムを設定していない（つまり、各ノードはそれ自体のローカル・ファイル・システムとの間のみで読取り / 書込みを実行できる）。
- クラスタ内のノードを使用可能にしてノード間相互の読取り / 書込みアクセス権を取得するために、NFS またはマップされたドライブを使用していない。

初期化パラメータ・ファイルは、次のように構成します。

```
sid1.log_archive_dest_1 = (location=/arc_dest)
sid2.log_archive_dest_1 = (location=/arc_dest)
sid3.log_archive_dest_1 = (location=/arc_dest)
```

次のアーカイブ・ログのファイル名が、制御ファイルに記録されていると仮定します。

```
/arc_dest/log_1_62.arc  
/arc_dest/log_2_100.arc  
/arc_dest/log_2_101.arc  
/arc_dest/log_3_70.arc  
/arc_dest/log_1_63.arc
```

この使用例では、Real Application Clusters データベースでリカバリを実行すると仮定します。ノード1 がリカバリの際にログを読み取る場合は、ノード1 のローカルの /arc_dest ディレクトリで、制御ファイルに記録されているファイル名を検索します。このデータを使用してノード1 が検出するログは、ローカルでアーカイブしたログのみです (/arc_dest/log_1_62.arc や /arc_dest/log_1_63.arc など)。ノード1 ではその他のログを適用することはできません。これは、その他のログのファイル名はノード1 のローカル・ファイル・システムでは読み取ることができないためです。したがって、リカバリは行き詰ることになります。この問題は、次の項で説明するスキームに従ってクラスタを構成することで回避できます。

Recovery Manager のアーカイブ構成スキーム

この章の2つの構成スキームでは、Real Application Clusters データベース用の3ノードのUNIXシステムについて説明します。いずれの構成スキームでも、リカバリを実行するインスタンスに指定する LOG_ARCHIVE_FORMAT は、ファイルをアーカイブするインスタンスに指定したフォーマットと同じである必要があります。

Real Application Clusters では、すべてのインスタンスに対して LOG_ARCHIVE_FORMAT パラメータを同じ値に設定します。アーカイブ REDO ログ・ファイルが元のアーカイブ先にならない場合は、リカバリ時に LOG_ARCHIVE_DEST_n に異なる値を指定できます。

この項の内容は次のとおりです。

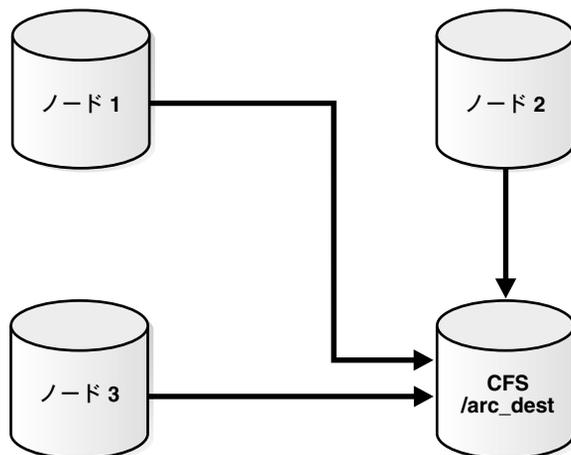
- クラスタ・ファイル・システムのアーカイブ・スキーム
- 非CFSのローカル・アーカイブ・スキーム
- Real Application Clusters のアーカイブ・ログ・ファイルのフォーマットと接続先

クラスタ・ファイル・システムのアーカイブ・スキーム

Real Application Clusters のデフォルト構成では、クラスタ・ファイル・システムのアーカイブ・スキームを使用します。図6-1に示すように、各ノードは、単一のCFSのアーカイブ・ログ宛先へ書き込み、他のノードのアーカイブ・ログ・ファイルを読み取ることができます。読取りアクセスは、CFSを使用して行われます。たとえば、ノード1がログをCFSの /arc_dest/log_1_100.arc にアーカイブした場合、クラスタ内の他のすべてのノードもこのファイルを読み取ることができます。

注意： UNIX 環境では、クラスタ・ファイル・システムを使用しない場合、アーカイブ・ログ・ファイルを共有ディスク上に置くことができません。これは、UNIX の共有ディスクは、アーカイブ・ログを使用するために簡単にパーティション化できない RAW デバイスであるためです。

図 6-1 クラスタ・ファイル・システムのアーカイブ・スキーム



クラスタ・ファイル・システムのアーカイブ・スキームのメリット

このスキームには、次のメリットがあります。

- いずれのノードもログのアーカイブでネットワークを使用しません。
- 各ノードにローカル・テープ・ドライブがある場合は、アーカイブ・ログのバックアップを分散できます。これによって、各ノードはネットワークにアクセスせずにローカルのログをバックアップできます。
- あるノードが書き込んだファイル名はクラスタ内の任意のノードで読み取ることができるため、Recovery Manager は、クラスタ内の任意のノードからすべてのログをバックアップできます。
- 各ノードには、すべてのアーカイブ・ログに対するアクセス権があるため、バックアップとリストアのスクリプトが簡素化されます。

クラスタ・ファイル・システムのアーカイブ・スキームに関する初期化パラメータの設定

CFS のスキームでの各ノードは、クラスタ・データベース内のすべてのインスタンスで同じ名前を使用して識別されるディレクトリにアーカイブします。これを構成するには、次の例のように `sid` 指定子を使用して各インスタンスの `LOG_ARCH_DEST_n` パラメータに値を設定します。

```
sid1.LOG_ARCHIVE_DEST_1="LOCATION=/arc_dest"  
sid2.LOG_ARCHIVE_DEST_1="LOCATION=/arc_dest"  
sid3.LOG_ARCHIVE_DEST_1="LOCATION=/arc_dest"
```

次のリストは、前述の例に基づいて、Recovery Manager のカタログまたは制御ファイルに表示されるアーカイブ・ログ・エントリの例を示しています。すべてのノードが任意のストレッドを使用してログをアーカイブできることに注意してください。

```
/arc_dest/arc_1_999.arc  
/arc_dest/arc_1_1000.arc  
/arc_dest/arc_1_1001.arc <- thread 1 archived in node 3  
/arc_dest/arc_3_1563.arc <- thread 3 archived in node 2  
/arc_dest/arc_2_753.arc <- thread 2 archived in node 1  
/arc_dest/arc_2_754.arc  
/arc_dest/arc_3_1564.arc
```

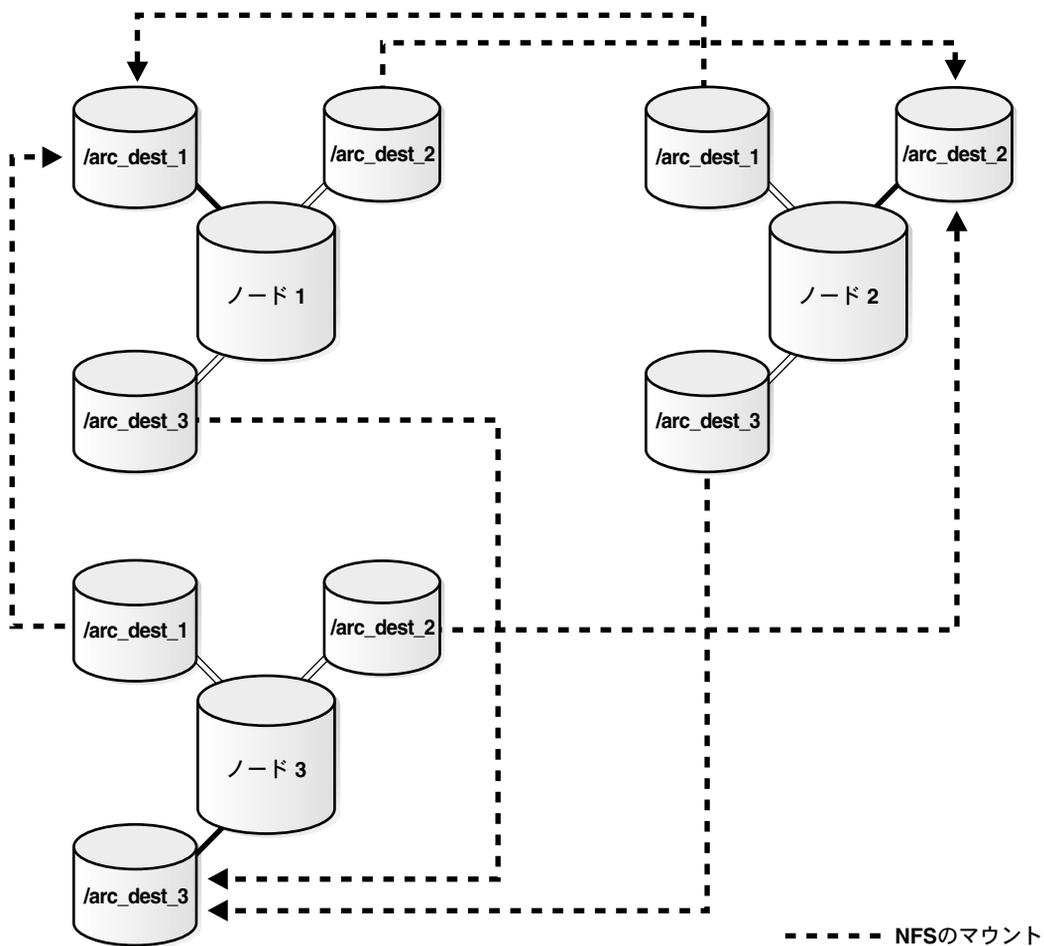
クラスタ・ファイル・システムのアーカイブ・スキームでのアーカイブ・ログの位置

ファイル・システムは共有化されており、各ノードはそのノード自体のアーカイブ・ログをクラスタ・ファイル・システムの `/arc_dest` ディレクトリに書き込むため、各ノードはそのノード自体、および他のノードが書き込んだログを読み取ることができます。

非 CFS のローカル・アーカイブ・スキーム

非 CFS のローカル・アーカイブ・スキームでの各ノードは、[図 6-2](#) のように、一意の名前のローカル・ディレクトリにアーカイブします。リカバリが必要な場合は、他のノードのディレクトリにリモートでアクセスできるように、リカバリ・ノードを構成できます。たとえば、UNIX プラットフォームの NFS または Windows プラットフォームの共有ドライブを使用します。したがって、各ノードはローカルの宛先にのみ書き込みますが、他のノード上のリモート・ディレクトリにあるアーカイブ・ログ・ファイルを読み取ることもできます。

図 6-2 非 CFS のローカル・アーカイブ・スキーム



Recovery Manager で Real Application Clusters データベースを 1 ステップでバックアップおよびリカバリできるようにするには、すべてのアーカイブ・ログに、クラスタ全体で一意に識別できる名前を指定する必要があります。ただし、このために 6-5 ページの「[クラスタ・ファイル・システムのアーカイブ・スキーム](#)」に記載されている技法を使用して、/arc_dest などの 1 つのディレクトリに複数のノードのアーカイブを指定することはできません。

注意： UNIX 環境では、アーカイブ・ログ・ファイルを共有ディスク上に置くことができません。これは、UNIX の共有ディスクは、アーカイブ・ログを使用するために簡単にパーティション化できない RAW デバイスであるためです。

非 CFS のアーカイブ・スキームのメリットとデメリット

このスキームには、次のメリットがあります。

- 各ノードにローカル・テープ・ドライブがある場合は、アーカイブ・ログのバックアップを分散できます。これによって、各ノードはネットワークにアクセスせずにローカルのログをバックアップできます。

このスキームには、次のデメリットがあります。

- メディア・リカバリの場合は、他のノード上のアーカイブ・ディレクトリにあるアーカイブ・ログ・ファイルを読み取ることができるように、リモート・アクセスのリカバリを実行するノードを構成する必要があります。
- 1つのノードにのみローカル・テープ・ドライブがある場合は、NFS を構成するか、または手動でログを転送しないかぎり、単一のノードですべてのログをバックアップすることはできません。
- このスキームには、シングル・ポイント障害が存在します。最後のバックアップ以降に1つのノードに障害が発生した場合、バックアップ後に生成されたこのノード上のアーカイブ・ログは消失します。リカバリを実行する際に、使用可能なすべてのログがない場合は、Point-in-Time リカバリを最初のアーカイブ・ログの順序番号が欠落している地点まで実行することを強制されます。

注意： このスキームのために特定の構成を使用する必要はありません。ただし、バックアップ処理を複数のノードに分散する最も簡単な方法は、[第7章「Real Application Clusters 環境でのバックアップとリカバリの実行」](#)のバックアップ使用例に記載されているように、チャンネルを構成することです。

非 CFS のローカル・アーカイブ・スキームに関する初期化パラメータの設定

アーカイブ先の値は、次のように[初期化パラメータ・ファイル](#)に設定できます。

```
sid1.LOG_ARCHIVE_DEST_1="LOCATION=/arc_dest_1"  
sid2.LOG_ARCHIVE_DEST_1="LOCATION=/arc_dest_2"  
sid3.LOG_ARCHIVE_DEST_1="LOCATION=/arc_dest_3"
```

次のリストは、データベース制御ファイルのアーカイブ・ログ・エントリを示しています。すべてのノードが任意のスレッドからログをアーカイブできることに注意してください。

```

/arc_dest_1/arc_1_1000.arc
/arc_dest_2/arc_1_1001.arc <- thread 1 archived in node 2
/arc_dest_2/arc_3_1563.arc <- thread 3 archived in node 2
/arc_dest_1/arc_2_753.arc <- thread 2 archived in node 1
/arc_dest_2/arc_2_754.arc
/arc_dest_3/arc_3_1564.arc
    
```

非 CFS のローカル・アーカイブ・スキームでのアーカイブ・ログの位置

表 6-1 に示すように、各ノードにはローカルのアーカイブ・ログを含むディレクトリがあります。また、NFS または共有ドライブを介して他のノード上のディレクトリをリモートでマウントしている場合、各ノードには、クラスタ内の他のノードがアーカイブしたアーカイブ・ログ・ファイルを Recovery Manager で読み取ることができる 2 つのリモート・ディレクトリがあります。

表 6-1 非 CFS のローカル・アーカイブ・スキームでのログの位置

ノード	アーカイブ・ログ・ファイルを読み取り可能なディレクトリ	ノードがアーカイブするログ
1	/arc_dest_1	1
1	/arc_dest_2	2 (NFS を介して)
1	/arc_dest_3	3 (NFS を介して)
2	/arc_dest_1	1 (NFS を介して)
2	/arc_dest_2	2
2	/arc_dest_3	3 (NFS を介して)
3	/arc_dest_1	1 (NFS を介して)
3	/arc_dest_2	2 (NFS を介して)
3	/arc_dest_3	3

非 CFS のローカル・アーカイブ・スキームに関するファイル・システムの構成

NFS はバックアップの実行には不要なため、ノード 1 はローカルのログをそれ自体のテープ・ドライブにバックアップし、ノード 2 はローカルのログをそれ自体のテープ・ドライブにバックアップする、ということが可能です。ただし、リカバリを実行していて、障害が発生しなかったインスタンスが、まだバックアップされていないディスク上のログをすべて読み取る必要がある場合は、表 6-2 に示すように NFS を構成する必要があります。

表 6-2 共有読取りローカル・アーカイブ・スキームに関する NFS の構成

ノード	ディレクトリ	構成	マウント先	ノード
1	/arc_dest_1	ローカルの読取り / 書込み	該当なし	該当なし
1	/arc_dest_2	NFS 読取り	/arc_dest_2	2
1	/arc_dest_3	NFS 読取り	/arc_dest_3	3
2	/arc_dest_1	NFS 読取り	/arc_dest_1	1
2	/arc_dest_2	ローカルの読取り / 書込み	該当なし	該当なし
2	/arc_dest_3	NFS 読取り	/arc_dest_3	3
3	/arc_dest_1	NFS 読取り	/arc_dest_1	1
3	/arc_dest_2	NFS 読取り	/arc_dest_2	2
3	/arc_dest_3	ローカルの読取り / 書込み	該当なし	該当なし

Real Application Clusters のアーカイブ・ログ・ファイルのフォーマットと接続先

アーカイブ・ログの構成では、LOG_ARCHIVE_FORMAT パラメータでアーカイブ REDO ログを一意に識別します。このパラメータのフォーマットは、オペレーティング・システム固有で、テキスト文字列、1つ以上の変数およびファイル名拡張子を指定できます。表 6-3 に示すように、LOG_ARCHIVE_FORMAT には、変数を指定することもできます。表 6-3 は、LOG_ARCHIVE_FORMAT=log_%parameter の例で、順序番号の値の長さ制限は 10 文字、スレッド値は 4 文字と仮定しています。

表 6-3 アーカイブ REDO ログ・ファイル名のフォーマット・パラメータ

パラメータ	説明	例
%T	左端に 0 が埋め込まれたスレッド番号	log_0001
%t	埋込みなしのスレッド番号	log_1
%S	左端に 0 が埋め込まれたログ順序番号	log_0000000251
%s	埋込みなしのログ順序番号	log_251

スレッド・パラメータの %t または %T は、Real Application Clusters では必須です。たとえば、REDO スレッド番号 1 に対応付けられたインスタンスによって、LOG_ARCHIVE_FORMAT が log_%t_%s.arc に設定されると、そのアーカイブ REDO ログ・ファイル名は次のようになります。

```
log_1_1000.arc
log_1_1001.arc
log_1_1002.arc
.
.
.
```

関連項目：

- アーカイブ REDO ログ・ファイル名のフォーマットとアーカイブ先の指定の詳細については、『Oracle9i データベース管理者ガイド』を参照してください。
- デフォルト・ログ・アーカイブのフォーマットとアーカイブ先の詳細については、プラットフォーム固有の Oracle マニュアルを参照してください。

Real Application Clusters でのアーカイブ・モードの変更

Recovery Manager に Real Application Clusters 環境を構成した後は、必要に応じてアーカイブ・モードを変更できます。たとえば、Real Application Clusters データベースで NOARCHIVELOG モードを使用している場合は、次の手順に従ってアーカイブ・モードを ARCHIVELOG モードに変更します。

1. すべてのインスタンスを停止します。
2. 1 つのインスタンス上で、CLUSTER_DATABASE パラメータを false にリセットします。サーバー・パラメータ・ファイルを使用している場合は、sid 固有のエントリを作成します。
3. パラメータ・ファイルに LOG_ARCHIVE_DEST_n、LOG_ARCHIVE_FORMAT および LOG_ARCHIVE_START の各パラメータの設定を追加します。使用するアーカイブ先は 10 箇所まで設定できます。LOG_ARCHIVE_FORMAT パラメータには %t パラメータを指定して、アーカイブ・ログ・ファイル名にスレッド番号を含める必要があります。アーカイブ・スキームの構成は、これらのパラメータ値の設定前に行う必要があります。
4. CLUSTER_DATABASE を false に設定したインスタンスを起動します。
5. SQL*Plus で次の文を実行します。

```
SQL>ALTER DATABASE ARCHIVELOG;
```
6. インスタンスを停止します。
7. CLUSTER_DATABASE パラメータの値を true に変更します。

8. インスタンスを再起動します。

アーカイブ・モードを ARCHIVELOG から NOARCHIVELOG に変更することもできます。アーカイブを使用禁止にするには、次の変更点に注意して前述の手順に従います。

1. 手順 3 で、作成したアーカイブ設定を削除します。
2. 手順 5 で、次のように NOARCHIVELOG キーワードを指定します。

```
ALTER DATABASE NOARCHIVELOG;
```

アーカイバ・プロセスの問合せ

Recovery Manager 構成が Real Application Clusters 環境で操作可能になった後、GV\$ARCHIVE_PROCESSES ビューと V\$ARCHIVE_PROCESSES ビューを使用してアーカイバ・プロセスのステータスを判断します。これらのビューには、問合せ対象がグローバル・ビューかローカル・ビューのいずれであるかに従って、Real Application Clusters データベース内のすべてのデータベース・インスタンスに関する情報、あるいは接続先のインスタンスのみに関する情報がそれぞれ表示されます。

関連項目： これらのビューの詳細については、『Oracle9i データベース・リファレンス』を参照してください。

Real Application Clusters 環境での バックアップとリカバリの実行

この章では、Real Application Clusters でのバックアップとリカバリの実行方法について説明します。Real Application Clusters 環境でのバックアップとリカバリ処理に **Recovery Manager (RMAN)** を使用するための情報も含まれています。内容は次のとおりです。

- Real Application Clusters でのバックアップの概要
- Real Application Clusters の Recovery Manager バックアップ・スキーム
- Real Application Clusters でのインスタンス・リカバリ
- Real Application Clusters でのメディア・リカバリの概要
- Real Application Clusters の Recovery Manager リストア・スキーム
- Real Application Clusters でのパラレル・リカバリ

Real Application Clusters でのバックアップの概要

Real Application Clusters 環境での Recovery Manager のバックアップ実行手順は、シングル・インスタンス環境での Recovery Manager のバックアップ手順とほぼ同じです。ただし、次の各項では、Real Application Clusters 固有の注意点について説明します。

- Real Application Clusters 構成のターゲット・データベースへの接続
- Real Application Clusters での Recovery Manager と Oracle Net
- クラスタ・インスタンスへのチャンネル接続
- ノード・アフィニティの認識
- ファイルのアクセス可能性とバックアップ・メディア

関連項目： シングル・インスタンスの Recovery Manager バックアップ手順の詳細については、『Oracle9i Recovery Manager ユーザーズ・ガイド』を参照してください。

Real Application Clusters 構成のターゲット・データベースへの接続

Recovery Manager を起動してターゲット・データベースに接続する場合、Recovery Manager が 1 度に接続できるのは、Real Application Clusters データベース内の 1 インスタンスのみです。この接続は、バックアップやリストアを実行しないユーティリティ接続であること、および Recovery Manager のコマンドラインから行われた接続に対してのみ適用されることに注意してください。

たとえば、Real Application Clusters 構成内の 3 つのインスタンスのネット・サービス名を、node1、node2 および node3 と仮定します。この場合、これらのネット・サービス名のうち 1 つのみを使用してターゲット・データベースに接続します。たとえば、次のコマンドを入力すると接続できます。

```
% rman TARGET SYS/oracle@node2 CATALOG rman/cat@catdb
```

Real Application Clusters での Recovery Manager と Oracle Net

ネット・サービス名を介して確立された Recovery Manager の接続では、各ネット・サービス名で、ただ 1 つのインスタンスのみが指定されている必要があります。このルールは、接続がコマンドラインから行われたか、ALLOCATE CHANNEL コマンドまたは CONFIGURE CHANNEL コマンドの CONNECT 句を介して行われたかに関係なく、すべての Recovery Manager の接続に適用されます。したがって、ネット・サービス名が、Oracle Net 機能を使用して Recovery Manager の接続を複数のインスタンスに割り当てている場合は、そのネット・サービス名を指定することはできません。

クラスタ・インスタンスへのチャンネル接続

Real Application Clusters 構成でバックアップを行う場合、割り当てられている各チャンネルは、クラスタ内の異なるインスタンスに接続できます。また、各チャンネル接続は1つのインスタンスにのみ接続する必要があります。たとえば、自動チャンネルを構成するには、次の構文を入力します。

```
CONFIGURE DEFAULT DEVICE TYPE TO sbt;  
CONFIGURE DEVICE TYPE sbt PARALLELISM 3;  
CONFIGURE CHANNEL 1 DEVICE TYPE sbt CONNECT = 'SYS/oracle@node1';  
CONFIGURE CHANNEL 2 DEVICE TYPE sbt CONNECT = 'SYS/oracle@node2';  
CONFIGURE CHANNEL 3 DEVICE TYPE sbt CONNECT = 'SYS/oracle@node3';
```

チャンネルの1つが接続しているインスタンスがデータベースをオープンしない場合、そのデータベースは、どのインスタンスによってもオープンされません。つまり、すべてのチャンネルは、オープン状態のインスタンスか、またはオープンされていないインスタンスに接続されている必要があります。たとえば、ノード2とノード3のインスタンスにはオープン状態のデータベースがあるが、ノード1のインスタンスにマウントされたデータベースがある場合は、バックアップに失敗します。

ノード・アフィニティの認識

一部の**クラスタ・データベース**構成では、クラスタの一部のノードは、他のデータ・ファイルに対するアクセスよりも高速に一部のデータ・ファイルにアクセスします。**Recovery Manager**は、このアフィニティを自動的に検出します。これは、ノード・アフィニティの認識と呼ばれています。

特定のデータ・ファイルのバックアップに使用するチャンネルを決定する場合、**Recovery Manager**は、バックアップするデータ・ファイルに対してアフィニティがあるノードで割り当てられたチャンネルを優先します。たとえば、3ノードのクラスタがあり、ノード1はデータ・ファイル7、8および9に対して他のノードよりも高速に読取り / 書込みアクセスを行う場合、ノード1は、ノード2および3に比べて、これらのファイルに対するノード・アフィニティが高いと言えます。

ノード・アフィニティを使用するには、バックアップするデータ・ファイルに対するアフィニティを持つクラスタのノードで、**Recovery Manager**のチャンネルを構成します。たとえば、次の構文を使用します。

```
CONFIGURE CHANNEL 1 DEVICE TYPE sbt CONNECT 'user1/password1@node1';  
CONFIGURE CHANNEL 2 DEVICE TYPE sbt CONNECT 'user2/password2@node2';  
CONFIGURE CHANNEL 3 DEVICE TYPE sbt CONNECT 'user3/password3@node3';
```

バックアップするデータ・ファイルとバックアップを行うチャンネルを指定することによって、自動ノード・アフィニティを手動でオーバーライドできます。次に例を示します。

```
BACKUP
# channel 1 gets datafile 1
(DATAFILE 1 CHANNEL ORA_SBT_TAPE_1)
# channel 2 gets datafiles 2-4
(DATAFILE 2,3,4 CHANNEL ORA_SBT_TAPE_2)
# channel 3 gets datafiles 5-10
(DATAFILE 5,6,7,8,9,10 CHANNEL ORA_SBT_TAPE_3);
```

関連項目： CONFIGURE CHANNEL コマンドの CONNECT 句の詳細は、『Oracle9i Recovery Manager ユーザーズ・ガイド』を参照してください。

ファイルのアクセス可能性とバックアップ・メディア

次の各項では、クラスタでの Recovery Manager のバックアップ実行に関するその他の重要な考慮事項について説明します。

- [バックアップするファイルの可読性](#)
- [バックアップ完了後のログの削除](#)
- [バックアップ・メディアの分散](#)

バックアップするファイルの可読性

バックアップを実行するノードでは、BACKUP コマンドに指定されているすべてのファイルの読取りが可能である必要があります。たとえば、3 ノード・クラスタのノード 1 で、次のコマンドを実行すると仮定します。

```
BACKUP DATABASE PLUS ARCHIVELOG;
```

この場合、Recovery Manager は、すべてのデータ・ファイルとアーカイブ・ログをバックアップしようとします。データ・ファイルは、クラスタ・ファイル・システムのファイルまたは共有ディスクのファイルのいずれかであるため、Recovery Manager はこれらのデータ・ファイルを読み取ることができます。ただし、ローカル・ノードが読み取れないログをバックアップすることはできません。バックアップを実行するノードがすべてのログにアクセスできる環境の構成方法については、6-5 ページの「[Recovery Manager のアーカイブ構成スキーム](#)」のアーカイブ・スキームを参照してください。

バックアップ完了後のログの削除

BACKUP コマンドは、バックアップ終了後にディスクからアーカイブ・ログを削除する必要があります。たとえば、DELETE INPUT 句は、バックアップした特定のログのみを削除するよう指定します。反対に、DELETE ALL INPUT 句は、ログのバックアップ時と同じスレッドと順序番号を待つログをすべて削除するよう指定します。6-7 ページに説明されている「[非](#)

