

Veritas™ High Availability Agent for Oracle インストール / 設定ガイド

Solaris

5.0

Veritas High Availability Agent for Oracle インストール / 設定ガイド

Copyright © 2006 Symantec Corporation. All rights reserved.

VCS 5.0

Symantec、Symantec ロゴ、Veritas は、Symantec Corporation または同社の米国およびその他の国における関連会社の商標または登録商標です。その他の会社名、製品名は各社の登録商標または商標です。

本書に記載する製品は、使用、コピー、頒布、逆コンパイルおよびリバース・エンジニアリングを制限するライセンスに基づいて頒布されています。Symantec Corporation からの書面による許可なく本書を複製することはできません。

Symantec Corporation が提供する技術文書は Symantec Corporation の著作物であり、Symantec Corporation が保有するものです。

保証の免責：技術文書は現状有姿で提供され、Symantec Corporation はその正確性や使用について何ら保証いたしません。技術文書またはこれに記載される情報はお客様の責任にてご使用ください。本書には、技術的な誤りやその他不正確な点を含んでいる可能性があります。Symantec は事前の通知なく本書を変更する権利を留保します。

使用を許諾されるソフトウェアおよび関連書類は、FAR section 12.212 および DFARS section 227.7202 に定義される「commercial computer software (商用コンピュータ・ソフトウェア)」および「commercial computer software documentation (商用コンピュータ・ソフトウェア説明書類)」であると見なされます。

サードパーティ（第三者）製ソフトウェアの権利に関する通知

本製品には、特定のサードパーティ製ソフトウェアが配布、組み込み、または同梱されている場合があります。また、本製品のインストールおよび使用にともない、サードパーティ製ソフトウェアの使用を推奨する場合があります。同サードパーティ製ソフトウェアのライセンスは、著作権の所有者により別途付与されます。サードパーティのソフトウェアの使用に必要なライセンスおよび著作権に関する情報については、本製品リリースノートのサードパーティに関する章を参照してください。

Solaris は Sun Microsystems, Inc. の商標です。
Oracle は Oracle Corporation の登録商標です。

ライセンスと登録

Veritas Cluster Server はライセンスが必要な製品です。ライセンスのインストールについては、『Veritas Cluster Server インストールガイド』を参照してください。

テクニカルサポート

製品のサポートを受けるには、<http://support.veritas.com> ページへアクセスし「Phone Support」または「E-mail Support」をクリックします。このページから TechNote、Software Alerts、ソフトウェアのダウンロード、ハードウェア互換性リスト、VERITAS Email Notifications サービスなどにアクセスすることもできます。「Knowledge Base Search」機能を使用し、製品ドキュメントのリリースなどの製品情報へアクセスすることができます。

目次

第 1 章	Veritas High Availability Agent for Oracle の概要	
	Veritas High Availability Agent for Oracle について	9
	このリリースの新機能	10
	サポートするソフトウェア	10
	Oracle エージェントの機能	11
	起動および停止オプション	12
	起動オプション	12
	停止オプション	13
	Oracle エージェントの監視オプション	14
	基本監視オプション	14
	詳細監視	14
	情報エントリポイント	14
	アクションエントリポイント	15
	Netlsnr エージェントの機能	17
	エージェントが Oracle の高可用性を保証するしくみ	18
	エージェントが Solaris 10 ゾーンで実行中の Oracle インスタンスを 監視するしくみ	18
	詳細監視での Oracle エラーコードの操作	19
	VCS cluster における典型的な Oracle 設定	20
	Oracle cluster の設定について	21
第 2 章	Oracle のインストールと設定	
	Oracle をインストールする前に	23
	Oracle をインストールするための VCS の必要条件	23
	\$ORACLE_HOME の場所	24
	複数の Oracle インスタンス (SID)	24
	データベース表領域の場所	25
	コアファイルの場所	25
	リスナーの透過的フェールオーバー	25
	\$ORACLE_HOME での長いパス名の制限	26
	属性内にある長いパス名の置換	26
	NLS 情報の定義	26
	Oracle データベースと VCS のホットバックアップ	26
	VCS 環境での Oracle のインストール	27
	Oracle インストールタスクの概要	27
	共有ディスクで行う \$ORACLE_HOME のインストールタスク	28

ローカルディスクで行う \$ORACLE_HOME の インストールタスク	29
Oracle バイナリのインストール	29
Oracle 10g クラスタデーモンの無効化	32
Oracle データベースの設定	32
\$ORACLE_BASE/admin/SID ディレクトリのコピー	35
データベースの起動	35
\$ORACLE_HOME が共有ディスク上にある場合の データベースの起動	35
\$ORACLE_HOME がローカルディスク上にある場合の データベースの起動	35

第 3 章

Oracle 用エージェントのインストール

Oracle 用エージェントをインストールまたはアップグレードする前に	37
エージェントソフトウェアのインストール	38
エージェントのアップグレード	39
VCS Agent 3.5 for Oracle のアップグレード	39
VCS Agent 4.0 for Oracle と VCS Agent 4.1 for Oracle の アップグレード	40

第 4 章

Oracle 用 VCS サービスグループの設定

Oracle 用サービスグループの設定について	41
VCS における Oracle の設定について	42
VCS における単一の Oracle インスタンスの設定	42
VCS における複数の Oracle インスタンス（単一リスナー）の設定	43
VCS における複数の Oracle インスタンス（複数リスナー）の設定	45
VCS における共有サーバーサポートの Oracle インスタンスの設定	46
サービスグループを設定する前に	48
OracleTypes.cf ファイルのインポート	48
サービスグループの設定	49
エージェント設定ウィザードを使ったサービスグループの設定	50
Cluster Manager (Java コンソール) を使った サービスグループの設定	60
コマンドラインを使ったサービスグループの設定	62
パスワードの暗号化	64
基本監視についてのその他の注意事項	65
詳細監視の設定	65
Oracle に対する詳細監視の設定	66
Oracle に対する詳細監視の有効化と無効化	68
Netlsnr に対する詳細監視の設定	68

第 5 章	Oracle 用 VCS サービスグループの管理	
	VCS サービスグループの管理について	69
	サービスグループのオンライン化	69
	サービスグループのオフライン	70
	サービスグループの切り替え	70
	サービスグループ設定の修正	71
	Veritas High Availability Agent for Oracle の無効化	71
	Veritas High Availability Agent for Oracle の削除	72
第 6 章	トラブルシューティング	
	Oracle および Netlsnr エージェント共通のエラーメッセージ	73
	Oracle エージェント固有のエラーメッセージ	75
	Netlsnr エージェント固有のエラーメッセージ	79
付録 A	リソースタイプの定義	
	リソースタイプの定義と属性の定義について	81
	Oracle リソースタイプ	81
	Oracle 属性の定義	82
	必須属性	82
	省略可能な属性	83
	内部属性	86
	Netlsnr リソースタイプ	87
	Netlsnr 属性の定義	87
	必須属性	87
	省略可能な属性	88
	内部属性	90
付録 B	設定例	
	Oracle エンタープライズエージェントの設定例について	91
	単一の Oracle インスタンスの設定例	91
	単一の Oracle インスタンスの設定ファイルの例	92
	Solaris ゾーンに設定された単一の Oracle インスタンス	94
	ローカルディスク上のゾーンルート	94
	共有ディスク上のゾーンルート	96
	複数の Oracle インスタンス (単一リスナー) の設定例	97
	複数の Oracle インスタンス (単一リスナー) の設定ファイルの例	99
	Solaris ゾーンに設定された複数の Oracle インスタンス	
	(単一リスナー)	102
	ローカルディスク上のゾーンルート	103
	共有ディスク上のゾーンルート	105

複数のインスタンス（複数リスナー）の設定例	107
設定ファイルの例	107
共有サーバーサポートの Oracle の設定例	111
共有サーバーサポートで設定された Oracle インスタンスの 設定ファイルの例	112
Solaris ゾーンに設定された共有サーバーサポートの Oracle インスタンス	113
ローカルディスク上のゾーンルート	113
共有ディスク上のゾーンルート	115

付録 C

ベストプラクティス

複数の Oracle インスタンスを設定する場合のベストプラクティス	117
------------------------------------------	-----

付録 D

Oracle 用 VCS クラスタにおける SPFILE の使用

Oracle 初期化パラメータファイルについて	119
Oracle インスタンスの起動	119
VCS クラスタでの SPFILE の使用	120
PFILE における SPFILE の位置指定	120
PFILE からの SPFILE 作成	120

索引

121

Veritas High Availability Agent for Oracle の概要

この章では、次の内容について説明します。

- [Veritas High Availability Agent for Oracle](#) について
- このリリースの新機能
- サポートするソフトウェア
- [Oracle](#) エージェントの機能
- [Netlsnr](#) エージェントの機能
- エージェントが [Oracle](#) の高可用性を保証するしくみ
- [VCS cluster](#) における典型的な [Oracle](#) 設定
- [Oracle cluster](#) の設定について

Veritas High Availability Agent for Oracle について

Veritas High Availability Agent for Oracle は、[Oracle](#) での高可用性を実現します。Veritas 高可用性エージェントは、エンタープライズアプリケーション内の特定のリソースを監視してその状態を特定し、外部のイベントに従ってこれらのリソースを起動または停止します。

Veritas High Availability Agent for Oracle は、[Oracle](#) とリスナープロセスの監視、オンライン化およびオフライン化を行います。

このパッケージには、次の 2 つのエージェントが含まれています。

- [Oracle](#) エージェントは、[Oracle](#) データベースプロセスを監視します。
- [Netlsnr](#) エージェントは、リスナープロセスを監視します。

これらのエージェントには、リソースタイプの宣言とエージェント実行ファイルが含まれており、[Oracle](#) エージェントは [Oracle](#) リソースタイプ、[Netlsnr](#) エー

エージェントは **Netlsnr** リソースタイプでそれぞれ表されます。この両方のエージェントが連携することにより、**Oracle** の高可用性が実現されます。

このリリースの新機能

Veritas High Availability Agent for Oracle には、次の新機能や拡張された機能が含まれています。

- 新しい監視オプション
Oracle エージェントの基本監視オプションによって、プロセス検査監視に加えて、ヘルスチェック監視を実行できるようになりました。Oracle 10g以降に対して、診断監視オプションを選択できます。
14 ページの「[Oracle エージェントの監視オプション](#)」を参照してください。
- 仮想ファイアドリルの実行
VCS では、基本になるインフラストラクチャと設定との同期をノードで維持する必要があります。仮想ファイアドリルは、特定のノードでサービスグループがオンライン状態になるのを妨げるような不一致を検出します。詳しくは『[Veritas Cluster Server ユーザーズガイド](#)』を参照してください。
Veritas High Availability Agent for Oracle は **action** エントリポイントを使って仮想ファイアドリル機能をサポートします。
15 ページの「[アクションエントリポイント](#)」を参照してください。
- Veritas High Availability Agent for Oracle 5.0 では、エージェントのバイナリディレクトリの場所は `/opt/VRTSagents/ha/bin` です。AgentDirectory 属性はこの場所を示します。
81 ページの「[リソースタイプの定義](#)」を参照してください。

サポートするソフトウェア

Veritas High Availability Agent for Oracle 5.0 は、Veritas Cluster Server (VCS) 環境で次のソフトウェアバージョンをサポートします。

Oracle Oracle9i、Oracle 10g R1、Oracle 10g R2 (64 ビットバージョンを含む)

VCS Solaris 環境の VCS 5.0

Solaris Solaris 8、9 および 10 (32 ビットおよび 64 ビット)

メモ: 各プラットフォームに対して、Sun から提供されている最新の Solaris オペレーティングシステムのパッチを適用することを推奨します。Sun の Web サイトにアクセスして、最新のパッチをダウンロードしてください。

メモ: cluster 内のすべてのシステムで、オペレーティングシステムのバージョンとパッチのレベルを同一にする必要があります。

Oracle エージェントの機能

Oracle エージェントは、データベースプロセスを監視します。表 1-1 に、Oracle エージェントの動作の一覧を示します。エージェントが実行する機能はエントリポイントと呼ばれます。

表 1-1 Oracle エージェントの動作

エージェントの動作	説明
Online	<p>次の svrmgr1 または sqlplus コマンドを使って Oracle データベースを起動します。</p> <pre>startup force pfile=\$PFile</pre> <p>デフォルトの起動オプションは STARTUP_FORCE です。Oracle の他の起動オプションを使ってデータベースを起動するようエージェントを設定することもできます。</p> <p>12 ページの「起動オプション」を参照してください。</p>
Offline	<p>次の svrmgr1 または sqlplus コマンドを使って、指定されたオプションで Oracle データベースを停止します。</p> <pre>shutdown immediate</pre> <p>デフォルトの停止オプションは IMMEDIATE です。Oracle の他の停止オプションを使ってデータベースを停止するようエージェントを設定することもできます。</p> <p>13 ページの「停止オプション」を参照してください。</p>
Monitor	<p>Oracle プロセスの状態を検証します。Oracle エージェントの監視には、基本と詳細の 2 種類のレベルがあります。</p> <p>14 ページの「Oracle エージェントの監視オプション」を参照してください。</p>
Clean	<p>次の svrmgr1 または sqlplus コマンドを使って Oracle データベースを強制的に停止します。</p> <pre>shutdown abort</pre> <p>プロセスが shutdown コマンドに 응답しない場合、エージェントはプロセステーブルをスキャンして設定済みインスタンスに関連付けられたプロセスを特定し、それらのプロセスに対して kill コマンドを実行します。</p>

表 1-1 Oracle エージェントの動作

エージェントの動作	説明
Info	データベースの状態に関する静的および動的な情報を提供します。 14 ページの「 情報エントリポイント 」を参照してください。
Action	リソースに対して、定義済みのアクションを実行します。 15 ページの「 アクションエントリポイント 」を参照してください。

起動および停止オプション

設定されている Oracle インスタンスに起動および停止オプションを指定できません。

起動オプション

表 1-2 に、エージェントがサポートする起動オプションの一覧を示します。

表 1-2 起動オプション

オプション	説明
STARTUP_FORCE (デフォルト)	pfile が設定されている場合は、 <code>startup force pfile='location_of_pfile'</code> を実行します。 pfile が設定されていない場合は、 <code>startup force</code> を実行します。デフォルトパラメータファイルを、デフォルトの場所から取得します。
STARTUP	pfile が設定されている場合は、 <code>startup pfile='location_of_pfile'</code> を実行します。 pfile が設定されていない場合、エージェントはデフォルトの場所からデフォルトパラメータファイルを取得し、 <code>startup</code> を実行します。
RESTRICTED	データベースを制限 (RESTRICTED) モードで起動します。
RECOVERDB	インスタンスの起動時に、データベースをリカバリします。

表 1-2 起動オプション

オプション	説明
CUSTOM	<p>定義済みの SQL スクリプト (start_custom_\$\$SID.sql) を使い、カスタム起動オプションを実行します。カスタムスクリプトは、/opt/VRTSagents/ha/bin/Oracle ディレクトリに置きます。ファイルの所有権をシステムに定義されている Oracle ユーザーに設定します。このファイルが存在しない場合、エージェントはログファイルにエラーメッセージを書き込みます。</p> <p>エージェントは、カスタムスクリプトを使って次の動作を実行します。</p> <pre>sqlplus /nolog <<! connect / as sysdba; @start_custom_\$\$SID.sql exit; !</pre>
SRVCTLSTART	<p>srvctl ユーティリティを使ってデータベースのインスタンスを起動します。</p> <p>Oracle9i R2 以降が稼動する RAC クラスタの場合は、デフォルトの起動オプションを手動で SRVCTLSTART に設定する必要があります。</p>

停止オプション

表 1-3 に、エージェントがサポートする停止オプションの一覧を示します。

表 1-3 停止オプション

オプション	説明
IMMEDIATE (デフォルト)	shutdown immediate を実行して Oracle インスタンスを停止します。
TRANSACTIONAL	shutdown transactional コマンドを実行します。このオプションは、それをサポートしているデータベースバージョンに対してのみ有効です。
CUSTOM	<p>定義済みの SQL スクリプト (shut_custom_\$\$SID.sql) を使い、カスタム停止オプションを実行します。カスタムスクリプトは、/opt/VRTSagents/ha/bin/Oracle ディレクトリに置きます。ファイルの所有権をシステムに定義されている Oracle ユーザーに設定します。このファイルがない場合、エージェントはデフォルトオプションで停止します。</p>

表 1-3 停止オプション

オプション	説明
SRVCTLSTOP	<p>srvctl ユーティリティを使ってデータベースのインスタンスを停止します。</p> <p>Oracle9i R2 以降が稼動する RAC クラスタの場合は、デフォルトのオプションを手動で SRVCTLSTOP に設定する必要があります。</p>

Oracle エージェントの監視オプション

Oracle エージェントの監視には、基本と詳細の 2 種類のレベルがあります。デフォルトでは、エージェントは基本監視を実行します。

基本監視オプション

基本監視モードには、プロセス検査とヘルスチェック検査という 2 つのオプションがあります。表 1-4 で、基本監視オプションについて説明します。

表 1-4 基本監視オプション

オプション	説明
0 (デフォルト)	<p>プロセス検査</p> <p>エージェントは、プロセステーブルで <code>ora_dbw</code>、<code>ora_smon</code>、<code>ora_pmon</code>、<code>ora_lgwr</code> の各プロセスをスキャンし、Oracle が実行中であることを検証します。</p>
1	<p>ヘルスチェック (Oracle 10g 以降)</p> <p>エージェントは Oracle の Health Check API を利用して SGA を監視し、インスタンスに関する情報を取り込みます。</p>

詳細監視

詳細監視レベルでは、データベースのテスト用表でトランザクションを実行し、Oracle が適切に機能するかどうかを確認します。

情報エントリポイント

Veritas High Availability Agent for Oracle は、データベースの状態に関する静的および動的な情報を提供する情報エントリポイントをサポートしています。

情報エントリポイントを呼び出すには、次のコマンドを入力します。

```
# hares -refreshinfo resource [-sys system] \  
[-clus cluster | -localclus]
```

エントリポイントでは、次の静的情報を取得できます。

- Version
- InstanceNo
- InstanceName
- DatabaseName
- HostName
- StartupTime
- Parallel
- Thread
- InstanceRole

エントリポイントでは、次の動的情報を抽出できます。

- InstanceStatus
- Logins
- OpenMode
- LogMode
- ShutdownPending
- DatabaseStatus
- Shared Pool Percent free
- Buffer Hits Percent

`/opt/VRTSagents/ha/bin/Oracle/resinfo.sql` ファイルに `sql` 文を追加して、他の属性を追加できます。たとえば、次のようにします。

```
select 'static:HostName:' || host_name from v$instance;
select 'dynamic:ShutdownPending:' || shutdown_pending from
v$instance;
```

選択されたレコードの形式は、次のようになります。

```
attribute_type:userkey_name:userkey_value
```

変数 `attribute_type` には、静的値または動的値のいずれも設定できます。

アクションエントリポイント

Veritas High Availability Agent for Oracle は、アクションエントリポイントをサポートしています。アクションエントリポイントを使うと、定義済みのアクションをリソースに対して実行できます。表 1-5 で、事前定義済みのエージェントのアクションについて説明します。

表 1-6 「事前定義済みの仮想ファイアドリルの処理」を参照してください。

リソースでアクションを実行するには、次のコマンドを入力します。

```
# hares -action res token [-actionargs arg1 ...] \
[-sys system] [-clus cluster]
```

このエージェントのカスタムアクションを追加することもできます。詳しくは『Veritas Cluster Server エージェント開発者ガイド』を参照してください。

表 1-5 事前定義済みのエージェントのアクション

アクション	説明
VRTS_GetInstanceName	設定されたインスタンスの名前を取得します。このオプションは、Oracle リソースおよび Netlsnr リソースに使えます。