CFS のローカル・アーカイブ・スキームを使用している場合は、DELETE INPUT または DELETE ALL INPUT のいずれかを指定できます。

このスキームには、各ノードにそれ自体のローカル・アーカイブ・ディレクトリに対する読取り / 書込みアクセス権がある場合、および各ノードに他のノードに対する読取りアクセス権 (NFS が未設定の場合) や読取り専用アクセス権 (リモート・ディレクトリに NFS が読取りでマウントされている場合) がない場合のアーカイブ・スキームが記述されています。このような場合の最良の選択は、BACKUP コマンドに DELETE ALL INPUT 句または DELETE INPUT 句を指定しないことです。かわりに、DELETE コマンドを使用します。

次のスクリプトは、バックアップ後に各ノードからアーカイブ・ログを削除する方法の一例です。

```
ALLOCATE CHANNEL FOR MAINTENANCE DEVICE TYPE DISK CONNECT 'SYS/oracle@node1';
DELETE ARCHIVELOG LIKE '%arc_dest_1%'
    BACKED UP 1 TIMES TO DEVICE TYPE sbt;
RELEASE CHANNEL;
```

```
ALLOCATE CHANNEL FOR MAINTENANCE DEVICE TYPE DISK CONNECT 'SYS/oracle@node2';
DELETE ARCHIVELOG LIKE '%arc_dest_2%'
    BACKED UP 1 TIMES TO DEVICE TYPE sbt;
RELEASE CHANNEL;
```

```
ALLOCATE CHANNEL FOR MAINTENANCE DEVICE TYPE DISK CONNECT 'SYS/oracle@node3';
DELETE ARCHIVELOG LIKE '%arc_dest_3%'
    BACKED UP 1 TIMES TO DEVICE TYPE sbt;
RELEASE CHANNEL;
```

バックアップ・メディアの分散

Real Application Clusters 構成でバックアップ・メディアを構成する場合、次の3つのオプションがあります。

- **ネットワーク・バックアップ・サーバー**: いずれのノードにもローカル・ドライブがない場合。かわりに、バックアップは、ネットワークを介してバックアップ・サーバーに転送されます。
- **1つのローカル・ドライブ**: 1つのノードのみがローカル・テープ・ドライブにアクセスできる場合。
- **複数のドライブ**: 各クラスター・ノードがローカル・テープ・ドライブにアクセスできる場合。

1つのノードのみにテープ・ドライブが連結されている場合、このノードでは、すべてのデータ・ファイルとアーカイブ・ログの読取りが可能であることが必要です。クラスターの1つのノードを使用してすべてのファイルをバックアップする例は、6-5 ページの「[クラスター・ファイル・システムのアーカイブ・スキーム](#)」および6-7 ページの「[非 CFS のローカル・アーカイブ・スキーム](#)」を参照してください。ただし、非 CFS のスキームの場合は、1つのノードですべてのログをバックアップできるように NFS を構成する必要があります。した

がって、ローカル・ドライブが1つのみの場合に、非 CFS のアーカイブ・スキームを使用することはお薦めしません。

かわりに、各ノードがそれ自体のローカル・テープ・ドライブに書き込みできる複数のドライブ・スキームを使用できます。CFS のスキームでは、いずれのノードでも、すべてのデータ・ファイルとアーカイブ・ログをバックアップできます。非 CFS のスキームでは、バックアップが**分散**されるようにバックアップ・スクリプトを記述する必要があります。分散バックアップでは、異なるファイルが各ノードに連結されているドライブに送信されます。たとえば、ノード1はパス名が /arc_dest_1 で開始するログを、ノード2はパス名が /arc_dest_2 で開始するログを、ノード3はパス名が /arc_dest_3 で開始するログをバックアップできます。

バックアップ・コマンドとリストア・コマンドのオートロケーション

Recovery Manager は、バックアップまたはリストアが必要なすべてのファイルの**オートロケーション**を自動的に実行します。この機能は、割り当てたチャンネルで、異なる CONNECT 設定または PARMS 設定を使用すると、自動的に使用可能になります。

オートロケーション機能は、アーカイブ REDO ログをバックアップする際に重要です。6-7 ページの「**非 CFS のローカル・アーカイブ・スキーム**」を使用している場合、各ノードが読み取ることができるのは、生成されたすべてのログのサブセットのみです。たとえば、ノード1はパス名が /arc_dest_1 で開始するログのみを、ノード2はパス名が /arc_dest_2 で開始するログのみを、ノード3はパス名が /arc_dest_3 で開始するログのみを読み取ることができます。Recovery Manager は、チャンネルを使用してログを読み取ることができない場合、そのチャンネルでのログのバックアップを試行しません。各チャンネルは、そのアーカイブ・ログのバックアップを、読取り可能なログに制限します。

リストアの操作時に、Recovery Manager はバックアップのオートロケーションを自動的に実行します。つまり、ノードにバックアップされたファイルのリストアが試行されるのは、特定のノードに接続されているチャンネルのみです。たとえば、ログ順序番号 1001 はノード1に連結されているドライブにバックアップされ、ログ 1002 はノード2に連結されているドライブにバックアップされるとします。各ノードに接続するチャンネルを割り当てる場合、ノード1に接続されたチャンネルは（ログ 1002 ではなく）ログ 1001 をリストアでき、ノード2に接続されたチャンネルは（ログ 1001 ではなく）ログ 1002 をリストアできます。

Real Application Clusters の Recovery Manager バックアップ・スキーム

この項では、次のバックアップ・スキームについて説明します。

- クラスタ・ファイル・システムのバックアップ・スキーム
- 非 CFS のバックアップ・スキーム

クラスタ・ファイル・システムのバックアップ・スキーム

このスキームは、6-5 ページの「[クラスタ・ファイル・システムのアーカイブ・スキーム](#)」に記載されているアーカイブ使用例を参照しています。CFS のバックアップ・スキームの場合、クラスタ内の各ノードには、すべてのデータ・ファイルとアーカイブ REDO ログに対する読取りアクセス権があります。

CFS で 1 つのローカル・ドライブにバックアップするアーカイブ・スキーム

このスキームは、クラスタ内の 1 つのノードにのみ 1 つのローカル・テープ・ドライブがあることを仮定しています。この場合、次のワнтаイム構成コマンドを実行します。

```
CONFIGURE DEVICE TYPE sbt PARALLELISM 1;  
CONFIGURE DEFAULT DEVICE TYPE TO sbt;
```

バックアップを実行するノードには、他のノードが書き込んだアーカイブ・ログへの読取り / 書き込みアクセス権があるため、いずれのノードについてもバックアップ・スクリプトは単純です。

```
BACKUP DATABASE PLUS ARCHIVELOG DELETE INPUT;
```

この場合、テープ・ドライブはすべてのデータ・ファイルとアーカイブ・ログを受け取りません。

CFS で複数のドライブにバックアップするアーカイブ・スキーム

このスキームは、クラスタ内の各ノードにそれぞれローカル・テープ・ドライブがあることを仮定しています。次のワнтаイム構成を実行して、クラスタ内の各ノードに 1 つのチャネルが構成されるようにします。たとえば、Recovery Manager のプロンプトで次のコマンドを入力します。

```
CONFIGURE DEVICE TYPE sbt PARALLELISM 3;  
CONFIGURE DEFAULT DEVICE TYPE TO sbt;  
CONFIGURE CHANNEL 1 DEVICE TYPE sbt CONNECT 'user1/password1@node1';  
CONFIGURE CHANNEL 2 DEVICE TYPE sbt CONNECT 'user2/password2@node2';  
CONFIGURE CHANNEL 3 DEVICE TYPE sbt CONNECT 'user3/password3@node3';
```

同様に、デバイス・タイプが DISK である場合もこの構成を実行できます。

注意： 前述のとおり、これはワンタイム構成です。バックアップのたびにこれらの構成コマンドを発行する必要はありません。

次のバックアップ・スクリプトは、クラスタ内のいずれのノードからでも実行でき、データ・ファイルとアーカイブ・ログのバックアップを複数のテープ・ドライブに分散します。

```
BACKUP DATABASE PLUS ARCHIVELOG DELETE INPUT;
```

たとえば、ディスク上のデータベースに 10 個のデータ・ファイルと 100 個のログがある場合、ノード 1 のテープ・ドライブにはデータ・ファイルの 1、3 および 7 とログの 1 ~ 33 を、ノード 2 のテープ・ドライブにはデータ・ファイルの 2、5 および 10 とログの 34 ~ 66 を、ノード 3 のテープ・ドライブにはデータ・ファイルの 4、6、8 および 9 とログの 67 ~ 100 をバックアップできます。

非 CFS のバックアップ・スキーム

このスキームは、6-7 ページの「[非 CFS のローカル・アーカイブ・スキーム](#)」に記載されているアーカイブ使用例を参照しています。非 CFS のバックアップ・スキームの場合、データ・ファイルは共有ディスク上にあり、クラスタ・データベース内のすべてのノードからアクセスできます。したがって、いずれのノードもすべてのデータ・ファイルをバックアップできます。非 CFS の環境では、各ノードがバックアップできるのはそれ自体のローカルのログのみです。つまり、リモート・アクセスに NFS を構成していないかぎり、ノード 1 は、ノード 2 やノード 3 のログにはアクセスできません。

バックアップ用に NFS を構成することを望まない場合は、バックアップを複数のドライブに分散します。ただし、バックアップ用に NFS を構成する場合は、1 つのドライブのみにバックアップできます。

非 CFS で複数のドライブにバックアップするスキーム

このスキームは、クラスタ内の各ノードにそれ自体のローカル・テープ・ドライブがあることを仮定しています。次のワンタイム構成を実行して、クラスタ内の各ノードに 1 つのチャンネルを構成します。たとえば、Recovery Manager のプロンプトで次のコマンドを入力します。

```
CONFIGURE DEVICE TYPE sbt PARALLELISM 3;  
CONFIGURE DEFAULT DEVICE TYPE TO sbt;  
CONFIGURE CHANNEL 1 DEVICE TYPE sbt CONNECT 'user1/password1@node1';  
CONFIGURE CHANNEL 2 DEVICE TYPE sbt CONNECT 'user2/password2@node2';  
CONFIGURE CHANNEL 3 DEVICE TYPE sbt CONNECT 'user3/password3@node3';
```

同様に、デバイス・タイプが DISK である場合もこの構成を実行できます。

注意： 前述のとおり、これはワнтаイム構成です。バックアップのたびにこれらの構成コマンドを発行する必要はありません。

データベース全体のバックアップ用に、すべてのノードから実行可能な本番バックアップ・スクリプトを作成します。Recovery Manager のオートロケーション機能は、各ノードに割り当てられているチャンネルがそのノードに配置されているログのみをバックアップできるようにします。次の例では、自動チャンネルを使用してデータベースとアーカイブ・ログのバックアップを作成しています。

```
BACKUP DATABASE PLUS ARCHIVELOG DELETE INPUT;
```

この例のデータ・ファイルのバックアップとログは、異なるテープ・ドライブに分散されています。ただし、チャンネル1で読み取ることができるのは、/arc_dest_1にローカルでアーカイブされたログのみです。これは、オートロケーション機能によって、チャンネル1は/arc_dest_1ディレクトリ内のログのみをバックアップし、チャンネル2は/arc_dest_2ディレクトリ内のログのみを読み取ることができ、チャンネル2は/arc_dest_2ディレクトリ内のログのみをバックアップできる、というように制限されるためです。重要な点は、すべてのログがバックアップされるが、異なるドライブに分散されることです。

非 CFS で1つのローカル・ドライブにバックアップするアーカイブ・スキーム

このスキームは、クラスタ内の1つのノードにのみ1つのローカル・テープ・ドライブがあることを仮定しています。このスキームでバックアップを行うには、他のノードにローカルでアーカイブされたログへの読取りアクセス権をバックアップ・ノードが持つように、NFSを構成する必要があります。したがって、非 CFS のアーカイブ・スキームでは1つのローカル・ドライブにバックアップすることはお薦めしません。この場合、次のワнтаイム構成のコマンドを実行できます。

```
CONFIGURE DEVICE TYPE sbt PARALLELISM 1;  
CONFIGURE DEFAULT DEVICE TYPE TO sbt;
```

バックアップを行うノードは、NFS を介してすべてのノードがアーカイブしたログを読み取ることができるため、バックアップ・スクリプトは、シングル・インスタンス・データベースの場合と同じです。どのノードがバックアップを実行するかに関係なく、同じスクリプトを実行できます。ただし、読取り / 書込みアクセス権を持つのは、各ノード上の1つのアーカイブ・ディレクトリのみです。したがって、DELETE ALL INPUT または DELETE INPUT を指定することはできません。ディスクから冗長ログを削除するには、各ノードで DELETE コマンドを実行する必要があります。

たとえば、データベース全体とアーカイブ・ログをバックアップ・ノードからバックアップするには、次の本番スクリプトを使用します。

```
BACKUP DATABASE PLUS ARCHIVELOG; # do not specify DELETE ... INPUT
```

アーカイブ・ログのみをバックアップするには、次のスクリプトを実行します。

```
BACKUP ARCHIVELOG ALL; # do not specify DELETE ... INPUT
```

Real Application Clusters でのインスタンス・リカバリ

インスタンス障害は、ソフトウェアまたはハードウェアの問題によってインスタンスが無効になった場合に発生します。インスタンス障害の後、Oracle はオンライン REDO ログ・ファイルを使用して、次の各項で説明するデータベース・リカバ리를自動的に実行します。

- [単一ノード障害](#)
- [Real Application Clusters での複数ノード障害](#)
- [Real Application Clusters でのインスタンス・リカバリのための制御ファイルとデータ・ファイルへのアクセス](#)
- [Real Application Clusters での Oracle インスタンス・リカバリの手順](#)

関連項目： インスタンス障害とリカバリの一般的な説明は、『Oracle9i バックアップおよびリカバリ概要』を参照してください。

単一ノード障害

Real Application Clusters のインスタンスは、障害が発生しなかったインスタンスの SMON プロセスを介して、リカバリを実行します。インスタンス・リカバリでは、障害が発生したインスタンスの再起動、または障害が発生したインスタンス上で実行していたアプリケーションのリカバリは実行されません。実行中のアプリケーションは、『Oracle9i Real Application Clusters セットアップおよび構成』に示すとおり、[フェイルオーバー](#)によって実行を継続する場合があります。

あるインスタンスが別のインスタンスのリカバリを実行する場合、障害が発生しなかったインスタンスは、障害が発生しているインスタンスによって生成された REDO ログ・エントリを読み取り、その情報を使用して、コミットされたすべてのトランザクションがデータベースに記録されるようにします。したがって、コミットされたトランザクションのデータが失われることはありません。リカバリを実行中のインスタンスは、障害発生時にアクティブだったトランザクションをロールバックし、それらのトランザクションによって使用されたリソースを解放します。

注意： すべてのオンライン REDO ログは、リカバリのためにアクセスできる必要があります。オンライン・ログをミラー化することをお勧めします。

関連項目： アプリケーション・フェイルオーバーおよび高可用性の概要については、『Oracle9i Real Application Clusters 概要』を参照してください。

Real Application Clusters での複数ノード障害

障害を受けなかったインスタンスが1つでもあるかぎり、複数ノード障害の後、その SMON プロセスは、障害が発生したすべてのインスタンスに対してインスタンス・リカバリを実行します。Real Application Clusters データベースのすべてのインスタンスに障害が発生した場合、Oracle では、次のインスタンスがデータベースをオープンするときに障害リカバリが自動的に実行されます。

リカバリを実行するインスタンスは、障害が発生したインスタンスの1つである必要はありません。また、そのインスタンスは、Real Application Clusters のどのノードからでも、共有または排他モードでデータベースをマウントできます。このリカバリ手順は、1つのインスタンスが、障害が発生したすべてのインスタンスのリカバリを実行するという点以外は、共有モードで実行している Oracle でも、排他モードで実行している Oracle でも同じです。

Real Application Clusters でのインスタンス・リカバリのための制御ファイルとデータ・ファイルへのアクセス

リカバリ・カタログを使用している場合、Recovery Manager はリカバリ・カタログを使用して障害が発生したインスタンスの制御ファイルをリカバリします。リカバリ・カタログを使用していない場合、Recovery Manager はリカバリを実行しているインスタンスの制御ファイルのコピーを使用して、障害が発生したインスタンスの制御ファイルをリカバリします。

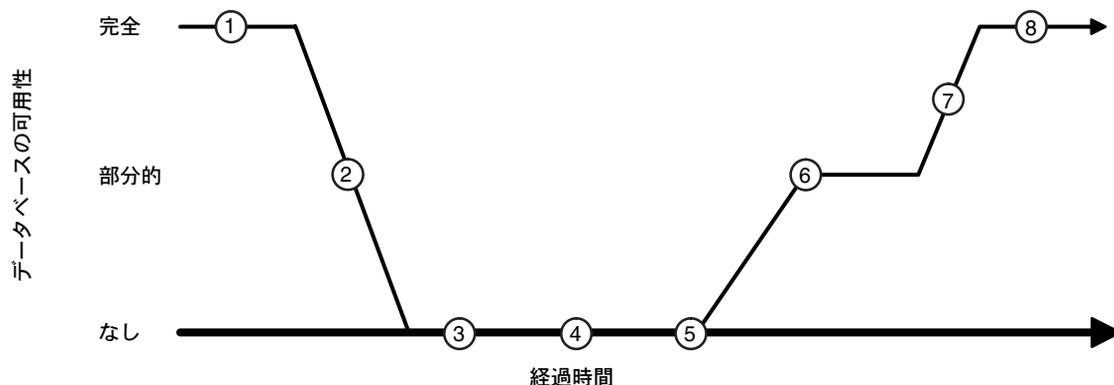
また、障害が発生したインスタンスのリカバリを実行するインスタンスには、障害が発生したインスタンスがアクセスしていた、すべてのオンライン・データ・ファイルに対するアクセス権限が必要です。データ・ファイルの検証に失敗したためにインスタンス・リカバリに障害が発生した場合、Oracle はアラート・ログにメッセージを書き込みます。ファイルへのアクセスを妨げていた問題を修正した後は、SQL 文 ALTER SYSTEM CHECK DATAFILES を使用してデータ・ファイルを検証し、インスタンスがファイルを使用できるようにします。

関連項目： ALTER SYSTEM 文の CHECK DATAFILES 句の詳細については、『Oracle9i SQL リファレンス』を参照してください。

Real Application Clusters での Oracle インスタンス・リカバリの手順

次に、インスタンスのリカバリ手順におけるデータベースの可用性の程度を示します。

図 7-1 Oracle インスタンスのリカバリ手順



図リカバリの手順は、次のとおりです。

1. Real Application Clusters が複数のノードで実行中です。
2. ノード障害が検出されます。
3. **グローバル・キャッシュ・サービス (GCS)** が再構成され、リソース管理が、障害が発生していないノードに再分散されます。
4. 障害が発生したインスタンスの REDO ログを SMON が読み取り、リカバリに必要なデータベース・ブロックを識別します。
5. SMON が、リカバリのために必要なすべてのデータベース・ブロックの取得を要求します。要求が完了すると、他のすべてのブロックがアクセス可能になります。

注意： **グローバル・キャッシュ・サービス・プロセス (LMSn)** では、マスターを失ったリソースのみを再マスター化します。

6. Oracle がロールフォワード・リカバリを実行します。障害が発生したスレッドの REDO ログが、データベースに適用されます。
7. Oracle がロールバック・リカバリを実行します。コミットされていないすべてのトランザクションの UNDO ブロックがデータベースに適用されます。
8. インスタンスのリカバリが完了し、すべてのデータがアクセス可能になります。

Real Application Clusters でのメディア・リカバリの概要

メディア障害は、Oracle ファイルの記憶メディアが破損した場合に発生します。通常はメディア障害によって、データの読取りまたは書き込みができなくなり、結果的に1つ以上のデータベース・ファイルが消失します。メディア・リカバリは、クライアント・アプリケーションを介してユーザーが起動する必要があります。一方、インスタンス・リカバリは、データベースによって自動的に実行されます。

この場合は、**Recovery Manager (RMAN)** を使用してデータ・ファイルのバックアップをリストアしてから、データベースをリカバリします。Real Application Clusters 環境での Recovery Manager のメディア・リカバリ手順は、シングル・インスタンス環境の Recovery Manager のメディア・リカバリ手順とほぼ同じです。

メディア・リカバリに関する問題点は、7-4 ページの「**ファイルのアクセス可能性とバックアップ・メディア**」に記載されている内容と同じです。リカバリを実行するノードは、必要なデータ・ファイルをすべてリストアできることが必要です。このノードは、ディスク上の必要なアーカイブ・ログをすべて読み取ることができるか、あるいはバックアップしたデータ・ファイルをリストアできることも必要です。

Real Application Clusters の Recovery Manager リストア・スキーム

この項では、次のリストア・スキームについて説明します。

- クラスタ・ファイル・システムのリストア・スキーム
- 非 CFS のリストア・スキーム

クラスタ・ファイル・システムのリストア・スキーム

CFS のスキームでバックアップを実行した場合のリストアとリカバリの手順は単純で、シングル・インスタンスの場合と実質的に同じです。

CFS スキームでの非分散バックアップのリストアとリカバリ

ここでは、7-7 ページの「**CFS で1つのローカル・ドライブにバックアップするアーカイブ・スキーム**」に記載されている非分散バックアップ・スキームを使用していると仮定します。この例では、次のチャンネル構成が必要です。

```
CONFIGURE DEVICE TYPE sbt PARALLELISM 1;
CONFIGURE DEFAULT DEVICE TYPE TO sbt;
```

ノード3がバックアップを実行するとします。リストアとリカバリの処理にノード3が使用可能で、既存のすべてのログがバックアップ済みか、ディスク上にある場合は、次のコマンドを実行して完全リカバリを実行します。

```
RESTORE DATABASE;
RECOVER DATABASE;
```

バックアップを実行したノード3でリストアとリカバリを処理できない場合は、残りのノードの1つに対してメディア管理デバイスを構成し、このデバイスでノード3のテープを使用可能にします。

CFS スキームでの分散バックアップのリストアとリカバリ

ここでは、7-7 ページの「[CFS で複数のドライブにバックアップするアーカイブ・スキーム](#)」に記載されている分散バックアップ・スキームを使用していると仮定します。次のワнтаイム構成を実行して、クラスタ内の各ノードに1つのチャンネルが構成されるようにします。たとえば、Recovery Manager のプロンプトで次のコマンドを入力します。

```
CONFIGURE DEVICE TYPE sbt PARALLELISM 3;  
CONFIGURE DEFAULT DEVICE TYPE TO sbt;  
CONFIGURE CHANNEL 1 DEVICE TYPE sbt CONNECT 'user1/password1@node1';  
CONFIGURE CHANNEL 2 DEVICE TYPE sbt CONNECT 'user2/password2@node2';  
CONFIGURE CHANNEL 3 DEVICE TYPE sbt CONNECT 'user3/password3@node3';
```

既存のすべてのログがバックアップ済みか、ディスク上にある場合は、次のコマンドを実行してクラスタ内の任意のノードから完全リカバリを実行します。

```
RESTORE DATABASE;  
RECOVER DATABASE;
```

リストアの前には、Recovery Manager によるバックアップのオートロケーションが行われるため、各ノードに接続されているチャンネルは、そのノードに連結されているテープ・ドライブにバックアップされたファイルのみをリストアします。

非 CFS のリストア・スキーム

このスキームでは、各ノードが異なるディレクトリにローカルでアーカイブします。たとえば、ノード1は /arc_dest_1 に、ノード2は /arc_dest_2 に、ノード3は /arc_dest_3 にアーカイブします。リカバリ・ノードが残りのノードでアーカイブ・ディレクトリを読み取ることができるように、NFS を構成する必要があります。リストアとリカバリの手順は、バックアップが分散されているかどうかによって異なります。

非 CFS で分散バックアップをリストアおよびリカバリするアーカイブ・スキーム

ここでは、7-8 ページの「[非 CFS で複数のドライブにバックアップするスキーム](#)」に記載されている分散バックアップ・スキームを使用していると仮定します。すべてのノードが使用可能で、すべてのアーカイブ・ログがバックアップされている場合は、データベースをマウントして任意のノードで次のコマンドを実行することで、完全なリストアとリカバリを実行できます。

```
RESTORE DATABASE;  
RECOVER DATABASE;
```

リカバリ・ノードは、クラスタ内の各ノードでサーバー・セッションを開始します。この例では、データベースが分散バックアップされていると仮定しているため、サーバー・セッションは、バックアップ・データ・ファイルを各ノードに連結されているテープ・ドライブからリストアします。NFS 構成では、各ノードにその他のノードに対する読取りアクセス権があるため、リカバリ・ノードは、ローカルとリモートのディスクにあるアーカイブ・ログを読取りおよび適用できます。手動によるログの転送は不要です。

非 CFS で非分散バックアップをリストアおよびリカバリするアーカイブ・スキーム

ここでは、7-9 ページの「非 CFS で 1 つのローカル・ドライブにバックアップするアーカイブ・スキーム」に記載されている非分散バックアップ・スキームを使用していると仮定します。次のチャンネル構成があります。

```
CONFIGURE DEVICE TYPE sbt PARALLELISM 1;  
CONFIGURE DEFAULT DEVICE TYPE TO sbt;
```

ノード 3 がバックアップを実行するとします。リストアとリカバリの操作にノード 3 を使用でき、リモート・ノードの NFS マウント・ポイントにアクセス可能な場合は、次のコマンドで完全リカバリを実行します。

```
RESTORE DATABASE;  
RECOVER DATABASE;
```

一部のノードが停止状態のために NFS を介してそのノードのログにアクセスできない場合、および完全リカバリに必要なログのバックアップがない場合は、次のように、データベース全体のバックアップ後に、最初にログが欠落している地点まで不完全リカバリを実行する必要があります。

```
RUN  
{  
  # in this example, sequence 1234 is the first missing log  
  SET UNTIL LOG SEQUENCE 1234 THREAD 3;  
  RESTORE DATABASE;  
  RECOVER DATABASE;  
}  
  
ALTER DATABASE OPEN RESETLOGS;
```

Real Application Clusters でのパラレル・リカバリ

パラレル・リカバリでは、複数 CPU および I/O のパラレル化を使用して、スレッドまたはメディアのリカバリの実行に必要な時間を削減します。パラレル・リカバリは、複数のディスク上の複数のデータ・ファイルを同時にリカバリする場合のリカバリ時間の削減に最も有効です。Real Application Clusters データベースでは、パラレル・インスタンス・リカバリ、パラレル障害リカバリおよびパラレル・メディア・リカバリを使用できます。

関連項目： 詳細については、『Oracle9i ユーザー管理バックアップおよびリカバリ・ガイド』を参照してください。

Recovery Manager を使用したパラレル・リカバリ

Recovery Manager の RESTORE コマンドおよび RECOVER コマンドを使用すると、この項で説明する 3 段階のリカバリ・プロセスが自動的にパラレル化されます。

- [データ・ファイルのリストア](#)
- [増分バックアップの適用](#)
- [REDO ログの適用](#)

データ・ファイルのリストア

データ・ファイルのリストアする場合、Recovery Manager のリカバリ・スクリプトに割り当てられているチャンネル数によって、Recovery Manager が使用するパラレル化が効果的に設定されます。たとえば、5 つのチャンネルを割り当てると、最大 5 つのパラレル・ストリームでデータ・ファイルのリストアできます。

増分バックアップの適用

同様に、増分バックアップを適用する場合、割り当てるチャンネル数によって潜在的なパラレル化が決定されます。

REDO ログの適用

Recovery Manager では、RECOVERY_PARALLELISM 初期化パラメータの設定で決定された数のパラレル・プロセスを使用して、REDO ログが適用されます。

関連項目： 7-17 ページ [「RECOVERY_PARALLELISM パラメータの設定」](#)

SQL*Plus を使用したパラレル・リカバリ

データベースのバックアップとリカバリに、ユーザー管理による方法を使用している場合は、次のいずれかの手順で、インスタンス・リカバリとメディア・リカバリをパラレル化できます。

- **RECOVERY_PARALLELISM** パラメータの設定
- **RECOVER** 文オプションの指定

Real Application Clusters は、1つのプロセスを使用して、ログ・ファイルを順次読み取り、REDO 情報をいくつかのリカバリ処理にディスパッチして、ログ・ファイルからの変更をデータ・ファイルに適用できます。Oracle は、リカバリ処理を自動的に開始するため、2つ以上のセッションを使用してリカバリを実行する必要はありません。

RECOVERY_PARALLELISM パラメータの設定

RECOVERY_PARALLELISM 初期化パラメータによって、インスタンス・リカバリまたはメディア・リカバリに関する REDO アプリケーション・サーバー・プロセスの数が指定されます。1つのプロセスによって、ログ・ファイルが順次読み取られ、REDO 情報がいくつかのリカバリ処理にディスパッチされます。このリカバリ処理では、ログ・ファイルの変更がデータ・ファイルに適用されます。値が 0 (ゼロ) または 1 の場合は、リカバリが 1つのプロセスによって逐次実行されることを意味します。このパラメータの値が、PARALLEL_MAX_SERVERS パラメータの値を超えることはできません。

RECOVER 文オプションの指定

RECOVER 文を使用してインスタンス・リカバリおよびメディア・リカバリをパラレル化する場合、インスタンスへのリカバリ処理の割当ては、オペレーティング・システム固有です。PARALLEL 句の DEGREE キーワードによって、Real Application Clusters データベースの各インスタンスの処理数またはすべてのインスタンスに分散する処理数を指定できます。

関連項目： インスタンスへのリカバリ処理の割当ての詳細については、システム固有の Oracle マニュアルを参照してください。

第IV部

Real Application Clusters 環境の拡張性

第IV部では、ノードとインスタンスの追加方法、およびインスタンスの削除方法について説明します。第IV部に含まれる章は、次のとおりです。

- [第8章「Real Application Clusters でのノードとインスタンスの追加およびインスタンスの削除」](#)

Real Application Clusters でのノードと インスタンスの追加およびインスタンス の削除

この章では、**Real Application Clusters** データベース環境で、ノードとインスタンスを追加する方法、およびインスタンスを削除する方法の詳細な手順について説明します。内容は次のとおりです。

- クラスタへのノードの追加
- ノードを追加する手順の概要
- クラスタウェア・レイヤーへのノードの追加
- Oracle レイヤーへのノードの追加
- インスタンスの削除

関連項目： **Database Configuration Assistant (DBCA)** を使用してデータベースを作成および削除する手順の詳細については、『Oracle9i Real Application Clusters セットアップおよび構成』を参照してください。

クラスタへのノードの追加

この項では、Real Application Clusters 環境に動的にノードを追加する方法について説明します。この項の内容は次のとおりです。

- ノードを追加する手順の概要
- クラスタウェア・レイヤーへのノードの追加
- Oracle レイヤーへのノードの追加

関連項目： インスタンスを削除する手順については、8-19 ページの「[インスタンスの削除](#)」を参照してください。

ノードを追加する手順の概要

Real Application Clusters にノードを追加する手順には、次の 2 つの主要な手順が含まれます。

- クラスタウェア・レイヤーへのノードの追加
- Oracle レイヤーへのノードの追加

この章で説明するように、クラスタウェア・レイヤーにノードを追加する方法はオペレーティング・システム固有です。Oracle レイヤーにノードを追加する 2 つ目の手順については、この章で説明する作業では **Oracle Universal Installer (OUI)** および **Database Configuration Assistant (DBCA)** を使用する必要があります。2 つ目の手順の作業は共通であり、すべてのプラットフォームに適用できます。

クラスタウェア・レイヤーへのノードの追加

UNIX では、追加ノードを収容できるようにクラスタウェアを事前に構成している場合を除いて、ノードを動的に追加できません。そのため、追加ノードを収容するには、クラスタウェアを停止して、使用しているクラスタウェアを再構成する必要があります。その後、「[UNIX のクラスタウェア・レイヤーに対するノードの追加](#)」の手順に従ってください。

注意： 一部のプラットフォームではクラスタウェアを事前に構成せずにノードを動的に追加できます。

Windows NT および Windows 2000 では、Oracle はクラスタ・ソフトウェアまたは **オペレーティング・システム依存 (OSD) のクラスタウェア** を提供しています。これによって、8-4 ページの「[Windows NT および Windows 2000 のクラスタウェア・レイヤーに対するノードの追加](#)」で説明するとおり、ノードを動的に追加できます。

この項の内容は次のとおりです。

- [UNIX のクラスタウェア・レイヤーに対するノードの追加](#)
- [Windows NT および Windows 2000 のクラスタウェア・レイヤーに対するノードの追加](#)

UNIX のクラスタウェア・レイヤーに対するノードの追加

Real Application Clusters データベースを使用する既存の UNIX クラスタに新しいノードを追加することによって、Real Application Clusters の**拡張性**および**高可用性**機能を利用できます。前述のとおり、追加ノードを収容できるように UNIX クラスタを事前に構成していれば、データベースを停止することなくノードを追加できます。

UNIX 環境で既存の Real Application Clusters に新しいノードを追加するには、次の手順を使用することをお薦めします。これらの手順では、Real Application Clusters データベースを実行している UNIX クラスタがすでに存在していると仮定しています。

新しいノードの準備として、そのノードが、ベンダー・オペレーティング・システムやクラスタウェアなどに関して既存のノードの正確なクローンであることを確認してください。その後、示されている順序で次の作業を実行して、UNIX のクラスタウェア・レイヤーにノードを追加します。

1. [クラスタへのノードの接続](#)
2. [ベンダーのクラスタ・ソフトウェアのインストール](#)
3. [RAW デバイスの作成 \(オプション\)](#)

クラスタへのノードの接続

既存のクラスタに新しいノードのハードウェアを接続します。この作業には、電気的な接続、ネットワーク・インターコネクト、共有ディスク・サブシステムの接続などが含まれます。この手順の詳細は、ハードウェア・ベンダーのマニュアルを参照してください。

ベンダーのクラスタ・ソフトウェアのインストール

ベンダーのインストール手順を参照して、クラスタ・ソフトウェアをインストールしてください。また、新しいノードをクラスタのアクティブなメンバーにする手順についても、ベンダーのマニュアルを参照してください。

RAW デバイスの作成 (オプション)

プラットフォームがクラスタ・ファイル・システム (CFS) をサポートしており、クラスタ・データベースに RAW デバイスを使用していない場合、この項で説明している RAW デバイスの作成は必要ありません。8-8 ページの「[Oracle レイヤーへのノードの追加](#)」に進んでください。

新しいノードに Real Application Clusters ソフトウェアをインストールする前に、新しいインスタンスの REDO ログを収容する新しいディスク・ボリュームを 2 つ以上作成してくだ

さい。これらの REDO ログおよびインストール前の作業の要件の詳細については、『Oracle9i Real Application Clusters セットアップおよび構成』を参照してください。

REDO ログ用の新しいディスク・ボリュームは、既存のノードのインスタンス用に構成した REDO ログのボリュームと同じサイズにしてください。ディスク・ボリュームおよび RAW デバイスの作成手順については、ベンダーのマニュアルを参照してください。

新しいノードとインスタンスを追加するクラスタ・データベースが自動 UNDO 管理を使用している場合は、新しいインスタンス用に新しい UNDO 表領域のための追加 RAW ボリュームを作成します。この手順の完了後、8-8 ページの「[Oracle レイヤーへのノードの追加](#)」を参照して、Oracle レイヤーにノードを追加してください。

関連項目： RAW デバイスの作成の詳細については、『Oracle9i Real Application Clusters セットアップおよび構成』を参照してください。

Windows NT および Windows 2000 のクラスタウェア・レイヤーに対するノードの追加

Real Application Clusters データベースを使用する既存の Windows クラスタに新しいノードを動的に追加することによって、Real Application Clusters の拡張性および高可用性機能を利用できます。

Windows 環境でノードを追加する場合、次の手順を使用することをお勧めします。これらの手順では、Real Application Clusters データベースを実行している Windows クラスタがすでに存在しており、新しいノードは、クラスタの既存のノードで実行中のオペレーティング・システムと同じオペレーティング・システムを使用すると仮定しています。たとえば、クラスタ内のすべてのノード上に、Windows NT 4.0 または Windows 2000 がインストールされていると仮定しています。

次の作業を順序どおりに実行して、Windows NT および Windows 2000 のクラスタウェア・レイヤーに新しいノードを追加します。

1. [クラスタへのノードの接続](#)
2. [クラスタ・ソフトウェアのインストール](#)
3. [ディスク・パーティションの作成](#)
4. [ディスク・パーティションへのリンクの作成](#)

クラスタへのノードの接続

既存のクラスタに新しいノードのハードウェアを接続します。この作業には、電気的な接続、ネットワーク・インターコネクト、共有ディスク・サブシステムの接続などが含まれます。この手順の詳細については、ハードウェア・ベンダーのマニュアルを参照してください。

クラスタ・ソフトウェアのインストール

オペレーティング・システムのベンダーが提供したクラスタウェアを既存のノード上で使用している場合、その同じベンダー・ソフトウェアを、ベンダーのインストール手順を使用して新しいノードにもインストールします。

ただし、オラクル社の試用版のクラスタウェアまたはオペレーティング・システム固有のソフトウェアを、クラスタの既存のノード上で使用している場合は、これらの Oracle OSD クラスタウェアを、Cluster Setup Wizard を使用して新しいノード上にもインストールする必要があります。これを実行するには、Cluster Setup Wizard を実行する前に、既存のノードの1つから次の手順を実行します。

1. 既存のクラスタ上で**仮想インタフェース・アーキテクチャ (VIA)**・インターコネクトを使用している場合、新しいノード上にも VIA インターコネクトがインストールされていることを確認します。
2. 既存のクラスタ上でプライベート・インターコネクトを使用している場合、新しいノード上にもプライベート・インターコネクトがインストールされていることを確認します。
3. クラスタ・ファイル・システムを使用している場合、この手順はスキップしてください。既存のノード上で、クラスタ・ソフトウェアがどのディスク・ドライブにインストールされているかを確認してください。OSD クラスタウェアおよび Object Link Manager をインストールするには、新しいノードの同一ディスク上に 2MB 以上の空き領域があることを確認します。
4. クラスタの既存の各ノードから次のコマンドが実行できることを確認します。

```
NET USE %host_name%C$
```

host_name は、新しいノードのパブリック・ネットワーク名です。オペレーティング・システムが次のように応答した場合、各ノードに対して管理者権限があります。

```
Command completed successfully.
```

5. 新しいノード上に、いずれかのバージョンのベンダー OSD クラスタウェアがすでにインストールされている場合に Oracle OSD クラスタウェアをインストールするには、ベンダー OSD クラスタウェアが停止していることを確認します。そのような例に当てはまらない場合は、次の手順に進んでください。
6. 新しいノードの CD-ROM ドライブに、Oracle9i リリース 2 (9.2) の CD セットの CD 1 を挿入します。
7. 次のように入力して、PREINSTALL¥CLUSTERSETUP ディレクトリに移動します。

```
cd ¥PREINSTALL¥CLUSTERSETUP
```

8. ¥PREINSTALL¥CLUSTERSETUP ディレクトリから次のように入力して、Cluster Setup Wizard を実行します。

```
clustersetup
```

9. Cluster Setup Wizard の「Welcome」ページが表示されます。「Next」をクリックして次に進みます。
10. 「Node Addition」オプションを選択して、「Next」をクリックします。
11. 新しく追加するノードにパブリック名を指定します。既存のクラスタも高速プライベート・ネットワークを使用している場合は、新しいノードにプライベート・ネットワーク・インタフェースの名前を指定し、「Next」をクリックします。
12. 「Finish」をクリックします。

関連項目：

- 詳細については、『Oracle9i Database for Windows インストール・ガイド』を参照してください。

ディスク・パーティションの作成

Oracle Real Application Clusters のノードに Oracle データベース・ソフトウェアをインストールする前に、RAW ディスク・パーティションを作成する必要があります。新しいインスタンス用の REDO ログを収容するために、2 つ以上の新しいディスク・パーティションを作成します。これらのディスク・パーティションは、既存のノードのインスタンス用に構成した REDO ログ・パーティションと同じサイズにしてください。ノードを追加するデータベースで自動 UNDO 管理を使用している場合、UNDO 表領域用に追加論理パーティションを作成してください。

関連項目： Windows NT および Windows 2000 プラットフォームでの、RAW パーティションの構成の詳細については、『Oracle9i Real Application Clusters セットアップおよび構成』を参照してください。

クラスタの既存のノードのいずれかから、次の手順を実行します。

1. Windows NT の場合は、パスを使用して「スタート」→「プログラム」→「Oracle - [HOME_NAME]」→「Administrative Tools」→「Disk Administrator」を選択し、ディスク・アドミニストレータを起動します。

Windows 2000 の場合は、パスを使用して「スタート」→「プログラム」→「Oracle - [HOME_NAME]」→「Administrative Tools」→「コンピュータの管理」を選択し、ディスクの管理を起動します。次に「記憶域」フォルダを展開して「ディスクの管理」を展開します。Windows 2000 の場合、拡張パーティションとして基本ディスクを使用して RAW パーティションを作成します。
2. 拡張パーティションの未割当て部分をクリックします。
3. 「パーティション」メニューから「作成」を選択します。ダイアログ・ボックスが表示されたら、パーティションのサイズを入力します。既存のノードで使用していたサイズと同じサイズを使用していることを確認してください。

4. 新しく作成されたパーティションをクリックし、「Tool」メニューから「Assign Drive Letter」を選択します。
5. 「ドライブ文字を割り当てない」を選択し、「OK」をクリックします。
6. 2つ目以降の追加パーティションを作成するには、手順2～5を繰り返します。
7. 「パーティション」メニューから「今すぐ変更を反映」を選択して、新しいパーティション情報を保存します。
8. Windows NT で 4.0 より前の Service Pack を使用している場合は、クラスタ内のすべてのノードを再起動する必要がある場合があります。これによって、すべてのノードが新しいパーティションを認識します。

注意： 5.0 より前の Service Pack は、西暦 2000 年問題に対応していません。したがって、6a の使用をお勧めします。

ディスク・パーティションへのリンクの作成

クラスタ・ファイル・システムがある場合は、これらの手順を実行する必要はありません。

「ディスク・パーティションの作成」で作成したパーティションにリンクを作成して、既存のノードがリンクを認識できるようにします。さらに、新しいノードも、新しいリンクのみでなく、事前に存在していた論理ドライブへのシンボリック・リンクを認識する必要があります。これらのリンクを作成するには、既存のノードのいずれかから次の手順を実行します。

1. %ORACLE_HOME%\bin ディレクトリから、次のように入力して Object Link Manager を起動します。

```
GUIOracleOBJManager
```

Object Link Manager は論理ドライブへのシンボリック・リンクを自動的に検出し、それらを OLM の GUI に表示します。

2. 前述の「ディスク・パーティションの作成」で作成したパーティションのディスク番号およびパーティション番号を再コールします。OLM の GUI でディスク番号およびパーティション番号を調べ、次の作業を実行します。
 - a. 「New Link」欄の下にあるボックスの横で右クリックし、最初のパーティションのリンク名を入力します。
 - b. 2つ目以降の追加パーティションにも手順 a を繰り返します。

たとえば、名前が db の Real Application Clusters データベースが 2 つのノード上で実行する 2 つのインスタンスで構成されているとします。このデータベースの 3 つ目のノードに 3 つ目のインスタンスを追加する場合、REDO ログのリンク名は db_redo3_1、db_redo3_2 のようになります。

3. 既存のデータベースで自動 UNDO 管理を使用している場合は、前述の「ディスク・パーティションの作成」で作成した UNDO 表領域用の論理パーティションへのリンク名を入力します。たとえば、名前が db の Real Application Clusters データベースが、2つのノード上で実行する2つのインスタンスで構成されているとします。このデータベースの3つ目のノードに3つ目のインスタンスを追加する場合、UNDO 表領域のリンク名は db_undotbs3 となります。
4. 「Options」メニューから「Commit」を選択します。これによって、現行のノードに新しいリンクが作成されます。
5. 「Options」メニューから「Sync Nodes」を選択します。これによって、クラスタの既存のノードが新しいリンクを認識できるようになります。
6. Object Link Manager を終了するには、「Options」メニューから「Exit」を選択します。

Oracle レイヤーへのノードの追加

この時点で、クラスタのクラスタウェア・レイヤーに新しいノードが追加されています。既存のクラスタの Oracle レイヤーにノードを追加するには、すべてのプラットフォームで次の手順を実行します。

- [新しいノードへの Oracle ソフトウェアのインストール](#)
- [インストール後の手順](#)
- [新しいノードへのデータベース・インスタンスの追加](#)

これらの手順の詳細を次に示します。

新しいノードへの Oracle ソフトウェアのインストール 新しいノードに Oracle ソフトウェアをインストールするには、次の手順を実行します。

1. プラットフォームがクラスタ・ファイル・システムをサポートしている場合は、8-9 ページの「インストール後の手順」に進みます。
2. 新しいノード上で、新しいノードの CD-ROM ドライブに Oracle9i リリース 2 (9.2) の CD-ROM を挿入します。
3. 次の手順のうち、ご使用のプラットフォーム用の手順を使用して、Oracle Universal Installer (OUI) を実行します。
 - UNIX では、Oracle CD 1 のルート・ディレクトリから runInstaller コマンドを実行して OUI を起動します。OUI の「Welcome」ページが表示されます。
 - Windows NT および Windows 2000 では、setup.exe コマンドを実行して OUI を起動します。OUI の「Welcome」ページが表示されます。

4. 「File Locations」 ページにある「Path」フィールド内の「Destination」に、Oracle Enterprise Edition および Real Application Clusters ソフトウェアをインストールする ORACLE_HOME を入力します。入力する ORACLE_HOME は、既存のノードが使用する ORACLE_HOME と同じにしてください。
5. 「Available Products」 ページで、「**Oracle9i Database**」を選択します。
6. 「Install Types」 ページで、「**Enterprise Edition**」を選択します。OUI の「Database Configuration」 ページが表示されます。
7. OUI の指示に従って、既存のノードにインストールしたのと同じ製品およびオプションをインストールします。または、「Database Configuration」 ページで「**Software Only**」構成オプションを選択します。
8. 「Node Selection」 ページで、現行ノードまたは新しいノードの名前のみを選択します。他の既存のノード名を選択しないでください。Oracle ソフトウェアはすでに既存のノード上に存在しています。
9. OUI が Oracle ソフトウェアのインストールを完了したら、インストール後のスクリプト root.sh を実行します。次に、OUI を終了し、次の「インストール後の手順」に進みます。

インストール後の手順 新しいノードから、次のインストール後の手順を実行します。

1. ORACLE_HOME の bin ディレクトリから Oracle Net Configuration Assistant を実行して、ノードの新しいリスナーを構成します。
2. ご使用のプラットフォーム固有の Oracle マニュアルを参照して、プラットフォーム固有のインストール後の手順を終了します。

新しいノードへのデータベース・インスタンスの追加 新しいノードにインスタンスを追加するには、新しいノードだけでなく、既存のすべてのノードで**グローバル・サービス・デーモン (GSD)** が実行されていることを確認します。GSD が実行されていない場合は、プラットフォーム固有の次の位置から、いずれかのノードで `gsdctl start` コマンドを実行します。

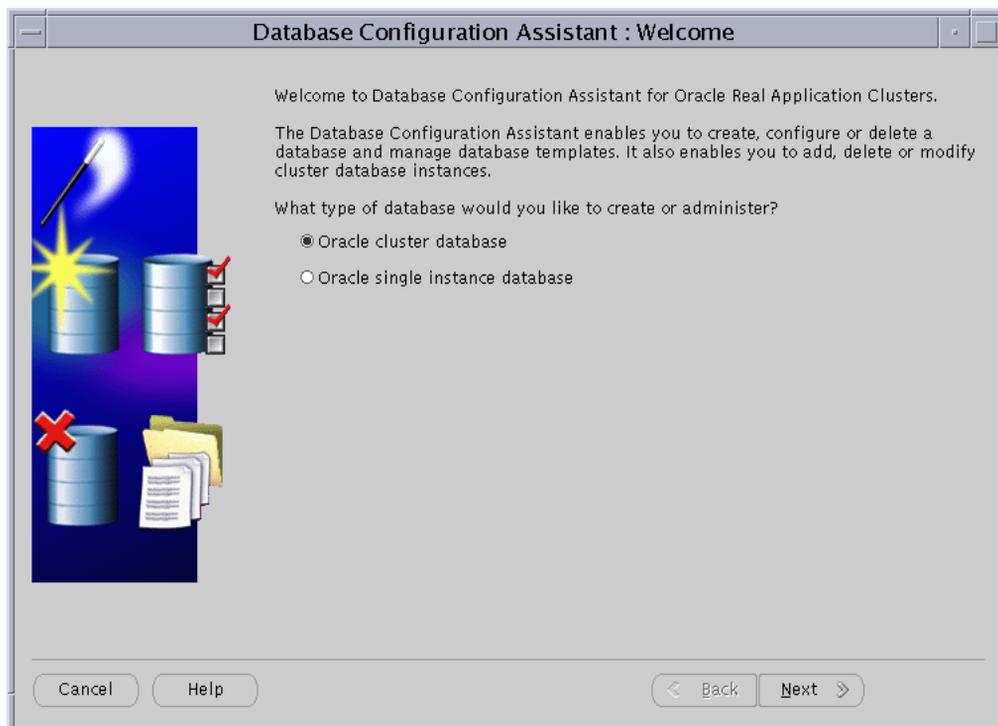
- UNIX の場合は、`$ORACLE_HOME/bin`
- Windows NT および Windows 2000 の場合は、`%ORACLE_HOME%\bin`

次に、既存のノードの 1 つに移動し、次の手順を実行します。

1. Database Configuration Assistant (DBCA) を起動します。UNIX の場合は、`$ORACLE_HOME` の bin ディレクトリで `dbca` と入力します。Windows NT および Windows 2000 の場合は、「スタート」→「プログラム」→「Oracle - HOME_NAME」→「Configuration and Migration Tools」→「Database Configuration Assistant」の順に選択します。

DBCA ウィザードが起動し、[図 8-1](#) に示す Real Application Clusters の「Welcome」ページが表示されます。

図 8-1 Real Application Clusters の DBCA 「Welcome」 ページ



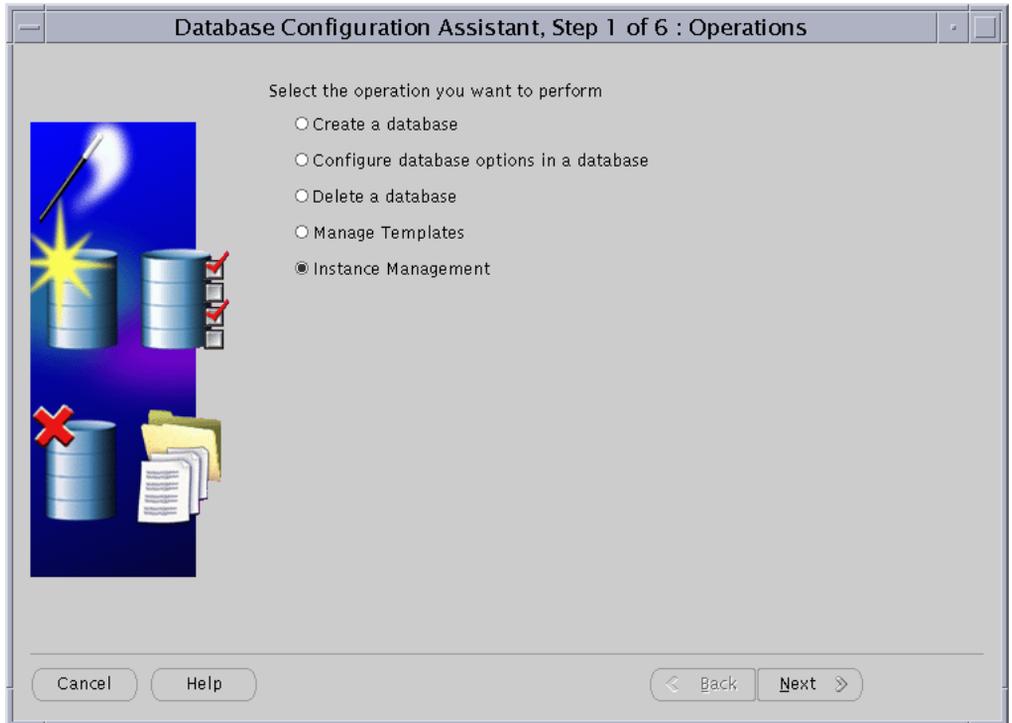
注意： Real Application Clusters の「Welcome」ページが表示されない場合は、DBCA で次の操作を実行できなかったことを意味します。

- Windows NT または Windows 2000 の場合、**Cluster Manager (CM)** ソフトウェアと通信できませんでした。
- UNIX オペレーティング・システムの場合、**グローバル・キャッシュ・サービス (GCS)** ソフトウェアまたはクラスタ内のノードのリストを検出できませんでした。

この問題を解決するには、ベンダーのオペレーティング・システム固有のマニュアルを参照して問題を修正し、DBCA を再起動してください。

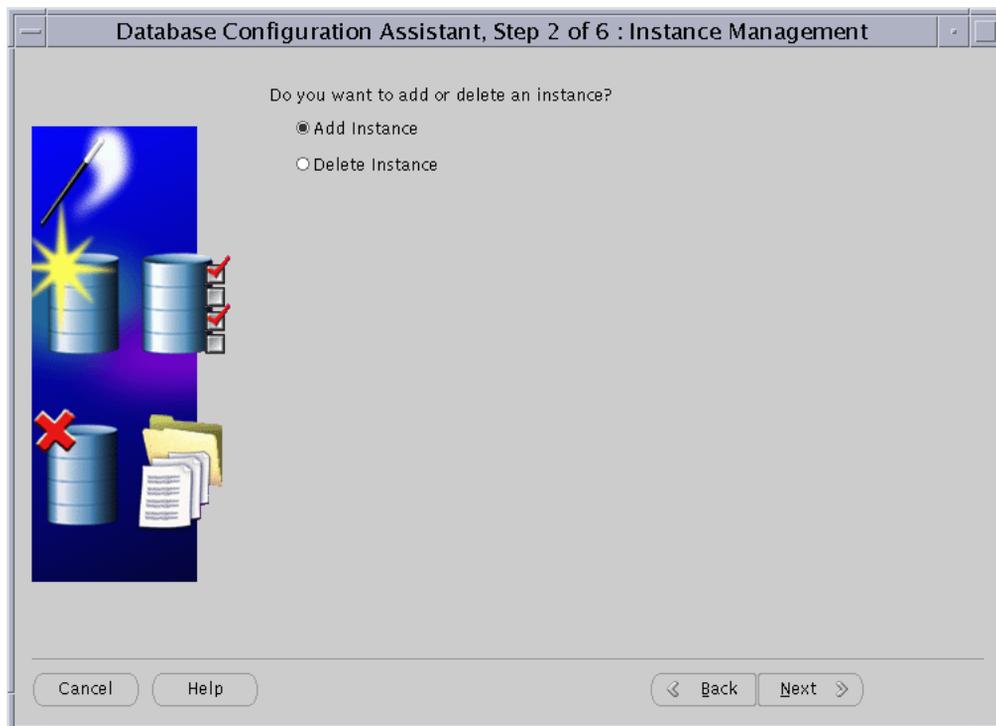
2. 「Oracle cluster database」を選択して、「Next」をクリックします。「Next」をクリックすると、[図 8-2](#) に示す「Operations」ページが表示されます。