表 1-5 事前定義済みのエージェントのアクション

アクション	説明
VRTS_GetRunningServices	エージェントによって監視されるプロセスのリストを取得します。このオプションは、Oracle リソースおよび Netlsnr リソースに使えます。
DBRestrict	データベースセッションを変更し、RESTRICTED モードを有効にします。
DBUndoRestrict	データベースセッションを変更し、RESTRICTED モードを無効にします。
DBSuspend	データベースを一時停止します。
DBResume	一時停止したデータベースを再開します。
DBTbspBackup	表領域をバックアップします。バックアップする表領域の名前は <code>actionargs</code> に格納されています。

表 1-6 に、Veritas High Availability Agent for Oracle の仮想ファイアドリルの処理の一覧を示します。これらの処理を使って、インフラストラクチャチェックを実行し、特定のエラーを修正できます。

表 1-6 事前定義済みの仮想ファイアドリルの処理

仮想ファイアドリルのアクション	説明
getid (Oracle エージェント)	Oracle 所有者がノードに存在することを検証します。
home.vfd (Oracle エージェント)	次の内容を検証します。 <ul style="list-style-type: none"> ■ ORACLE_HOME がノードにマウントされており、対応するエントリが <code>fstab</code> にあること。 ORACLE_HOME がマウントされていない場合、<code>action</code> エントリポイントは、その他のリソースがすでに ORACLE_HOME をマウントしていないかどうかをチェックします。 ■ <code>Pfile</code> が指定されており、その <code>Pfile</code> がノードに存在すること。 ■ <code>\$ORACLE_HOME/dbs/orapw[SID]</code> のパスワードファイルが存在すること。
owner.vfd (Oracle エージェント)	Oracle 所有者属性の <code>uid</code> と <code>gid</code> を検証します。Oracle リソースが現在 ONLINE であるノードで、所有者属性の <code>uid</code> と <code>gid</code> が同じであるかどうかをチェックします。

表 1-6 事前定義済みの仮想ファイアドリルの処理

仮想ファイアドリルのアクション	説明
tnsadmin.vfd (Netlsnr エージェント)	listener.ora ファイルが存在するかどうかをチェックします。 listener.ora ファイルが存在する場合、ORACLE_HOME がマウントされているかどうかをチェックし、適切なメッセージを表示します。

Netlsnr エージェントの機能

リスナーとは、クライアントの接続要求を待ち受け、データベースへのトラフィックを管理するサーバープロセスです。Netlsnr エージェントは、リスナーサービスをオンラインにし、その状態を監視し、オフラインにします。表 1-7 に、Netlsnr エージェントの動作の一覧を示します。

表 1-7 Netlsnr エージェントの動作

エージェントの動作	説明
Online	次のコマンドを使ってリスナープロセスを開始します。 <code>lsnrctl start \$LISTENER</code>
Offline	次のコマンドを使ってリスナープロセスを停止します。 <code>lsnrctl stop \$LISTENER</code> リスナーにパスワードが設定されている場合は、そのパスワードを使ってリスナーを停止します。
Monitor	リスナープロセスの状態を検証します。 Netlsnr エージェントの監視には、基本と詳細の 2 種類のレベルがあります。 <ul style="list-style-type: none"> ■ 基本監視モードでは、プロセステーブルで tnslistener プロセスをスキャンし、このリスナープロセスが実行中かどうかを確認します。(デフォルト) ■ 詳細監視モードでは、lsnrctl status \$LISTENER コマンドでリスナープロセスの状態を確認します。
Clean	tnslistener \$Listener をプロセステーブルでスキャンし、このプロセスに対して kill コマンドを実行します。
Action	リソースに対して、定義済みのアクションを実行します。 15 ページの「 アクションエントリポイント 」を参照してください。

エージェントが Oracle の高可用性を保証するしくみ

Veritas High Availability Agent for Oracle は、Oracle データベースとリスナープロセスを絶えず監視し、正常に機能しているかどうかを確認します。このエージェントには、次の 2 つのアプリケーション監視レベルがあります。

- プライマリまたは基本監視
基本監視モードでは、デフォルトのプロセス検査オプションを使うと、エージェントは Oracle およびリスナープロセスがプロセステーブルに存在しているかどうかを検証します。プロセス検査では、プロセスのハングまたは停止状態を検知できません。
14 ページの「[Oracle エージェントの監視オプション](#)」を参照してください。
- セカンダリまたは詳細監視
詳細監視モードでは、Perl スクリプトを実行し、データベースおよびリスナーに対してコマンドを発行してその状態を確認します。

監視ルーチンによって Oracle プロセスまたはリスナープロセスが正しく機能していないことが報告されると、エージェントはアプリケーションに障害があると判断します。この状態になると、Oracle サービスグループは cluster 内の別のノードにフェールオーバーされるため、Oracle サービスおよび Oracle データベースの高可用性が保証されます。

エージェントが Solaris 10 ゾーンで実行中の Oracle インスタンスを監視するしくみ

Solaris では、仮想オペレーティングシステムサービスの手段が用意されており、システム上の他のアクティビティと分離して、1 つまたは複数のプロセスを実行できます。このようなサンドボックスは、非グローバルゾーンと呼ばれています。各ゾーンは、カスタマイズした様々なサービスを提供できます。グローバルゾーンもあり、このゾーンで実行しているプロセスには、現在の Solaris システムで利用できる権限のセットと同じものが与えられます。

VCS では、フェールオーバー機能をゾーンに拡張することにより、非グローバルゾーン上で実行しているアプリケーションの高可用性を実現します。VCS はグローバルゾーンにインストールされ、すべての VCS エージェントとエンジンコンポーネントはグローバルゾーンで実行されます。非グローバルゾーン内で実行しているアプリケーションでは、エージェントにより、このゾーン内でスクリプトエントリポイントが実行されます。VCS による制御下で設定されたゾーンが障害が発生した場合、VCS はゾーンを含むサービスグループ全体をフェールオーバーします。

Veritas High Availability Agent for Oracle はゾーンに対応しており、非グローバルゾーンで実行されている Oracle インスタンスを監視できます。

詳細監視での Oracle エラーコードの操作

Veritas High Availability Agent for Oracle は、詳細監視中に Oracle のエラーを処理します。エージェントは、Oracle のエラーを重大度別に分類し、定義済みのアクションを各エラーコードに関連付けます。

エージェントには `oraerror.dat` という参照ファイルが付属しており、Oracle エラーとそのエラーが発生した場合に実行するアクションが一覧表示されています。

このファイルには、次の形式で情報が格納されています。

```
Oracle_error_string:action_to_be_taken
```

次に例を示します。

```
01035:WARN
```

```
01034:FAILOVER
```

表 1-8 に、Oracle エラーが発生したときにエージェントが実行する事前定義済みアクションの一覧を示します。

表 1-8 Oracle エラーの事前定義済みエージェントのアクション

アクション	説明
IGNORE	エラーを無視します。
UNKNOWN	リソースの状態を UNKNOWN と判断し、Notifier リソースが設定されている場合は通知を送信します。VCS の通知の詳細については、『Veritas Cluster Server ユーザーズガイド』を参照してください。 このアクションは、通常、設定エラーまたはプログラムインターフェースのエラーに関連付けられます。
WARN	リソースの状態を ONLINE と判断し、Notifier リソースが設定されている場合は通知を送信します。 このアクションは、通常、クォータの制限、セッションの制限または制限されたセッション数を越えたことによるエラーに関連付けられます。
FAILOVER (デフォルト)	リソースの状態を OFFLINE と判断します。これはサービスグループのエラーで、次の使用可能なシステムにフェールオーバーされます。 これがエージェントのデフォルトの動作です。oraerror.dat ファイルがない場合、エージェントは Oracle エラーが発生するたびにこのデフォルトの動作を実行します。
NOFAILOVER	サービスグループを一時的にフリーズし、リソースの状態を OFFLINE と判断します。Notifier リソースが設定されている場合は、通知も送信します。 このアクションは、通常、システム固有ではないエラーに関連付けられます。たとえば、Oracle ファイルが壊れているためノードからデータベースをオープンできない場合、その処理を他のノードにフェールオーバーしても役に立ちません。

VCS cluster における典型的な Oracle 設定

VCS cluster における典型的な Oracle 設定には次の特徴があります。

- VCS は 2 ノードの cluster で設定されます。
- Oracle データは共有ストレージにインストールされます。
- Oracle バイナリは、両方のノードでローカルにインストールされるか、共有ディスクにインストールされます。
- Veritas High Availability Agent for Oracle は両方のノードにインストールされます。

図 1-1 は、Oracle バイナリとデータがすべて共有ディスクにインストールされる設定を示しています。

図 1-1 共有ディスクに置かれた Oracle バイナリとデータ

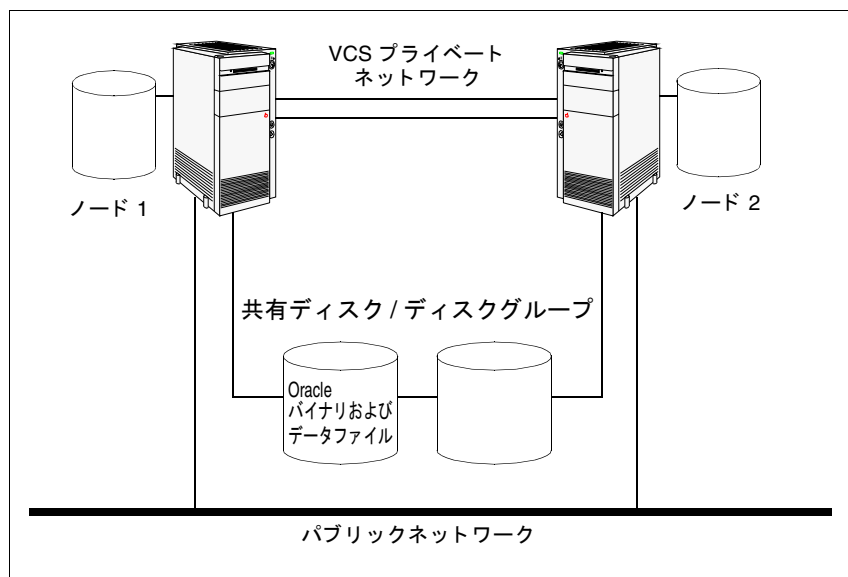
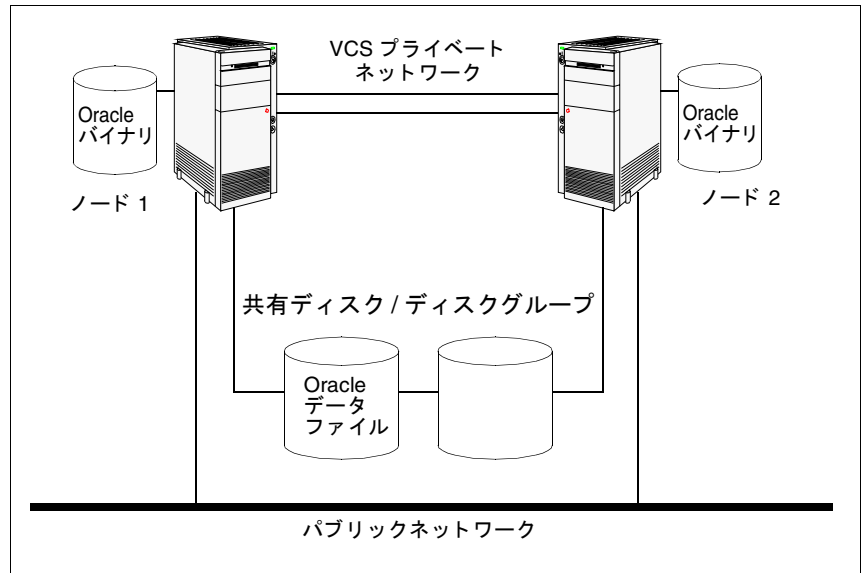


図 1-2 は、Oracle バイナリが各ノードでローカルにインストールされ、Oracle データが共有ディスクにインストールされる設定を示しています。

図 1-2 ローカルディスクに置かれた Oracle バイナリと共有ディスクに置かれた Oracle データ



Oracle cluster の設定について

VCS 環境での Oracle の設定に関するタスクには次が含まれます。

- VCS cluster の設定
VCS のインストールと設定について詳しくは『Veritas Cluster Server インストールガイド』を参照してください。
- Oracle のインストールと設定
- Oracle 用エージェントのインストール
- Oracle 用 VCS サービスグループの設定

Oracle のインストールと設定

この章では、次の内容について説明します。

- [Oracle をインストールする前に](#)
- [Oracle をインストールするための VCS の必要条件](#)
- [VCS 環境での Oracle のインストール](#)

Oracle をインストールする前に

- クラスタ内のすべてのノードに VCS がインストールされていることを確認します。
- クラスタ内のすべてのノードに、Oracle と VCS を実行するための十分なリソースがあることを確認してください。
- ネットワークが TCP/IP プロトコルをサポートしていることを確認してください。
- Oracle をインストールするための VCS の必要条件を満たしていることを確認してください。
23 ページの「[Oracle をインストールするための VCS の必要条件](#)」を参照してください。

Oracle をインストールするための VCS の必要条件

Solaris 上の VCS クラスタに Oracle をインストールする前に必要条件を見直し、それらを満たしていることを確認します。

\$ORACLE_HOME の場所

Oracle バイナリファイルと設定ファイルが格納された Oracle ホームディレクトリ（\$ORACLE_HOME）は、各サーバーのディスクにローカルに配置できます。また、Oracle ホームディレクトリを共有ストレージに配置することもできます。Oracle バイナリの適切な場所は、環境によって異なります。次に、それぞれの場合の利点について説明します。

共有ディスク上に \$ORACLE_HOME ディレクトリを配置した場合

Oracle データベースサーバー（\$ORACLE_HOME）を共有ディスクにインストールすると、共有ファイルシステムのマウントポイントディレクトリはクラスタ内のどのノードでも同じになります。Oracle バイナリを共有ストレージにインストールすると、クラスタ内の任意のノード上でインスタンスが実行できるよう、簡単にセットアップできます。それぞれのデータベースサービスグループは、自己完結しています。インスタンスを、クラスタ内でストレージを共有している新しいノードに移動することも可能です。

たとえば、4つのノードを持つクラスタで、3つのデータベースインスタンス（またはサービスグループ）でそれぞれバージョンの異なる Oracle を実行することが可能です。Oracle バイナリを共有ストレージに配置している場合は、共有ストレージ上に各バージョンにつき1つ、合計3つの Oracle バイナリが必要になります。これとは対照的に、Oracle バイナリをローカルストレージに配置している場合は、バイナリのコピーが最大で12個（4ノードにそれぞれ3つのバージョン）必要です。

この方法の欠点は、共有ストレージ上の Oracle バイナリのローリングアップグレードが行えないことです。

ローカルディスク上に \$ORACLE_HOME ディレクトリを配置した場合

ローカルディスクに Oracle データベースサーバー（\$ORACLE_HOME）をインストールすると、他のノードでデータベースサーバーが稼働している間に、オフラインのノードで Oracle データベースのバイナリをアップグレードできるという利点があります。データベースに互換性があれば、最小限のダウンタイムで、アップグレードしたノードに後で切り替えることができます。

この方法の欠点は、多数のノードを使った場合、様々な Oracle のインストール環境の保守が困難になることです。

複数の Oracle インスタンス（SID）

1つのクラスタ設定に複数の Oracle インスタンスが定義されている場合は、サービスグループの SystemList 属性に示されているすべてのノードから、各インスタンス用のパラメータファイルにアクセスする必要があります。

データベース表領域の場所

表領域が通常の (UFS または VxFS) ファイル上に作成されている場合、これらのファイルを含むファイルシステムは共有ディスク上に配置する必要があります。また、各ノードで、共有ディスク上のファイルシステム用に同一のマウントポイントを作成します。

共有ディスク上で Oracle 表領域用に RAW デバイスを使っている場合は、そのデバイスの所有権を Oracle dba ユーザーに変更し、Oracle データを含む raw デバイスのアクセス権モードを 660 に変更します。

たとえば、Veritas Volume Manager を使っている場合は、次のように入力します。

```
# vxedit -g diskgroup_name set group=dba \  
user=oracle mode=660 volume_name
```

メモ: ユーザー **oracle** とグループ **dba** は、Network Information Service (NIS および NIS+) ユーザーではなくローカルユーザーである必要があります。

コアファイルの場所

VCS エージェントフレームワークは、Oracle バイナリを実行する Oracle エージェントのスクリプトまたはプログラムを実行する前に、現在のディレクトリを /opt/VRTSagents/ha/bin/Oracle に設定します。ユーザー oracle として実行される Oracle バイナリは、/opt/VRTSagents/ha/bin/Oracle に書き込む権限を持っていません。このため、Oracle バイナリによって生成される「コア」ファイルはすべて失われます。Solaris で利用可能な coreadm (1M) コマンドを使って、異常終了したプロセスによって生成されるコアファイルの名前と場所を指定することを推奨します。

リスナーの透過的フェールオーバー

Oracle Server クライアントを、ノード切り替え後に再設定せずもう一度接続しなおせるようにするには、最低 1 つの仮想 IP を設定するリソースを Oracle のサービスグループに含めておきます。このリソースが設定する IP アドレスにマッピングされているホスト名は、\$TNS_ADMIN/listener.ora ファイルの Host フィールドに定義されている必要があります。

Oracle のクライアント / サーバー通信に TCP/IP プロトコルを使っている場合は、サービスグループの SystemList 属性に定義されている各ノード上のファイル /etc/services に、Oracle Net Service のサービス名へのエントリが含まれていることを確認してください。

\$ORACLE_HOME での長いパス名の制限

Solaris のプロセステーブルでは、プロセスのパス名が 79 文字に制限されます。しかし、\$ORACLE_HOME 内のプロセスの絶対パス名は 80 文字以上になる可能性があります。この場合は、\$ORACLE_HOME ディレクトリへのソフトリンクを作成し、それを main.cf ファイルの Home 属性内の長いファイル名の代わりに使うことができます。

属性内にある長いパス名の置換

クラスタ内の各ノードでソフトリンクを作成する必要があります。

長いパス名を置換するには

- 1 長いパス名へのソフトリンクを作成します。次に例を示します。

```
# ln -s /opt/apps/oracle/home/directory/is/longer/than\  
/eighty/characters/oracle /opt/link_to_longpath
```
- 2 ファイル /etc/VRTSvcs/conf/config/main.cf で、Oracle リソースタイプと Netlsnr リソースタイプの両方に対する Home 属性と Pfile 属性を修正します。
91 ページの「[設定例](#)」を参照してください。
次に例を示します。

```
Home = "/opt/link_to_longpath"  
Pfile = "/opt/link_to_longpath/dbs/initVRT.ora"
```

NLS 情報の定義

NLS 情報は、Oracle パラメータファイルで適切なパラメータを定義するか、または EnvFile で適切な環境変数を定義することにより定義できます。

81 ページの「[リソースタイプの定義](#)」を参照してください。

Oracle パラメータファイルでパラメータを定義すると、Oracle サーバー用の NLS 設定に影響します。環境変数を定義すると、クライアントユーティリティの NLS 入力および出力に影響します。

Oracle データベースと VCS のホットバックアップ

Oracle データベースで、ホットバックアップ実行中のクラスタ内のノードに障害が発生すると、Oracle リソースの AutoEndBkup 属性に 0 以外の値が定義されている場合にのみ、他のノードへフェールオーバーを行います。

メモ: VCS 環境でホットバックアップ機能を使うためには、Oracle エージェントの起動オプションを STARTUP または STARTUP_FORCE に設定する必要があります。12 ページの「[起動および停止オプション](#)」を参照してください。

81 ページの「リソースタイプの定義」を参照してください。

それ以外の場合は、フェールオーバーノードにあるバックアップモードのデータベースがオープンできないため、VCS は Oracle リソースグループをオンラインにできません。

この状態の場合には次のエラーが表示されます。

```
$ ORA-1110 "data file %s: '%s'"
```

または

```
$ ORA-1113 "file %s needs media recovery"
```

VCS がフェールオーバーノードの Oracle リソースグループをオンラインにできるようにするには、データベースのデータファイルをバックアップモードから解除し、データベースを停止して再起動する必要があります。データベースファイルの状態を変更する方法については、Oracle のマニュアルを参照してください。