図 8-2 Database Configuration Assistant の「Operations」ページ



3. 「Instance Management」を選択して、「Next」をクリックします。
「Next」をクリックすると、[図 8-3](#) に示す「Instance Management」ページが表示されます。

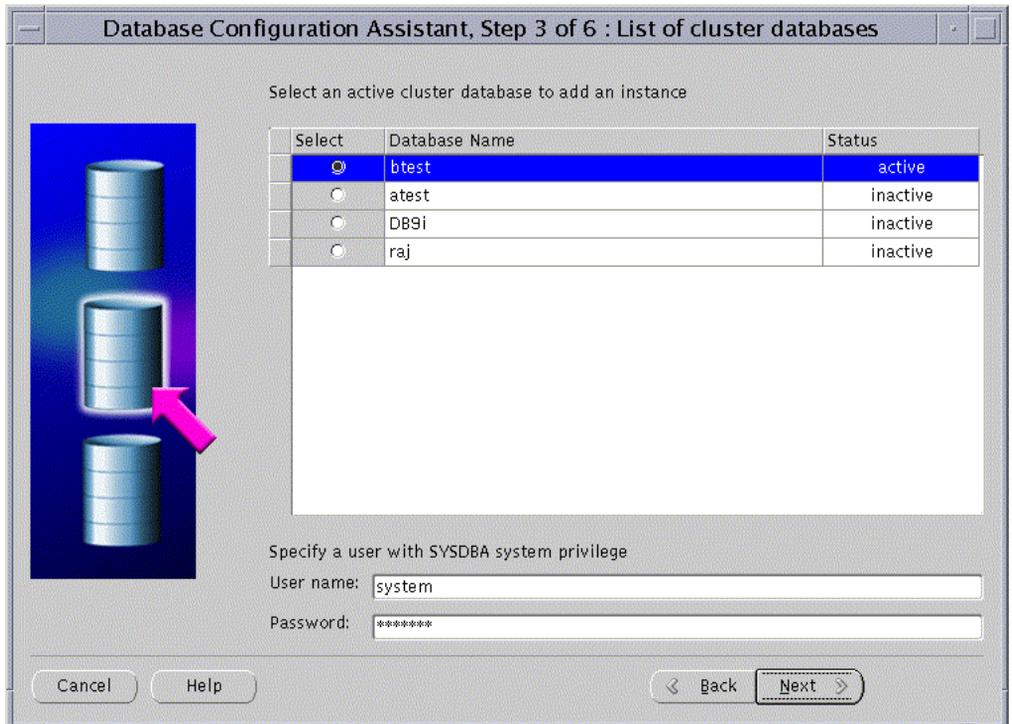
図 8-3 Database Configuration Assistant の「Instance Management」ページ



4. 「Add Instance」を選択して、「Next」をクリックします。

「Next」をクリックすると、[図 8-4](#) に示す「List of Cluster Databases」ページに、データベースとその現在のステータス（ACTIVE や INACTIVE など）が表示されます。

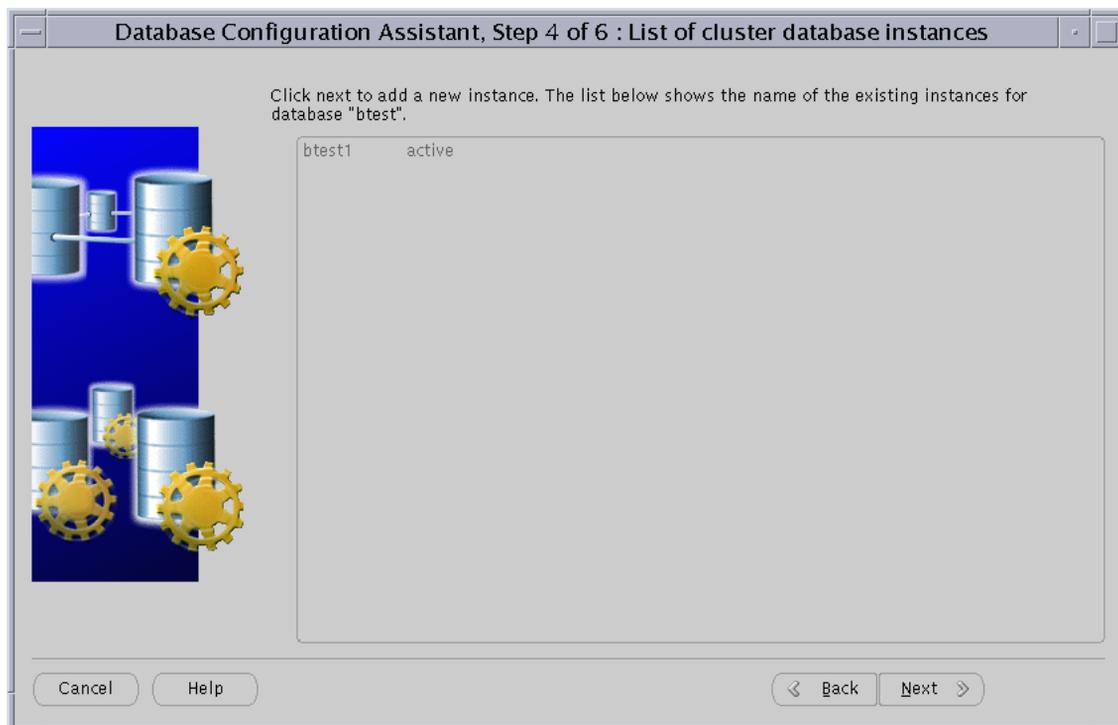
図 8-4 Database Configuration Assistant の「List of Cluster Databases」ページ



5. 「List of Cluster Databases」ページで、インスタンスを追加するアクティブな Real Application Clusters データベースを選択します。ユーザー ID がオペレーティング・システムで認証されていない場合は、DBCА でも **SYSDBA** 権限を持つデータベース・ユーザーのユーザー ID とパスワードの入力を求めるプロンプトが表示されます。このような DBCА のプロンプトが表示された場合は、有効なユーザー ID とパスワードを入力して、「Next」をクリックしてください。

「Next」をクリックすると、「List of Cluster Database Instances」ページが表示されます。[図 8-5](#) に示すとおり、選択した Real Application Clusters データベースのインスタンスの名前が表示されます。

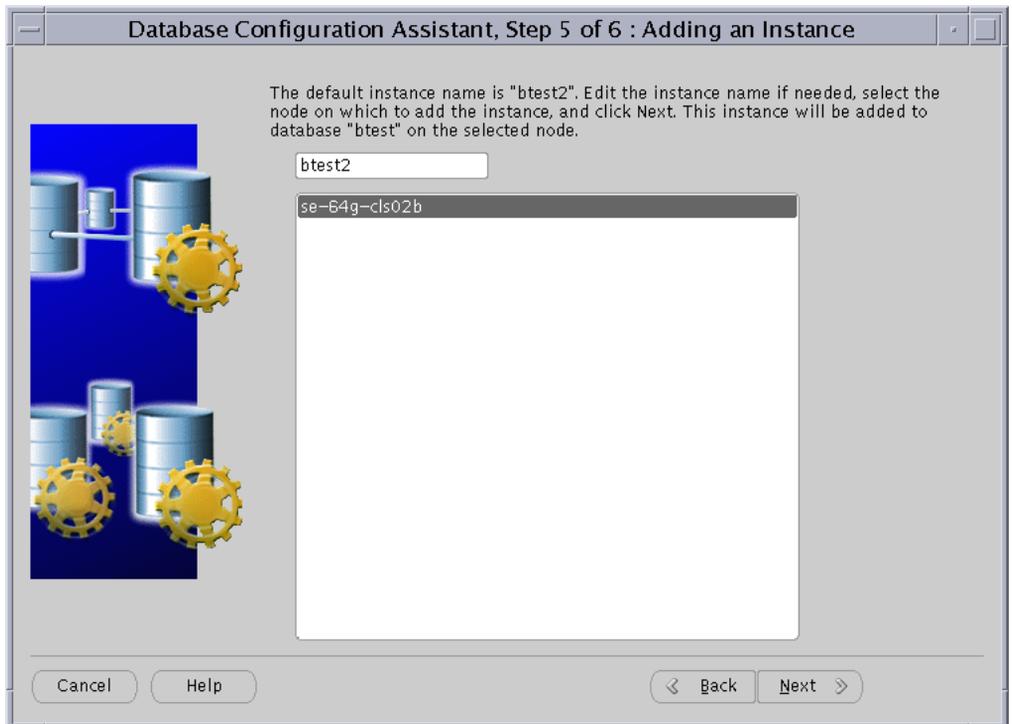
図 8-5 Database Configuration Assistant の「List of Cluster Database Instances」ページ



6. 「Next」をクリックします。

「Next」をクリックすると、図 8-6 に示す「Adding an Instance」ページが表示されます。

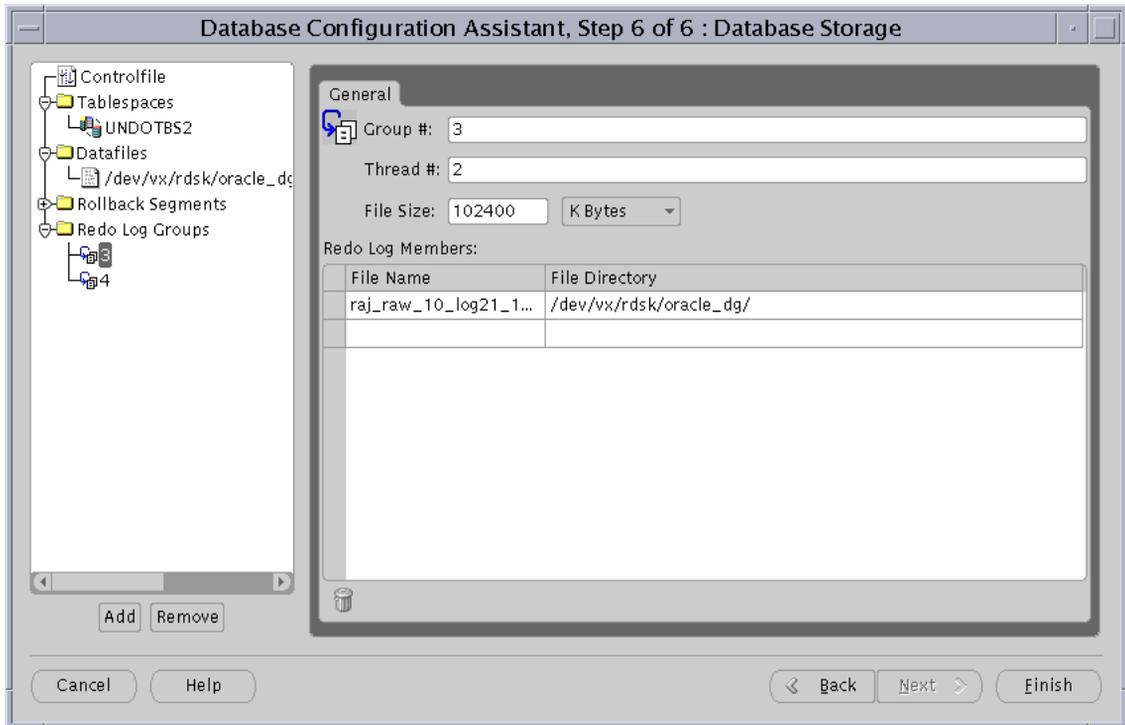
図 8-6 Database Configuration Assistant の「Adding an Instance」ページ



7. DBCA に表示されるインスタンス名が既存のインスタンス名順序に一致しない場合、このページの一番上のフィールドに**インスタンス名**を入力します。次に、リストから新しいノード名を選択し、「Next」をクリックします。

「Next」をクリックすると、図 8-7 に示す「Database Storage」ページが表示されます。

図 8-7 Database Configuration Assistant の「Database Storage」ページ



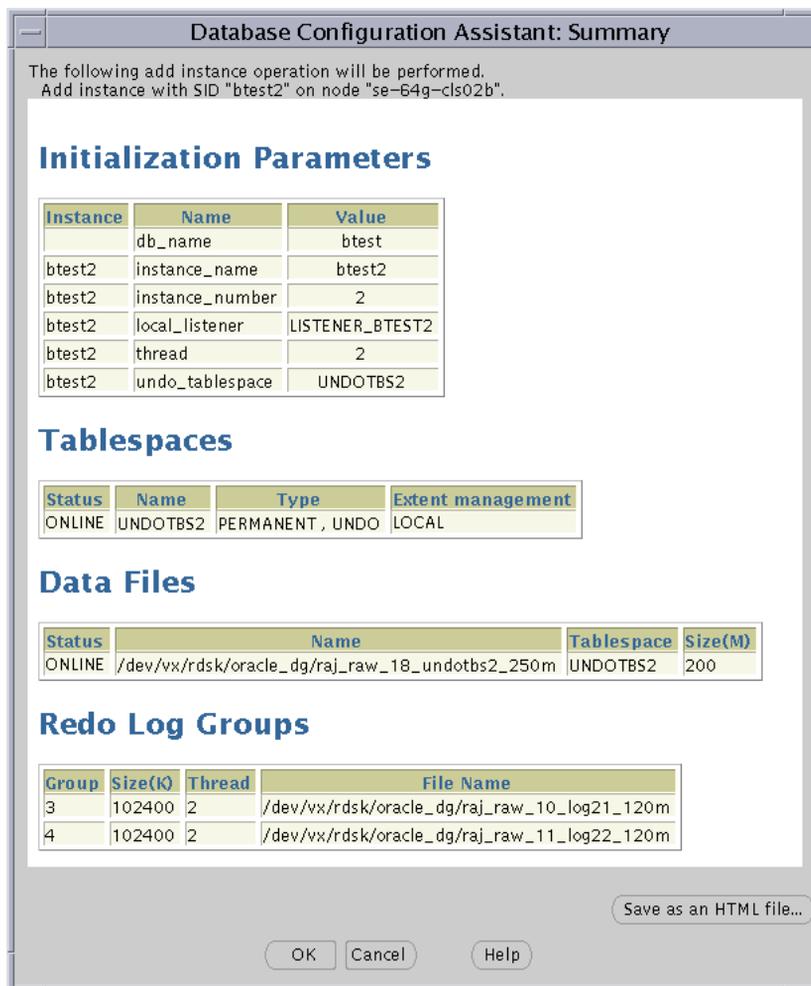
8. データベースで自動 UNDO 管理を使用しており、プラットフォームがクラスタ・ファイル・システムをサポートしている場合、この手順は省略できます。ただし、DBCA が提供するデフォルトの UNDO 表領域のデータ・ファイル名を変更する場合は、「**Tablespaces**」フォルダを選択して展開し、「undo tablespace storage object」を選択します。ダイアログ・ボックスが右側に表示されます。デフォルトのデータ・ファイル名を表領域用の RAW デバイス名に変更します。デフォルトの名前を使用しない場合は、名前をクラスタ・ファイル・システムのファイル名に変更することもできます。UNDO 表領域の記憶域プロパティは変更しないことをお勧めします。
9. データベースでロールバック UNDO 管理を使用しており、DBCA が示すデフォルトのロールバック・セグメント名を使用しない場合は、「**Rollback Segment**」フォルダを選択してロールバック・セグメント名を変更します。ロールバック・セグメントの記憶域プロパティは変更しないことをお勧めします。
10. プラットフォームがクラスタ・ファイル・システムをサポートしている場合、この手順は省略できます。デフォルトの REDO ログ・グループのファイル名を変更する場合は、「**Redo Log Groups**」フォルダを展開します。選択する REDO ログ・グループ番号ごとに、DBCA には別のダイアログ・ボックスが表示されます。UNIX プラットフォームの

場合は、8-3 ページの「RAW デバイスの作成 (オプション)」で作成した RAW デバイス名を「**File Name**」フィールドに入力します。Windows NT プラットフォームまたは Windows 2000 プラットフォームの場合は、8-7 ページの「ディスク・パーティションへのリンクの作成」で作成したシンボリック・リンクの名前を入力します。

11. 他のすべての REDO ログ・グループに対して前述の手順を繰り返して、「**Finish**」をクリックします。

「**Finish**」をクリックすると、[図 8-8](#) に示すような「**Summary**」ダイアログ・ボックスが表示されます。

図 8-8 Database Configuration Assistant の「Summary」ダイアログ・ボックス



12. ダイアログ・ボックスの情報を確認して「OK」をクリックするか、または「Cancel」をクリックしてインスタンスの追加操作を終了します。

「OK」をクリックすると、DBCA がインスタンスの追加操作を実行していることを示す進捗ダイアログ・ボックスが表示されます。この操作の中で DBCA は、インスタンスを追加し、そのインスタンスの Oracle Net 構成を作成します。DBCA によるインスタンス追加操作が終了すると、別の操作を実行するかどうかを尋ねるダイアログ・ボックスが表示されます。「No」をクリックすると DBCA が終了します。「Yes」をクリックすると、8-12 ページの図 8-3 に示す「Operations」画面が表示されます。

「No」をクリックしてインスタンスの追加操作を継続する場合は、この時点で、次の作業が完了しています。

- 新しいノード上での新しいデータベース・インスタンスの作成
- Windows NT または Windows 2000 用の新しいインスタンスの必須サービスの作成
- Oracle Net の構成
- Windows NT または Windows 2000 用の必須サービスの起動
- 新しいインスタンスおよびそのリスナーの起動

インスタンスの削除

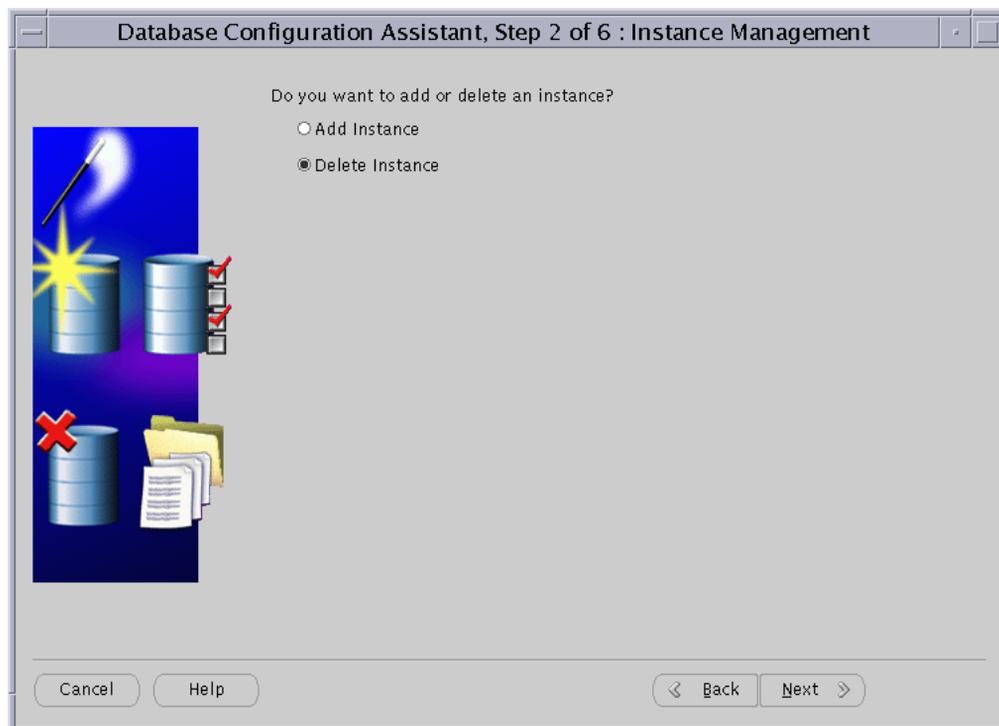
次の手順では、Database Configuration Assistant を使用してインスタンスを削除する方法について説明します。必要に応じてこの章の前半で示した図を参照してください。インスタンスを削除するには、次の手順を実行します。

1. 図 8-2 に示す DBCA の「Operations」ページに移動して「Instance Management」を選択し、「Next」をクリックします。

「Next」をクリックすると、図 8-9 に示す「Instance Management」ページが表示されます。

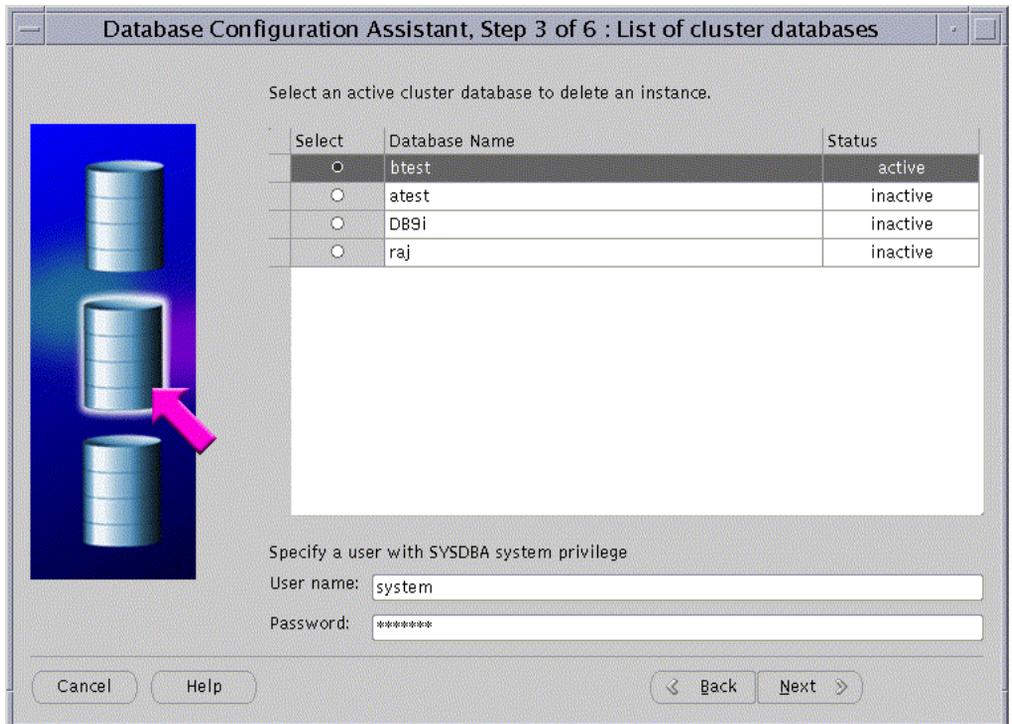
2. 「Delete Instance」を選択して、「Next」をクリックします。

図 8-9 DBCA の「Instance Management」ページ



「Next」をクリックすると、図 8-10 に示す「List of Cluster Databases」ページが表示されます。

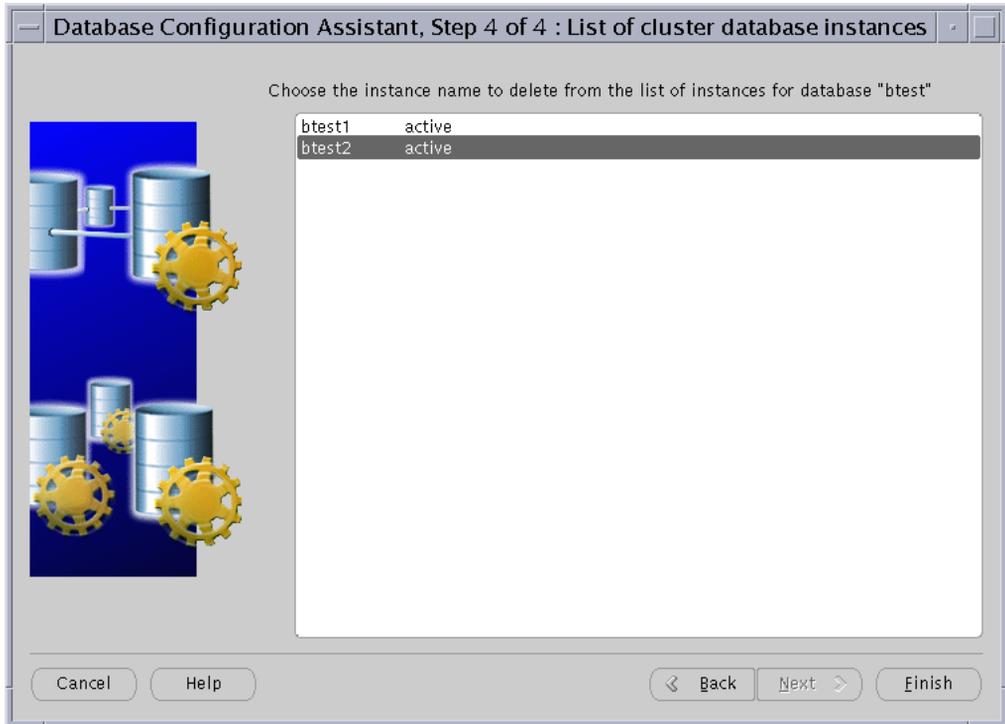
図 8-10 Database Configuration Assistant の「List of Cluster Databases」ページ



3. インスタンスを削除する Real Application Clusters データベースを選択します。ユーザー ID がオペレーティング・システムで認証されない場合、DBCA でも SYSDBA 権限を持つデータベース・ユーザーのユーザー ID およびパスワードの入力を求めるプロンプトが表示されます。このような DBCA のプロンプトが表示されたら、有効なユーザー ID およびパスワードを入力してください。「Next」をクリックします。

「Next」をクリックすると、図 8-11 に示す「List of Cluster Database Instances」ページが表示されます。この「List of Cluster Database Instances」ページには、選択した Real Application Clusters データベースに対応付けられたインスタンスと各インスタンスのステータスが表示されます。

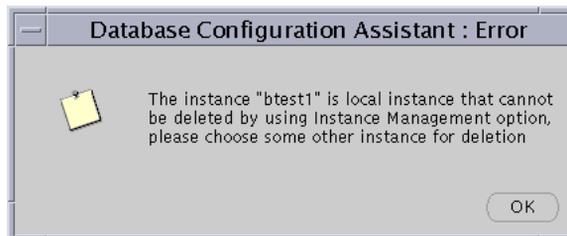
図 8-11 Database Configuration Assistant の「List of Cluster Database Instances」ページ



4. 削除するインスタンスを選択して、「Finish」をクリックします。

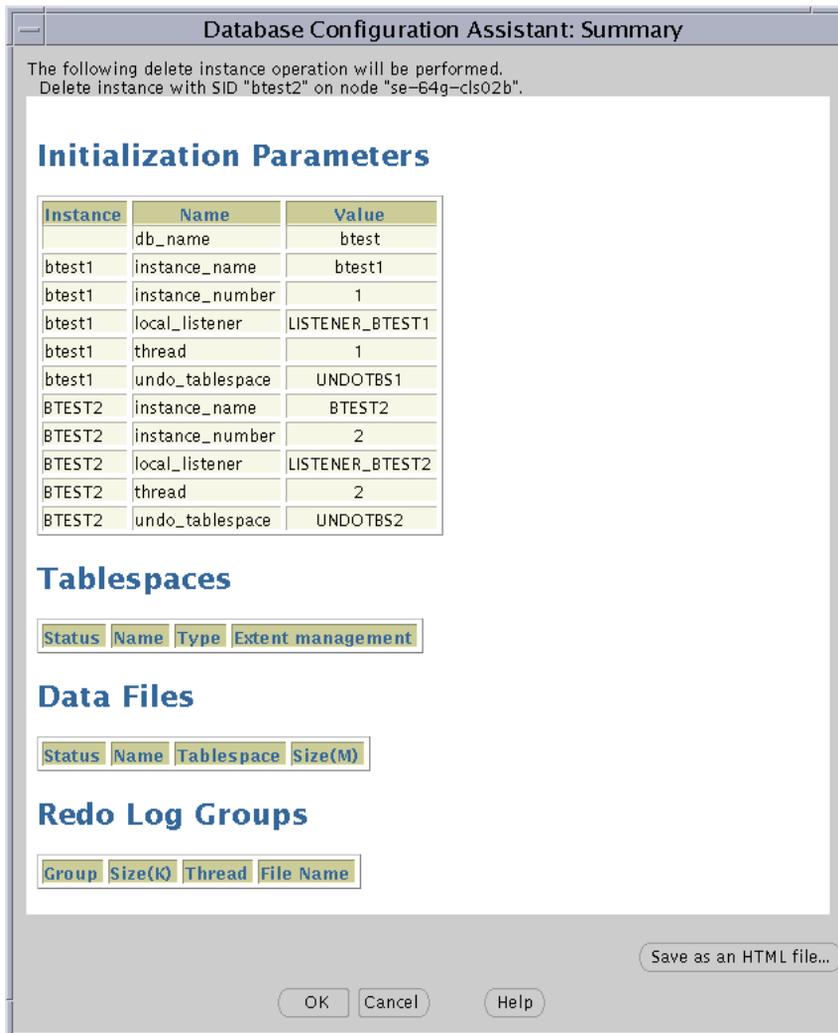
ローカル・インスタンスを選択すると、図 8-12 に示す「Error」ダイアログ・ボックスが表示されます。

図 8-12 Database Configuration Assistant の「Error」ダイアログ・ボックス



5. この「Error」ダイアログ・ボックスが表示された場合は、「OK」をクリックして「List of Cluster Database Instances」ページで別のインスタンスを選択し、「Finish」をクリックします。
6. 図 8-13 に示すような DBCA の「Summary」ダイアログ・ボックスが表示されます。

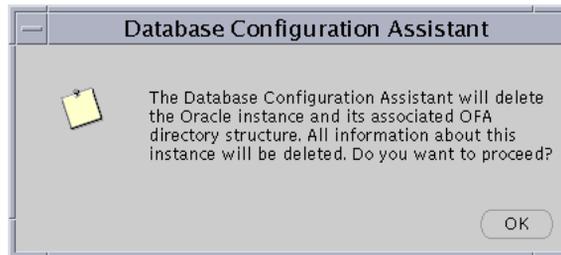
図 8-13 Database Configuration Assistant の「Summary」ダイアログ・ボックス



7. 「Summary」ダイアログ・ボックスで、インスタンスの削除操作に関する情報を確認して「OK」をクリックします。または、「Cancel」をクリックして、インスタンスの削除操作を終了します。

「OK」をクリックすると、[図 8-14](#) に示すような「Confirmation」ダイアログ・ボックスが表示されます。

図 8-14 Database Configuration Assistant の「Confirmation」ダイアログ・ボックス



8. 「Confirmation」ダイアログ・ボックスで「OK」をクリックして、インスタンスの削除操作を続行します。

「OK」をクリックすると、DBCA がインスタンスの削除操作を実行していることを示す進捗ダイアログ・ボックスが表示されます。この操作の中で DBCA は、インスタンスとそのインスタンスの Oracle Net 構成を削除します。DBCA がインスタンスの削除操作を終了すると、別の操作を実行するかどうかを尋ねるダイアログ・ボックスが表示されます。DBCA を終了する場合は「No」を、別の操作を実行する場合は「Yes」をクリックします。「Yes」をクリックすると、[8-12 ページの図 8-3](#) に示す「Operations」ページが表示されます。

この時点で、次の作業が完了しています。

- 選択されたインスタンスに対応付けられたリスナーの停止
- インスタンスの構成ノードからの選択されたデータベース・インスタンスの削除
- Windows NT または Windows 2000 用の選択されたインスタンスの必須サービスの削除
- Oracle Net 構成の削除
- インスタンスの構成ノードからの Oracle Flexible Architecture ディレクトリ構造の削除

第 V 部

リファレンス

第 V 部では、Real Application Clusters のリファレンス情報を示します。第 V 部に含まれる章は、次のとおりです。

- [付録 A 「トラブルシューティング」](#)
- [付録 B 「インスタンスとユーザーの、空きリストと空きリスト・グループ（オプション）への対応付け」](#)
- [付録 C 「Real Application Clusters 管理ツールのエラー・メッセージ」](#)
- [用語集](#)

トラブルシューティング

この付録では、トレース・ファイルの管理方法およびオラクル社カスタマ・サポート・センターへの連絡方法について説明します。内容は次のとおりです。

- [トレース・ファイルの使用](#)
- [オラクル社カスタマ・サポート・センターへの連絡方法](#)

関連項目： [Oracle Enterprise Manager](#) のサービス検出障害のトラブルシューティングについては、『[Oracle9i Real Application Clusters セットアップおよび構成](#)』を参照してください。

トレース・ファイルの使用

この項では、トレース・ファイルについて説明します。内容は次のとおりです。

- バックグラウンド・スレッド・トレース・ファイル
- ユーザー・スレッド・トレース・ファイル
- アラート・ファイル
- エラー・コール・トレース・スタック

バックグラウンド・スレッド・トレース・ファイル

Real Application Clusters のバックグラウンド・スレッドは、トレース・ファイルを使用して、データベース処理のエラー、状態変化および例外を記録します。これらの詳細なトレース・ログによって、オラクル社カスタマ・サポート・センターは、ユーザーの**クラスタ・データベース**構成の問題を効果的にデバッグできます。バックグラウンド・スレッドのトレース・ファイルは、**initdbname.ora 初期化パラメータ・ファイル**で `BACKGROUND_DUMP_DEST` パラメータが設定されているかどうかにかかわらず作成されます。`BACKGROUND_DUMP_DEST` を設定した場合、トレース・ファイルは指定されたディレクトリに格納されます。パラメータを設定しない場合、トレース・ファイルは次に示すディレクトリに格納されます。

- UNIX オペレーティング・システムの場合：`$ORACLE_HOME/rdbms/log`
- Windows NT および Windows 2000 オペレーティング・システムの場合：
`%ORACLE_HOME%\RDBMS\TRACE`

Oracle データベースでは、バックグラウンド・スレッドごとに異なるトレース・ファイルが作成されます。トレース・ファイルの名前は、バックグラウンド・スレッド名の後に拡張子 `.trc` を付けたものです。次に例を示します。

- `sidbwr.trc`
- `sidsmon.trc`

トレース情報は、[表 A-1](#) に示すトレース・ファイルにレポートされます。

表 A-1 バックグラウンド・スレッド・トレース・ファイル

トレース・ファイル	説明
<code>sidlmsn.trc</code>	グローバル・キャッシュ・サービス・プロセス (LMSn) のトレース・ファイルです。他のバックグラウンド・プロセスに対するロック要求を示します。
<code>sidlmdn.trc</code>	LMD n プロセスのトレース・ファイルです。ロック要求を示します。

表 A-1 バックグラウンド・スレッド・トレース・ファイル (続き)

トレース・ファイル	説明
<i>sidlmon.trc</i>	LMON プロセスのトレース・ファイルです。クラスタの状態を示します。
<i>sidp00n.trc</i>	パラレル実行プロセスのトレース・ファイルです。

ユーザー・スレッド・トレース・ファイル

初期化パラメータ・ファイルで `USER_DUMP_DEST` パラメータを設定すると、ユーザー・スレッドに対してもトレース・ファイルは作成されます。ユーザー・スレッド・トレース・ファイルの形式は、`oraxxxx.trc` です。xxxx は 5 桁の数字で、UNIX の場合はプロセス識別子 (PID)、Windows NT および Windows 2000 の場合は **スレッド番号** を示します。

アラート・ファイル

アラート・ファイル (`sidalrt.log`) には、データベース処理中に発生したエラー・メッセージと例外についての重要な情報が含まれています。各 **インスタンス** にはアラート・ファイルが 1 つずつあり、インスタンスを起動するたびに、ファイルに情報が追加されます。すべての処理スレッドは、アラート・ファイルに書き込むことができます。

`sidalrt.log` ファイルは、`initdb_name.ora` 初期化パラメータ・ファイルの `BACKGROUND_DUMP_DEST` パラメータで指定するディレクトリに作成されます。`BACKGROUND_DUMP_DEST` パラメータを設定しない場合、`sidalrt.log` ファイルは次に示すディレクトリに作成されます。

- UNIX オペレーティング・システムの場合: `$ORACLE_HOME/rdbms/log`
- Windows NT および Windows 2000 オペレーティング・システムの場合: `%ORACLE_HOME%\rdbms\trace`

エラー・コール・トレース・スタック

オラクル社カスタマ・サポート・センターでは、特定のトレース・ファイルに対するエラー・コール・トレース・スタックを作成するようにお薦めする場合があります。エラー・コール・トレース・スタックは、データベースの特定のバックグラウンド・スレッドまたはユーザー・スレッドのプログラム・トレースを提供します。

エラー・コール・トレース・スタックを作成するには、次の手順を実行します。

1. バックグラウンド・プロセスの Oracle プロセス ID を取得します。

```
SELECT pid "Oracle Process Id",
       name
   from v$PROCESS, V$BGPROCESS
  where V$PROCESS.addr = V$BGPROCESS.paddr;
```

表示される出力結果は次のようになります。

```
Oracle Pro NAME
-----
      2 PMON
      3 LMON
      4 DBW0
      5 LGWR
      6 CKPT
      7 SMON
      8 RECO
      9 SNP0
     10 SNP1
     11 GCS0
```

2. トレース・スタックをトレース・ファイルにダンプします。たとえば、LMON のトレース・スタックをダンプするには、次の手順を実行します。
 - a. Oracle プロセス ID を LMON に設定します。この例では、プロセス ID は 3 です。


```
ORADEBUG setorapid 3
```
 - b. エラー・スタックを *sidlmon.trc* にダンプします。


```
ORADEBUG dump errorstack 3
```

オラクル社カスタマ・サポート・センターへの連絡方法

この付録で説明した処理を行っても問題を解決できない場合は、Metalink で TAR を発行して、オラクル社カスタマ・サポート・センターにエラーを連絡してください。連絡する際は、次の情報を用意してください。

- クラスタ・ハードウェアの仕様（たとえば、Dell PowerEdge 6100 サーバーの 2 ノード・クラスタ）
- オペレーティング・システムのバージョン
- Oracle RDBMS の 5 桁のリリース番号（たとえば、9.2.0.0.1）
- Oracle9i Real Application Clusters データベースの 5 桁のリリース番号
- HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\OSD キーの内容（Windows NT および Windows 2000 の場合）
- ベンダー提供のクラスタ OSD アップグレード製品に関する情報
- 正常に実行されなかった特定の操作に関する情報（たとえば、データベースの起動や問合せ）
- 問題を再現するための一連の手順

重大なエラー

ORA-00600 エラーが発生した場合、`sidalrt.log` ファイルに示されます。ORA-00600 エラーまたは他の重大なエラーが `sidalrt.log` ファイルに示された場合、次のディレクトリ内のすべてのファイルを提供してください。

- UNIX オペレーティング・システムの場合：`$ORACLE_HOME/admin/db_name/bdump`
- Windows NT および Windows 2000 オペレーティング・システムの場合：
`%ORACLE_HOME%\admin\%db_name%\bdump`

インスタンスとユーザーの、空きリストと 空きリスト・グループ（オプション）への 対応付け

この付録では、インスタンスとユーザーを、空きリストと空きリスト・グループに対応付ける方法について説明します。SQL 固有の空きリスト・オプションおよびエクステン트의空きリスト・グループへの事前割当てについても説明します。この付録の情報は、自動セグメント領域管理が使用できない場合にのみ利用してください。内容は次のとおりです。

- インスタンスと空きリストの対応付け
- ユーザー・プロセスの、空きリストと空きリスト・グループへの対応付け
- 空きリストのある空き領域を管理するための SQL オプション
- 空きリスト・グループへのエクステン트의事前割当て

注意： オラクル社では自動セグメント領域管理をお勧めします。空きリストと空きリスト・グループの使用はお勧めしません。

インスタンスと空きリストの対応付け

通常、すべての表には同じ数の**空きリスト・グループ**が必要ですが、1つのグループ内の空きリストの数は、各表のアクティビティの種類および量によって変わります。1つの空きリスト・グループのブロックを処理する**キャッシュ・フュージョン**・リソースは、主に、その空きリスト・グループを使用するインスタンスで保持される傾向があります。これは、データを変更するインスタンスは、通常、他のインスタンスよりそのデータを再使用する可能性が高いためです。複数のインスタンスが同じエクステントから空き領域を取得する場合は、それらのインスタンスで挿入したデータが後で変更されると、エクステントのブロックに競合が発生しやすくなります。一方、自動セグメント領域管理では、Real Application Clustersにおけるこのようなパフォーマンス上の問題は排除されます。

既存の空きリスト・グループへの新しいインスタンスの割当て

MAXINSTANCES が表またはクラスタ内の空きリスト・グループの数を超える場合、**インスタンス番号**は、次の計算式に対応付けられる空きリスト・グループにマップされます。

$$\text{instance_number modulo number_of_free_list_groups}$$

modulo (または剰余を表す *rem*) は、剰余値を計算して、使用する空きリスト・グループを決定するための計算式です。たとえば、2つの空きリスト・グループと10のインスタンスがあるとします。インスタンス6が使用する空きリスト・グループを判断するには、 $6 \bmod 2 = 0$ という計算式を使用します。6を2で割ると3で、剰余は0 (ゼロ) になります。したがって、インスタンス6は空きリスト・グループ0 (ゼロ) を使用します。同様に、インスタンス5の場合、 $5 \bmod 2 = 1$ であるため、空きリスト・グループ1を使用します。5は剰余を1として2で割り切れません。

空きリスト・グループの数がMAXINSTANCESを超える場合、別のハッシング・メカニズムが使用されます。複数インスタンスが1つの空きリスト・グループを共有する場合、それらのインスタンスは、その空きリスト・グループを共有しているインスタンスに対して限定的に割り当てられた、すべてのエクステントへのアクセスを共有します。

FREELIST GROUPS および MAXINSTANCES

通常、比較的少ない数の**ノード**を含むシステムでは、表に対するFREELIST GROUPS オプションは、CREATE DATABASE のMAXINSTANCES オプションと同じ値になります。MAXINSTANCES オプションは、データベースに同時アクセスできるインスタンスの数を制限します。ただし、大規模パラレル処理システムでは、MAXINSTANCES の値がFREELIST GROUPS の数倍になり得るため、1つの空きリスト・グループを多数のインスタンス間で共有することになります。

関連項目： インスタンスおよびユーザーの、空きリスト・グループへの対応付けの詳細については、『Oracle9i Real Application Clusters 配置およびパフォーマンス』を参照してください。

ユーザー・プロセスの、空きリストと空きリスト・グループへの対応付け

ユーザー・プロセスは、Oracle プロセス ID に基づいて空きリストに対応付けされます。各ユーザー・プロセスは、そのプロセスが実行されているインスタンスの空きリスト・グループ内にある、1つの空きリストのみにアクセスできます。すべてのユーザー・プロセスは、空きブロックの**マスター空きリスト**にもアクセスできます。1つの表に複数の空きリストがあっても複数の空きリスト・グループがない場合、または空きリスト・グループの数がインスタンスの数より少ない場合、空きリストおよび空きリスト・グループは、異なるインスタンスのユーザー・プロセス間で共有されます。

ALTER SESSION INSTANCE_NUMBER 文を使用して、インスタンス番号の値に MAXINSTANCES より大きい値を指定できます。また、ALTER TABLE 文を使用して、表の空きリストの割当てを動的に変更できます。ただし、空きリスト・グループにはこの文を使用できません。

空きリストのある空き領域を管理するための SQL オプション

いくつかの SQL オプションでは、表、クラスタおよび索引用の空きリストおよび空きリスト・グループを作成できます。オブジェクト用の新しい領域は、特定のデータ・ファイルから取得するように明示的に指定できます。空き領域を特定の空きリスト・グループと対応付けることもできます。この空きリスト・グループは、後で特定のインスタンスと対応付けることができます。

使用できる SQL 文は次のとおりです。

```
CREATE [TABLE | CLUSTER | INDEX]
    STORAGE
    FREELISTS
    FREELIST GROUPS
ALTER [TABLE | CLUSTER | INDEX]
    ALLOCATE EXTENT
    SIZE
    DATAFILE
    INSTANCE
```

これらの SQL オプションに INSTANCE_NUMBER パラメータを指定すると、データ・ブロックをインスタンスと対応付けることができます。

関連項目： これらの文の完全な構文については、『Oracle9i SQL リファレンス』を参照してください。

空きリスト・グループへのエクステントの事前割当て

エクステントの事前割当ては、Oracle によるエクステントの自動割当てを回避するための静的な方法です。空きリスト・グループを含む表にエクステントを事前に割り当てることができます。これは、エクステントを事前に割り当てるインスタンスの空きリスト・グループ内に常駐する空きリストに、すべての空きブロックがフォーマットされることを意味します。この方法は、サイズの拡張が予想されるオブジェクトを保存する必要がある場合に有効です。

Real Application Clusters 管理ツールの エラー・メッセージ

この付録では、Real Application Clusters 管理ツールのエラー・メッセージについて説明します。エラー・メッセージのグループをアルファベット順に示します。次のエラー・メッセージ・グループがあります。

- グローバル・サービス・デーモンのエラー・メッセージ (PRKA)
- クラスタ・コマンドのエラー・メッセージ (PRKC)
- Windows NT の Cluster Setup エラー・メッセージ (PRKI)
- サーバー制御ユーティリティ (SRVCTL) のエラー・メッセージ (PRKO)
- クラスタ・データベース管理のエラー・メッセージ (PRKP)
- リポジトリのエラー・メッセージ (PRKR)

関連項目： Windows ベースのエラー・メッセージについては『Oracle9i Database for Windows 管理者ガイド』を、その他のエラー・メッセージについては『Oracle9i データベース・エラー・メッセージ』を参照してください。

グローバル・サービス・デーモンのエラー・メッセージ (PRKA)

PRKA-2001: GSD はすでに存在します

原因: グローバル・サービス・デーモン (GSD) ・アプリケーションがすでに実行中であるのに、GSD アプリケーションを作成しようとしました。

処置: GSD アプリケーションがすでに存在している場合、GSD アプリケーションは作成できません。実行中の GSD を停止してから、'srvctl add nodeapps' を使用して GSD アプリケーションを再作成してください。既存の GSD アプリケーションを使用することもできます。

PRKA-2002: リスナーはすでに存在します

原因: リスナー・アプリケーションがすでに実行中であるのに、リスナー・アプリケーションを作成しようとしました。

処置: リスナー・アプリケーションがすでに存在している場合、リスナー・アプリケーションは作成できません。実行中のリスナーを停止してから、'srvctl add nodeapps' を使用してリスナー・アプリケーションを再作成してください。既存のリスナー・アプリケーションを使用することもできます。

PRKA-2003: エージェントはすでに存在します

原因: エージェント・アプリケーションがすでに実行中であるのに、エージェント・アプリケーションを作成しようとしました。

処置: エージェント・アプリケーションがすでに存在している場合、エージェント・アプリケーションは作成できません。実行中のエージェントを停止してから、'srvctl add nodeapps' を使用してエージェント・アプリケーションを再作成してください。既存のエージェント・アプリケーションを使用することもできます。

PRKA-2004: VIP はすでに存在します

原因: 仮想 IP アプリケーションがすでに実行中であるのに、仮想 IP アプリケーションを作成しようとしました。

処置: 仮想 IP アプリケーションがすでに存在している場合、仮想 IP アプリケーションは作成できません。実行中の仮想 IP アプリケーションを停止してから、'srvctl add nodeapps' を使用して仮想 IP アプリケーションを再作成してください。既存の仮想 IP アプリケーションを使用することもできます。

PRKA-2006: ネットマスクが異なります

原因: 'srvctl add vip_range' の実行でクラスタに指定した VIP アドレスの範囲が無効です。

処置: 追加する VIP 範囲の開始と終了のアドレスが同じネットマスクに属しているかどうかをチェックしてください。

PRKA-2007: 使用可能な空き VIP はありません

原因: クラスタ・データベースに追加した仮想 IP アドレスのリストに、使用可能な空き仮想 IP がありません。追加した仮想 IP アドレスはすべて使用中です。

処置: 'srvctl add vip_range' を使用して、クラスタ・データベースに使用する新規の仮想 IP アドレスを追加してください。

PRKA-2008: VIP は見つかりません

原因: クラスタ・データベースに追加した仮想 IP 範囲に、仮想 IP アドレスはありません。

処置: 'srvctl add vip_range' を使用して、新規の仮想 IP アドレスをクラスタに追加してください。

PRKA-2009: リモート・コマンドが失敗しました

原因: ローカル・ノードまたはリモート・ノードのグローバル・サービス・デーモンが操作を実行しようとしてエラーが発生しました。

処置: 'srvctl config' を実行して、グローバル・サービス・デーモンが起動中で応答するかどうかをチェックしてください。

PRKA-2010: OCR からの VIP Range の構成情報の取得に失敗しました

原因: 構成リポジトリから仮想 IP 範囲の構成を取得しようとしてエラーが発生しました。グローバル・サービス・デーモンが起動していないか、または構成リポジトリがアクセス可能ではありません。リポジトリが破損しているか、またはリポジトリに対する権限が不十分な場合は、構成リポジトリにアクセスできない可能性があります。

処置: 'srvctl config' を実行して、グローバル・サービス・デーモンが起動中で構成リポジトリがアクセス可能であるかどうかをチェックしてください。

PRKA-2011: OCR への VIP Range の構成情報の追加に失敗しました

原因: 構成リポジトリに仮想 IP 範囲の構成を追加しようとしてエラーが発生しました。グローバル・サービス・デーモンが起動していないか、または構成リポジトリがアクセス可能ではありません。リポジトリが破損しているか、またはリポジトリに対する権限が不十分な場合は、構成リポジトリにアクセスできない可能性があります。

処置: 'srvctl config' を実行して、グローバル・サービス・デーモンが起動中で構成リポジトリがアクセス可能であるかどうかをチェックしてください。

クラスタ・コマンドのエラー・メッセージ (PRKC)

PRKC-1000 クラスタ内にアクティブなノードが検出できません

原因: クラスタのアクティブなノードのリストを取得できませんでした。このエラーは、オペレーティング・システム依存のクラスタウェアが、クラスタ内の1つ以上のノードで適切に機能していなかった場合、またはクラスタウェアがマシンにインストールされていない場合に発生する可能性があります。

処置: ORACLE_HOME/bin から 'lsnodes' バイナリを実行してクラスタの状態をチェックし、ノード・リストが正しく出力されるかどうかを確認してください。ベンダーのマニュアルを参照し、オペレーティング・システム依存のクラスタウェアの現在の状態に関する補足情報を入手してください。

PRKC-1001 バッファ内のコマンドを送信中にエラーが発生しました

原因: 内部エラーが発生しました。

処置: オラクル社カスタマ・サポート・センターに連絡してください。

PRKC-1002 送信されたコマンドで正常に実行されなかったコマンドがあります

原因: このエラーは、管理操作中にクラスタ内の一部のノードに障害が発生した場合、またはノード間の通信チャンネルに障害が発生した場合に発生する可能性があります。

処置: すべてのノードが設定され、ノード間の通信が機能していることをチェックしてください。

PRKC-1004 ノードへのファイルのコピーで問題が発生しました

原因: ローカル・ノードから、クラスタ内の1つ以上のノードへのファイルのコピー操作を試行して失敗しました。このエラーは、操作中に1つ以上のノードに障害が発生した場合、接続先ディレクトリに1つ以上のノードのユーザーに対する書き込み権限がなかった場合、ソース・ファイルが存在しなかった場合、あるいは1つ以上のノードに対する UNIX プラットフォームでの 'rcp' コマンドに失敗した場合に発生する可能性があります。

処置: ソース・ファイルが存在していることをチェックしてください。クラスタ内のすべてのノードが起動していることをチェックしてください。接続先ディレクトリにユーザーに対する書き込み権限があることをチェックしてください。UNIX ベースのプラットフォームの場合は、すべてのノードについて、ユーザーに 'rcp' の実行権限があることをチェックしてください。

PRKC-1005 ノードからのファイルの削除で問題が発生しました

原因: ローカル・ノードから、クラスタ内の1つ以上のノードへのファイルの削除操作を試行して失敗しました。このエラーは、操作中に1つ以上のノードに障害が発生した場合、接続先ディレクトリに1つ以上のノードのユーザーに対する書き込み権限がなかった場合、ソース・ファイルが存在しなかった場合、あるいは1つ以上のノードに対する UNIX プラットフォームでの 'rcp' コマンドに失敗した場合に発生する可能性があります。

処置: 詳細については、前述のエラー・メッセージを参照してください。

PRKC-1006 ノードへのファイルの移動で問題が発生しました

原因: ローカル・ノードから、クラスタ内の1つ以上のノードへのファイルの移動操作を試行して失敗しました。このエラーは、操作中に1つ以上のノードに障害が発生した場合、接続先ディレクトリに1つ以上のノードのユーザーに対する書き込み権限がなかった場合、ソース・ファイルが存在しなかった場合、あるいは1つ以上のノードに対するUNIXプラットフォームでの 'rcp' コマンドに失敗した場合に発生する可能性があります。

処置: 詳細については、前述のエラー・メッセージを参照してください。

PRKC-1007 ノードへのディレクトリの作成で問題が発生しました

原因: ローカル・ノードからクラスタ内の1つ以上のノードへのディレクトリの作成操作を試行して失敗しました。このエラーは、操作中に1つ以上のノードに障害が発生した場合、接続先ディレクトリに1つ以上のノードのユーザーに対する書き込み権限がなかった場合、ソース・ファイルが存在しなかった場合、あるいは1つ以上のノードに対するUNIXプラットフォームでの 'rcp' コマンドに失敗した場合に発生する可能性があります。