VCS 環境での Oracle のインストール

Oracle を VCS クラスタにインストールする場合は、クラスタ内のすべてのノードでのインストールを確実に統一することが必要になります。Oracle のインストールについて詳しくは、Solaris に関する Oracle のマニュアルを参照してください。

Oracle を VCS 環境にインストールする方法には、次の 2 種類があります。

共有ディスク上に \$ORACLE_HOME を配置した場合 Oracle バイナリと Oracle データは共有ディスクにインストールされます。

ローカルディスク上に \$ORACLE_HOME を配置した場合 Oracle バイナリは各ノードでローカルにインストールされ、Oracle データは共有ディスクにインストールされます。

Oracle データには、データファイル、制御ファイル、REDO ログファイル、アーカイブログファイルが含まれます。

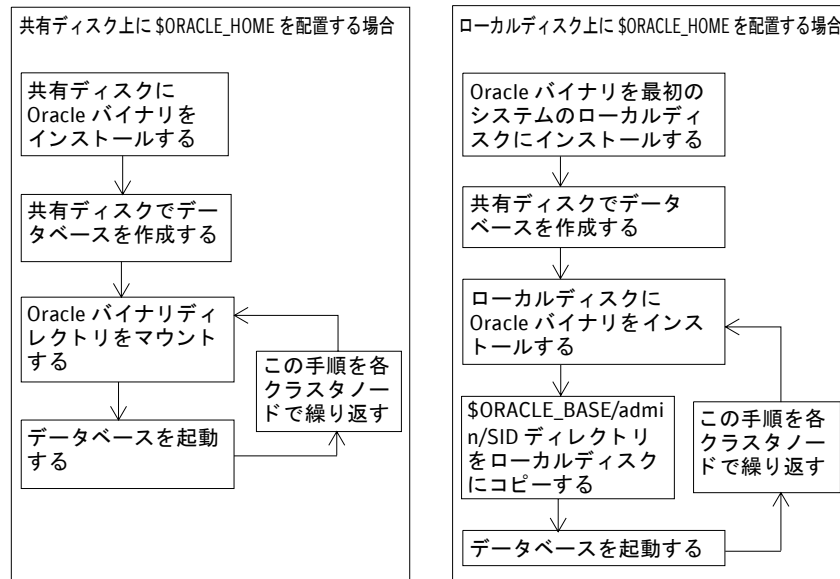
Oracle をインストールする場合は、必ず Oracle の所有者の login_id、id_name、group_id および group_name がすべてのノードで同じであることを確認してください。ユーザー oracle とグループ dba は、Network Information Service (NIS および NIS+) ユーザーではなくローカルユーザーである必要があります。

Oracle インストールタスクの概要

VCS クラスタで Oracle のインストールを完了するためのタスクは、\$ORACLE_HOME を共有ディスクとローカルディスクのどちらに配置するかによって異なります。

図 2-3 は、\$ORACLE_HOME が共有ディスク上にある場合とローカルディスク上にある場合のインストールの流れを示しています。

図 2-3 インストールの流れの比較



共有ディスクで行う \$ORACLE_HOME のインストールタスク

表 2-9 に、\$ORACLE_HOME が共有ディスク上に配置されるように Oracle をインストールするためのタスクの一覧を示します。手順としては、初回インストール時に、共有ディスク上に Oracle バイナリをインストールし、Oracle データベースを作成します。Oracle バイナリディレクトリをマウントし、すべてのノードからデータベースを起動できることを確認してください。

表 2-9 共有ディスクで行う \$ORACLE_HOME のインストールタスク

タスク	参照
クラスタ内の任意のノードから、共有ディスクに Oracle バイナリをインストールします。	29 ページの「 Oracle バイナリのインストール 」を参照してください。
Oracle をインストールしたノードから、共有ディスクにデータベースを作成します。	32 ページの「 Oracle データベースの設定 」を参照してください。
<ul style="list-style-type: none"> Oracle クラスタの一部になる各ノードから、共有ディスク上のデータベースにアクセスできることを確認します。 任意のノードで、Oracle データベースを起動します。 	35 ページの「 データベースの起動 」を参照してください。

ローカルディスクで行う \$ORACLE_HOME のインストールタスク

表 2-10 に、\$ORACLE_HOME がローカルディスク上に配置されるように Oracle をインストールするためのタスクの一覧を示します。手順としては、初回インストール時に、ローカルディスク上に Oracle バイナリをインストールし、共有ディスク上に Oracle データベースを作成します。次に、他のノードのローカルディスクに Oracle バイナリをインストールします。この作業により、すべての Oracle インストールはまったく同じになり、共有ディスクの同じ場所からデータベースにアクセスできるようになります。

表 2-10 ローカルディスクで行う \$ORACLE_HOME のインストールタスク

タスク	参照
クラスタ内の最初のノードで、ローカルディスクに Oracle バイナリをインストールします。	29 ページの「 Oracle バイナリのインストール 」を参照してください。
Oracle をインストールした最初のノードから、共有ディスクにデータベースを作成します。	32 ページの「 Oracle データベースの設定 」を参照してください。
クラスタ内の各ノードで、次のタスクを実行します。 <ul style="list-style-type: none"> ■ ローカルディスクに Oracle バイナリをインストールする。 ■ \$ORACLE_BASE/admin/SID ディレクトリをローカルディスクにコピーする。 	<ul style="list-style-type: none"> ■ 29 ページの「Oracle バイナリのインストール」を参照してください。 ■ 35 ページの「\$ORACLE_BASE/admin/SID ディレクトリのコピー」を参照してください。
任意のノードで、Oracle データベースを起動します。	35 ページの「 データベースの起動 」を参照してください。

Oracle バイナリのインストール

Veritas High Availability Agent for Oracle がサポートするバージョンの Oracle をインストールします。VCS 環境で Oracle9i または Oracle 10g をインストールする手順を確認してください。画面およびオプションは、Oracle のほかのバージョンの表示と異なる場合があります。

VCS 環境への Oracle9i のインストール

- 1 Oracle CD を挿入します。

- 2 *DISPLAY* 変数を設定し、CD ドライブをマウントし、Oracle ユーザーで Oracle インストーラを実行します。
`# /mnt/cdrom/runInstaller`
- 3 [ようこそ (Welcome)] 画面を読み、[次へ (Next)] をクリックします。
- 4 初めて Oracle をインストールする場合は、[インベントリの場所 (Inventory Location)] ダイアログボックスが表示されます。ベースディレクトリの場所を指定し、[OK] をクリックします。
 - インストール時に、`$ORACLE_HOME` をローカルディスク上に置く予定の場合は、そのローカルディスク上の場所を指定してください。
 - `$ORACLE_HOME` を共有ディスク上に置く予定の場合は、その共有ディスク上の場所を指定してください。
- 5 [UNIX グループ名 (UNIX Group Name)] ダイアログボックスに Oracle ソフトウェアの更新権限を持つ UNIX グループの名前を入力し、[次へ (Next)] をクリックします。
- 6 スクリプト `/tmp/orainstRoot.sh` の実行を指示するメッセージが表示されます。このスクリプトを実行し、[続行 (Continue)] をクリックして Oracle インストーラに戻ります。
- 7 [ファイルの場所 (File Locations)] ダイアログボックスで Oracle ホームディレクトリの名前とパスを入力または選択し、[次へ (Next)] をクリックします。
- 8 [使用可能な製品 (Available Products)] ダイアログボックスで [Oracle9i データベース (Oracle9i Database)] オプションを選択し、[次へ (Next)] をクリックします。
- 9 [インストールタイプ (Installation Types)] ダイアログボックスでインストールの種類を選択し、[次へ (Next)] をクリックします。
- 10 [データベースの設定 (Database Configuration)] ダイアログボックスで [ソフトウェアのみ (Software Only)] オプションを選択し、[次へ (Next)] をクリックします。
- 11 [概略 (Summary)] ダイアログボックスで、選択した内容を確認します。変更する場合は、[戻る (Back)] をクリックします。変更の必要がなければ、[インストール (Install)] をクリックします。
- 12 インストールが完了したら、[次へ (Next)] をクリックします。
- 13 [インストールの終了 (End of Installation)] ダイアログボックスで、[終了 (Exit)] をクリックします。
- 14 共有ディスクでデータベースを設定するタスクに進みます。
32 ページの「[Oracle データベースの設定](#)」を参照してください。

VCS 環境への Oracle 10g のインストール

- 1 Oracle CD を挿入します。
- 2 *DISPLAY* 変数を設定し、CD ドライブをマウントし、Oracle ユーザーで Oracle インストーラを実行します。

```
# /mnt/cdrom/runInstaller -ignoreSysPrereqs
```
- 3 [ようこそ (Welcome)] 画面を読み、[次へ (Next)] をクリックします。
- 4 初めて Oracle をインストールする場合は、[インベントリの場所 (Inventory Location)] ダイアログボックスが表示されます。ベースディレクトリの場所を指定し、[OK] をクリックします。次のいずれかを実行します。
 - インストール時に、`$ORACLE_HOME` をローカルディスク上に置く予定の場合は、そのローカルディスク上の場所を指定してください。
 - `$ORACLE_HOME` を共有ディスク上に置く予定の場合は、その共有ディスク上の場所を指定してください。
- 5 Oracle インストーラにより、スクリプト `/tmp/orainstRoot.sh` の実行を指示するメッセージが表示されます。このスクリプトを実行し、[続行 (Continue)] をクリックして Oracle インストーラに戻ります。
- 6 [ファイルの場所 (File Locations)] ダイアログボックスで Oracle ホームディレクトリの名前とパスを入力または選択し、[次へ (Next)] をクリックします。
- 7 [インストールタイプ (Installation Types)] ダイアログボックスでインストールの種類を選択し、[次へ (Next)] をクリックします。
- 8 [Select Database Configuration] ダイアログボックスで [Do not create a starter database] オプションを選択し、[次へ (Next)] をクリックします。
- 9 [概略 (Summary)] ダイアログボックスで、選択した内容を確認します。変更する場合は、[戻る (Back)] をクリックします。変更の必要がなければ、[インストール (Install)] をクリックします。
- 10 インストールが完了したら、[次へ (Next)] をクリックします。
- 11 [インストールの終了 (End of Installation)] ダイアログボックスで、[終了 (Exit)] をクリックします。
- 12 Oracle 10g クラスタデーモンを無効にします。
32 ページの「[Oracle 10g クラスタデーモンの無効化](#)」を参照してください。
- 13 共有ディスクでデータベースを設定するタスクに進みます。
32 ページの「[Oracle データベースの設定](#)」を参照してください。

Oracle 10g クラスターデーモンの無効化

Oracle 10g を共有ディスクにインストールした場合は、Oracle クラスターデーモンを無効にする必要があります。

Oracle 10g には、Oracle Cluster Synchronization Service Daemon (CSSD) と呼ばれるクラスターデーモンが用意されています。Oracle バイナリファイルが共有ストレージ上に置かれている場合、init コマンドでこのデーモンを起動すると、エラーが起きることがあります。Oracle 用の VCS クラスターではこのデーモンを必要としないので、デーモンを無効にすることを推奨します。

Solaris 9 以前でデーモンを無効にするには

- ◆ Oracle インストールウィザードを実行したノードの `/etc/inittab` ファイルから、次の行を削除します。

```
h1:23:respawn:/etc/init.d/init.cssd run >/dev/null 2>&1 >  
</dev/null
```

Solaris 10 でデーモンを無効にするには

- ◆ Oracle インストールウィザードを実行したノードで次を行います。
 - CSSD デーモンの Fault Management Resource Identifier (FMRI) を探します。コマンドプロンプトから次のように入力します。

```
# svcs | grep cssd
```

CSSD デーモンの FMRI が表示されます。
 - SVC がデーモンをコントロールしないように、サービス設定リポジトリを修正します。コマンドプロンプトから次のように入力します。

```
# svccfg delete -f <FMRI>
```

FMRI は、前のステップで取得した CSSD の FMRI です。

Oracle データベースの設定

Database Configuration Assistant を使って共有ディスク上に Oracle データベースを設定します。データベースの設定は、最初に Oracle をインストールしたノード上で、一度だけ行います。Oracle9i データベースまたは Oracle 10g データベースを設定する手順を確認してください。

Oracle9i データベースを設定するには

- 1 `DISPLAY` 変数を設定し、Oracle ユーザーで Oracle Database Configuration Assistant を起動します。

```
# dbca
```
- 2 [ようこそ (Welcome)] 画面を読み、[次へ (Next)] をクリックします。
- 3 [操作 (Operations)] ダイアログボックスで [データベースの作成 (Create database)] オプションを選択し、[次へ (Next)] をクリックします。

- 4 [データベーステンプレート (Database Templates)] ダイアログボックスで、データベースを作成するテンプレートを選択し、[次へ (Next)] をクリックします。
- 5 [データベース識別情報 (Database Identification)] ダイアログボックスで一意的グローバルデータベース名と SID を入力または選択し、[次へ (Next)] をクリックします。
- 6 [データベース接続オプション (Database Connection Options)] ダイアログボックスで [専用サーバーモード (Dedicated Server Mode)] または [共有サーバーモード (Shared Server Mode)] を選択し、[次へ (Next)] をクリックします。
- 7 [初期化パラメータ (Initialization Parameters)] ダイアログボックスで、アーカイブログファイルと初期化パラメータファイルの場所を指定します。
 - データベースのアーカイブログを有効にするには、[アーカイブ (Archive)] タブをクリックし、[アーカイブログモード (Archive Log Mode)] チェックボックスを選択します。[アーカイブログの宛先 (Archive Log Destination)] リストに共有ディスクへのパスを入力します。これにより、アーカイブログが共有ディスクに作成されます。
 - [ファイルの場所 (File Locations)] タブをクリックします。
 - すべてのシステムでローカルに Oracle をインストールする場合は、初期化パラメータファイルとトレースファイルのディレクトリをローカルディスク上に配置してください。
 - 必要に応じてその他のパラメータを変更します。
 - [次へ (Next)] をクリックします。
- 8 [データベース記憶域 (Database Storage)] ダイアログボックスで、制御ファイル、データファイルおよび REDO ログファイルの場所を指定します。
 - 左のペインで、[記憶域 (Storage)] フォルダの横にある [+] をクリックし、フォルダを展開します。
 - [制御ファイル (Controlfile)] をクリックし、右のペインで [一般 (General)] タブをクリックします。
 - [ファイルディレクトリ (File Directory)] フィールドで、各制御ファイルの共有ディスクのパスを入力します。
 - [データファイル (Datafiles)] フォルダの横にある [+] をクリックし、フォルダを展開します。
 - 左のペインでデータファイルをクリックし、右のペインで [一般 (General)] タブをクリックします。[名前 (Name)] フィールドで、各データファイルの共有ディスクのパスを入力します。
 - [REDO ロググループ (Redo Log Groups)] フォルダの横にある [+] をクリックし、フォルダを展開します。

- 左のペインで REDO ログファイルを選択します。右のペインの [ファイルディレクトリ (File Directory)] フィールドで、各 REDO ログファイルの共有ディスクのディレクトリのパスを入力します。必要に応じて REDO ログファイルのサイズを指定します。
 - [次へ (Next)] をクリックします。
- 9 [作成オプション (Creation Operations)] ダイアログボックスで [データベースの作成 (Create database)] チェックボックスを選択し、[完了 (Finish)] をクリックします。
 - 10 [概略 (Summary)] ダイアログボックスで選択した内容を確認し、[OK] をクリックします。

Oracle 10g データベースを設定するには

- 1 *DISPLAY* 変数を設定し、Oracle ユーザーで Oracle Database Configuration Assistant を起動します。
dbca
- 2 [ようこそ (Welcome)] 画面を読み、[次へ (Next)] をクリックします。
- 3 [操作 (Operations)] ダイアログボックスで [データベースの作成 (Create database)] オプションを選択し、[次へ (Next)] をクリックします。
- 4 [データベーステンプレート (Database Templates)] ダイアログボックスで、データベースを作成するテンプレートを選択し、[次へ (Next)] をクリックします。
- 5 [データベース識別情報 (Database Identification)] ダイアログボックスで一意的グローバルデータベース名と SID を入力または選択し、[次へ (Next)] をクリックします。
- 6 ウィザードの指示に従い、その他のオプションを選択します。
- 7 [Database File Locations] ダイアログボックスで、作成するデータベースファイルのために共有ディスク上の場所を指定し、[次へ (Next)] をクリックします。
- 8 [Recovery Configuration] ダイアログボックスで、共有ディスク上の [Flash Recovery Area] を指定し、[次へ (Next)] をクリックします。
- 9 ウィザードの指示に従ってデータベースの作成を完了します。

\$ORACLE_BASE/admin/SID ディレクトリのコピー

\$ORACLE_HOME がローカルディスクにある場合にのみ、クラスタ内のノードごとに、次の手順に従って作業を進めてください。

\$ORACLE_BASE/admin/SID ディレクトリをコピーするには

- ◆ 共有ディスクから、\$ORACLE_BASE/admin/SID ディレクトリをローカルディスクにコピーします。

変数 *SID* は、データベースインスタンスを表します。

詳しくは Oracle の OFA の構成を参照してください。

データベースの起動

\$ORACLE_HOME の場所に応じてどちらかの手順を実行し、データベースを起動します。

SQL クエリーを発行すると、データベースへのアクセス状態を確認できます。

\$ORACLE_HOME が共有ディスク上にある場合のデータベースの起動

各ノードから共有データベースにアクセスできることを確認します。Oracle クラスタの一部であるそれぞれのノードから次の手順を実行します。

\$ORACLE_HOME が共有ディスク上にある場合にデータベースを起動するには

- 1 Oracle バイナリが他のノードでマウント済みの場合は、マウント解除します。
- 2 共有ディスクのデータファイルが任意のノードでマウント済みの場合は、マウント解除します。
- 3 Oracle バイナリとデータファイルをマウントします。
- 4 データベースを起動します。

\$ORACLE_HOME がローカルディスク上にある場合のデータベースの起動

Oracle クラスタの一部になる各ノードで、個別にデータベースを起動します。

\$ORACLE_HOME がローカルディスク上にある場合にデータベースを起動するには

- 1 Oracle バイナリをクラスタ内の各ノードにインストールしたことを確認します。

- 2 共有ディスクのデータファイルが任意のノードでマウント済みの場合は、マウント解除します。
- 3 データファイルをマウントします。
- 4 データベースを起動します。

Oracle 用エージェントのインストール

この章では、次の内容について説明します。

- [Oracle 用エージェントをインストールまたはアップグレードする前に](#)
- [エージェントソフトウェアのインストール](#)
- [エージェントのアップグレード](#)

Oracle 用エージェントをインストールまたはアップグレードする前に

Veritas High Availability Agent for Oracle をインストールまたはアップグレードするための前提条件を満たします。

- VCS がクラスタにインストールされていることを確認します。
Symantec では、VCS GUI をインストールすることを推奨します。『[Veritas Cluster Server インストールガイド](#)』を参照してください。
- Oracle Server for Solaris とアドオンのリスナーがインストールされ、設定されていることを確認します。詳しくは [Oracle のマニュアル](#)を参照してください。
23 ページの「[Oracle をインストールするための VCS の必要条件](#)」を参照してください。
27 ページの「[VCS 環境での Oracle のインストール](#)」を参照してください。

エージェントソフトウェアのインストール

Veritas High Availability Agent for Oracle は製品ディスクからインストールできます。Oracle サービスグループをホストする予定のすべてのノードに Oracle エンタープライズエージェント をインストールする必要があります。

これにより、次の RPM がインストールされます。

- VRTSvcsor (エージェントのバイナリ)
- VRTScsocw (エージェント設定ウィザード)

Solaris ノードにエージェントをインストールするには

- 1 スーパーユーザーとしてログインします。
- 2 インストール用の一時ディレクトリを作成します。

```
# mkdir /tmp/install
```
- 3 ノードに接続されているドライブに、適切なディスクを挿入します。
 - Solaris ボリューム管理ソフトウェアを実行している場合は、このソフトウェアがディスクを /cdrom/cdrom0 として自動的にマウントします。次のコマンドを入力して、場所を移動します。

```
# cd /cdrom/cdrom0
```
 - Solaris ボリューム管理ソフトウェアを実行していない場合は、ディスクを手動でマウントする必要があります。次に例を示します。

```
# mount -F hsfs -o ro /dev/dsk/c0t6d0s2 /cdrom
```

この例では、/dev/dsk/c0t6d0s2 がデフォルトの CD ドライブです。
ディスクをマウントしたら、パッケージファイルがある場所に移動します。

```
# cd /cdrom
```
- 4 圧縮されたパッケージファイルを、ソフトウェアディスクから一時ディレクトリへコピーします。

```
# cp -r oracle_agent/pkgs/* /tmp/install
```
- 5 システムに gunzip ユーティリティがインストールされていない場合は、次のコマンドでディスクからコピーできます。

```
# cp /cdrom_path/gnu/gunzip /tmp/install
```
- 6 一時ディレクトリに移動し、圧縮されたパッケージファイルを解凍します。

```
# cd /tmp/install  
# gunzip VRTS*.gz
```
- 7 tar ファイルで圧縮されたファイルを解凍します。

```
# tar -xvf VRTSvcsor.tar  
# tar -xvf VRTScsocw.tar
```
- 8 パッケージをインストールします。

```
# pkgadd -d . VRTSvcsor  
# pkgadd -d . VRTScsocw
```

言語パックをインストールするには

- 1 エージェントをインストールしたら、システムに接続しているドライブに言語ディスクを挿入します。
- 2 システムに接続されているドライブに、適切なディスクを挿入します。
 - Solaris ボリューム管理ソフトウェアを実行している場合は、このソフトウェアがディスクを `/cdrom/cdrom0` として自動的にマウントします。次のコマンドを入力して、場所を移動します。

```
# cd /cdrom/cdrom0
```
 - Solaris ボリューム管理ソフトウェアを実行していない場合は、ディスクを手動でマウントする必要があります。次に例を示します。

```
# mount -F hsfs -o ro /dev/dsk/c0t6d0s2 /cdrom
```

この例では、`/dev/dsk/c0t6d0s2` がデフォルトの CD ドライブです。ディスクをマウントしたら、パッケージファイルがある場所に移動します。

```
# cd /cdrom
```
- 3 圧縮されたパッケージファイルを、ソフトウェアディスクから一時ディレクトリへコピーします。

```
# cp -r ja/oracle_agent/pkgs/* /tmp/install
```
- 4 一時ディレクトリに移動し、圧縮されたパッケージファイルを解凍します。

```
# cd /tmp/install  
# gunzip VRTSjacso.tar.gz
```
- 5 `tar` ファイルで圧縮されたファイルを解凍します。

```
# tar -xvf VRTSjacso.tar
```
- 6 言語パックをインストールします。

```
# pkgadd -d . VRTSjacso
```

エージェントのアップグレード

VCS クラスタで、Veritas High Availability Agent for Oracle をバージョン 3.5、4.0、4.1 からアップグレードできます。

VCS Agent 3.5 for Oracle のアップグレード

VCS Agent 3.5 for Oracle をアップグレードするには

- 1 VCS の設定を保存し、VCS エンジンを停止します。

```
# haconf -dump -makero  
# hastop -all -force
```
- 2 Oracle エージェントがインストールされているすべてのノードで次を実行します。

- Oracle エージェントを削除します。
pkgrm VRTSvcsor
 - ファイル `/etc/VRTSvcs/conf/config/OracleTypes.cf` を削除します。
 - VCS エンタープライズエージェント 5.0 for Oracle をインストールします。
38 ページの「[エージェントソフトウェアのインストール](#)」を参照してください。
- 3 OracleTypes.cf ファイルを `/etc/VRTSagents/ha/conf/Oracle` ディレクトリから `/etc/VRTSvcs/conf/config` ディレクトリにコピーします。
 - 4 クラスタ内のノードから、`/etc/VRTSvcs/conf/config/main.cf` 設定ファイルを編集します。
Sqlnet タイプのリソースをすべて Netlsnr タイプのリソースに置換します。
81 ページの「[リソースタイプの定義](#)」を参照してください。
 - 5 設定を確認します。
hacf -verify config
 - 6 最初に、ローカルノードの VCS を起動します。
 - 7 次に、他のノードの VCS を起動します。

VCS Agent 4.0 for Oracle と VCS Agent 4.1 for Oracle のアップグレード

VCS クラスタ内の各ノードで、次の手順を実行します。

VCS Agent for Oracle を 5.0 にアップグレードするには

- 1 VCS をローカルで停止します。
hastop -local -force
- 2 VCS エンタープライズエージェント 4.0 for Oracle を削除します。
pkgrm VRTSvcsor
pkgrm VRTSscsw
- 3 VCS エンタープライズエージェント 5.0 for Oracle をインストールします。
38 ページの「[エージェントソフトウェアのインストール](#)」を参照してください。
- 4 OracleTypes.cf ファイルを `/etc/VRTSagents/ha/conf/Oracle` ディレクトリから `/etc/VRTSvcs/conf/config` ディレクトリにコピーします。
- 5 VCS エンジン を再起動します。
hastart

Oracle 用 VCS サービスグループの設定

この章では、次の内容について説明します。

- [Oracle 用サービスグループの設定について](#)
- [VCS における Oracle の設定について](#)
- [サービスグループを設定する前に](#)
- [サービスグループの設定](#)
- [基本監視についてのその他の注意事項](#)
- [詳細監視の設定](#)

Oracle 用サービスグループの設定について

Oracle サービスグループを設定するときには、Oracle サービスグループとそのリソースを作成し、設定されたリソースに対して属性値を定義する必要があります。サービスグループの作成および設定には、Administrator 権限が必要です。VCS では、次のようないくつかの方法でエンタープライズエージェントを設定できます。

- エージェント設定ウィザードを使う
 - Cluster Manager (Java コンソール) を使う
 - コマンドラインを使う
- 49 ページの「[サービスグループの設定](#)」を参照してください。

VCS における Oracle の設定について

VCS 環境下では、Oracle を次のいくつかの方法で設定できます。表 4-11 では、VCS 環境で設定して高可用性を実現できる、Oracle の様々な設定を示します。VCS 用に Oracle を設定するときは、Oracle ファイルの listener.ora と tnsnames.ora を VCS の必要条件に従って設定する必要があります。

表 4-11 VCS における Oracle の設定

設定の種類	参照
単一の Oracle インスタンスの設定	42 ページの「 VCS における単一の Oracle インスタンスの設定 」を参照してください。
複数の Oracle インスタンス（単一リスナー）の設定	43 ページの「 VCS における複数の Oracle インスタンス（単一リスナー）の設定 」を参照してください。
複数の Oracle インスタンス（複数のリスナー）の設定	45 ページの「 VCS における複数の Oracle インスタンス（複数リスナー）の設定 」を参照してください。
共有サーバーサポートの設定	46 ページの「 VCS における共有サーバーサポートの Oracle インスタンスの設定 」を参照してください。

117 ページの「[複数の Oracle インスタンスを設定する場合のベストプラクティス](#)」を参照してください。

VCS における単一の Oracle インスタンスの設定

単一の Oracle インスタンスが関係するサービスグループのリソースの依存関係グラフと設定例を見直します。

91 ページの「[単一の Oracle インスタンスの設定例](#)」を参照してください。

VCS で Oracle インスタンスを設定するには

- 1 Oracle と Netlsnr のリソースタイプと属性の定義を見直します。
81 ページの「[リソースタイプの定義](#)」を参照してください。
- 2 VCS の必要条件に従って Oracle ファイル tnsnames.ora を設定します。ファイルで必要な変更は、Oracle の設定によって異なります。クライアントがフェールオーバーインスタンスに接続できるようにするには、\$TNS_ADMIN に定義されている tnsnames.ora ファイルで、TCP プロトコルのアドレスデータベースのホスト名を、サービスグループで設定する仮想 IP アドレスに変更します。

次の例では、データベースのホスト名を、サービスグループの仮想 IP アドレスを示す **oraprod** に変更すると想定しています。

```
prod =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP) (HOST = oraprod) (PORT = 1521))
    )
    (CONNECT_DATA =
      (SERVICE_NAME = prod)
    )
  )
```

- 3 VCS の必要条件に従って Oracle ファイル **listener.ora** を設定します。ファイルで必要な変更は、Oracle の設定によって異なります。

\$TNS_ADMIN 内のファイル **listener.ora** の **ADDRESS_LIST** セクションにある「**HOST =**」の行を編集し、サービスグループの高可用性アドレスの名前（この場合は、**oraprod**）を追加します。

```
LISTENER_PROD =
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (ADDRESS_LIST =
        (ADDRESS = (PROTOCOL = TCP) (HOST = oraprod) (PORT = 1521))
      )
    )
  )
```

- 4 エージェント設定ウィザード、**Cluster Manager (Java コンソール)**、コマンドラインを使って、Oracle サービスグループを作成します。
49 ページの「**サービスグループの設定**」を参照してください。

- 5 Oracle サービスグループをオンラインにします。
69 ページの「**サービスグループのオンライン化**」を参照してください。

VCS における複数の Oracle インスタンス（単一リスナー）の設定

複数の Oracle インスタンスが関係するサービスグループのリソースの依存関係グラフと設定例を見直します。

- 97 ページの「**複数の Oracle インスタンス（単一リスナー）の設定例**」を参照してください。

複数の Oracle インスタンス（単一リスナー）を設定するには

- 1 Oracle と Netlsnr のリソースタイプと属性の定義を見直します。
81 ページの「**リソースタイプの定義**」を参照してください。
- 2 VCS の必要条件に従って Oracle ファイル **tnsnames.ora** を設定します。ファイルで必要な変更は、Oracle の設定によって異なります。
クライアントがフェールオーバーインスタンスに接続できるようにするには、**\$TNS_ADMIN** に定義されている **tnsnames.ora** ファイルで、TCP プロ

トコルのアドレスデータベースのホスト名を、サービスグループで設定する仮想 IP アドレスに変更します。

次の例では、データベースのホスト名を、サービスグループの仮想 IP アドレスを示す **oraprod** に変更すると想定しています。

```
prod =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP) (HOST = oraprod) (PORT = 1521))
    )
    (CONNECT_DATA =
      (SERVICE_NAME = prod)
    )
  )
prod =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP) (HOST = oramktg) (PORT = 1522))
    )
    (CONNECT_DATA =
      (SERVICE_NAME = mktg)
    )
  )
```

- 3 VCS の必要条件に従って Oracle ファイル **listener.ora** を設定します。ファイルで必要な変更は、Oracle の設定によって異なります。

\$TNS_ADMIN 内のファイル **listener.ora** の **ADDRESS_LIST** セクションにある「**HOST =**」の行を編集し、サービスグループの高可用性アドレスの名前（この場合は、**oraprod**）を追加します。

```
LISTENER_ORACLE =
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (ADDRESS_LIST =
        (ADDRESS = (PROTOCOL = TCP) (HOST = oraprod) (PORT = 1521))
        (ADDRESS = (PROTOCOL = TCP) (HOST = oramktg) (PORT = 1522))
      )
    )
  )
```

- 4 Cluster Manager (Java コンソール) を使って、Oracle サービスグループとリスナーサービスグループを作成します。コマンドラインを使ってサービスグループを作成することもできます。

49 ページの「[サービスグループの設定](#)」を参照してください。

- 5 Oracle サービスグループをオンラインにします。

69 ページの「[サービスグループのオンライン化](#)」を参照してください。

VCS における複数の Oracle インスタンス（複数リスナー）の設定

複数の Oracle インスタンスが関係するサービスグループのリソースの依存関係グラフと設定例を見直します。

107 ページの「[複数のインスタンス（複数リスナー）の設定例](#)」を参照してください。

複数の Oracle インスタンス（複数リスナー）を設定するには

- 1 Oracle と Netlsnr のリソースタイプと属性の定義を見直します。
81 ページの「[リソースタイプの定義](#)」を参照してください。
- 2 VCS の必要条件に従って Oracle ファイル `tnsnames.ora` を設定します。ファイルで必要な変更は、Oracle の設定によって異なります。
クライアントがフェールオーバーインスタンスに接続できるようにするには、`$TNS_ADMIN` に定義されている `tnsnames.ora` ファイルで、TCP プロトコルのアドレスデータベースのホスト名を、サービスグループで設定する仮想 IP アドレスに変更します。

```
prod =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP) (HOST = oraprod) (PORT = 1521))
    )
    (CONNECT_DATA =
      (SERVICE_NAME = prod)
    )
  )
mktg =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP) (HOST = oramktg) (PORT = 1522))
    )
    (CONNECT_DATA =
      (SERVICE_NAME = mktg)
    )
  )
```

- 3 VCS の必要条件に従って Oracle ファイル `listener.ora` を設定します。ファイルで必要な変更は、Oracle の設定によって異なります。

各リスナーに対して個別の設定を行うよう、`listener.ora` ファイルを編集します。

```
LISTENER_PROD =
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (ADDRESS_LIST =
        (ADDRESS = (PROTOCOL = TCP) (HOST = oraprod) (PORT = 1521))
      )
    )
  )
```

```
LISTENER_MKTG =  
  (DESCRIPTION_LIST =  
    (DESCRIPTION =  
      (ADDRESS_LIST =  
        (ADDRESS = (PROTOCOL = TCP) (HOST = oramktg) (PORT = 1522))  
      )  
    )  
  )
```

- 4 エージェント設定ウィザード、**Cluster Manager (Java コンソール)**、コマンドラインを使って、**Oracle** サービスグループを作成します。
49 ページの「[サービスグループの設定](#)」を参照してください。
- 5 **Oracle** サービスグループをオンラインにします。
69 ページの「[サービスグループのオンライン化](#)」を参照してください。

VCS における共有サーバーサポートの Oracle インスタンスの設定

リソースの依存関係グラフと設定例を見直します。

- 111 ページの「[共有サーバーサポートの Oracle の設定例](#)」を参照してください。

共有サーバーサポートの Oracle を設定するには

- 1 **Oracle** と **Netlsnr** のリソースタイプと属性の定義を見直します。
81 ページの「[リソースタイプの定義](#)」を参照してください。
- 2 VCS の必要条件に従って **Oracle** ファイル **tnsnames.ora** を設定します。ファイルで必要な変更は、**Oracle** の設定によって異なります。
クライアントがフェールオーバーインスタンスに接続できるようにするには、**\$TNS_ADMIN** に定義されている **tnsnames.ora** ファイルで、**TCP** プロトコルのアドレスデータベースのホスト名を、サービスグループで設定する仮想 IP アドレスに変更します。
次の例では、データベースのホスト名を、サービスグループの仮想 IP アドレスを示す **oraprod** に変更すると想定しています。

```
prod =  
  (DESCRIPTION =  
    (ADDRESS_LIST =  
      (ADDRESS = (PROTOCOL = TCP) (HOST = oraprod) (PORT = 1521))  
    )  
    (CONNECT_DATA =  
      (SERVICE_NAME = prod)  
    )  
  )
```

- 3 VCS の必要条件に従って Oracle ファイル `listener.ora` を設定します。ファイルで必要な変更は、Oracle の設定によって異なります。

`$TNS_ADMIN` 内のファイル `listener.ora` の `ADDRESS_LIST` セクションにある「`HOST =`」の行を編集し、サービスグループの高可用性アドレスの名前（この場合は、`oraprod`）を追加します。

```
LISTENER_PROD =
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (ADDRESS_LIST =
        (ADDRESS = (PROTOCOL = TCP)(HOST = oraprod)(PORT = 1522))
      )
    )
  )
```

- 4 初期化パラメータファイルで、`dispatchers` パラメータについて、すべての TCP プロトコルアドレスディスパッチャのホスト名をサービスグループの仮想 IP アドレスに設定します。

次の例では、ディスパッチャホスト名として `oraprod` を設定しています。

```
dispatchers =
  "(ADDRESS = (PROTOCOL = TCP)(HOST = oraprod))
  (MUL=ON)(TICK=15)(POOL=true)"
```

- 5 初期化パラメータファイルで、`LOCAL_LISTENER` パラメータにリスナーのエイリアスを設定し、Oracle のプロセスモニタ (PMON) にリスナーの情報を登録させます。

```
local_listener=listener_alias
```

ここで定義するリスナーのエイリアス名の後には、`sqlnet.ora` ファイルで定義されているデフォルトのドメイン名が付けられます。

- 6 `tnsnames.ora` ファイルで、アドレス情報を解決するためのリスナーエイリアスのエントリを作成します。アドレスパラメータで、ホスト名をサービスグループの仮想 IP アドレスに設定します。ただし、接続記述子の `CONNECT_DATA` の部分は除外します。

次の例で、リスナー `listener_alias` は、TCP/IP ポート 1521 を使用しており、サービスグループの仮想 IP アドレスは `oraprod` です。

```
listener_alias=
  (address = (PROTOCOL = TCP)(HOST = oraprod)
  (PORT= 1521))
```

- 7 エージェント設定ウィザード、Cluster Manager (Java コンソール)、コマンドラインを使って、Oracle サービスグループを作成します。

48 ページの「サービスグループを設定する前に」を参照してください。

- 8 Oracle サービスグループをオンラインにします。

69 ページの「サービスグループのオンライン化」を参照してください。

サービスグループを設定する前に

Oracle サービスグループを設定する前に、次のことを行う必要があります。

- サービスグループを設定する クラスタ内のすべてのノードに VCS がインストールされ、設定されていることを確認します。
詳しくは『Veritas Cluster Server インストールガイド』を参照してください。
- クラスタ内のすべてのノードに Oracle がまったく同様にインストールされ、設定されていることを確認します。
23 ページの「[Oracle のインストールと設定](#)」を参照してください。
- クラスタ内のすべてのノードに Veritas High Availability Agent for Oracle がインストールされていることを確認します。
37 ページの「[Oracle 用エージェントのインストール](#)」を参照してください。
- Veritas High Availability Agent for Oracle のタイプ定義が VCS エンジンにインポートされていることを確認します。
48 ページの「[OracleTypes.cf ファイルのインポート](#)」を参照してください。
- Oracle 10g をインストールした場合は、init スクリプトによる再ブート時に Oracle サービスが起動していないことを確認します。

OracleTypes.cf ファイルのインポート

Veritas High Availability Agent for Oracle を設定する前に、OracleTypes.cf ファイルを VCS エンジンにインポートする必要があります。

Cluster Manager を使ってインポートするには

- 1 Cluster Manager を開始して、クラスタにログオンします。
- 2 Cluster Explorer のファイル (File) メニューで、[タイプのインポート (Import Types)] を選択します。
- 3 [タイプのインポート (Import Types)] ダイアログボックスで、
/etc/VRTSagents/ha/conf/Oracle/OracleTypes.cf ファイルを選択します。
- 4 [インポート (Import)] をクリックします。
- 5 設定を保存します。

コマンドラインを使ってインポートするには

- 1 スーパーユーザーとして `sysa` にログインします。
- 2 既存の設定に対するすべての変更が保存されたことを確認し、`main.cf` の修正中にこれ以上の変更が行われないようにします。

```
# haconf -dump -makero
```
- 3 `main.cf` の編集中に VCS が実行されないようにするには、すべてのノードで VCS エンジンを停止し、リソースは利用可能なままにします。

```
# hastop -all -force
```
- 4 `main.cf` ファイルのバックアップコピーを作成します。

```
# cd /etc/VRTSvcs/conf/config  
# cp main.cf main.cf.orig
```
- 5 Oracle タイプ設定ファイルを次の場所にコピーします。

```
# cp /etc/VRTSagents/ha/conf/Oracle/OracleTypes.cf \  
/etc/VRTSvcs/conf/config/OracleTypes.cf
```
- 6 `OracleTypes.cf` ファイルを含むように `main.cf` ファイルを編集します。

```
include "OracleTypes.cf"
```