処置: 詳細については、前述のエラー・メッセージを参照してください。

PRKC-1009 すべてのノードでサービスの開始に失敗しました

原因: クラスタ内のすべてのノードでのサービスの開始、および1つ以上のサービスの開始が正常に行われませんでした。

処置: クラスタ内のすべてのノードが起動しているかどうかをチェックしてください。

PRKC-1010 すべてのノードでサービスの停止に失敗しました

原因: クラスタ内のすべてのノードでのサービスの停止、および1つ以上のサービスの停止が正常に行われませんでした。

処置: 前述のエラー・メッセージを参照してください。

PRKC-1011 すべてのノードからのサービスの削除に失敗しました

原因: クラスタ内のすべてのノードでのサービスの削除、および1つ以上のサービスの削除が正常に行われませんでした。

処置: 前述のエラー・メッセージを参照してください。

PRKC-1012 すべてのノードでサービスの作成に失敗しました

原因: クラスタ内のすべてのノードでのサービスの作成、および1つ以上のサービスの作成が正常に行われませんでした。

処置: 前述のエラー・メッセージを参照してください。

PRKC-1016 環境変数の値の取得で問題が発生しました

原因: 未定義の環境変数の値を取得しようとしました。

処置: 環境変数を定義してください。

PRKC-1017 GSD の再起動に問題が発生しました

原因: ローカル・ノードでグローバル・サービス・デーモン (GSD) を起動できませんでした。このエラーは、オペレーティング・システム依存のクラスタウェアが適切に機能していない場合、またはすでに実行中の GSD がある場合に発生する可能性があります。

処置: ORACLE_HOME/bin から 'lsnodes' を実行してクラスタウェアの状態をチェックし、ベンダーのマニュアルに従ってください。これによって、クラスタ内のノードがリストされます。'gsdctl stop' を実行してから、'gsdctl start' を再度実行します。

PRKC-1018 コーディネータ・ノードの取得中にエラーが発生しました

原因: このエラーは、オペレーティング・システム依存のクラスタウェアが適切に機能していなかった場合に発生する可能性があります。

処置: 各ノードで 'gsdctl stop' を使用して GSD を停止します。クラスタ内の各ノードで 'gsdctl start' を使用します。

PRKC-1019 ノード {0} のデーモンへのハンドルの作成中にエラーが発生しました

原因: ノードでグローバル・サービス・デーモンが実行されていません。

処置: 'gsdctl stat' を使用してデーモンの状態をチェックしてください。'gsdctl start' を使用して GSD を起動してください。

PRKC-1020 リモート・ノード {0} でのオペレーションの実行中に例外が発生しました

原因: リモート・ノード {0} でグローバル・サービス・デーモンが起動されていないときに、リモート・ノードで操作を実行しようとした。

処置: 'gsdctl start' を使用してリモート・ノードでグローバル・サービス・デーモンを起動してください。

PRKC-1021 クラスタウェアで問題が発生しました

原因: オペレーティング・システム依存のクラスタウェアに接続できません。このエラーは、オペレーティング・システム依存のクラスタウェアが適切に機能していなかったために発生する可能性があります。

処置: 'lsnodes' を使用してオペレーティング・システム依存のクラスタウェアの状態をチェックし、ベンダーのマニュアルに従ってください。

PRKC-1022 {1} でノード {0} の "ノード名" を取得できませんでした

原因: オペレーティング・システム依存のクラスタウェアに接続できません。このエラーは、オペレーティング・システム依存のクラスタウェアが適切に機能していなかったために発生する可能性があります。

処置: 'lsnodes' を使用してオペレーティング・システム依存のクラスタウェアの状態をチェックし、ベンダーのマニュアルに従ってください。

Windows NT の Cluster Setup エラー・メッセージ (PRKI)

PRKI-2059 ノードにファイルをコピーできません

原因: 宛先ファイルを作成したり、ファイルを宛先位置にコピーする権限が不十分です。

処置: 宛先ファイルを作成する権限を確認してください。また、宛先ファイルが現在使用中でないことも確認してください。

PRKI-2119 {0} のファイルを削除できません。ファイルを削除できない場合、インストールが成功しない場合があります。

原因: 削除しようとしたファイルが存在しないか、または現在使用中です。

処置: 削除しようとしたファイルが存在しない場合、このエラーは無視してください。削除するファイルが現在使用中の場合は、ファイルを使用している処理が終了してから再試行してください。

PRKI-2060 ノードにサービスを作成できません

原因: サービスを作成する権限が不十分か、またはサービスがすでに存在しています。

処置: 宛先ノードでサービスを作成する権限を確認してください。また、サービスが存在していないことも確認してください。

PRKI-2061 ノードでサービスを開始できません

原因: サービスが未作成、実行中、削除マークが設定されている、あるいはサービスを適切に開始できない、などが原因です。

処置: サービス制御パネルからサービスを手動で起動し、表示された Windows のエラー・メッセージをチェックしてください。

PRKI-2064 ノードでレジストリ・エントリを更新できません

原因: ノードのレジストリ・エントリを変更する権限が不十分か、またはレジストリの更新でエラーが発生しました。

処置: レジストリ・エントリを変更する権限を確認してください。

PRKI-2066 Oracle ディスク・パーティションが見つかりません。ウィザードを終了し、Oracle パーティションを作成してから再試行してください。

原因: Oracle ディスク・パーティションが存在しないか、または OLM DLL のロードに失敗しました。

処置: OLM DLL (oobjlib.dll) がロード・パスにあることを確認してください。ディスク・アドミニストレータに移動してパーティションを検索し、Oracle ディスク・パーティションの存在も確認してください。

PRKI-2114 ノードのハードウェア情報で収集および検証できないものがあります。「強制終了」を押してウィザードを終了するか、または無視して続行してください。

原因: ノードのクラスタへの追加で、リモート・ノードへの接続に失敗しました。VIA が選択されていた場合は、VIA 情報の収集に失敗した可能性もあります。

処置: ノードを既存のクラスタに追加しようとしている場合は、「net use」などのコマンドを使用して) ドライブ文字を各リモート・ノードのドライブにマップしてください。VIA を選択していた場合は、VIA 情報を収集するリモート・ノードで一時サービスを起動する権限があることも確認してください。

PRKI-2116 ノード {0} のリモート・ドライブに接続できません

原因: リモート・ノードへの接続に失敗しました。

処置: 「net use」などのコマンドを使用して、ローカル・ノードのドライブ文字をリモート・ノードにマップしてください。

PRKI-2120 {1} のサービス {0} を削除できません。サービスを削除できない場合、インストーラが成功しない場合があります。

原因: サービスが存在しないか、またはサービスがすでに削除されていた可能性があります。

処置: サービス制御パネルをチェックして、サービスの存在を調べてください。サービスが存在しない場合は、このエラーを無視してください。サービスが存在する場合は、「net」や「sc」などのユーティリティを使用して、そのサービスをコマンドラインから削除してください。削除に失敗した場合は、このコマンドによって戻された Windows のエラー・メッセージをチェックしてください。

サーバー制御ユーティリティ (SRVCTL) のエラー・メッセージ (PRKO)

PRKO-2001: コマンドライン構文が無効です。

原因: 無効な SRVCTL コマンドラインが入力されました。

処置: -h SRVCTL コマンドライン・オプションを使用して正しいコマンドライン構文を確認し、コマンドを再入力してください。

PRKO-2002: コマンドライン・オプションが無効です:

原因: 無効な SRVCTL コマンドライン・オプションが入力されました。

処置: -h SRVCTL コマンドライン・オプションを使用して正しいコマンドライン構文を確認し、コマンドを再入力してください。

PRKO-2003: コマンドラインのオプション値が無効です:

原因: 無効な SRVCTL コマンドライン・オプション値が入力されました。

処置: -h SRVCTL コマンドライン・オプションを使用して正しいコマンドライン構文を確認し、コマンドを再入力してください。

PRKO-2004: コマンドライン・オプションが重複しています:

原因: 重複する SRVCTL コマンドライン・オプションが入力されました。

処置: 重複を消去してコマンドを再入力してください。

PRKO-2005: アプリケーション・エラー: クラスタ・データベース構成の取得で障害が発生しました:

原因: 指定したデータベースのクラスタ・データベース構成の取得でエラーが発生しました。

処置: データベースがクラスタ・データベース構成リポジトリに構成されていることを確認してください。また、GSD がクラスタ内の各ノードで実行されていることも確認してください。

PRKO-2006: ノード名が無効です:

原因: 無効なノード名が入力されました。

処置: 正しいノード名を使用してください。有効なノード名は 'lsnodes' の出力と一致します。ノード名にはドメイン名を含めないでください。

PRKO-2007: インスタンス名が無効です:

原因: 無効なインスタンス名が入力されました。

処置: データベースの正しいインスタンス名を使用してください。'srvctl config database -d <db_name>' コマンドを実行して、クラスタ・データベース構成内のデータベースの全インスタンスを確認してください。

PRKO-2008: 接続文字列が無効です:

原因: 無効な接続文字列が入力されました。

処置: 正しい接続文字列の構文 (<user>/<password>[as <role>]) を使用してください。

PRKO-2009: 名前 / 値文字列が無効です:

原因: SRVCTL setenv コマンドで、無効な環境の名前 / 値のペアが入力されました。

処置: 適切な名前 / 値の文字列フォーマット (<name>=<value>) が使用されていることを確認してください。

PRKO-2010: ノードへのインスタンスの追加でエラーが発生しました:

原因: クラスタ・データベース構成へのインスタンスの追加でエラーが発生しました。

処置: 'srvctl config database -d <db_name>' コマンドを使用して、データベースがクラスタ・データベース構成リポジトリに構成されていることをチェックしてください。GSD がクラスタ内の各ノードで実行されていることも確認してください。

PRKO-2011: インスタンスの削除でエラーが発生しました：

原因： クラスタ・データベース構成からのインスタンスの削除でエラーが発生しました。

処置： 'srvctl config database -d <db_name>' コマンドを使用して、データベースとインスタンスがクラスタ・データベース構成リポジトリに構成されているかどうかをチェックしてください。GSD がクラスタ内の各ノードで実行されていることも確認してください。

PRKO-2012: ノードへのインスタンスの移動でエラーが発生しました：

原因： クラスタ・データベース構成のインスタンス変更とノードのマッピング変更でエラーが発生しました。

処置： 'srvctl config database -d <db_name>' コマンドを使用して、データベースとインスタンスがクラスタ・データベース構成リポジトリに構成されているかどうかをチェックしてください。GSD がクラスタ内の各ノードで実行されていることも確認してください。

PRKO-2013: env の設定でエラーが発生しました：

原因： データベースまたはインスタンスに関する環境変数の設定中にエラーが発生しました。

処置： 'srvctl config database -d <db_name>' コマンドを使用して、データベースまたはインスタンス（あるいはその両方）がクラスタ・データベース構成リポジトリに構成されているかどうかをチェックしてください。GSD がクラスタ内の各ノードで実行されていることも確認してください。

PRKO-2014: env の設定解除でエラーが発生しました：

原因： データベースまたはインスタンスに関する環境変数の設定解除中にエラーが発生しました。

処置： 'srvctl getenv' コマンドを使用してデータベースまたはインスタンスに対する環境変数の存在をチェックし、データベースまたはインスタンス（あるいはその両方）がクラスタ・データベース構成リポジトリに構成されていることを確認してください。GSD がクラスタ内の各ノードで実行されていることも確認してください。

PRKO-2015: ノードのインスタンスの条件チェックでエラーが発生しました：

原因： インスタンスのステータス情報を取得できませんでした。

処置： 'srvctl config database -d <db_name>' コマンドを使用して、インスタンスが Oracle データベースに構成されているかどうかをチェックしてください。GSD がクラスタ内の各ノードで実行されていることも確認してください。

PRKO-2016: ノードのリスナーの条件チェックでエラーが発生しました:

原因: リスナーのステータス情報を取得できませんでした。

処置: リスナーがリスナー構成ファイルに構成されているかどうか、およびノードのデータベース・インスタンスがリスナー構成の SID_NAME に記載されているかどうかをチェックしてください。GSD がクラスタ内の各ノードで実行されていることも確認してください

クラスタ・データベース管理のエラー・メッセージ (PRKP)

PRKP-1000 クラスタ・データベース {0} の構成を取り出せません

原因: リポジトリからクラスタ・データベース構成を取得できません。このエラーは、データベースが未登録の場合、またはリポジトリ自体が作成されていない場合に発生する可能性があります。

処置: 'srvctl config' を使用してすべてのクラスタ・データベースのリストを出力し、データベースが構成されているかどうかをチェックしてください。リポジトリが未作成の場合は、'srvconfig-init' を使用して作成してください。

PRKP-1001 ノード {1} でのインスタンス {0} の開始でエラーが発生しました

原因: SQL*Plus の startup コマンドでインスタンスが起動しません。

処置: SQL*Plus を使用して指定のインスタンスを手動で起動し、障害の原因を確認してください。

PRKP-1002 ノード {1} でのインスタンス {0} の停止でエラーが発生しました

原因: インスタンスをシャットダウンする際、SQL*Plus の shutdown コマンドでエラーが戻りました。

処置: SQL*Plus を使用して指定のインスタンスを手動でシャットダウンし、障害の原因を確認してください。

PRKP-1003 起動操作が一部失敗しました

原因: クラスタ・データベースの一部のコンポーネントを起動できませんでした。

処置: 詳細については、前述のエラー・メッセージを参照してください。

PRKP-1004 シャットダウン操作が一部失敗しました

原因: 停止する際、クラスタ・データベースの一部のコンポーネントでエラーがレポートされました。

処置: 詳細については、前述のエラー・メッセージを参照してください。

PRKP-1005 クラスタ・データベース {0} の起動に失敗しました

原因: クラスタ・データベースを起動できませんでした。

処置: 詳細については、前述のエラー・メッセージを参照してください。

PRKP-1006 クラスタ・データベース {0} のシャットダウンに失敗しました

原因: シャットダウンの際、クラスタ・データベースでエラーがレポートされました。

処置: 詳細については、前述のエラー・メッセージを参照してください。

PRKP-1007 クラスタ・データベース {0} のすべてのインスタンスに対応付けられたすべてのリスナーの起動に失敗しました

原因:

処置: オラクル社カスタマ・サポート・センターに連絡してください。

PRKP-1008 ノード {1} のインスタンス {0} に対応付けられたリスナーの開始に失敗しました

原因:

処置: オラクル社カスタマ・サポート・センターに連絡してください。

PRKP-1009 クラスタ・データベース {0} のすべてのインスタンスに対応付けられたすべてのリスナーの停止に失敗しました

原因: インスタンスに対応付けられたリスナー名を判断できなかったか、またはリスナーの「lsnrctl start」に失敗しました。

処置: listener.ora に、指定したデータベースの各インスタンスに対する SID_LIST エントリが含まれていること、およびそれらのリスナーに対する lsnrctl start コマンドが正常に終了していることを確認してください。

PRKP-1010 ノード {1} のインスタンス {0} に対応付けられたすべてのリスナーの停止に失敗しました

原因: インスタンスに対応付けられたリスナー名を判断できなかったか、またはリスナーの「lsnrctl stop」に失敗しました。

処置: listener.ora に、指定したデータベースの各インスタンスに対する SID_LIST エントリが記載されていること、およびそれらのリスナーに対する lsnrctl stop コマンドが正常に終了していることを確認してください。

PRKP-1011 ノード {1} のインスタンス {0} に対応付けられたすべてのリスナーの取得に失敗しました

原因: インスタンスに対応付けられたリスナー名を判断できませんでした。

処置: 指定したインスタンスの SID_LIST エントリが listener.ora に含まれていることを確認してください。

PRKP-1012 クラスタ・データベース {1} に対する環境変数 {0} の設定が無効です

原因: -t オプションに対する引数が <name>=<value> のフォームでないか、または特殊文字が含まれています。

処置: -t オプションに <name>=<value> のフォームの引数があることを確認してください。-t フラグの引数は引用符で囲みます。

PRKP-1013 {0}: クラスタ・データベース {1} に対する未定義の環境変数

原因: 指定したクラスタ・データベースに指定した環境変数が定義されていません。

処置: 「srvctl set env」で変数の値を設定してください。

PRKP-1014 {0}: クラスタ・データベース {2} のインスタンス {1} に対する未定義の環境変数

原因: 指定したインスタンスに指定した環境変数が定義されていません。

処置: 「srvctl set env」で変数の値を設定してください。

PRKP-1015 {0}: 未定義の環境変数

原因: 指定した環境変数が定義されていません。

処置: 「srvctl set env」で、指定した環境変数の値を設定してください。

PRKP-1040 ノード {1} のインスタンス {0} に対応付けられたリスナーのステータスの取得に失敗しました

原因: インスタンスに対応付けられたリスナー名を判断できなかったか、またはそのリスナーの「lsnrctl status」に失敗しました。

処置: 指定したインスタンスの SID_LIST エントリが listener.ora に含まれていること、および lsnrctl status コマンドが正常に終了していることを確認してください。

リポジトリのエラー・メッセージ (PRKR)

PRKR-1001 クラスタ・データベース {0} は存在しません

原因: クラスタ・データベースがリポジトリに登録されていません。

処置: 'srvctl config' を使用してすべてのクラスタ・データベースのリストを出力し、データベースが構成されているかどうかをチェックしてください。

PRKR-1002 クラスタ・データベース {0} はすでに存在しています

原因: リポジトリ内の既存のクラスタ・データベースを登録しようとした。

処置: 'srvctl config' を使用してすべてのクラスタ・データベースのリストを出力し、データベースがすでに構成されているかどうかをチェックしてください。

PRKR-1003 インスタンス {0} は存在しません

原因: 指定したインスタンスがリポジトリに登録されていません。

処置: 'srvctl config' を使用してすべてのデータベースとそのインスタンスを検出してください。

PRKR-1004 インスタンス {0} はすでに存在しています

原因: 既存のインスタンスを登録しようとした。

処置: 前述のメッセージを参照してください。

PRKR-1005 クラスタ・データベース {0} の構成の追加に失敗しました、{1}

原因: 構成リポジトリに新規クラスタ・データベースを追加しようとして、エラーが発生しました。グローバル・サービス・デーモンがクラスタ内のすべてのノードで起動されていないか、構成リポジトリがアクセス可能ではなかったか、または構成リポジトリが初期化されていません。

処置: 各ノードで 'gsdctl stat' を使用して、グローバル・サービス・デーモンが起動しているかどうかをチェックしてください。いずれかのノードでグローバル・サービス・デーモンが起動されていない場合は、'gsdctl start' を使用して起動します。'srvctl config' を使用して、構成リポジトリがアクセス可能かどうかをチェックしてください。構成リポジトリが初期化されていない場合は、'srvconfig -init' を使用して構成リポジトリを初期化します。

PRKR-1006 クラスタ・データベース {0} の構成の削除に失敗しました、{1}

原因: PRKR-1005 と同じです。

処置: 前述のエラー・メッセージを参照してください。

PRKR-1007 クラスタ・データベース {0} の構成の取得に失敗しました、{1}

原因: PRKR-1005 と同じです。

処置: 前述のエラー・メッセージを参照してください。

PRKR-1008 ノード {1} でクラスタ・データベース {2} へのインスタンス {0} の追加に失敗しました、{3}

原因: PRKR-1005 と同じです。

処置: 前述のエラー・メッセージを参照してください。

PRKR-1009 クラスタ・データベース {1} からのインスタンス {0} の削除に失敗しました、{2}

原因: PRKR-1005 と同じです。

処置: 前述のエラー・メッセージを参照してください。

PRKR-1010 クラスタ・データベース {2} のノード {1} へのインスタンス {0} の移動に失敗しました、{3}

原因: PRKR-1005 と同じです。

処置: 前述のエラー・メッセージを参照してください。

PRKR-1011 クラスタ・データベース {2} のインスタンス {1} の名前へのインスタンス {0} の名前の変更失敗しました、{3}

原因: PRKR-1005 と同じです。

処置: オラクル社カスタマ・サポート・センターに連絡してください。

**PRKR-1012 クラスタ・データベース {1} の構成のシリアル化中にエラー {0} が発生しました
原因:**

処置: オラクル社カスタマ・サポート・センターに連絡してください。

PRKR-1013 クラスタ・データベース {1} の構成のデシリアライズ中にエラー {0} が発生しました

原因:

処置: オラクル社カスタマ・サポート・センターに連絡してください。

PRKR-1014 ディレクトリのシリアル化中にエラー {0} が発生しました。

原因:

処置: オラクル社カスタマ・サポート・センターに連絡してください。

PRKR-1015 ディレクトリのデシリアライズ中にエラー {0} が発生しました。

原因:

処置: オラクル社カスタマ・サポート・センターに連絡してください。

PRKR-1016 クラスタ・データベース {0} の構成の読み込みに失敗しました、{1}、{2}

原因: PRKR-1005 と同じです。

処置: 前述のエラー・メッセージを参照してください。

PRKR-1017 クラスタ・データベース {0} の構成の書き込みに失敗しました、{1}、{2}

原因: PRKR-1005 と同じです。

処置: 前述のエラー・メッセージを参照してください。

PRKR-1018 ディレクトリの読み込みに失敗しました、{0}、{1}

原因:

処置: オラクル社カスタマ・サポート・センターに連絡してください。

PRKR-1019 ディレクトリの書き込みに失敗しました、{0}、{1}

原因:

処置: オラクル社カスタマ・サポート・センターに連絡してください。

PRKR-1020 バージョン情報の読み込みに失敗しました、{0}、{1}

原因: PRKR-1005 と同じです。

処置: 前述のエラー・メッセージを参照してください。

PRKR-1021 バージョン情報の書き込みに失敗しました、{0}、{1}

原因: PRKR-1005 と同じです。

処置: 前述のエラー・メッセージを参照してください。

PRKR-1022 ロー・デバイス {0} は非互換バージョンを含んでいます、{1} != {2}

原因: 構成リポジトリの非互換バージョンを使用しようとしてしました。

処置: オラクル社カスタマ・サポート・センターに連絡してください。

PRKR-1023 ファイル {0} は存在しません

原因: 指定したファイルは存在していません。

処置: ファイルの存在をチェックしてください。

PRKR-1024 ファイル {0} には {1} 権限がありません

原因: 指定したファイルには、指定された権限がありません。

処置: ファイルのアクセス権を指定された権限に変更してください。

PRKR-1025 ファイル {0} にはプロパティ {1} が含まれていません

原因: ファイルには、指定したプロパティが含まれていません。

処置: オラクル社カスタマ・サポート・センターに連絡してください。

PRKR-1026 プロパティ {0} はファイル {1} に設定されていません

原因: ファイルには、指定したプロパティが含まれていません。

処置: オラクル社カスタマ・サポート・センターに連絡してください。

PRKR-1027 クラスタ・データベースのリストの取得に失敗しました

原因: PRKR-1005 と同じです。

処置: 前述のエラー・メッセージを参照してください。

PRKR-1028 ロー・デバイス {0} が無効です \n[ヒント: "srvconfig" ツールを使用してロー・デバイスを初期化してください]

原因: 管理リポジトリが初期化されていません。

処置: 'srvconfig -init' を使用してリポジトリを初期化してください。

PRKR-1038 無効な引数 {0} が -init オプションに指定されました

原因: 'srvconfig -init' に無効な引数が指定されました。

処置: 詳細は、'srvconfig -init' の使用方法を参照してください。

PRKR-1039 無効なオプション {0} が指定されました

原因: 指定したオプションは無効です。

処置: 使用方法をチェックしてください。

PRKR-1040 {0} オプションの <file> 引数が不明です

原因: srvconfig に指定したオプションは無効です。

処置: 詳細は使用方法をチェックしてください。

PRKR-1045 ロー・デバイスバージョン {0}

原因: リポジトリのバージョンを取得しようとした。

処置: 処置は不要です。

PRKR-1046 "srvconfig" によって有効なロー・デバイスが検出されました \n [ヒント: 強制的に初期化を行う場合は "-init -f" オプションを指定してください]

原因: 有効な構成リポジトリが検出されました。

処置: 処置は不要です。

PRKR-1047 ノード {1} のデーモンがロー・デバイス {0} を使用中です。

原因: グローバル・サービス・デーモンがクラスタ内の 1 つ以上のノードで起動されている間に、管理リポジトリを初期化しようとした。

処置: 各ノードで 'gsdctl stop' コマンドを使用し、クラスタ内の全ノードのグローバル・サービス・デーモンをすべて停止してください。'srvconfig -init' 操作を再試行してください。

PRKR-1050 {1} ディレクトリでのファイル {0} の作成が失敗しました。権限などをチェックしてください。

原因: 存在しないディレクトリまたは適切な権限がないディレクトリで、ファイルを作成しようとした。

処置: ディレクトリが存在していない場合は、ディレクトリを作成するか、ディレクトリの権限を変更してください。

PRKR-1051 ファイル {0} には dbname {1} へのエントリが含まれていません

原因:

処置: オラクル社カスタマ・サポート・センターに連絡してください。

PRKR-1052 ファイル名 {0} は <cluster database name>.conf フォームではありません

原因: 管理リポジトリで 8.1.7 以前のデータベースの登録を試み、<cluster database name>.conf のフォームではないファイル引数を渡しました。

処置: 詳細については、srvconfig の使用方法を参照してください。

PRKR-1053 node_list = {1} に無効な範囲 {0} が指定されました。

原因: リリース 8.1.7 以前のリリースの Oracle Parallel Server を RAC に変換しようとして、<opsName>.conf ファイルに指定されている無効なパラメータを検出しました。

処置: リリース 8.1.7 のマニュアルをチェックし、<opsName>.conf ファイルのフォーマットがマニュアルと一致していることを確認してください。

PRKR-1054 inst_oracle_sid = {1} に無効なパラメータ {0} が指定されました

原因: 構成リポジトリで 8.1.7 以前のリリースの Oracle Parallel Server のデータベースを追加しようとして、<opsName>.conf ファイルに指定されている無効なパラメータを検出しました。

処置: リリース 8.1.7 のマニュアルをチェックし、<opsName>.conf ファイルのフォーマットがマニュアルと一致していることを確認してください。

PRKR-1055 無効な余分の引数 {0} が {1} オプションに指定されました

原因: 余分な引数を 'srvconfig' に指定しました。

処置: 詳細については、'srvconfig' の使用方法を参照してください。

PRKR-1056 無効なレジストリ・エントリ {0} が見つかりました。{1} の形式にしてください

原因: 構成リポジトリで 8.1.7 以前のリリースの Oracle Parallel Server を追加しようとして、無効なレジストリ・エントリを検出しました。

処置: 前述のエラー・メッセージを参照してください。

PRKR-1057 環境変数は存在しません

原因: 存在しない環境変数を取得しようとしてしました。

処置: 環境変数を設定してください。

用語集

Cluster Manager (CM)

オペレーティング・システム固有のコンポーネント。[クラスタ](#)上のメンバーシップに関する共通ビューを提供して、各[ノード](#)のメンバーシップ状態を検出して追跡する。CMは、プロセスの状態、具体的にはデータベース・インスタンスの状態も監視する。[グローバル・キャッシュ・サービス \(GCS\)](#)の状態を監視するバックグラウンド・プロセスである[グローバル・エンキュー・サービス・モニター \(LMON\)](#)は、CM への登録と登録解除を行う。

CM

「[Cluster Manager \(CM\)](#)」を参照。

Database Configuration Assistant (DBCA)

データベースを作成および削除し、データベース・テンプレートを管理するための Oracle のツール製品。

DSS

「[意思決定支援システム \(decision support system: DSS\)](#)」を参照。

Enterprise Manager Configuration Assistant (EMCA)

[Oracle Enterprise Manager](#) の構成および設定を作成、削除および変更するためのツール。

Enterprise Manager

「[Oracle Enterprise Manager](#)」を参照。

「Event Management」 ウィンドウ (Event Management Window)

[コンソール](#)の一部。「Event Management」ウィンドウで、管理者は、重要なデータベースおよびシステム・イベントをリモートで監視できる。

「Group」 ウィンドウ (Group Window)