サービスグループの設定

VCS 環境の Oracle は、VCS がサポートするいずれかの方法で設定できます。

42 ページの「[VCS における Oracle の設定について](#)」を参照してください。

次のことを見直してサービスグループを設定してください。

- Oracle サービスグループの設定ファイルの例とリソース依存関係グラフ
91 ページの「[設定例](#)」を参照してください。
- Oracle リソースタイプと、Oracle エージェントや Netlsnr エージェントの属性定義
81 ページの「[リソースタイプの定義](#)」を参照してください。

VCS では、次のようないくつかの方法でエンタープライズエージェントを設定できます。

- [エージェント設定ウィザードを使ったサービスグループの設定](#)
- [Cluster Manager \(Java コンソール\) を使ったサービスグループの設定](#)
- [コマンドラインを使ったサービスグループの設定](#)

エージェント設定ウィザードを使ったサービスグループの設定

VCS にはエージェント設定の手順を説明するエージェント設定ウィザードがあります。このウィザードでは、Oracle サービスグループの作成と修正を行います。既存のサービスグループを修正することもできます。

71 ページの「[サービスグループ設定の修正](#)」を参照してください。

次の必要条件を満たしていることを確認してください。

- 設定対象の Oracle インスタンスとリスナーが、グローバルまたは非グローバルゾーンで実行中であること。設定対象のすべてのリスナーが、同じ仮想 IP アドレスをリスニングする必要があります。

メモ: デフォルトのリスナーの場合は、設定ウィザードを使うには、リスナーパラメータファイル (`listener.ora`) が `$ORACLE_HOME/network/admin` に存在する必要があります。この条件は、デフォルトのリスナー以外には適用されません。

- 非グローバルゾーンで実行中の Oracle インスタンスに対してサービスグループを設定する場合は、サービスグループとして含めるすべてのシステムにゾーンが作成済みであること。
- Oracle ファイル（コントロール、データ、REDO ログ、アーカイブの各ファイル）が共有ディスクに存在すること。
- `listener.ora` ファイルと `tnsnames.ora` ファイルに指定された IP アドレスとホスト名が同じであること。
- データベースインスタンスに詳細監視を使う場合は、割り当てられたユーザーとパスワードを使って、詳細監視に使うテーブルが設定されていること。

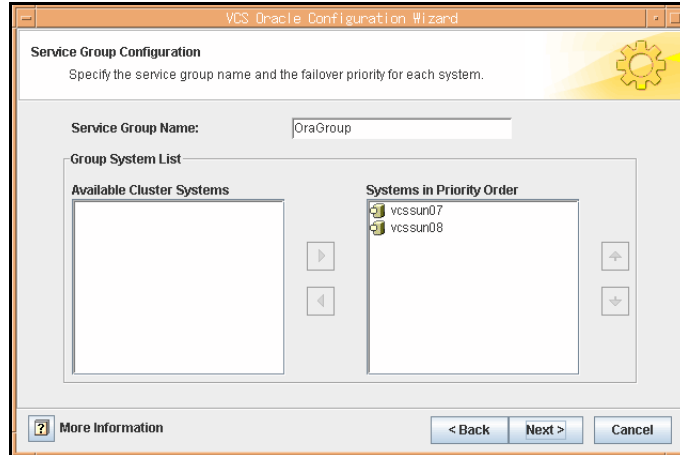
65 ページの「[詳細監視の設定](#)」を参照してください。

エージェント設定ウィザードを使ってサービスグループを設定するには

- 1 `DISPLAY` 変数を設定し、`root` として VCS Oracle 設定ウィザードを起動します。

```
# hawizard oracle
```
- 2 [ようこそ (Welcome)] 画面を読み、[次へ (Next)] をクリックします。
- 3 [ウィザードオプション (Wizard Options)] ダイアログボックスで [Oracle サービスグループを作成します (Create Oracle Service Group)] オプションを選択し、[次へ (Next)] をクリックします。

- 4 [サービスグループ設定 (Service Group Configuration)] ダイアログボックスで、サービスグループに関する情報を指定します。



次の情報を指定します。

サービスグループ名
(Service Group Name)

Oracle サービスグループの名前を入力します。

使用可能なクラスタ
システム (Available
Cluster Systems)

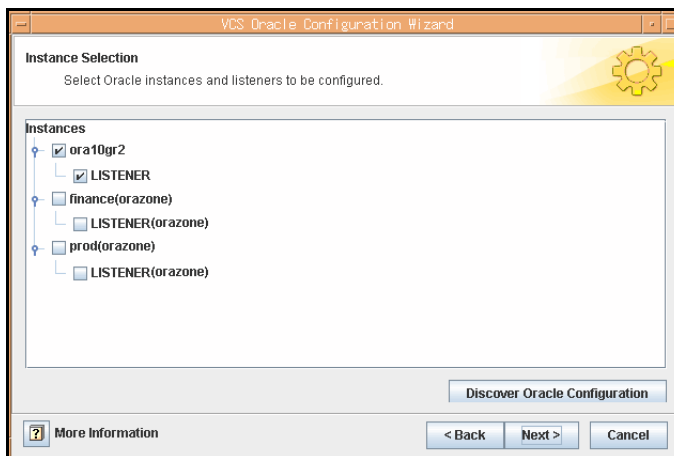
サービスグループを設定するシステムを選択し、右矢印アイコンをクリックしてそのシステムをサービスグループのシステムリストに移動させます。

システム (優先度順)
(Systems in Priority
Order)

- サービスグループのシステムリストからシステムを削除するには、[システム (優先度順) (Systems in Priority Order)] ボックスでシステムを選択し、左矢印アイコンをクリックします。
- サービスグループのシステムリストでシステムの優先度を変更するには、[システム (優先度順) (Systems in Priority Order)] ボックスでシステムを選択し、上下矢印アイコンをクリックします。リストの一番上のシステムは最も優先度の高いシステムであり、リストの一番下のシステムは最も優先度の低いシステムです。

- 5 [次へ (Next)] をクリックします。

- 6 [インスタンス選択 (Instance Selection)] ダイアログボックスで、設定対象の Oracle インスタンスとリスナーを選択します。必要に応じて、[Oracle の設定を検出 (Discover Oracle Configuration)] をクリックします。



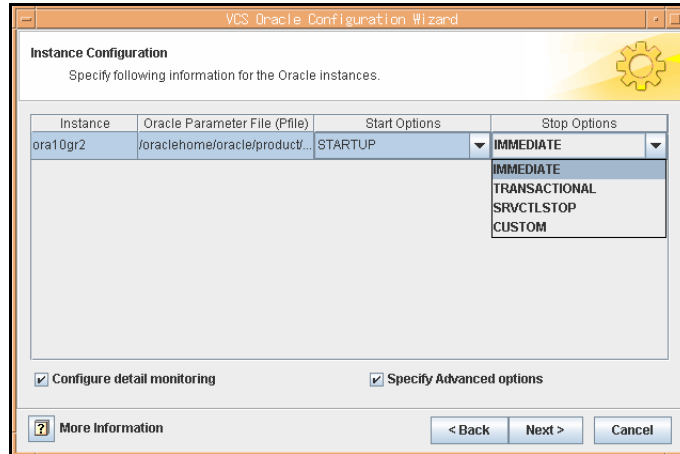
ウィザードに、インスタンスとインスタンスを実行中の非グローバルゾーンの名前が表示されます。ただし、グローバルゾーンで実行中のインスタンスの場合は、ゾーン名は表示されません。

インスタンスを選択して、[次へ (Next)] をクリックします。

複数のインスタンスを選択する場合は、それらのインスタンスが同じゾーンで実行されていることを確認してください。

[次へ (Next)] をクリックします。

- 7 [インスタンス設定 (Instance Configuration)] ダイアログボックスで、選択したすべてのインスタンスに関する情報を指定します。



選択した Oracle インスタンスについて次の情報を指定します。

Oracle パラメータファイル (Pfile) Oracle パラメータファイルの場所を確認します。
(Oracle Parameter File (Pfile))

手順 6 で、非グローバルゾーンで実行中のインスタンスを選択した場合は、指定した場所が、インスタンスを実行中のゾーン環境にあることを確認します。

起動オプション
(Start Options)

必要に応じて、起動オプションを選択します。デフォルトは `STARTUP_FORCE` です。

12 ページの「[起動オプション](#)」を参照してください。

停止オプション
(Stop Options)

必要に応じて、停止オプションを選択します。デフォルトは `IMMEDIATE` です。

13 ページの「[停止オプション](#)」を参照してください。

詳細監視の設定
(Configure detail monitoring)

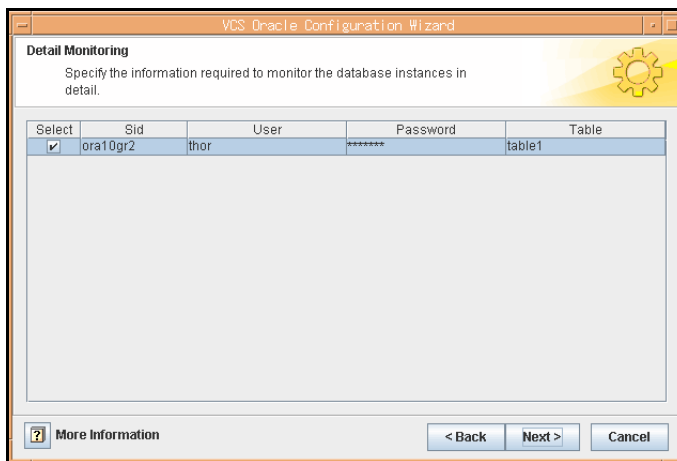
データベースの詳細を監視する場合は、チェックボックスにチェックマークを付けます。

14 ページの「[詳細監視](#)」を参照してください。

拡張オプションの指定
(Specify Advanced Options)

データベースインスタンスの拡張設定情報を入力する場合は、チェックボックスにチェックマークを付けます。

- 8 [次へ (Next)] をクリックします。
- 9 データベースの詳細を監視するように選択した場合は、[詳細監視 (Detail Monitoring)] ダイアログボックスが表示されます。



エージェントで詳細を監視するデータベースインスタンスについて次の情報を指定し、[次へ (Next)] をクリックします。

選択 (Select) 詳細を監視するデータベースに対応するチェックボックスにチェックマークを付けます。

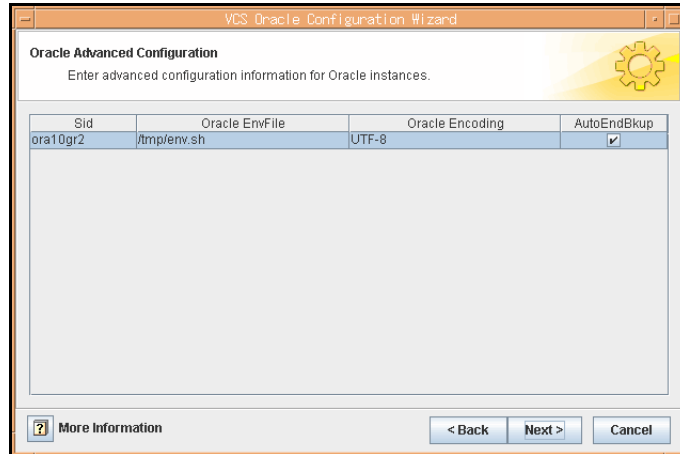
ユーザー (User) データベースの有効なユーザー名を入力します。

パスワード (Password) データベースユーザーの有効なパスワードを入力します。

メモ: エージェント設定ウィザードを使ってパスワードを入力するときは、パスワードを暗号化しないでください。パスワードの暗号化はウィザードが行います。

テーブル (Table) データベースの状態を検証するためにクエリーするテーブルの名前を入力します。

- 10 拡張オプションを指定するように選択した場合は、[Oracle 拡張設定 (Oracle Advanced Configuration)] ダイアログボックスが表示されます。



Oracle インスタンスについて次の情報を指定し、[次へ (Next)] をクリックします。

Oracle EnvFile Oracle Envfile の場所を入力します。

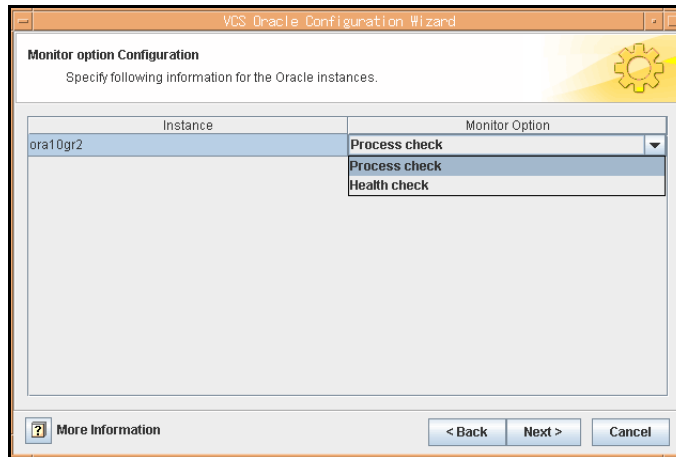
手順 6 で、非グローバルゾーンで実行中のインスタンスを選択した場合は、指定した場所が、インスタンスを実行中のゾーン環境にあることを確認します。

Oracle エンコード エンコードを入力します。
(Oracle Encoding)

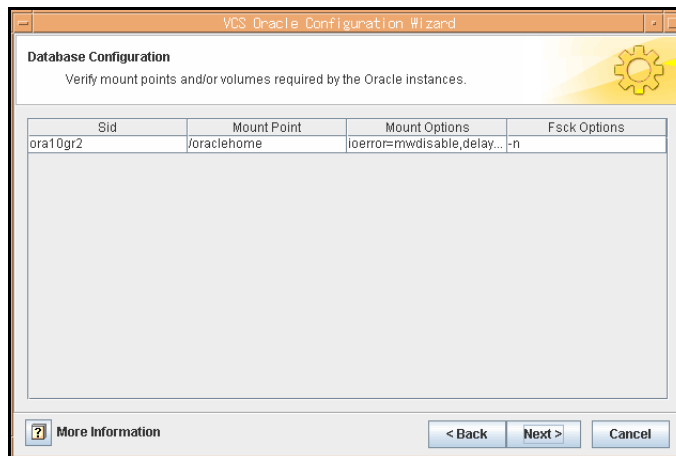
AutoEndBkup 必要に応じて、チェックボックスにチェックマークを付けます。
26 ページの「Oracle データベースと VCS のホットバックアップ」を参照してください。

- 11 [監視オプション設定 (Monitor option Configuration)] ダイアログボックスで、Oracle インスタンスの監視オプションを指定し、[次へ (Next)] をクリックします。
デフォルトの監視オプションは、[プロセスチェック (Process check)] です。

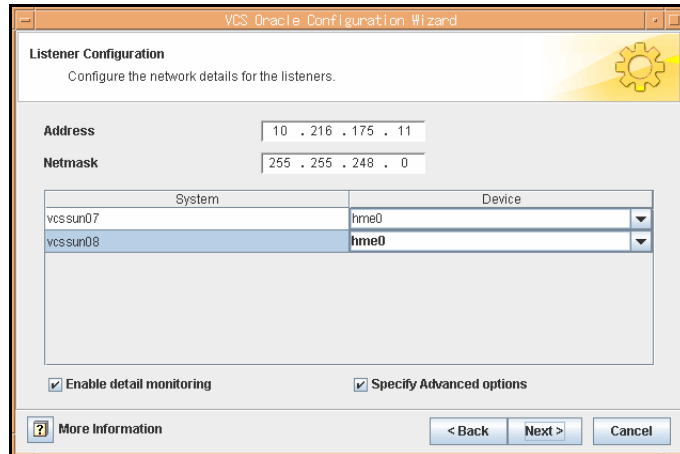
14 ページの「基本監視オプション」を参照してください。



- 12 [データベース設定 (Database Configuration)] ダイアログボックスで、各データベースで必要なマウントポイントを確認し、Mount オプションと Fskc オプションを指定して、[次へ (Next)] をクリックします。



- 13 [リスナー設定 (Listener Configuration)] ダイアログボックスで、データベースのリスナーを設定します。

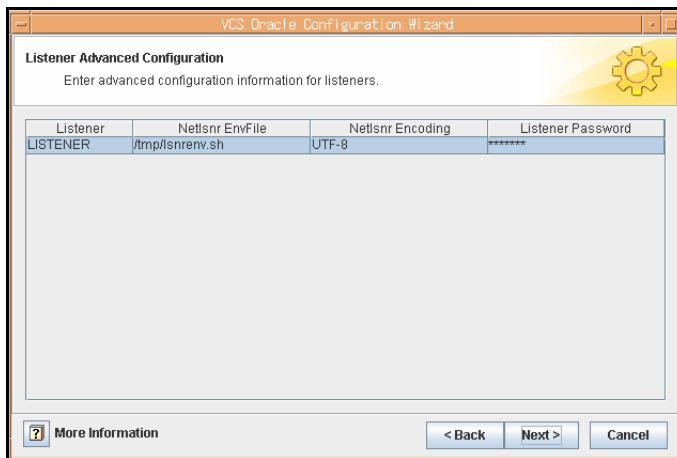


リスナーについて次の情報を指定します。

- | | |
|---------------------------------------|--------------------------------------------|
| アドレス (Address) | 仮想 IP アドレスを確認します。 |
| ネットマスク (Netmask) | ネットマスクを確認します。 |
| デバイス (Device) | システムごとにデバイスを選択します。 |
| 詳細監視の有効化 (Enable detail monitoring) | 詳細監視を有効にするには、チェックボックスにチェックマークを付けます。 |
| 拡張オプションの指定 (Specify Advanced Options) | リスナーの拡張設定情報を入力する場合は、チェックボックスにチェックマークを付けます。 |

- 14 [次へ (Next)] をクリックします。

- 15 リスナーの拡張オプションを指定するように選択した場合は、[リスナー拡張設定 (Listener Advanced Configuration)] ダイアログボックスが表示されます。



各リスナーについて次の情報を指定し、[次へ (Next)] をクリックします。

Netlsnr EnvFile リスナーの **Envfile** のパスを入力します。

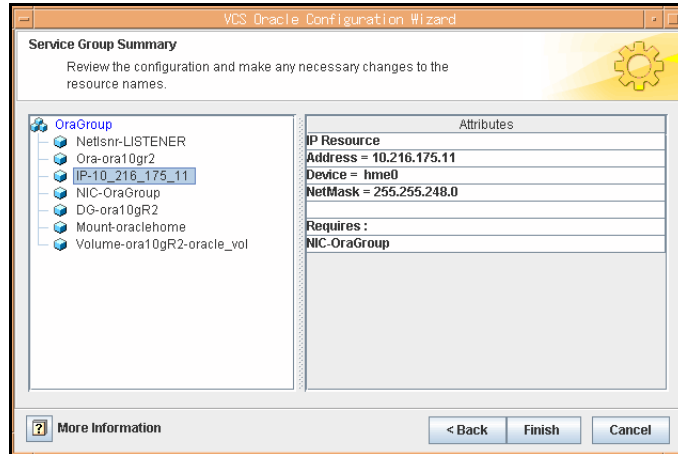
手順 6 で、非グローバルゾーンで実行中のリスナーを選択した場合は、指定した場所が、リスナーを実行中のゾーン環境にあることを確認します。

Netlsnr エンコード (Netlsnr Encoding) エンコードを入力します。

リスナーパスワード (Listener Password) リスナーの有効なパスワードを入力します。

メモ : エージェント設定ウィザードを使ってパスワードを入力するときは、パスワードを暗号化しないでください。パスワードの暗号化は、ウィザードが行います。

- 16 サービスグループの概略（Service Group Summary）で、設定を見直します。



リソースをクリックすると、[属性 (Attributes)] ボックスに属性と属性の設定値が表示されます。

- 必要に応じてリソース名を変更します。ウィザードは、それぞれの名前の規則に基づいて、リソースに一意的な名前を割り当てます。
リソース名を編集するには、リソース名を選択してクリックし、各属性を編集してから **Enter** キーを押します。
- 設定を見直して、[完了 (Finish)] をクリックします。
ウィザードは、**Oracle** サービスグループを作成するコマンドの実行を開始します。これらのコマンドの状態を示す様々なメッセージが表示されます。

- 17 [Oracle 設定ウィザードを終了します。(Completing the Oracle Configuration wizard.)] ダイアログボックスで、[サービスグループをオンラインにする (Bring the service group online)] チェックボックスにチェックマークを付けて、ローカルシステムでサービスグループをオンライン状態にします。

- 18 [閉じる (Close)] をクリックします。
Oracle サービスグループがクラスタに作成されました。

Cluster Manager (Java コンソール) を使ったサービスグループの設定

Oracle リソースグループのテンプレートは、Oracle エンタープライズエージェントをインストールすると自動的にインストールされます。VCS Cluster Manager を使ってテンプレートを表示できます。テンプレートには、Oracle サービスグループ、サービスグループのリソース、リソースの属性が表示されます。設定に応じて、属性値を動的に修正できます。

次のような設定タスクがあります。

- 新しいサービスグループの作成
- リソース属性の編集

メモ: Java コンソールを使って、Solaris 非グローバルゾーンで実行中の Oracle インスタンスのサービスグループを設定することはできません。このようなサービスグループの設定には、設定ウィザードを使います。手順について詳しくは 50 ページの「[エージェント設定ウィザードを使ったサービスグループの設定](#)」を参照してください。

新しいサービスグループを作成するには

- 1 Oracle のタイプ定義ファイル (OracleTypes.cf) が設定にインポートされていることを確認します。
- 2 クラスタ設定ウィザードを起動します。次のいずれかの操作をします。
 - Cluster Explorer メニューで、[ツール (Tools)]、[設定ウィザード (Configuration Wizard)] の順に選択します。
 - システムにサービスグループが存在しない場合は、[Cluster Explorer] に、クラスタ設定ウィザードの起動を求めるプロンプトが表示されます。このプロンプトで [はい (Yes)] をクリックします。
[テンプレート情報を読み込んでいます (Loading Templates Information...)] ウィンドウが表示され、ウィザードが起動します。
- 3 [ようこそ (Welcome)] 画面を読み、[次へ (Next)] をクリックします。
- 4 サービスグループ名とそのサービスグループを設定する対象システムを指定します。
 - サービスグループ名を入力します。
 - [使用可能なシステム (Available Systems)] ボックスで、サービスグループを設定するシステムを選択します。
 - 選択したシステムを [サービスグループのシステム (Systems for Service Group)] ボックスに移動するには、右矢印をクリックします。

ボックスからシステムを削除するには、システムを選択して左矢印をクリックします。

- システムをフェールオーバーする順序（優先度）を指定します。システムの優先度には、順番に番号が割り当てられます。1 は最初にフェールオーバーするシステムを表します。
 - [次へ (Next)] をクリックします。
- 5 [テンプレートをを使ってサービスグループを設定しますか？ (Would you like to use a template to configure the service group?)] ダイアログボックスで、[次へ (Next)] をクリックします。
 - 6 [サービスグループを設定するために、テンプレートを選択します。 (Select a template to configure the service group.)] ダイアログボックスで、サービスグループの基礎になる OracleGroup テンプレートを選択します。
場合によっては、新しいサービスグループ内のリソースの名前の中にすでに使われている名前があることを通知するウィンドウが表示されることがあります。このウィンドウには、代わりの名前の一覧も表示されます。代わりの名前を選択するか、またはその名前を修正することができます。名前の重複を解決したら、[次へ (Next)] をクリックします。
 - 7 選択したテンプレートに基づいてサービスグループを作成するには、[次へ (Next)] をクリックします。
グループ、そのグループのリソース、テンプレートに指定された属性と依存関係を追加するコマンドを送信中であることを示すウィンドウが開きます。進行状況インジケータに、発行中のコマンドの進行状況がパーセンテージで示されます。
現在のコマンドがインジケータの上部に表示されます。
 - 8 サービスグループが正常に作成されたことを示すプロンプトが表示されたら、[次へ (Next)] をクリックします。
サービスグループのリソースとそれらのリソースに関連する属性の一覧のウィンドウが開きます。
 - 9 デフォルト値を受け入れて設定を完了する場合は、[完了 (Finish)] をクリックします。
特定の設定の必要条件に従って、リソースのデフォルト値を修正できます。これについては、次で説明します。

リソース属性を編集するには

- 1 左ペインの一覧からリソースを選択します。
リソースの属性が右ペインに表示されます。
- 2 修正する属性を選択します。
81 ページの「[リソースタイプの定義](#)」を参照してください。
- 3 [編集 (Edit)] 列の編集アイコンをクリックします。

- 4 [属性の編集 (Edit Attribute)] ダイアログボックスで、属性値を入力します。属性の範囲を修正するには、ラジオボタンの [グローバル (Global)] または [ローカル (Local)] をクリックします。
- 5 [OK] をクリックします。
- 6 各リソースと属性について同じ手順を繰り返し、[完了 (Finish)] をクリックします。設定に応じてすべてのリソースの属性を編集します。
- 7 ウィザードの指示に従って設定を完了します。
- 8 [完了 (Finish)] をクリックしてウィザードを終了します。

コマンドラインを使ったサービスグループの設定

Veritas High Availability Agent for Oracle には、現在の設定ファイルを直接修正するための参考として使える設定ファイルの例が付いています。この方法を使うときは、VCS を停止し、再起動してから設定を実装する必要があります。

設定ファイルの例を使ってサービスグループを設定するには

- 1 スーパーユーザーとして `sysa` にログインします。
- 2 既存の設定に対するすべての変更が保存されたことを確認し、`main.cf` の修正中にこれ以上の変更が行われないようにします。

```
# haconf -dump -makero
```
- 3 `main.cf` の編集時に VCS が実行されないようにするには、次のコマンドを実行することによって、すべてのシステムで VCS エンジン を停止し、リソースは利用可能なままにします。

```
# hastop -all -force
```
- 4 `main.cf` ファイルのバックアップコピーを作成します。

```
# cd /etc/VRTSvcs/conf/config
# cp main.cf main.cf.orig
```
- 5 Oracle タイプ設定ファイルを次の場所にコピーします。

```
# cp /etc/VRTSagents/ha/conf/Oracle/OracleTypes.cf \
/etc/VRTSvcs/conf/config/OracleTypes.cf
```
- 6 `main.cf` ファイルを編集します。
`/etc/VRTSagents/ha/conf/Oracle/sample_main.cf` ファイルは次のような処理の参考として使えます。
 - `OracleTypes.cf` ファイルの `include` 行を追加。

```
include "OracleTypes.cf"
```
 - Oracle リソースと `Netlsnr` リソースを作成。
`/etc/VRTSagents/ha/conf/Oracle/sample_main.cf` ファイルに示されている例を参照できます。

- 設定内のパラメータに一致するようにデフォルトの属性を編集。
81 ページの「リソースタイプの定義」を参照してください。
 - 新しく作成されたリソースへの依存関係の割り当て。例として、
/etc/VRTSagents/ha/conf/Oracle/sample_main.cf ファイルと依存関係
グラフを参照できます。依存関係の割り当てについては、
『Veritas Cluster Server ユーザーズガイド』を参照してください。
91 ページの「設定例」を参照してください。
- 7 ファイルを保存して閉じます。
 - 8 /etc/VRTSvcs/conf/config/main.cf ファイルの構文を検証します。
cd /etc/VRTSvcs/conf/config/
hacf -verify .
 - 9 VCS エンジンを実行します。
hastart
 - 10 すべての Oracle サービスグループリソースがオンライン状態になっている
ことを確認します。
hagr -state
 - 11 サービスグループをオフライン状態にして、すべてのリソースが停止したこ
とを確認します。
hagr -offline service_group -sys system_name
hagr -state
 - 12 サービスグループをオンライン状態にして、すべてのリソースが利用可能で
あることを確認します。
hagr -online service_group -sys system_name
hagr -state
 - 13 sysb で VCS エンジンを実行します。
hastart
 - 14 Oracle サービスグループを sysb に切り替えます。
hagr -switch service_group -to sysb
 - 15 すべての Oracle サービスグループリソースが sysb でオンライン状態になっ
ていることを確認します。
hagr -state
 - 16 すべてのノードで、次のログファイルを確認してエラーや状態を調べます。
/var/VRTSvcs/log/engine_A.log
/var/VRTSvcs/log/Oracle_A.log
/var/VRTSvcs/log/Netlsnr_A.log

パスワードの暗号化

VCS には、データベースユーザーのパスワードやリスナーパスワードを暗号化するためのユーティリティが用意されています。Oracle エージェントの **Pword** 属性、Netlsnr エージェントの **LsnrPwd** 属性は、暗号化してから設定する必要があります。

Oracle には、リスナーパスワードをクリアテキストまたは暗号化形式で **listener.ora** ファイルに格納するオプションが用意されています。どちらの形式で Oracle に格納されているかに関係なく、パスワードは、**LsnrPwd** 属性を設定する前に、**vcscrypt** ユーティリティを使って暗号化する必要があります。このパスワードは、解読を行うと、**listener.ora** ファイルに格納された時のフォーマットになります。

メモ : 設定ウィザードまたは **VCS Cluster Manager (Java コンソール)** を使って属性を設定しているときは、パスワードを暗号化する必要はありません。Oracle データベースの詳細監視に使うユーザーパスワードは暗号化されます。リスナーの状態の問い合わせとリスナーの停止に使うリスナーパスワードも暗号化されません。

パスワードを暗号化するには

- 1 パス `$VCS_HOME/bin/` から、**vcscrypt** ユーティリティを実行します。
 - 次のコマンドを入力します。

```
# vcscrypt -agent
```
 - パスワードを入力したら、もう一度入力してパスワードを確認します。**Enter** キーを押します。

```
# パスワードを入力してください。(Enter New Password) :  
# もう一度パスワードを入力してください。(Enter Again) :
```
- 2 ユーティリティによりパスワードが暗号化され、暗号化パスワードが表示されたら見直します。
- 3 この暗号化パスワードを属性値として入力します。
- 4 今後の参照用に、暗号化パスワードのコピーを保存します。

基本監視についてのその他の注意事項

Oracle プロセスの基本監視の対象になるのはユーザーに固有の操作です。したがって、別のユーザー環境で起動された Oracle インスタンスをオンラインとして検出することはできません。たとえば、ある Oracle インスタンスがユーザー oraVRT の環境で起動されており、エージェントがユーザー oracle 用に設定されていると、そのエージェントは oraVRT の環境で起動されたインスタンスをオンラインとして検出しません。

このため、次のような状況が生じる場合があります。

- ノード上のリソースをオンラインにするために発行したコマンドによって、そのノード（または別のノード）上ですでに実行中のインスタンスがオンライン化されます。
- フェールオーバーサービスグループ内で、管理者がノード上の Oracle リソースをオンラインにしたときに、クラスタ内の別のノードで Oracle インスタンスがオンラインになっている場合、そのインスタンスが起動されます。ただし、データベースはマウントされません。このようなケースでは、この障害は基本監視のヘルスチェック監視オプション、または詳細監視でしか検出されません。詳細監視では障害の検出後にデータベース表が更新されますが、診断監視ではデータベース表の更新は行われません。基本監視のヘルスチェック監視オプションまたは詳細監視が設定されていない場合、このような競合は検出されません。
14 ページの「[Oracle エージェントの監視オプション](#)」を参照してください。

そのため、VCS の制御外で起動されるインスタンスには、そのインスタンスの OS ユーザーに対応した正しい Owner 属性を設定することを推奨します。

詳細監視の設定

Veritas High Availability Agent for Oracle でのアプリケーション監視には、プライマリ（基本監視）とセカンダリ（詳細監視）の 2 つのレベルがあります。基本監視モードでは、Oracle のプロセスを監視し、プロセスが継続的にアクティブであるかどうかを確認します。

詳細監視モードでは、エージェントは Oracle または Netlsnr リソースの Monscript 属性で定義されたスクリプトを実行します。スクリプトが正常に実行された場合は、リソースが使用可能であると判断されます。デフォルトスクリプトは、設定に応じてカスタマイズできます。

エージェントの詳細監視機能を使うと、データベースやリスナーの状態を監視し、それらの可用性への信頼度を高めることができます。詳細監視を設定するには、エージェントが基本レベルの監視モードで正しく動作することが前提となります。

メモ: データベースのメンテナンスで外部ユーザーのデータベースアクセスを無効にする必要がある場合は、事前に詳細監視を無効にしてください。

Oracle に対する詳細監視の設定

Oracle リソースの詳細監視では、データベース内の表に対して更新トランザクションを実行することによって、トランザクションに対するデータベースの準備が整っているかどうか確認します。この更新アクションは、**Veritas High Availability Agent for Oracle** に付属する 2 つのスクリプト、`SqlTest.pl` と `SimpleTest.pl` によって実行されます。これらのスクリプトは、`/opt/VRTSagents/ha/bin/Oracle/` ディレクトリの下にあります。どちらのスクリプトも、データベースを監視するためにタイムスタンプを更新します。

`SqlTest.pl` スクリプトは、タイムスタンプを更新する前に、データベースがオープンしているかどうかチェックします。データベースが制限モード、非活動モードまたはサスペンドモードであることが判明した場合も、監視成功と返されます。この場合は、基本監視のみが行われます。`SimpleTest.pl` スクリプトは、データベースのチェックは行わず、表に対する更新ステートメントを実行するだけです。

Oracle に対して詳細監視を有効にするには、Oracle データベースにテスト用の表（タイムスタンプ付きの）を作成する必要があります。詳細監視用スクリプト `MonScript` が存在し、`root` にこのスクリプトの実行許可がある必要があります。ユーザーが監視スクリプトを作成するか、エージェント付属のスクリプトを使えます。監視スクリプトのリターンコード **100** は、異常終了を意味します。リターンコードの **101 - 110** は、正常終了を意味します。

詳細監視の設定例では、付属のスクリプトを基に、詳細監視で使う表の作成とテストの方法および詳細監視を有効にする方法を示します。

Oracle に対する詳細監視を設定するには

- 1 VCS 設定を書き込み可能にします。
`haconf -makerw`
- 2 不完全な再設定によって VCS が自動的に処理を行うのを避けるために、サービスグループをフリーズします。
`hagrps -freeze service_group`
- 3 Oracle ユーザーでログインします。
`su - <Owner>`
- 4 ORACLE_HOME および ORACLE_SID の環境変数を設定します。
`export ORACLE_HOME=<Home>`
`export ORACLE_SID=<Sid>`

- 5 データベースの表を設定するために `svrmgr1` または `sqlplus` ユーティリティを起動します。

```
$ORACLE_HOME/bin/svrmgr1
```

または

```
$ORACLE_HOME/bin/sqlplus /nolog
```
- 6 データベース管理者で、`svrmgr1` または `sqlplus` プロンプトで次のステートメントを発行してテスト用の表を作成します。

```
connect / as sysdba
```



```
create user <User>
identified by <Pword>
default tablespace USERS
temporary tablespace TEMP
quota 100K on USERS;

grant create session to <User>;

create table <User>.<Table> ( tstamp date );
insert into <User>.<Table> (tstamp) values (SYSDATE);
```
- 7 次のように入力して、作成したデータベース表が使えることを確認します。

```
disconnect
connect <User>/<Pword>
update <User>.<Table> set ( tstamp ) = SYSDATE;

select TO_CHAR(tstamp, 'MON DD, YYYY HH:MI:SS AM')
from <User>.<Table>;
exit
```
- 8 次の VCS コマンドを使って Oracle リソースに対する詳細監視を有効にします。

```
hares -modify OracleResource User User
hares -modify OracleResource Pword Pword
hares -modify OracleResource Table Table
hares -modify OracleResource MonScript "./bin/Oracle/SqlTest.pl"
hares -modify OracleResource DetailMonitor 1
haconf -dump -makero
hagrp -unfreeze service_group
```

Cluster Manager (Java コンソール) を使ってこれらの属性を設定することもできます。

Oracle に対する詳細監視の有効化と無効化

詳細監視を有効にするには

- ◆ DetailMonitor 属性を 1 に設定します。
hares -modify OracleResource DetailMonitor 1

詳細監視を無効にするには

- ◆ DetailMonitor 属性を 0 に設定します。
hares -modify OracleResource DetailMonitor 0

Netlsnr に対する詳細監視の設定

MonScript 属性に詳細監視用のスクリプトまたは実行プログラムを指定することで、Netlsnr に対する詳細監視を有効にすることができます。詳細監視の設定例では、Netlsnr 用の付属の監視スクリプト `/opt/VRTSagents/ha/bin/Netlsnr/LsnrTest.pl` を使います。Netlsnr リソース用の詳細監視スクリプトでは、リスナーコマンド `lsnrctl status $Listener` を使ってリスナープロセスをテストします。

Netlsnr に対する詳細監視を設定するには

- 1 VCS 設定を書き込み可能にします。
haconf -makerw
- 2 不完全な再設定によって VCS が自動的に処理を行うのを避けるために、サービスグループをフリーズします。
hagrp -freeze service_group
- 3 次のコマンドを入力して詳細監視を有効にします。
hares -modify LsnrResource MonScript "./bin/Netlsnr/LsnrTest.pl"
haconf -dump -makero
hagrp -unfreeze service_group
- 4 詳細監視を無効にするには、MonScript 属性に空文字列を設定します。
haconf -makerw
hagrp -freeze service_group
hares -modify SqlResource MonScript ""
haconf -dump -makero
hagrp -unfreeze

Oracle 用 VCS サービスグループの管理

この章では、次の内容について説明します。

- [VCS サービスグループの管理について](#)
- [サービスグループのオンライン化](#)
- [サービスグループのオフライン](#)
- [サービスグループの切り替え](#)
- [サービスグループ設定の修正](#)
- [Veritas High Availability Agent for Oracle の無効化](#)
- [Veritas High Availability Agent for Oracle の削除](#)

VCS サービスグループの管理について

Cluster Manager またはコマンドラインを使って、Cluster Server 内のサービスグループを管理できます。Cluster Manager を使ってサービスグループを管理する手順を見直します。

サービスグループのオンライン化

サービスグループを オンラインにするには、次の手順を実行します。

サービスグループを オンラインにするには

- 1 Cluster Explorer の設定ツリーで、新規に作成したサービスグループを選択します。

- 2 サービスグループ名を右クリックし、[リソースの有効化 (Enable Resources)] を選択します。これにより、サービスグループ内のすべてのリソースが有効になります。
- 3 サービスグループ名を右クリックし、サービスグループを有効にするシステムを選択します (右クリック、[有効化 (Enable)]、*system_name* または右クリック、[有効化 (Enable)]、[すべて (All)])。
- 4 設定を保存します ([ファイル (File)]、[設定を閉じる (Close Configuraiton)])。
- 5 サービスグループを右クリックし、システムのサービスグループをオンライン化することを選択します (右クリック、[オンライン (Online)]、*system_name*)。

サービスグループのオフライン

サービスグループをオフラインにするには、次の手順を実行します。

サービスグループをオフラインにするには

- 1 Cluster Explorer の設定ツリーの [サービスグループ (Service Groups)] タブで、サービスグループを右クリックします。
または
Cluster Explorer の設定ツリーでクラスタを選択し、[サービスグループ (Service Groups)] タブをクリックします。その後、表示パネルのサービスグループアイコンを右クリックします。
- 2 [オフライン (Offline)] を選択し、ポップアップメニューから適切なシステムを選択します (右クリック、[オフライン (Offline)]、*system_name*)。

サービスグループの切り替え

サービスグループの切り替えでは、サービスグループを現在のシステムでオフライン化し、別のシステムでオンライン化する処理を行います。

サービスグループを切り替えるには

- 1 Cluster Explorer の設定ツリーの [サービスグループ (Service Groups)] タブで、サービスグループを右クリックします。
または
Cluster Explorer の設定ツリーでクラスタを選択し、[サービスグループ (Service Groups)] タブをクリックします。その後、表示パネルのサービスグループアイコンを右クリックします。
- 2 [切り替え (Switch To)] を選択し、ポップアップメニューから適切なシステムを選択します (右クリック、[切り替え (Switch To)]、*system_name*)。