コンソールの一部。「Group」ウィンドウでは、カスタマイズされたグラフィカル表示で主なオブジェクトを参照できる。このウィンドウは管理者が作成する。

GSD

「**グローバル・サービス・デーモン (Global Services Daemon: GSD)**」を参照。

GV\$

「**グローバル動的パフォーマンス・ビュー (global dynamic performance view: GV\$)**」を参照。

IFILE

初期化パラメータ・ファイルの継続ファイルを設計するパラメータ。

initdbname.ora

データベース・パラメータを含み、インスタンス間で共有される共通のデータベース初期化ファイルのパラメータ。

initsid.ora

インスタンス固有のパラメータを含み、データベース・パラメータ用の **initdbname.ora** を指すインスタンス初期化ファイルのパラメータ。

Intelligent Agent

「**Oracle Intelligent Agent**」を参照。

「Job」 ウィンドウ (job window)

コンソールの一部。「Job」ウィンドウを使用すると、管理者は繰り返し行われる処理を自動化できる。

LDAP

「**Lightweight Directory Access Protocol (LDAP)**」を参照。

Lightweight Directory Access Protocol (LDAP)

オンライン・ディレクトリ・サービスにアクセスするためのプロトコル。

listener.ora

リスナーが、接続要求およびリスニング対象のサービスを受け入れるためのプロトコル・アドレスを識別するリスナー構成ファイル。

LMD

「**グローバル・エンキュー・サービス・デーモン (Global Enqueue Service Daemon: LMD)**」を参照。

LMON

「**グローバル・エンキュー・サービス・モニター (Global Enqueue Service Monitor: LMON)**」を参照。

LMSn

「**グローバル・キャッシュ・サービス・プロセス (Global Cache Service Process: LMSn)**」を参照。

Management Server

Oracle Enterprise Manager の Management Server は、**コンソール**と管理される**ノード**の間で、集中化されたインテリジェント機能と分散制御を提供する。また、コンソールから送信されたシステム管理タスクを処理し、これらのタスクを社内に分散する処理を管理する。Management Server では、すべてのシステム・データ、アプリケーション・データおよび管理されるノードの状態情報が、リポジトリに格納される。このリポジトリは、データベースに格納された表の集合である。複数の Management Server が存在する場合は、ワークロードが自動的に共有され、負荷が平衡化されるため、高度なパフォーマンスと拡張性が保証される。

MTBF

「**平均障害間隔時間 (mean time between failure: MTBF)**」を参照。

MTTF

「**平均障害時間 (mean time to failure: MTTF)**」を参照。

MTTR

「**平均リカバリ時間 (mean time to recover: MTTR)**」を参照。

「Navigator」ウィンドウ (Navigator Window)

コンソールの一部。「Navigator」ウィンドウには、ネットワーク内のデータベース・オブジェクトを構成して一貫性を持たせた階層ビューを提供するオブジェクト・ブラウザが含まれている。

NULL (N)

ブロックまたはリソースにアクセス・モードが割り当てられていないことを示す。

N

「**NULL (N)**」を参照。

OLTP

「**オンライン・トランザクション処理 (online transaction processing: OLTP)**」を参照。

Oracle Enterprise Edition

オブジェクト・リレーショナル・データベース管理システム (ORDBMS)。データベース管理用のアプリケーションおよびファイルを提供する。他の Real Application Clusters コンポーネントはすべて、Oracle Enterprise Edition の最上部の層に位置している。

Oracle Enterprise Manager

異機種間環境を集中管理するための統合されたソリューションを提供する、システム管理ツール。Oracle Enterprise Manager によって、グラフィカルな **コンソール**、**Management Server**、**Oracle Intelligent Agent**、**リポジトリ・データベース** およびツール製品が組み合わされ、Oracle 製品を管理するための統合された包括的なシステム管理プラットフォームが提供される。

Oracle Intelligent Agent

各ノードで実行するプロセス。**Management Server** 経由で **コンソール** から送信されるジョブとイベントの実行者として機能する。エージェントは **コンソール** やネットワーク接続の状態に関係なく機能できるため、高可用性が保証される。

Oracle Net

接続を可能にするソフトウェア・コンポーネント。Oracle Net Foundation レイヤーと呼ばれるコア通信レイヤーおよびネットワーク・プロトコル・サポートが含まれる。Oracle Net Services とそのアプリケーションは、異なるコンピュータに常駐し、ピア・アプリケーションとして通信を行う。

Oracle Performance Manager

Real Application Clusters のパフォーマンス統計を様々な表形式やグラフ形式で提供する **Oracle Enterprise Manager** のアドオン・アプリケーション。統計は、Real Application Clusters で実行されている全インスタンスの集計パフォーマンスを表す。

Oracle Real Application Clusters

クラスタが共有データベースにアクセスできる画期的なアーキテクチャ。Oracle9i Enterprise Edition を Oracle9i Real Application Clusters データベースとして活用するために必要な、Real Application Clusters スクリプト、初期化ファイルおよびデータ・ファイルが含まれている。

Oracle Universal Installer (OUI)

Oracle リレーショナル・データベース・ソフトウェアをインストールするためのツール。Oracle Universal Installer を使用して、**Database Configuration Assistant (DBCA)** を起動することもできる。

Oracle システム識別子 (Oracle System Identifier: *sid*)

Oracle ソフトウェアが動作しているときに、その特定のインスタンスを識別する。Real Application Clusters データベースの場合、クラスタ内の各 **ノード** に、データベースを参照するインスタンスがある。

各ノードの *sid* は、`initdb_name.ora` ファイルの `DB_NAME` パラメータで指定されたデータベース名と、一意のスレッド ID からなる。スレッド ID はクラスタ内の最初のインスタンスが 1 で始まり、2 番目以降のインスタンスの ID は 1 ずつ大きくなる。「**インスタンス名 (instance name)**」を参照。

OSD

「**オペレーティング・システム依存 (OSD) のクラスタウェア (operating system-dependent (OSD) clusterware)**」を参照。

Performance Manager

「**Oracle Performance Manager**」を参照。

Real Application Clusters

「**Oracle Real Application Clusters**」を参照。

Recovery Manager (RMAN)

Oracle のツール製品の 1 つ。データ・ファイル、制御ファイルおよびアーカイブ REDO ログのバックアップ、コピー、リストアおよびリカバリを行う。Recovery Manager は Oracle サーバーに含まれているため、個別にインストールする必要はない。Recovery Manager は、オペレーティング・システム (O/S) のプロンプトからコマンドライン・ユーティリティとして起動するか、または GUI ベースの Enterprise Manager の一部である Backup Manager を使用できる。

RMAN

「**Recovery Manager (RMAN)**」を参照。

Server Management (SRVM)

Real Application Clusters での **Oracle Enterprise Manager** 操作に必要なコンポーネントが含まれる。Intelligent Agent、グローバル・サービス・デーモン、SRVCTL などの SRVM コンポーネントでは、Oracle Enterprise Manager を使用したオープンなクライアント / サーバー・アーキテクチャを介して異機種間環境で実行するクラスタ・データベースを管理できる。

sid

「**Oracle システム識別子 (Oracle System Identifier: *sid*)**」を参照。

spfile.ora

Oracle サーバーに常駐するバイナリ・パラメータ・ファイル。

sqlnet.ora ファイル (sqlnet.ora File)

次のことを指定する、クライアントまたはサーバー用の構成ファイル。

- 修飾されていないサービス名またはネット・サービス名に付加されるクライアント・ドメイン
- 名前の変換時にクライアントが使用するネーミング・メソッドの順序
- 使用するロギングおよびトレース機能
- 接続ルート
- 優先 Oracle Names Server
- 外部ネーミング・パラメータ
- Oracle Advanced Security パラメータ

通常、sqlnet.ora ファイルは、UNIX プラットフォームでは \$ORACLE_HOME/network/admin ディレクトリに、Windows プラットフォームでは %ORACLE_HOME%\network\admin ディレクトリに常駐する。

SRVCTL

「[サーバー制御ユーティリティ \(SRVCTL\) \(Server Control \(SRVCTL\) Utility\)](#)」を参照。

SYSDBA

ADMIN OPTION を持つすべてのシステム権限と SYSOPER システム権限を含むデータベース管理ロール。SYSDBA では、CREATE DATABASE と時間ベースのリカバリも許可される。

SYSOPER

データベース管理者に、STARTUP、SHUTDOWN、ALTER DATABASE OPEN/MOUNT、ALTER DATABASE BACKUP、ARCHIVE LOG および RECOVER の各コマンドの実行を許可する特別なデータベース管理ロール。RESTRICTED SESSION 権限が含まれる。

TAF

「[透過的アプリケーション・フェイルオーバー \(Transparent Application Failover: TAF\)](#)」を参照。

tnsnames.ora ファイル (tnsnames.ora file)

ネット・サービス名が記述されているファイル。このファイルは、クライアント、ノード、[コンソール](#)および Oracle Performance Manager マシン上で必要である。

X

「[排他的 \(X\) アクセス・モード \(exclusive \(X\) access mode\)](#)」を参照。

空きリスト・グループ (free list group)

1つ以上のインスタンスで使用可能な空きリストの集合。

意思決定支援システム (decision support system: DSS)

意思決定支援システムまたはデータ・ウェアハウス・システムに有効なデータベース環境およびアプリケーション環境。

インスタンス (instance)

Real Application Clusters データベースの場合、通常、クラスタ内の各ノードには、データベースを参照する実行中の Oracle ソフトウェアのインスタンスが1つある。データベースを起動すると、Oracle によってシステム・グローバル領域 (System Global Area: SGA) と呼ばれるメモリー領域が割り当てられ、1つ以上の Oracle プロセスが開始される。このような、SGA と Oracle プロセスの組合せをインスタンスと呼ぶ。各インスタンスには、一意の Oracle システム識別子 (*sid*)、インスタンス名、ロールバック・セグメントおよびスレッド ID がある。

インスタンス番号 (instance number)

特定のインスタンスとデータ・ブロックのエクステンントを対応付ける番号。インスタンス番号を使用すると、インスタンスを起動でき、そのインスタンスに割り当てられたエクステンントを挿入と更新に対して使用することが保証される。これによって、他のインスタンスに割り当てられた領域を使用しないことが保証される。

インスタンス名 (instance name)

インスタンスの名前を表し、クラスタが共通のサービス名を共有する場合は、特定のインスタンスを一意に識別するために使用される。インスタンス名は、インスタンス初期化ファイル (initsid.ora) の INSTANCE_NAME パラメータによって識別される。このインスタンス名は、Oracle システム識別子 (*sid*) と同じである。「**Oracle システム識別子 (Oracle System Identifier: sid)**」を参照。

インスタンス・メンバーシップ・リカバリ (instance membership recovery: IMR)

すべてのクラスタ・メンバーが機能している、またはアクティブであることを保証するために **Real Application Clusters** で使用される方法。IMR はメンバーシップをポーリングし、判定を行う。制御ファイルを通してハートビートを示さないメンバー、または定期的なアクティビティ照会メッセージに応答しないメンバーは終了したとみなされる。

インターコネクト (interconnect)

ノード間の通信リンク。

エンキュー (enqueue)

データベース・リソースへのアクセスをシリアル化し、セッションまたはトランザクションと対応付けられる共有メモリー構造。Real Application Clusters では、エンキューはデータベースに対してグローバルになる。Real Application Clusters が使用不可の場合は、1つのインスタンスに対してローカルである。

オペレーティング・システム依存 (OSD) のクラスタウェア (operating system-dependent (OSD) clusterware)

Oracle 社または他のベンダーが開発した様々なソフトウェア・コンポーネントで構成されたソフトウェア。OSD クラスタウェアによって、**Real Application Clusters** の正常操作に必要な、主要なオペレーティング・システムとクラスタウェア・サービスがマップされる。

オンライン・トランザクション処理 (online transaction processing: OLTP)

コンピュータによるリアルタイムでのトランザクション処理。

拡張性 (scalability)

Real Application Clusters アプリケーションにノードを追加し、大幅にスピードアップおよびスケールアップさせる機能。

仮想インタフェース・アーキテクチャ (Virtual Interface Architecture: VIA)

メモリーベースのネットワーキング・インタフェース。

キャッシュ一貫性 (cache coherency)

複数キャッシュにおけるデータの同期化。どのキャッシュを通してメモリー位置を読み取っても、その位置に書き込まれた最新のデータが他のキャッシュを通して戻される。「キャッシュの整合性」と呼ばれることもある。

キャッシュ・フュージョン (Cache Fusion)

Real Application Clusters におけるディスクレスのキャッシュの一貫性メカニズム。保持側インスタンスのメモリー・キャッシュから要求側インスタンスのメモリー・キャッシュにブロックのコピーを直接提供する。

共有カレント (Shared Current: SCUR)

ブロックへの共有アクセス・モードに対するバッファの状態名。

共有サーバー (shared server)

多数のユーザー・プロセスが、非常に少数のサーバー・プロセスを共有できるように構成されたサーバー。これによって、サポートされるユーザーの数が増える。共有サーバー構成では、多数のユーザー・プロセスがディスパッチャに接続する。

共有モード (Shared Mode: S)

保護付きの読取りブロック・アクセス・モード。書込みは許可されない。共有モードでは、どのユーザーでも、数に関係なく、リソースに同時読込みアクセスを持つことができる。「**排他的 (X) アクセス・モード (exclusive (X) access mode)**」を参照。

クラスタ (cluster)

同じタスクを協力して実行するための一連のインスタンス。

クラスタ化データベース (clustered database)

「[クラスタ・データベース \(cluster database\)](#)」を参照。

クラスタ・データベース (cluster database)

[Real Application Clusters](#) データベースの総称。

クラスタリング (clustering)

「[クラスタ・データベース \(cluster database\)](#)」を参照。

グローバル化・サポート (Globalization Support)

データの格納、処理および取出しを各国の言語で実行するアーキテクチャ。データベースのユーティリティやエラー・メッセージ、ソート順序、日付、時刻、通貨単位、数値およびカレンダーの規則には、各国の言語やロケールの設定が自動的に適応される。

グローバル・エンキュー・サービス (Global Enqueue Service: GES)

グローバルに共有されるエンキューを調整するサービス。

グローバル・エンキュー・サービス・デーモン (Global Enqueue Service Daemon: LMD)

リソース・エージェント・プロセス。これは、[グローバル・キャッシュ・サービス \(GCS\)](#) のリソースに対する要求を管理して、ブロックへのアクセスを制御する。LMD プロセスは、デッドロック検出およびリモート・リソース要求も処理する。リモート・リソース要求とは、別のインスタンスから発生する要求のことである。

グローバル・エンキュー・サービス・モニター (Global Enqueue Service Monitor: LMON)

バックグラウンド LMON プロセスは、クラスタ全体を監視して、グローバル・リソースを管理する。LMON は、インスタンスの「異常終了」、および[グローバル・キャッシュ・サービス \(GCS\)](#) に関連するリカバリを管理する。特に、グローバル・リソースに関連するリカバリの部分进行处理する。LMON が提供するサービスはクラスタ・グループ・サービスとも呼ばれる。

グローバル・キャッシュ・サービス (GCS)・リソース (Global Cache Service (GCS) resource)

キャッシュの一貫性を維持するために、複数の [Real Application Clusters](#) インスタンスのバッファ・キャッシュにあるデータ・ブロックへのアクセスを調整するグローバル・リソース。

グローバル・キャッシュ・サービス (Global Cache Service: GCS)

[キャッシュ・フュージョン](#)を実装するプロセス。グローバル・ロールのブロックに対するブロック・モードを保持する。インスタンス間のブロック送信を担当する。グローバル・キャッシュ・サービスでは、[グローバル・キャッシュ・サービス・プロセス \(LMSn\)](#) や [グローバル・エンキュー・サービス・デーモン \(LMD\)](#) など、様々なバックグラウンド・プロセスが使用される。

グローバル・キャッシュ・サービス・プロセス (Global Cache Service Process: LMSn)

グローバル・キャッシュ・サービス (GCS) のリモート・メッセージを管理するプロセス。**Real Application Clusters** では、最大 10 個のグローバル・キャッシュ・サービス・プロセスが提供される。LMSn の数は、クラスタのノード間のメッセージ通信量によって異なる。

グローバル・サービス・デーモン (Global Services Daemon: GSD)

SRVCTL から要求を受信し、起動や停止などの管理ジョブ・タスクを実行するコンポーネント。コマンドは各ノードでローカルに実行され、結果が SRVCTL に戻される。GSD は、デフォルトでノードにインストールされる。

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・データベース名のフォームは、database_name.database_domain (たとえば、OP.US.ORACLE.COM)。

グローバル動的パフォーマンス・ビュー (global dynamic performance view: GV\$)

Real Application Clusters のクラスタにあるすべてのオープン・インスタンス (ローカル・インスタンスのみではない) に関する情報が格納されている動的パフォーマンス・ビュー。これに対して、標準の動的パフォーマンス・ビュー (V\$) には、ローカル・インスタンスに関する情報のみが格納されている。

グローバル・リソース (global resource)

Real Application Clusters にキャッシュの一貫性を提供するインスタンス間の同期化メカニズム。この用語は、**グローバル・キャッシュ・サービス (GCS)**・リソースと**グローバル・エンキュー・サービス (GES)**・リソースの両方を表す。

高可用性 (high availability)

ハードウェアまたはソフトウェア障害が発生しても、一貫性のある連続的なサービスを提供する、冗長コンポーネントを持つシステム。ある程度の冗長性を伴う。

コンソール (Console)

Oracle Enterprise Manager のコンソールは、強力で確実なシステム管理を実現する直観的な GUI を介して、Oracle 環境の集中管理ポイントとなる。コンソールには、メニュー、ツールバーおよび起動パレットがあり、これらを使用して Oracle のツール製品にアクセスする。コンソールは 4 つの別々のウィンドウで構成されている。「**Navigator ウィンドウ (Navigator Window)**」、「**Group ウィンドウ (Group Window)**」、「**Event Management ウィンドウ (Event Management Window)**」および「**Job ウィンドウ (job window)**」を参照。

サーバー・クラスタリング (server clustering)

「**Real Application Clusters**」を参照。

サーバー制御ユーティリティ (SRVCTL) (Server Control (SRVCTL) Utility)

Server Management では、SRVCTL ユーティリティ (各ノードにインストールされている) を使用して、一部の Oracle のツール製品が使用する構成情報を管理できる。たとえば、SRVCTL は、Oracle Intelligent Agent とノードの間における単一の制御ポイントとして機能する。SRVCTL との通信には、1つのノードの Oracle Intelligent Agent のみが使用される。そのノード上の SRVCTL が、Oracle Net を介して他のノードと通信する。

サーバー・パラメータ・ファイル (server parameter file)

Oracle サーバーに常駐するバイナリ・パラメータ・ファイル。このファイルには、グローバルなパラメータ設定とインスタンス固有のパラメータ設定の両方が含まれている。これらのパラメータ設定は、インスタンス停止およびインスタンス起動イベントが発生しても維持される。

サービス検出 (service discovery)

コンソールから Discover Node (ノード検出) コマンドを実行すると、Management Server はそのノードにインストールされている Oracle Intelligent Agent に接続してインストール済の Oracle サービスを検出する。次に、Management Server は、新規情報をリポジトリに配置してコンソールの「Navigator」ウィンドウ内の階層ツリーを更新し、すべてのノードとそれぞれのノードが提供するサービスの全体図を表示する。

サービス登録 (service registration)

PMON がリスナーに情報を自動的に登録する機能。この情報はリスナーに登録されるため、listener.ora ファイルはこの静的情報で構成する必要はない。

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示する方法。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実装できる。サービス名は、グローバル・データベース名を表す文字列である。グローバル・データベース名は、インストール時またはデータベースの作成時に入力したデータベース名 (DB_NAME) とドメイン名 (DB_DOMAIN) で構成される。

シード・データベース (seed database)

データベース作成時のユーザー入力を最小限に抑えた、すぐに使用できる構成済のデータベース。「初期データベース (starter database)」を参照。

自動検出 (auto-discovery)

コンソールから Discover Node (ノード検出) コマンドを実行すると、Management Server はそのノードにインストールされている Oracle Intelligent Agent に接続してインストール済の Oracle サービスを検出する。次に、Management Server は、新規情報をリポジトリに配置してコンソールの「Navigator」ウィンドウ内の階層ツリーを更新し、すべてのノードとそれぞれのノードが提供するサービスの全体図を表示する。

初期化パラメータ・ファイル (initialization parameter file)

spfile.ora、initdbname.ora、initsid.ora など、データベースを初期化するために必要な情報を含むファイル。

初期データベース (starter database)

「シード・データベース (seed database)」を参照。

スナップショット制御ファイル (snapshot control file)

Recovery Manager は、スナップショット制御ファイルを作成して、読み取り一貫性バージョンの制御ファイルから再同期化する。これは、一時的なスナップショット制御ファイルである。リカバリ・カタログとの再同期化、または現行の制御ファイルのバックアップを実行する場合、Recovery Manager には 1 つのスナップショット制御ファイルのみが必要である。

スレッド (thread)

各 Oracle インスタンスには、独自のオンライン REDO ログ・グループのセットがある。このようなグループを、オンライン REDO のスレッドと呼ぶ。Real Application Clusters 以外の環境では、1 つのデータベースには 1 つのスレッドしか存在せず、このスレッドは、そのデータベースにアクセスしているインスタンスに属している。Real Application Clusters 環境では、インスタンスごとに別々のスレッドがある。つまり、各インスタンスは、独自のオンライン REDO ログを持っている。各スレッドには、独自の現行ログ・メンバーがある。

スレッド番号 (thread number)

インスタンスが使用する REDO スレッドの番号。THREAD 初期化パラメータまたは ALTER DATABASE ADD LOGFILE 文の THREAD 句に指定されている。使用可能な REDO スレッド番号は任意に使用できるが、別のインスタンスと同じスレッド番号は使用できない。

接続記述子 (connect descriptor)

特別にフォーマットされた、ネットワーク接続の接続先の説明。接続記述子には、宛先サービスおよびネットワーク・ルート情報が含まれる。

接続時フェイルオーバー (connect-time failover)

「フェイルオーバー (failover)」を参照。

接続時ロード・バランシング (connection load balancing)

同じサービスの様々なインスタンスと共有サーバー・ディスパッチャの間で、アクティブな接続数を平衡化する機能。サービス登録機能を使用してリモート・リスナーに登録できるため、リスナーはすべてのインスタンスおよびディスパッチャを常に認識できる。このように、リスナーは、特定のサービスに対するクライアントからの受信要求を、その位置に関係なく最もロードの少ないインスタンスとディスパッチャに送信できる。

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。サーバー・プロセスは、クライアントごとに1つある。Oracle Net 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返信する。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。「**共有サーバー (shared server)**」と対比。

デーモン (daemon)

Disk and Execution Monitor の略。明示的には起動されず、一定の条件が満たされるまで待機している休止状態のプログラム。

ディスパッチャ (dispatcher)

多くのクライアントが、クライアントごとの**専用サーバー**・プロセスなしで同じサーバーに接続できるようにするプロセス。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・プロセスに送る。「**共有サーバー (shared server)**」を参照。

透過的アプリケーション・フェイルオーバー (Transparent Application Failover: TAF)

Real Application Clusters や Oracle Real Application Clusters Guard など、高可用性の環境を目的としたランタイム・フェイルオーバー。TAF は、アプリケーションとサービス間の接続のフェイルオーバーおよび再確立を示す。これによって、接続に障害が発生した場合、クライアント・アプリケーションは自動的にデータベースに再接続され、処理中の SELECT 文を再開できる。この再接続は、Oracle Call Interface (OCI) ライブラリ内から自動的に実行される。

ネーミング・メソッド (naming method)

ネット・サービス名を接続記述子に変換するために、クライアント・アプリケーションによって使用されるメソッド。

ネット・サービス名 (net service name)

接続記述子に変換されるサービスの単純な名前。ユーザーは、接続するサービスに対する接続文字列でネット・サービス名とともにユーザー名とパスワードを渡し、接続要求を開始する。

ネットワーク・ファイル・システム (network file system: NFS)

Sun 社が開発したプロトコルで、RFC 1094 で定義されている。このプロトコルを使用すると、コンピュータで、ローカル・ディスク上のファイルと同様に、ネットワーク上のファイルにアクセスできる。

ノード (node)

インスタンスが常駐するマシン。

排他的 (X) アクセス・モード (exclusive (X) access mode)

書き込み専用のグローバル・ブロック・アクセス・モード。このモードでは、他のアクセスは許可されない。

ハイブリッド (hybrid)

OLTP とデータ・ウェアハウス処理の両方の特長を備えたハイブリッド・データベース。「汎用 (General Purpose)」を参照。

パラメータ・ファイル (parameter file: PFILE)

Oracle サーバーが使用するファイル。データベースの起動時に使用する特定の値および構成の設定を指定する。キーワード PFILE は、起動コマンドで使用される。

汎用 (General Purpose)

ハイブリッド・データベース環境のために事前に構成された、Database Configuration Assistant のデータベース・テンプレート。このテンプレートにはデータ・ファイルが含まれる。

フェイルオーバー (failover)

障害の認識とリカバリのプロセス。

複数の Oracle ホーム (multiple Oracle homes)

1 台のマシン上に複数の Oracle ホームのディレクトリを所有できる機能。

プロセス間通信 (Inter-Process Communication: IPC)

オペレーティング・システム固有の高速な転送コンポーネント。様々なノード上のインスタンス間でメッセージを転送する。**インターコネクト**とも呼ばれる。

分散リカバリ (distributed recovery)

クラスタをリカバリするプロセス。

平均障害間隔時間 (mean time between failure: MTBF)

コンポーネントが障害なしに動作する平均時間 (通常は時間単位で表される)。計測した合計動作時間数を合計障害数で割って算出する。この用語を使用してユーザーは、障害が発生するまでのデバイスまたはシステムの動作期間を予想できる。

平均障害時間 (mean time to failure: MTTF)

コンポーネントに障害が発生するまでの平均時間。

平均リカバリ時間 (mean time to recover: MTTR)

ハードウェアの障害をオンラインに戻すまでの平均時間。Real Application Clusters 関係以外で使用される場合、頭文字の MTTR は平均修復時間 (Mean Time to Repair) の意味でもある。

マスター空きリスト (master free list)

表内のすべてのエクステンツから引き出した使用可能領域が格納されているブロックのリスト。

優先接続情報 (preferred credential)

各 **Oracle Enterprise Manager** 管理者は、ネットワーク上で管理するノード、リスナー、データベースおよび他のサービスに対して、特定のユーザー名、パスワードおよびロールを設定できる。

ラッチ (latch)

低レベルのシリアル化メカニズムで、システム・グローバル領域 (System Global Area: SGA) のメモリー内データ構造を保護する。ラッチはデータ・ファイルを保護せず、自動的に実行され、非常に短い時間のみ排他モードで保持される。ラッチは1つのノード内で同期化されるため、**ノード**間の同期化には使用できない。

リスナー (listener)

サーバーに常駐するプロセス。クライアントの着信接続要求をリスニングし、サーバーへの通信量を管理する。クライアントがサーバーとのネットワーク・セッションを要求すると、リスナーが要求を受け取る。クライアント情報がリスナー情報と一致すると、リスナーはサーバーへの接続を認める。

リポジトリ・データベース (repository database)

Oracle Enterprise Manager に必要なデータが格納されている Oracle データベース内の表の集合。これは、ノード上のデータベースとは別のデータベースである。