サービスグループ設定の修正

Veritas High Availability Agent for Oracle は、設定ウィザード、コマンドラインインターフェース、Cluster Manager (Java コンソール)、クラスタ管理コンソールなど、いくつかの方法で動的に設定できます。詳しくは『Veritas Cluster Server ユーザーズガイド』を参照してください。

クラスタ設定ウィザードを使って Oracle サービスグループを修正するには

- 1 [Oracle 設定ウィザード (Oracle Configuration Wizard)] を起動します。
`# hawizard oracle`
- 2 [ようこそ (Welcome)] 画面を読み、[次へ (Next)] をクリックします。
- 3 [ウィザードオプション (Wizard Options)] ダイアログボックスで [サービスグループの変更 (Modify Service Group)] オプションを選択し、変更するサービスグループを選択して、[次へ (Next)] をクリックします。
- 4 ウィザードの指示に従い、設定に応じた変更を行います。
50 ページの「[エージェント設定ウィザードを使ったサービスグループの設定](#)」を参照してください。

Veritas High Availability Agent for Oracle の無効化

Veritas High Availability Agent for Oracle を無効にするには、Oracle サービスグループを OFFLINE 状態に変更する必要があります。アプリケーションを完全に停止したり、エージェントを他のシステムに切り替えることが可能です。

エージェントを無効にするには

- 1 サービスグループの SystemList からシステムを削除するには、そのサービスグループがオンラインになっているかどうかを確認します。
`hagrp -state service_group -sys system_name`
- 2 サービスグループがオンラインの場合は、オフラインにします。次のコマンドのいずれかを使います。
 - あるノードでサービスグループをオフラインにし、別のノードでオンラインにするには、`-switch` オプションを使えます。
`hagrp -switch service_group -to system_name`
 - サービスグループをオフラインにして、クラスタ内の他のノードでオンラインにしない場合は、次のように入力します。
`hagrp -offline service_group -sys system_name`
- 3 ノード上のエージェントを停止します。
`haagent -stop Oracle -sys system_name`

- 4 [ログファイルの中からメッセージを検索してください。(Please look for messages in the log file)] というメッセージが表示された場合は、`/var/VRTSvcs/log/engine_A.log` ファイルを調べ、エージェントが停止したことを示すメッセージがあるかどうかを確認してください。
また、`ps` コマンドを使って、エージェントが停止していることを確認することもできます。

すべてのノード上のエージェントを無効にしたら、サービスグループまたはリソースタイプ、またはその両方を VCS 設定から削除できます。詳しくは、『Veritas Cluster Server ユーザーズガイド』内のコマンドラインからの VCS の再設定に関する章を参照してください。

Veritas High Availability Agent for Oracle の削除

エージェントを削除するには、エージェント設定ウィザードとエージェントファイルを各システムから削除する必要があります。

ノード上のエージェントを削除するには

- 1 エージェント設定ウィザードを削除します。
`# pkgrm VRTSscow`
- 2 エージェントを削除します。プロンプトに対して適宜応答してください。
`# pkgrm VRTSvcsor`

トラブルシューティング

この章では、次の内容について説明します。

- Oracle および Netlsnr エージェント共通のエラーメッセージ
- Oracle エージェント固有のエラーメッセージ
- Netlsnr エージェント固有のエラーメッセージ

Oracle および Netlsnr エージェント共通のエラーメッセージ

表 6-12 に、Oracle エンタープライズエージェントのエラーメッセージとその説明、推奨される解決策（利用できる場合）の一覧を示します。

表 6-12 Oracle エンタープライズエージェントのエラーメッセージ

メッセージ	説明と解決方法
ORACLE_HOME が定義されていません。(No ORACLE_HOME specified)	Oracle または Netlsnr タイプの Home 属性が設定されていません。 解決方法: Home 属性に Oracle ホームディレクトリの正しい絶対パス名を設定します。
Oracle のホームディレクトリ %s が存在しません。(Oracle home directory %s does not exist)	Oracle または Netlsnr タイプの Home 属性に指定された文字列が正しくありません。 解決方法: Home 属性に Oracle ホームディレクトリの正しい絶対パス名を設定します。

表 6-12 Oracle エンタープライズエージェントのエラーメッセージ

メッセージ	説明と解決方法
ファイル %s は適切なテキストファイルではありません。(File %s is not a valid text file)	環境変数のソースとして EnvFile 属性で指定されたファイルが存在しない、読み込めないまたはテキストファイルではない、のいずれかです。 解決方法: EnvFile 属性に正しい絶対パス名を設定します。ファイルの書式が有効であることを確認してください。
詳細テストを実行中に、VCSAgExec はエラーを返しました。(VCSAgExec returned failure when trying to execute in-depth test)	内部エラーです。 解決方法: テクニカルサポートまでご連絡ください。
%s からパイプを開けません。(Unable to open pipe from %s)	内部エラーです。 解決方法: テクニカルサポートまでご連絡ください。
プロセス %s を再起動しました。(Process %s restarted)	Oracle プロセスとして指定された PID が、前回の監視サイクルで登録されたものと異なることを示す警告メッセージ。
監視プロシージャ %s から %s が返されました。(Monitor procedure %s returned %s)	MonScript の実行に失敗しました。 解決方法: MonScript をデバッグし、失敗の原因を特定します。
監視プロシージャ %s は終了しませんでした。戻り値は %s です。(Monitor procedure %s did not exit, return value is %s)	MonScript の実行中に発生した内部エラー。 解決方法: テクニカルサポートまでご連絡ください。
Oracle を実行するための所有者が定義されていません。(No owner for Oracle executables was specified)	Oracle タイプの Owner 属性が設定されていません。 解決方法: Owner 属性にデータベースバイナリの正しい所有者を設定します。
無効な所有者 %s が、Oracle の所有者として定義されています。(Invalid owner %s for Oracle executables was specified)	Owner 属性で指定された OS ユーザーが無効です。 解決方法: Owner 属性にデータベースバイナリの正しい所有者を設定します。

表 6-12 Oracle エンタープライズエージェントのエラーメッセージ

メッセージ	説明と解決方法
Monscript %s へのアクセスは拒否されました。詳細監視は有効にされません。有効なファイルを指定してください。(Access to Monscript %s denied. Detail Monitoring will not be enabled!! Please specify a valid file.)	<p>MonScript によって指定されたファイルがアクセスできないか、または見つかりません。</p> <p>解決方法: ファイル名が有効なアクセス可能なファイルであることを確認します。</p>
パスワードの解読中にエラーが発生しました。(Encountered errors while decrypting password!)	<p>エージェントは、指定したパスワードを解読できません。</p> <p>解決方法: <code>vsencrypt</code> ユーティリティを使って新しい暗号化パスワードを作成し、パスワードを指定します。</p>

Oracle エージェント固有のエラーメッセージ

表 6-13 に、Oracle エージェントのエラーメッセージとその説明、推奨される解決策（利用可能な場合）の一覧を示します。

表 6-13 Oracle エージェントのエラーメッセージ

メッセージ	説明と解決方法
SID が定義されていません。(No SID specified)	<p>Oracle タイプの Sid 属性が設定されていません。</p> <p>解決方法: Sid 属性に正しいデータベースインスタンスを設定します。</p>
%s/bin で sqlplus/svrmgrl が見つかりません (sqlplus/svrmgrl not found in %s/bin)	<p>クライアントユーティリティ <code>svrmgrl</code> または <code>sqlplus</code> が <code>\$ORACLE_HOME/bin</code> ディレクトリの中に見つかりません。</p> <p>解決方法: Oracle ホームディレクトリが正しく指定されており、これらの実行ファイルが存在することを確認します。</p>
%s/bin で srvctl が見つかりません (srvctl not found in %s/bin)	<p>クライアントユーティリティ <code>srvctl</code> が <code>\$ORACLE_HOME/bin</code> ディレクトリの中に見つかりません。</p> <p>解決方法: Oracle ホームディレクトリが正しく指定されており、この実行ファイルが存在することを確認します。</p>

表 6-13 Oracle エージェントのエラーメッセージ

メッセージ	説明と解決方法
Oracle %s の停止に失敗しました。 (Oracle %s failed to stop)	clean または offline エントリポイントで、 Shutdown immediate または Shutdown abort コマンドが Oracle インスタンスを終了できなかったことを示す警告メッセージ。
Oracle データベース %s は起動していません (Oracle database %s not running)	clean または offline エントリポイントの実行前からデータベースインスタンスが動作していなかったことを示す警告メッセージ。 解決方法: 解決する必要はありません。
Oracle (%s) を TERM %s で強制終了します。(Oracle (%s) kill TERM %s)	Oracle プロセスに SIGTERM が発行されたことを示す警告メッセージ。 解決方法: 解決する必要はありません。
Oracle (%s) を KILL %s で強制終了します。(Oracle (%s) kill KILL %s)	Oracle プロセスに SIGKILL が発行されたことを示す警告メッセージ。 解決方法: 解決する必要はありません。
データベースは QUIESCING 状態または QUIESCED 状態です。 (Database in QUIESCING/QUIESCED state)	データベースが QUIESCING または QUIESCED 状態であることを示す警告メッセージ。
データベースは RESTRICTED モードです。(Database in RESTRICTED mode)	データベースが RESTRICTED モードであることを示す警告メッセージ。
データベースは SUSPENDED 状態です。(Database in SUSPENDED state)	データベースが SUSPENDED 状態であることを示す警告メッセージ。

表 6-13 Oracle エージェントのエラーメッセージ

メッセージ	説明と解決方法
リソース %s - 監視プロシージャが、設定した時間内に完了しませんでした。(Resource %s- monitor procedure did not complete within the expected time.)	<p>詳しくは Oracle のアラートログを参照してください。</p> <p>監視が、指定された値と同じ回数タイムアウトすると、clean エントリポイントが呼び出され、対応するリソースが強制的に停止されます。次に、RestartLimit 属性で設定されている値に応じて、リソースは FAULTED とマーク付けされるか、再起動されます。</p> <p>解決方法 : FaultOnMonitorTimeouts 属性を 0 に設定し、監視のエラーがリソースの障害と見なされないようにします。</p> <p>考えられる別の原因としては、データベースの設定中に自動アーカイブが有効でなかったことが挙げられます。</p> <p>解決方法 : データベースを手動でアーカイブします。自動アーカイブが有効な場合は、init.ora ファイルの LOG_ARCHIVE_START パラメータを TRUE に設定します。</p>
カスタムスクリプト /opt/VRTSagents/ha/bin/Oracle/start_custom_%.sql が存在しません。データベースが起動できなくなります。(Custom script /opt/VRTSagents/ha/bin/Oracle/start_custom_%.sql does not exist. Will not be able to start the database.)	<p>データベースを起動するためのカスタムスクリプトが指定された場所にありません。</p> <p>解決方法 : カスタムファイルが指定された場所にあり、有効なアクセス権が付与されていることを確認します。</p>
カスタムスクリプト /opt/VRTSagents/ha/bin/Oracle/shut_custom_%.sql が存在しません。デフォルトの shutdown オプションを使います。(Custom script /opt/VRTSagents/ha/bin/Oracle/shut_custom_%.sql does not exist. Using default shutdown option.)	<p>データベースを起動するためのカスタムスクリプトが指定された場所にありません。</p> <p>解決方法 : カスタムファイルが指定された場所にあり、有効なアクセス権が付与されていることを確認します。</p>
oraerror.dat は、解析可能なレコードではありません。(oraerror.dat did not have records that could be parsed)	<p>oraerror.dat ファイルが存在しないか、サポートされていない書式でレコードを保持しています。</p> <p>解決方法 : oraerror.dat ファイルが存在し、そのデータがサポートされている書式であることを確認します。</p>

表 6-13 Oracle エージェントのエラーメッセージ

メッセージ	説明と解決方法
監視オプションが正しくありません。(Incorrect Monitor Option)	MonitorOption の値が、0 未満または 1 より大きくなっています。 解決方法: MonitorOption の値を 0 または 1 に設定します。
この Oracle バージョンに MonitorOption 値を適用できません (MonitorOption value not applicable for this Oracle Version)	Oracle のバージョンが Oracle 10g 以降ではないのにヘルスチェック監視オプションが選択されました。 解決方法: MonitorOption の値を 0 に設定して、プロセス検査の監視オプションを選択します。
ヘルスチェック監視のテストを実行中に VCSAgExec がエラーを返しました (VCSAgExec returned failure when trying to execute health check monitor test)	内部エラーです。 解決方法: テクニカルサポートまでご連絡ください。
Oracle のバージョンを検索中に VCSAgExec がエラーを返しました (VCSAgExec returned failure while trying to find Oracle version)	内部エラーです。 解決方法: テクニカルサポートまでご連絡ください。
属性 User:Pword:Table:MonScript の 1 つまたは複数が正しく設定されていません。詳細監視が有効になりません。詳細監視を行わない場合は、詳細監視に関する属性を設定しないようにしてください。(One or more of the attributes User:Pword:Table:MonScript are not set correctly. Detail monitoring will not be enabled!! Unset the DetailMonitor attribute if you want to disable DetailMonitoring.)	詳細監視が有効に設定されましたが、詳細監視に必要な属性が正しく設定されていません。 解決方法: 詳細監視に必要な属性を正しく設定するか、または DetailMonitor 属性に 0 を設定して詳細監視を無効にします。

Netlsnr エージェント固有のエラーメッセージ

表 6-14 に、Netlsnr エージェントのエラーメッセージとその説明、推奨される解決策（利用可能な場合）の一覧を示します。

表 6-14 Netlsnr エージェントのエラーメッセージ

メッセージ	説明と解決方法
プロセスディレクトリを開くことができません。(Cannot open process directory.)	特定の監視サイクルで、 <code>/proc</code> エントリを処理できませんでした。 解決方法: 解決する必要はありません。
リスナープロセス %s が、動作していません。(Listener process %s not running)	<code>clean</code> または <code>offline</code> エントリポイントの実行前からリスナープロセスが動作していなかったことを示す警告メッセージ。 解決方法: 解決する必要はありません。
Listener %s を TERM %s で強制終了します。(Listener %s kill TERM %s)	リスナープロセスに <code>SIGTERM</code> が発行されたことを示す警告メッセージ。 解決方法: 解決する必要はありません。
Listener %s を KILL %s で強制終了します。(Listener %s kill KILL %s)	リスナープロセスに <code>SIGKILL</code> が発行されたことを示す警告メッセージ。 解決方法: 解決する必要はありません。
%s/bin で、lsnrctl が見つかりません。(lsnrctl not found in %s/bin)	クライアントユーティリティ <code>lsnrctl</code> が <code>\$ORACLE_HOME/bin</code> ディレクトリの中に見つかりません。 解決方法: Oracle ホームディレクトリが正しく指定されており、この実行ファイルが存在することを確認します。
lsnrctl がタイムアウトしました。(lsnrctl operation timed out)	<code>tnslsnr</code> プロセスが応答していません。 解決方法: 基本となるネットワークプロトコルを確認します。

リソースタイプの定義

この付録では、次の内容について説明します。

- [リソースタイプの定義と属性の定義について](#)
- [Oracle リソースタイプ](#)
- [Netlsnr リソースタイプ](#)

リソースタイプの定義と属性の定義について

リソースタイプとは、エージェントの VCS 設定の定義を表すものであり、設定ファイル `main.cf` でのエージェントの定義方法を指定しています。属性の定義には、エージェントに関連付けられた属性が示されています。必須属性では、エージェントが正しく機能するために設定する必要がある属性について説明されています。

Oracle リソースタイプ

Veritas High Availability Agent for Oracle の Oracle エージェントは、VCS の Oracle リソースタイプによって表されます。

```
type Oracle (
    static str AgentDirectory = "/opt/VRTSagents/ha/bin/Oracle"
    static str ContainerType = Zone
    static keylist SupportedActions = {VRTS_GetInstanceName,
        VRTS_GetRunningServices, DBRestrict, DBUndoRestrict,
        DBResume, DBSuspend, DBTbspBackup }
    static str ArgList[] = { Sid, Owner, Home, Pfile, StartUpOpt,
        ShutDownOpt, EnvFile, AutoEndBkup, MonitorOption,
        DetailMonitor, User, Pword, Table, MonScript,
        AgentDebug, Encoding }
    str Sid
    str Owner
    str Home
    str Pfile
```

```
str StartUpOpt = STARTUP_FORCE
str ShutDownOpt = IMMEDIATE
str EnvFile
boolean AutoEndBkup = 1
int MonitorOption = 0
int DetailMonitor = 0
str MonScript = "./bin/Oracle/SqlTest.pl"
str User
str Pword
str Table
boolean AgentDebug = 0
str Encoding
str ContainerName
)
```

Oracle 属性の定義

Oracle エージェントの属性の説明を見直してください。エージェントの属性は、必須、省略可能、内部に分類されます。

必須属性

表 A-15 は必須属性の一覧です。必須属性には値を割り当てる必要があります。

表 A-15 Oracle エージェントの必須属性

必須属性	データ形式と値の種類	定義
Sid	文字列 - スカラー	変数 \$ORACLE_SID は、Oracle インスタンスを表します。Sid は、Oracle エージェントおよび Oracle データベースサーバーにより大文字または小文字が区別されます。
Owner	文字列 - スカラー	Oracle ユーザー。/etc/passwd ファイルで、実行ファイルとデータベースファイルの所有者として定義されます。
Home	文字列 - スカラー	Oracle バイナリおよび設定ファイルへの \$ORACLE_HOME パス。たとえば、/opt/ora_home のように指定します。 メモ: パスの末尾には、スラッシュ (/) を付けずにください。

省略可能な属性

表 A-16 は Oracle エージェントの省略可能な属性の一覧です。省略可能な属性は必要に応じて設定できます。

表 A-16 Oracle エージェントの省略可能な属性

省略可能な属性	データ形式と値の種類	定義
ContainerType	文字列 - スカラー	この属性を Zone に設定すると、Oracle エージェントは、Solaris 10 ゾーンで動作している Oracle インスタンスを監視します。この属性を空のままにした場合は、監視しません。
StartUpOpt	文字列 - スカラー	Oracle インスタンスの起動オプション。この属性は次の値をとることができます。 <ul style="list-style-type: none">■ STARTUP■ STARTUP_FORCE■ RESTRICTED■ RECOVERDB■ SRVCTLSTART■ CUSTOM デフォルトは STARTUP_FORCE です。 12 ページの「 起動オプション 」を参照してください。
ShutDownOpt	文字列 - スカラー	Oracle インスタンスの停止オプション。この属性は次の値をとることができます。 <ul style="list-style-type: none">■ IMMEDIATE■ TRANSACTIONAL■ SRVCTLSTOP■ CUSTOM デフォルトは IMMEDIATE です。 13 ページの「 停止オプション 」を参照してください。

表 A-16 Oracle エージェントの省略可能な属性

省略可能な属性	データ形式と値の種類	定義
EnvFile	文字列 - スカラー	<p>エントリポイントスクリプトが参照するファイルの絶対パス名。このファイルには、Oracle データベースサーバーの実行環境用にユーザーが設定する LD_LIBRARY_PATH、NLS_DATE_FORMAT などの環境変数が含まれます。</p> <p>ファイル内での環境変数を定義するための構文は、Owner のログインシェルに応じて異なります。このファイルは、Owner が読み取りアクセス権を持っているファイルであることが必要です。このファイルには、ユーザーに入力を促すようなメッセージを含まないようにしてください。</p>
Pfile	文字列 - スカラー	<p>初期化パラメータファイルの絶対パス名。</p> <p>サーバーパラメータファイルを使うこともできます。SPFILE パラメータだけを格納した、1 行のテキストからなる初期化パラメータファイルを作成します。詳しくは Oracle のマニュアルを参照してください。</p> <p>120 ページの「VCS クラスタでの SPFILE の使用」を参照してください。</p>
AutoEndBkup	整数 - スカラー	<p>AutoEndBkup 属性を 0 以外の値に設定すると、オンライン化の実行時に、データベースのデータファイルがバックアップモードから解除されます。</p> <p>デフォルトは 1 です。</p> <p>26 ページの「Oracle データベースと VCS のホットバックアップ」を参照してください。</p>
MonitorOption	整数 - スカラー	<p>Oracle インスタンスの監視オプション。この属性は 0 または 1 の値をとることができます。</p> <ul style="list-style-type: none"> ■ 0 - プロセス検査監視 (推奨) ■ 1 - ヘルスチェック監視 <p>デフォルトは 0 です。</p> <p>14 ページの「Oracle エージェントの監視オプション」を参照してください。</p>