A

Abort モード、警告, 5-10
ACTIVE_INSTANCE_COUNT パラメータ, 2-14
「Adding an Instance」 ページ, 8-15
ALERT ファイル, 7-11
ALLOCATE EXTENT オプション
 DATAFILE オプション, B-3
 INSTANCE オプション, B-3
 SIZE オプション, B-3
 インスタンス番号, B-2
alrt_sid.log ファイル, A-3
ALTER DATABASE ADD LOGFILE 文, 4-21
ALTER DATABASE 文
 CLOSE 句, 4-21
ALTER ROLLBACK SEGMENT 文, 3-6
ALTER SESSION 文
 SET INSTANCE 句, B-3
ALTER SYSTEM ARCHIVE LOG CURRENT 文, 4-21
ALTER SYSTEM ARCHIVE LOG 文, 4-21
 THREAD 句, 4-21
ALTER SYSTEM CHECK DATAFILES 文
 インスタンス・リカバリ, 7-11
ALTER SYSTEM CHECKPOINT LOCAL 文, 4-21
ALTER SYSTEM CHECKPOINT 文
 インスタンスの指定, 4-21
 グローバル対ローカル, 4-21
ALTER SYSTEM SWITCH LOGFILE 文, 4-21
ARCHIVE LOG コマンド, 4-21
ARCHIVE_LOG_TARGET パラメータ, 2-13

B

BACKGROUND_DUMP_DEST パラメータ, A-2, A-3

C

CACHE オプション、CREATE SEQUENCE, 3-10
CHECK DATAFILES 句
 インスタンス・リカバリ, 7-11
「Cluster Database Instances」 フォルダ
 コンソールの「Navigator」 ウィンドウ, 5-6
「Cluster Database Operation Results」 ダイアログ・ボックス, 5-11
「Cluster Database Shutdown Progress」 ダイアログ・ボックス, 5-10
「Cluster Database Shutdown」 ダイアログ・ボックス, 5-9
「Cluster Database Started」 メッセージ・ボックス, 5-9
「Cluster Database Startup Results」 ダイアログ・ボックス, 5-9
「Cluster Database Startup/Shutdown Results」 ダイアログ・ボックス, 5-11
「Cluster Database Startup」 ダイアログ・ボックス, 5-8
「Cluster Database Stopped」 メッセージ・ボックス, 5-10
Cluster Manager (CM)
 ソフトウェア, 4-14
CLUSTER_DATABASE_INSTANCES パラメータ, 2-15
CLUSTER_DATABASE パラメータ, 2-15
CLUSTER_INTERCONNECTS パラメータ, 2-14, 2-16
「Confirmation」 ダイアログ・ボックス, 8-24
CONNECT SYS
 例, 2-20
CONNECT コマンド, 4-17, 4-18, 4-21
CONTROL_FILES パラメータ, 2-13
 すべてのインスタンスで同一, 2-13

CREATE DATABASE 文
 MAXINSTANCES 句, B-2
 MAXLOGHISTORY 句, 3-8
「Create Job」プロパティ・シート, 5-19, 5-24
 「General」タブ, 5-19
 「Parameters」タブ, 5-21
 「Tasks」タブ, 5-21
CREATE SEQUENCE 文, 3-9, 3-11
 CACHE 句, 3-10, 3-11
 CYCLE 句, 3-10
 ORDER 句, 3-10, 3-11
 説明, 3-10
CREATE TABLE 文
 FREELIST GROUPS 句, B-3
 FREELISTS 句, B-3
CYCLE オプション、CREATE SEQUENCE, 3-10

D

Database Configuration Assistant
 「Adding an Instance」ページ, 8-15
 「Confirmation」ダイアログ・ボックス, 8-24
 「Database Storage」ページ, 8-15
 「Error」ダイアログ・ボックス, 8-22
 IFILE パラメータの配置, 2-10
 「Instance Management」ページ, 8-12, 8-19
 「List of Cluster Database Instances」ページ, 8-13, 8-21
 「List of Cluster Databases」ページ, 8-13, 8-20
 「Operations」ページ, 8-11
 「Summary」ダイアログ・ボックス, 8-17, 8-23
 「Welcome」ページ, 8-9
 インスタンスの削除, 8-19
 インスタンスの追加, 8-9
 トラブルシューティング, 8-10
 ログ・ファイル, 3-8
 「Database Storage」ページ, 8-15
DB_BLOCK_SIZE パラメータ, 2-13
 すべてのインスタンスで同一, 2-13
DB_DOMAIN パラメータ, 2-13
DB_FILES パラメータ, 2-13
 すべてのインスタンスで同一, 2-13
DB_NAME パラメータ, 2-8, 2-13, 2-16
 すべてのインスタンスで同一, 2-13
DISCONNECT コマンド, 4-18

DISPATCHERS パラメータ, 2-17
 共有サーバー, 2-17
DML_LOCKS パラメータ, 2-13, 2-17

E

「Edit Cluster Database」ダイアログ・ボックス, 5-15
EM コンソールの起動, 5-2
「Error」ダイアログ・ボックス, 8-22

F

FORCE モード、警告, 5-8
FREELIST GROUPS
 記憶域オプション、インスタンス番号, B-2
FREELIST GROUPS オプション, B-2

G

GC_FILES_TO_LOCKS パラメータ, 2-14
GCS
 リカバリの手順, 7-12
 「General」タブ、「Create Job」プロパティ・シート, 5-19
GLOBAL 句
 チェックポイントの強制, 4-21
GSD
 UNIX での実装, 4-3
 Windows での実装, 4-4
 管理ツールのバックグラウンド・プロセス, 4-3
gsdctl remove
 サービスの停止および削除, 4-14
gsdctl start
 サービスの起動, 4-14

H

HOST コマンド, 4-21

I

IFILE パラメータ
 値のオーバーライド, 2-10
 同一パラメータの指定, 2-20
 複数のファイル, 2-11
init.ora ファイル
 定義, 2-7

initdb_name.ora ファイル
BACKGROUND_DUMP_DEST パラメータ, A-2,
A-3
DB_NAME パラメータ, 2-8
USER_DUMP_DEST パラメータ, A-3
initdb_name.ora ファイル
説明, 2-9
initsid.ora ファイル
説明, 2-7
initsid.ora ファイル
説明, 2-8
「Instance Management」 ページ, 8-12, 8-19
INSTANCE_NAME パラメータ, 2-14
インスタンスに対する一意の値, 2-17
INSTANCE_NUMBER
設定, 2-21
INSTANCE_NUMBER パラメータ, 2-14, 2-21
SQL オプション, B-3
インスタンスに対する一意の値, 2-21
推奨設定, 2-21
排他モードまたは共有モード, 2-21
INSTANCE 句
SHOW INSTANCE コマンド, 4-18

L

「List of Cluster Database Instances」 ページ, 8-13,
8-21
「List of Cluster Databases」 ページ, 8-13, 8-20
LISTENER
共有サーバーのパラメータ, 2-17
LOCAL 句
チェックポイントの強制, 4-21
LOG_ARCHIVE_DEST_n パラメータ, 6-12
LOG_ARCHIVE_DEST パラメータ, 2-13, 6-5
リカバリのための指定, 6-5
LOG_ARCHIVE_FORMAT パラメータ, 2-17, 6-11,
6-12
すべてのインスタンスで同一, 6-5
リカバリへの使用, 6-5
LOG_ARCHIVE_START パラメータ
自動アーカイブ, 2-11
LSNRCTL ユーティリティ
START コマンド, 4-14, 4-15

M

MAX_COMMIT_PROPAGATION_DELAY パラ
メータ, 2-14, 2-18
MAXEXTENTS 記憶域パラメータ
自動割当て, B-4
MAXINSTANCES オプション, B-2
MAXINSTANCES 句, B-2
MAXINSTANCES パラメータ, B-3
空きリストのインスタンスへの割当て, B-2
MAXLOGHISTORY オプション, 3-8
MINEXTENTS 記憶域パラメータ
自動割当て, B-4
modulo, B-2

N

「Navigator」 ウィンドウ
「Cluster Database Instances」 フォルダ, 5-6
Oracle Cluster Database データベースの起動, 5-8
起動結果の表示, 5-11
クラスタ・データベース・イベントの設定, 5-24
クラスタ・データベース・オブジェクト, 5-3
クラスタ・データベースの状態の表示, 5-15
コンテキスト・メニュー, 5-7, 5-11
Edit, 5-15
Related Tools, 5-7
Results, 5-7
Shutdown, 5-7, 5-9
Startup, 5-7, 5-8
View Edit Details, 5-7
停止結果の表示, 5-11
データベースの停止, 5-9
NOORDER オプション、CREATE SEQUENCE, 3-11

O

「Operations」 ページ, 8-11
ORA-00600, A-5
Oracle Cluster Database
停止, 5-11
Oracle Cluster Database データベース
マウント, 5-8
Oracle Cluster Database の停止, 5-11
Oracle Enterprise Manager
イベント
クラスタ・データベース・イベント, 5-24

管理の実行, 1-3

起動

Oracle Cluster Database, 5-8

ジョブのスケジューリング, 5-18

データベースの停止, 5-9

表示

起動結果, 5-11

クラスタ・データベースの状態, 5-15

停止結果, 5-11

Oracle Real Application Clusters

ジョブの作成, 5-18

ORACLE_SID パラメータ, 2-14

Oracle8i

Oracle9i へのアップグレード, 4-13

Oracle9i

Oracle8i からのアップグレード, 4-13

Oracle9i Real Application Clusters

概要, 1-1

Oracle レイヤー

ノードの追加, 8-8

oraxxxx.trc ファイル, A-3

ORDER オプション, 3-10, 3-11

「Output」タブ

「Cluster Database Operation Results」ダイアログ・ボックス, 5-14

P

PARALLEL_MAX_SERVERS パラメータ, 7-17

「Parameters」タブ

「Create Job」プロパティ・シート, 5-21

PRKA メッセージ, C-2

PRKC メッセージ, C-4

PRKI メッセージ, C-7

PRKO

メッセージ, C-8

PRKP

メッセージ, C-11

PRKR

メッセージ, C-13

PROCESSES パラメータ, 2-18

PROTOCOL

共有サーバーのパラメータ, 2-17

R

RAW デバイスの構成情報

インポート, 4-13

エクスポート, 4-13

Real Application Clusters

初期化パラメータ・ファイル, 2-7

Real Application Clusters データベース

起動, 4-14

RECOVERY_PARALLELISM パラメータ, 2-18, 7-16, 7-17

RECOVER コマンド, 4-21, 7-17

REDO ログ

フォーマットおよび接続先の指定, 6-11

REDO ログ・グループ, 3-7

REDO ログ・ファイル

インスタンス・リカバリ, 7-10

使用, 3-7

制御ファイルでの識別, 3-8

ログ順序番号, 6-11

REDO ログ・ファイルのアーカイブ

制御ファイルでの識別, 3-8

ログ順序番号, 6-11

RETRY オプション

STARTUP PARALLEL コマンド, 4-16

ROLLBACK_SEGMENTS パラメータ, 2-14, 2-18, 3-6

ROW_LOCKING パラメータ, 2-13

S

「Select Instances to Start」ダイアログ・ボックス, 5-8

「Select Instances to Stop」ダイアログ・ボックス, 5-10

SERIALIZABLE パラメータ, 2-13

Server Control

使用, 4-2

Server Management

インスタンスの管理, 5-2

SERVICE_NAMES パラメータ, 2-14

SESSIONS_PER_USER パラメータ, 2-18

SET INSTANCE コマンド, 2-20, 4-17

インスタンスの起動, 4-18

例, 2-20

SHOW INSTANCE コマンド, 4-18, 4-21

SHOW PARAMETERS コマンド, 4-21

例, 2-21

SHOW PARAMETER コマンド, 4-21

SHOW SGA コマンド, 4-21

SHUTDOWN ABORT コマンド, 4-20
SHUTDOWN TRANSACTIONAL, 4-20
SHUTDOWN コマンド, 4-21
 ABORT オプション, 4-20
 IMMEDIATE オプション, 4-20
 インスタンスの指定, 4-18
siddbwr.trc ファイル, A-2
sidlckn.trc ファイル, A-2, A-4
sidlmdn.trc ファイル, A-2
sidlmon.trc ファイル, A-3
sidp00n.trc ファイル, A-3
sidsmon.trc ファイル, A-2
SMON プロセス
 SHUTDOWN ABORT 後のリカバリ, 4-20
 インスタンス・リカバリ, 7-10, 7-11
SPFILE
 作成, 2-2
 パラメータ, 2-19
SQL*Plus セッション
 複数, 4-20
SQL 文
 インスタンス固有, 4-21
SRVCONFIG
 RAW デバイスの構成のインポートおよびエクスポート, 4-13
SRVCTL ユーティリティ
 インスタンス構成の管理のための使用, 4-4
 使用, 4-2
STARTUP コマンド, 2-20, 4-21
 インスタンスの指定, 4-18
「Status Details」タブ, 5-12
 「Cluster Database Operation Results」ダイアログ・ボックス, 5-12
「Status」タブ
 「Edit Cluster Database」ダイアログ・ボックス, 5-16
「Summary」ダイアログ・ボックス, 8-17, 8-23
SYSDBA
 接続のための権限, 4-18
SYSDBA 権限, 5-22
SYSOPER 権限, 5-22
 接続, 4-18
SYSTEM ロールバック・セグメント, 3-5

T

「Tasks」タブ
 「Create Job」プロパティ・シート, 5-21
THREAD 句, 4-21
THREAD パラメータ, 2-14, 2-19

U

UNDO_MANAGEMENT パラメータ, 3-3, 3-6
UNDO_TABLESPACE パラメータ, 2-14, 3-3
UNDO 表領域
 切替え, 3-4
USER_DUMP_DEST パラメータ, A-3

V

V\$ACTIVE_INSTANCES, 4-19
V\$ACTIVE_INSTANCES 表, 5-16
V\$LOGFILE ビュー, 3-8

W

「Welcome」ページ, 8-9
Windows NT 版 Cluster Setup のエラー・メッセージ (PRKI), C-7

あ

アーカイブ・ロギング
 使用可能化, 6-12
アーカイブ・ログ
 ファイル・フォーマットおよび接続先, 6-11
空きリスト, B-3
空きリスト・グループ
 インスタンスへの割当て, B-2
空き領域
 SQL による管理, B-3
アップグレード
 Oracle8i 構成から Oracle9i, 4-13
アフィニティ
 認識, 7-3
アラート・ログ, A-3
 管理, 3-9

い

インスタンス

- DBCA で削除, 8-19
- initsid.ora ファイル, 2-7
- Oracle レイヤーへの追加, 8-9
- Server Management, 5-2
- sid, 2-8
- 空きリスト・グループとの対応付け, B-2
- インスタンス番号, B-2
- 確認, 4-19
- 起動および停止の結果, 5-11, 5-15
- 起動順序, 2-21
- 現行, 4-18
- 最大数, B-2
- 障害, 7-11
- 障害、リカバリ, 7-10
- 初期化パラメータ・ファイル, 2-2
- スレッド番号, 2-21
- 接続, 4-17
- 設定, 4-17
- 停止, 4-19
- データ・ブロックとの対応付け, B-3
- リカバリ, 4-20, 7-10
- リカバリ、異常終了, 4-20
- リカバリ、データ・ファイルのアクセス, 7-11
- リカバリ、ファイルへのアクセス, 7-11
- リカバリ、複数障害, 7-11
- リモート, 2-20, 4-18
- インスタンスからの切断, 4-18
 - 複数のセッション, 4-20
 - ユーザー・プロセス, 4-19
- インスタンスの管理
 - Server Management, 5-2
- インスタンスの設定, 4-17
- インスタンスの停止, 4-19
 - 異常終了, 4-20
- 失われた順序番号, 3-10
- インスタンスへの接続, 4-17
- インポート
 - RAW デバイスの構成情報, 4-13

え

エクスポート

- RAW デバイスの構成情報, 4-13
- エラー
 - ORA-00600, A-5
 - コール・トレース・スタック, A-4
- エラー・メッセージ
 - 管理ツール, C-1
 - パラメータ値, 2-21

お

オペレーティング・システム

- 権限, 4-22
- オペレーティング・システム固有の Oracle マニュアル
 - アーカイブ REDO ログ名, 6-11
- オンライン REDO ログ・ファイル
 - REDO スレッド, 2-21
- オンライン・リカバリ, 7-10, 7-11

か

拡張性

- インスタンスの追加, 8-9
- 概要, 1-4
- ノードおよびインスタンスの追加, 8-1
- ノードの追加, 8-2

可用性

- 単一ノード障害, 7-10
- データ・ファイル, 7-11
- リカバリの手順, 7-12

管理

- SQL および SQL*Plus, 4-14
- SRVCTL, 4-2
- 問題、一般, 1-2

き

起動

- Oracle Cluster Database, 5-8
- Real Application Clusters データベース, 4-14
- 起動順序, 2-21
- クラスタ・データベース・パラメータの設定, 5-21
- 結果, 5-11
- パラメータ, 2-19
- リスナー, 4-14, 4-15

- リモート・インスタンス, 2-20, 4-18
- リモート・ノード, 2-20
- 起動タイプ, 5-8
 - Force, 5-8
 - Mount, 5-8
 - No Mount, 5-8
 - Open, 5-8
 - Restrict, 5-8
- 機能、新規, xxiii
- キャッシュ
 - 順序キャッシュ, 3-9, 3-10
- 競合
 - 順序番号, 3-9, 3-10
- 共通パラメータ・ファイル
 - 複数の使用, 2-11
- 共有サーバー
 - パラメータ, 2-17
- 共有モード
 - インスタンス番号, 2-21
 - インスタンス・リカバリ, 7-10

く

- クライアント側のパラメータ・ファイル
 - 使用, 2-6
 - ネーミング規則, 2-6
- クラスタウェア・レイヤー
 - ノードの追加, 8-2
- クラスタ・コマンドのエラー・メッセージ (PRKC), C-4
- クラスタ・データベース
 - コンソールへのオブジェクトの表示, 5-3
 - 切断, 5-9
 - 宛先タイプに関するタスク, 5-21
- クラスタ・データベース管理のエラー・メッセージ (PRKP), C-11
- クラスタへのノードの追加, 8-2
- グループ
 - REDO ログ・ファイル, 3-8
 - V\$LOGFILE, 3-8
 - 一意の番号, 3-8
- グローバルゼーション・サポート・パラメータ, 2-18
- グローバル・キャッシュ・サービス・リソース
 - 順序, 3-9
- グローバル・サービス・デーモン (GSD), 4-3
- グローバル・サービス・デーモンのエラー・メッセージ (PRKA), C-2

- グローバル定数パラメータ
 - すべてのインスタンスで同一, 2-13

こ

- コミットされたデータ
 - インスタンス障害, 7-10
 - 順序番号, 3-10
- コメント
 - サーバー・パラメータ・ファイルでの指定, 2-5
- コンソール
 - Oracle Cluster Database データベースの起動, 5-8
 - 起動結果の表示, 5-11
 - クラスタ・データベース・イベントの設定, 5-24
 - クラスタ・データベース・オブジェクトが表示された「Navigator」ウィンドウ, 5-3
 - クラスタ・データベースの状態の表示, 5-15
 - コンテキスト・メニュー, 5-7
 - Edit, 5-15
 - Related Tools, 5-7
 - Results, 5-7, 5-11
 - Shutdown, 5-7, 5-9
 - Startup, 5-7, 5-8
 - View Edit Details, 5-7
 - ジョブのスケジューリング, 5-18
 - 停止結果の表示, 5-11
 - データベースの停止, 5-9
- コンソール、起動, 5-2

さ

- サーバー制御ユーティリティ (SRVCTL) のエラー・メッセージ (PRKO), C-8
- サーバー・パラメータ・ファイル
 - 値の設定, 2-2
 - インスタンス固有の設定, 2-5
 - エクスポート, 2-4
 - 使用, 2-2
 - バックアップ, 2-2
- サービスの起動
 - gsdctl start, 4-14
- サービスの停止および削除
 - gsdctl, 4-14
- 作成
 - SPFILE, 2-2

し

- システム・グローバル領域 (SGA)
 - 順序キャッシュ, 3-10
- システム変更番号
 - アーカイブ・ファイル形式, 6-11
- 自動 UNDO 管理, 2-18, 3-3
 - オーバーライド, 3-6
 - 使用, 3-2, 3-3
- 順序
 - CACHE オプション, 3-10
 - ORDER オプション, 3-11
 - キャッシュされない, 3-11
 - タイムスタンプ, 3-11
 - データ・ディクショナリ・キャッシュ, 3-9, 3-10
 - ログ順序番号, 6-11
- 順序番号ジェネレータ, 3-9
 - Real Application Clusters データベース, 3-10
 - 順序番号のスキップ, 3-10
 - 制限, 3-11
 - 分散リソース, 3-10
- 障害
 - インスタンス, 7-10
 - インスタンス・リカバリ, 7-11
 - ノード, 7-10
 - ファイルへのアクセス, 7-11
 - 複数ノード, 7-11
- 初期化パラメータ
 - RECOVERY_PARALLELISM, 7-17
 - クラスタ・データベースの設定, 2-13
 - クラスタ・データベースの問題, 2-15
 - 重複する値, 2-10
 - すべてのインスタンスで一意, 2-14
 - すべてのインスタンスで同一, 2-13
- 初期化パラメータ・ファイル
 - initdb_name.ora, 2-9
 - initsid.ora, 2-8
 - インスタンス, 2-2
 - 定義, 2-7
- ジョブ
 - クラスタ・データベースに対する作成, 5-18
 - サーバー管理タスクに関するパラメータの指定, 5-21
 - 実行するための権利の設定, 5-22
 - 情報の指定, 5-19
 - タスクの選択, 5-21
- 新機能, xxiii

す

- スレッド
 - REDO, 3-7
 - アーカイブ・ファイル形式, 6-11
 - グループの番号, 3-8
 - 排他モード, 2-21
 - 例, 3-7

せ

- 制御ファイル
 - MAXLOGHISTORY, 3-8
- 制限
 - キャッシュされた順序, 3-11
- セッション
 - 複数, 4-18, 4-20
- 接続
 - インスタンス, 4-17
 - リモート・インスタンス, 4-17
- 接続文字列, 4-18

た

- 大規模パラレル処理システム, B-2
- 多重 REDO ログ・ファイル, 3-7
- 例, 3-8

て

- 停止
 - インスタンス, 4-19
 - クラスタ・データベース、パラメータの設定, 5-23
 - データベース, 5-9
- 停止タイプ, 5-9
 - Abort, 5-10
 - Immediate, 5-9
 - Normal, 5-9
 - Shutdown Database and Other Services, 5-10
 - Shutdown Database Only, 5-10
- データ・ディクショナリ
 - 順序キャッシュ, 3-10
- データ・ファイル
 - インスタンス・リカバリ, 7-11
 - インスタンス・リカバリのためのアクセス, 7-11
 - 追加, 3-2

データベース
アーカイブ・ログ・ファイルの数, 3-8
インスタンス数, B-2
停止, 4-20
データベースの停止
Real Application Clusters, 4-20
結果の表示, 5-11

と

トラブルシューティング, A-1 ~ A-5
Database Configuration Assistant, 8-10
概要, 1-4
トレース・ファイル, A-2
トランザクション
インスタンス障害, 7-10
順序番号, 3-10
リカバリ待機, 7-10
ロールバック, 7-10
トレース・ファイル, A-2
alrt_sid.log, A-3
oraxxxx.trc, A-3
siddbwr.trc ファイル, A-2
sidlckn.trc, A-2, A-4
sidlmdn.trc, A-2
sidlmon.trc, A-3
sidp00n.trc, A-3
sidsmon.trc ファイル, A-2
エラー・コール・トレース・スタック, A-4
管理, 3-9
バックグラウンド・スレッド・トレース・ファイル, A-2

ね

ネーミング規則
クライアント側のパラメータ・ファイル, 2-6

の

ノード
Oracle レイヤーへの追加, 8-8
アフィニティの認識, 7-3
クラスタウェア・レイヤーへの追加, 8-2
クラスタへの追加, 8-2
障害, 7-10

リモート, 4-18
ローカル, 2-20

は

排他モード
スレッド番号の指定, 2-21
バックアップ
概要, 1-3
バックグラウンド・スレッド・トレース・ファイル, A-2
バックグラウンド・プロセス
SMON, 4-20, 7-10
バッファ・キャッシュ
インスタンス・リカバリ, 7-10
パフォーマンス
順序のキャッシュ, 3-10
順序番号, 3-11
パブリック・ロールバック・セグメント, 3-5
パラメータ
管理, 1-2
クラスタ・データベースの設定, 2-13
サーバー・パラメータ・ファイルのインスタンス固有の設定, 2-5
初期化, 2-2, 3-1
すべてのインスタンスで同一, 2-13
タイプ, 2-13
データベースの作成, B-2
パラメータ・ファイル
PFILE, 2-20
位置, 2-12
インスタンス固有, 2-12
共通ファイル, 2-20
クライアント側、使用, 2-6
サーバー・ファイル、エクスポート, 2-4
サーバー・ファイル、使用, 2-2
サーバー・ファイル、バックアップ, 2-2
サーバー・ファイル、パラメータの設定, 2-2
重複する値, 2-10
リモート・インスタンス, 2-20, 4-18
パラレル・モード
順序の制限, 3-11
パラレル・リカバリ, 7-17
番号ジェネレータ, 3-10

ひ

- 表
 - エクステンツの割当て, B-3
- 表領域
 - UNDO の切替え, 3-4

ふ

- ファイル
 - ALERT, 7-11
 - REDO ログ, 6-11
 - REDO ログのアーカイブ, 6-11
 - 制御ファイル, 3-8
- ファイル管理, 3-2
- フォアグラウンド・プロセス
 - インスタンスの停止, 4-19
- 複数ノード障害, 7-11
- 複数のノード
 - 1つのノードからの開始, 2-20
- プライベート・ロールバック・セグメント, 2-14, 3-5
 - 指定, 3-6
- ブロック
 - インスタンスとの対応付け, 7-10, B-3
- 分散リソース
 - 順序, 3-10

へ

- 並行性
 - インスタンスの最大数, B-2
 - 順序, 3-11
- 変更データ
 - インスタンス・リカバリ, 7-10

め

- メッセージ
 - ALERT ファイル, 7-11
 - インスタンスの停止, 4-19
 - 管理ツールのエラー, C-1
 - ファイルへのアクセス, 7-11
- メディア障害
 - リカバリ, 7-13
- メモリー
 - SGA, 3-10

ゆ

- ユーザー
 - 空きリスト・グループとの対応付け, B-2
- ユーザー・トレース・ファイル, A-3
- ユーザー・プロセス
 - 空きリスト, B-3
 - 空きリストとの対応付け, B-3
 - インスタンス停止エラー, 4-19

り

- リカバリ
 - PARALLEL_MAX_SERVERS パラメータ, 7-17
 - SHUTDOWN ABORT 後, 4-20
 - インスタンス, 4-20, 7-10
 - オンライン, 7-10
 - 概要, 1-3
 - 単一ノード障害, 7-10
 - 手順, 7-12
 - パラレル, 7-17
 - ファイルへのアクセス, 7-11
 - 複数ノード障害, 7-11
 - メディア障害, 7-13
- リソース
 - 空きリスト・グループとの対応付け, B-2
 - 解放, 7-10
- リポジトリのエラー・メッセージ (PRKR), C-13
- リモート・インスタンス, 2-20, 4-18
- 領域
 - 管理, 1-2

ろ

- ローカル・インスタンス
 - ノード, 4-18
- ロールバック
 - インスタンス・リカバリ, 7-10
- ロールバック・セグメント
 - SYSTEM, 3-5
 - パブリック, 3-5
 - プライベート, 3-5
- ログ順序番号, 6-11
- ログ・スイッチ, 3-8

わ

割当て

空き領域, B-3

順序番号, 3-10