表 A-16 Oracle エージェントの省略可能な属性

省略可能な属性	データ形式と値の種類	定義
DetailMonitor	整数 - スカラー	このフラグを 0 以外の値に設定すると、Oracle に対する詳細監視機能が有効になります。この値は、エージェントが Oracle の詳細監視を開始するまでの監視サイクルの回数です。たとえば、値 5 は、監視サイクル 5 回目ごとに毎回、エージェントが Oracle を詳細に監視することを示します。 デフォルトは 0 です。
MonScript	文字列 - スカラー	詳細監視用に用意されたスクリプトのパス名。デフォルト（基本監視）では、データベース PID のみが監視されます。 メモ : MonScript 属性値に無効な値または空の文字列を設定すると、詳細監視機能は無効になります。 詳細監視用にあらかじめ用意されているスクリプトのパス名は、 /opt/VRTSagents/ha/bin/Oracle/SqlTest.pl です。 MonScript には、/opt/VRTSagents/ha の相対パス名も使えます。相対パス名は、 ./bin/Oracle/SqlTest.pl のように「./」で始まる必要があります。
User	文字列 - スカラー	内部データベースユーザー。詳細監視を行う時に使うデータベースユーザー名。
Pword	文字列 - スカラー	内部データベースユーザーを認証するための暗号化パスワード。 パスワードは、コマンドラインを使って入力する場合にのみ暗号化してください。パスワードは VCS Encrypt ユーティリティを使って暗号化する必要があります。 64 ページの「パスワードの暗号化」を参照してください。
Table	文字列 - スカラー	User/Pword によってアップグレードされる表。

表 A-16 Oracle エージェントの省略可能な属性

省略可能な属性	データ形式と値の種類	定義
Encoding	文字列 - スカラー	Oracle の出力表示で使う Oracle のエンコードに対応する、オペレーティングシステムのエンコードを指定します。たとえば、Oracle の出力が "JAPANESE_JAPAN.JA16EUC" で行われている場合は、"eucJP" が Encoding に対する Solaris の値になります。各エンコードの値については、Oracle および Solaris のマニュアルを参照してください。デフォルトは "" です。
AgentDebug	ブール - スカラー	このフラグが設定されると、追加のデバッグメッセージがログに記録されます。 デフォルトは 0 です。
ContainerName	文字列 - スカラー	監視対象の Oracle インスタンスが動作しているゾーンの名前を指定します。

内部属性

表 A-17 は Oracle エージェントの内部属性の一覧です。この属性は内部での使用のみを目的としています。この属性の値は修正しないことを推奨します。

表 A-17 Oracle エージェントの内部属性

内部属性	データ形式と値の種類	定義
AgentDirectory	静的 - 文字列	Oracle エージェントに関連するバイナリ、スクリプト、他のファイルの場所を指定します。 デフォルトは、/opt/VRTSagents/ha/bin/Oracle です。

Netlsnr リソースタイプ

Veritas High Availability Agent for Oracle の Netlsnr エージェントは、VCS の Netlsnr リソースタイプによって表されます。

```
type Netlsnr (  
    static str AgentDirectory = "/opt/VRTSagents/ha/bin/Netlsnr"  
    static str ContainerType = Zone  
    static keylist SupportedActions = { VRTS_GetInstanceName,  
        VRTS_GetRunningServices }  
    static str ArgList[] = { Owner, Home, TnsAdmin, Listener,  
        EnvFile, MonScript, LsnrPwd, AgentDebug, Encoding }  
    str Owner  
    str Home  
    str TnsAdmin  
    str Listener  
    str EnvFile  
    str MonScript = "./bin/Netlsnr/LsnrTest.pl"  
    str LsnrPwd  
    boolean AgentDebug = 0  
    str Encoding  
    str ContainerName  
)
```

Netlsnr 属性の定義

Netlsnr エージェントの属性の説明を見直してください。エージェントの属性は、必須、省略可能、内部に分類されます。

必須属性

表 A-18 は Netlsnr エージェントの必須属性の一覧です。必須属性には値を割り当てる必要があります。

表 A-18 Netlsnr エージェントの必須属性

必須属性	データ形式と値の種類	定義
Owner	文字列 - スカラー	Oracle ユーザー。/etc/passwd ファイルで、実行ファイルとデータベースファイルの所有者として定義されます。
Home	文字列 - スカラー	Oracle バイナリおよび設定ファイルへの \$ORACLE_HOME パス。たとえば、/opt/ora_home のように指定します。 パスの末尾には、スラッシュ (/) を付けしないでください。

省略可能な属性

表 A-19 は Netlsnr エージェントの省略可能な属性の一覧です。省略可能な属性は必要に応じて設定できます。

表 A-19 Netlsnr エージェントの省略可能な属性

省略可能な属性	データ形式と値の種類	定義
ContainerType	静的 - 文字列	この属性を Zone に設定すると、Netlsnr エージェントは、Solaris 10 ゾーンで動作しているリスナープロセスを監視します。この属性を空のままにした場合は、監視しません。
TnsAdmin	文字列 - スカラー	リスナー設定ファイル (listener.ora) が配置されているディレクトリへの \$TNS_ADMIN パス。 デフォルトは、 /var/opt/oracle です。
Listener	文字列 - スカラー	リスナー名。Netlsnr エージェントおよび Oracle データベースサーバーでは、Listener の名前の大文字または小文字は区別されます。 デフォルトは LISTENER です。
LsnrPwd	文字列 - スカラー	リスナーの停止および監視に使われる、VCS の暗号化パスワード。このパスワードは、リスナー設定ファイルで設定されます。 パスワードは、コマンドラインを使って入力する場合にのみ暗号化してください。パスワードは VCS Encrypt ユーティリティを使って暗号化する必要があります。 64 ページの「 パスワードの暗号化 」を参照してください。
EnvFile	文字列 - スカラー	エントリポイントスクリプトが参照するファイルを絶対パスで指定します。このファイルには、Oracle リスナー環境用にユーザーが設定する LD_LIBRARY_PATH などの環境変数が含まれます。 ファイル内での環境変数を定義するための構文は、Owner のログインシェルに応じて異なります。このファイルは、Owner が読み取りアクセス権を持っているファイルであることが必要です。このファイルには、ユーザーに入力を促すようなメッセージを含まないようにしてください。

表 A-19 Netlsnr エージェントの省略可能な属性

省略可能な属性	データ形式と値の種類	定義
MonScript	文字列 - スカラー	<p>詳細監視用に用意されたスクリプトのパス名。デフォルト（基本監視）では、リスナープロセスのみが監視されます。</p> <p>メモ : MonScript 属性値に無効な値または空の文字列を設定すると、詳細監視機能は無効になります。</p> <p>詳細監視用にあらかじめ用意されているスクリプトのパス名は、 /opt/VRTSagents/ha/bin/Netlsnr/LsnrTest.pl です。</p> <p>MonScript には、/opt/VRTSagents/ha の相対パス名も使えます。相対パス名は、 ./bin/Netlsnr/LsnrTest.pl のように「./」で始まる必要があります。</p>
Encoding	文字列 - スカラー	<p>Oracle の出力表示で使う Oracle のエンコードに対応する、オペレーティングシステムのエンコードを指定します。たとえば、Oracle の出力が "JAPANESE_JAPAN_JA16EUC" で行われている場合は、"eucJP" が Encoding に対する Solaris の値になります。各エンコードの値については、Oracle および Solaris のマニュアルを参照してください。デフォルトは "" です。</p>
AgentDebug	ブール	<p>このフラグが設定されると、追加のデバッグメッセージがログに記録されます。</p> <p>デフォルトは 0 です。</p>
ContainerName	文字列 - スカラー	<p>監視対象のリスナープロセスが動作しているゾーンの名前を指定します。</p>

内部属性

表 A-20 は Netlsnr エージェントの内部属性の一覧です。この属性は内部での使用のみを目的としています。この属性の値は修正しないことを推奨します。

表 A-20 Oracle エージェントの内部属性

内部属性	データ形式と値の種類	定義
AgentDirectory	静的 - 文字列	Netlsnr エージェントに関連するバイナリ、スクリプト、他のファイルの場所を指定します。 デフォルトは、 <code>/opt/VRTSagents/ha/bin/Netlsnr</code> です。

設定例

この付録では、次の内容について説明します。

- 単一の Oracle インスタンスの設定例
- 複数の Oracle インスタンス（単一リスナー）の設定例
- 複数のインスタンス（複数リスナー）の設定例
- 共有サーバーサポートの Oracle の設定例

Oracle エンタープライズエージェントの設定例について

設定例には、VCS クラスタ内の Oracle の状態を監視するよう設定された典型的なサービスグループについての説明が含まれています。

42 ページの「[VCS における Oracle の設定について](#)」を参照してください。

依存関係のグラフのサンプルは、サービスグループ間におけるリソースタイプ、リソースおよびリソースの依存関係を示しています。参照用として、設定ファイル（main.cf）の例も記載されています。

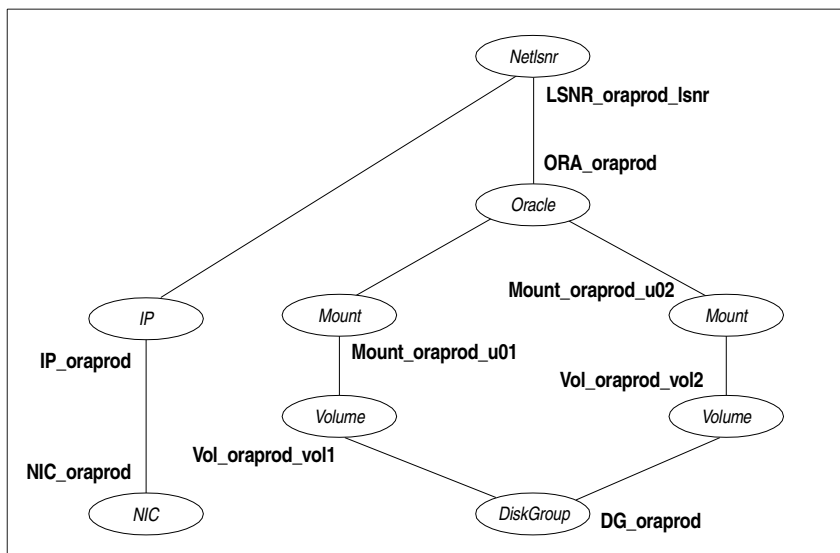
エージェントを設定する前に、これらの依存関係を十分確認してください。VCS のリソースタイプの詳細については、『Veritas Cluster Server 付属エージェントリファレンスガイド』を参照してください。

単一の Oracle インスタンスの設定例

[図 B-4](#) は、VCS クラスタ内の Oracle インスタンスの状態を監視するよう設定された典型的なサービスグループを示しています。

94 ページの「[Solaris ゾーンに設定された単一の Oracle インスタンス](#)」を参照してください。

図 B-4 単一の Oracle インスタンスの依存関係グラフ



クラスタ内の共有ディスクグループとボリュームは、それぞれ **DiskGroup** タイプおよび **Volume** タイプのリソースとして設定します。これらのボリュームは、**Mount** エージェントによりマウントされます。サービスグループ用の仮想 IP アドレスは、**IP** および **NIC** リソースタイプのリソースにより設定されます。これらの各リソースがオンラインにされた後で、**Oracle** サーバーを起動できます。

Veritas Volume Manager を使う設定になっていない場合は、**DiskGroup** および **Volume** リソースタイプの代わりに、**DiskReservation** リソースタイプを使って共有ストレージを設定します。

単一の Oracle インスタンスの設定ファイルの例

Oracle リソースタイプの設定例を見直してください。main.cf ファイルで次のように設定されます。

```
include "types.cf"
include "OracleTypes.cf"
```

```
cluster vcs (
)

system vcssun48 (
)

system vcssun49 (
)
```

```

group ORA_PROD_Group (
  SystemList = { vcssun48 = 0, vcssun49 = 1 }
  AutoStartList = { vcssun48 }
)

DiskGroup DG_oraprod (
  DiskGroup = ora_prod_dg
  StartVolumes = 0
  StopVolumes = 0
)

IP IP_oraprod (
  Device = hme0
  Address = "10.212.102.13"
)

Mount Mount_oraprod_u01 (
  MountPoint = "/prod/u01"
  BlockDevice = "/dev/vx/dsk/ora_prod_dg/u01-vol"
  FSType = vxfs
  FsckOpt = "-n"
)

Mount Mount_oraprod_u02 (
  MountPoint = "/prod/u02"
  BlockDevice = "/dev/vx/dsk/ora_prod_dg/u02-vol"
  FSType = vxfs
  FsckOpt = "-n"
)

NIC NIC_oraprod (
  Device = hme0
)

Netlsnr LSNR_oraprod_lsnr (
  Owner = oraprod
  Home = "/orahome/Oracle"
  TnsAdmin = "/orahome/Oracle/network/admin"
  Listener = LISTENER_PROD
  MonScript = "./bin/Netlsnr/LsnrTest.pl"
  LsnrPwd = cqfOdoOolOo
)

Oracle ORA_oraprod (
  Sid = PROD
  Owner = oraprod
  Home = "/orahome/Oracle"
  EnvFile = "/tmp/env.sh"
  DetailMonitor = 1
  MonScript = "./bin/Oracle/SqlTest.pl"
  User = thor
  Pword = hvlTptWvj
)

```

```
Table = thor
MonitorOption = 0
)

Volume Vol_oraprod_vol1 (
  Volume = u01-vol
  DiskGroup = ora_prod_dg
)

Volume Vol_oraprod_vol2 (
  Volume = u02-vol
  DiskGroup = ora_prod_dg
)

IP_oraprod requires NIC_oraprod
LSNR_oraprod_lsnr requires IP_oraprod
LSNR_oraprod_lsnr requires ORA_oraprod
Mount_oraprod_u01 requires Vol_oraprod_vol1
Mount_oraprod_u02 requires Vol_oraprod_vol2
ORA_oraprod requires Mount_oraprod_u01
ORA_oraprod requires Mount_oraprod_u02
Vol_oraprod_vol1 requires DG_oraprod
Vol_oraprod_vol2 requires DG_oraprod
```

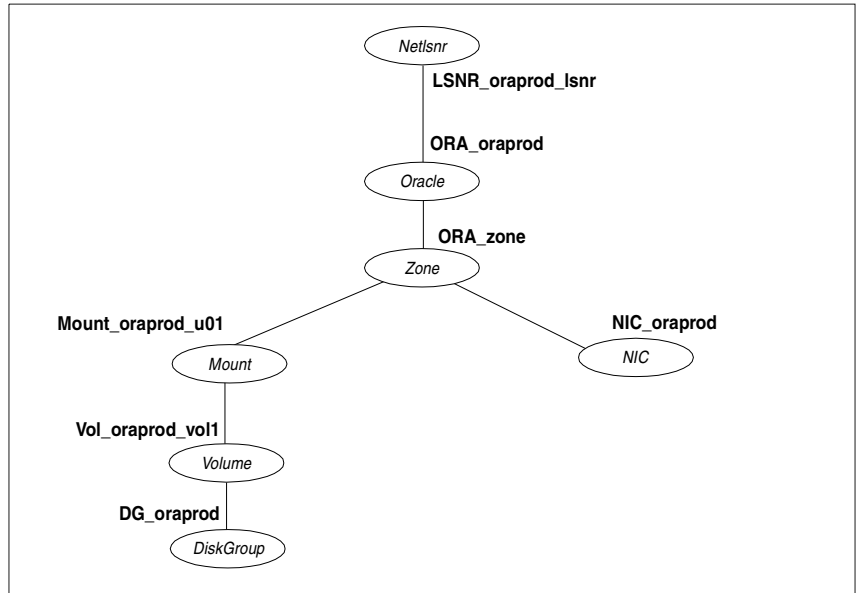
Solaris ゾーンに設定された単一の Oracle インスタンス

次の例は、Solaris ゾーンに設定された Oracle インスタンスの状態を監視するように設定された典型的なサービスグループを示しています。

ローカルディスク上のゾーンルート

図 B-5 は、ローカルディスク上にある Solaris ゾーンの典型的なサービスグループを示しています。ゾーンのルートファイルシステムが、各ノードのローカルディスク上にある場合、システムの起動時に、このファイルシステムがマウントされます。したがって、サービスグループは、ゾーン用に個別の DiskGroup リソースおよび Volume リソースを必要としません。

図 B-5 ローカルディスク上のゾーンにある単一の Oracle インスタンス



クラスタ内の共有ディスクグループとボリュームは、それぞれ **DiskGroup** タイプおよび **Volume** タイプのリソースとして設定します。これらのボリュームは、**Mount** エージェントによりマウントされます。**Solaris** ゾーンは、ゾーンリソースにより監視されます。このゾーンリソースは、**Mount** リソースおよび **NIC** リソースに依存します。これらの各リソースがオンラインにされた後で、**Oracle** サーバーを起動できます。

Veritas Volume Manager を使う設定になっていない場合は、**DiskGroup** および **Volume** リソースタイプの代わりに、**DiskReservation** リソースタイプを使って共有ストレージを設定します。

設定例

リソースタイプ **Zone** は、**main.cf** で次のように設定できます。

```

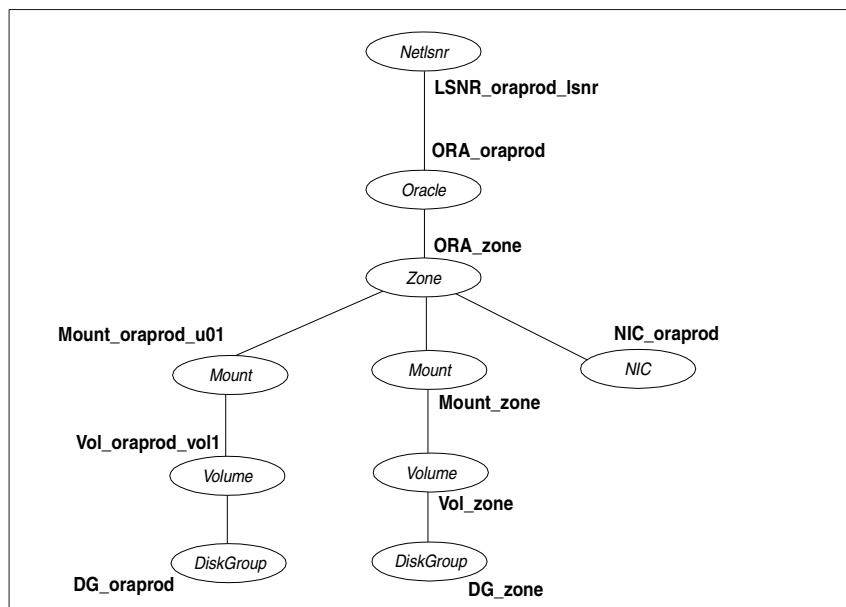
Zone ORA_zone (
    ZoneName = z-ora
)
  
```

このリソースの追加に伴って、**NIC** リソースおよび **Mount** リソースが **Zone** リソースに依存するように、サービスグループ設定も変更されます。

共有ディスク上のゾーンルート

図 B-6 は、共有ディスク上にある Solaris ゾーン内の典型的なサービスグループを示しています。ゾーンのルートファイルシステムが共有ディスク上に置かれている場合、このファイルシステムは、VCS によりマウントされる必要があります。したがって、ゾーン用に個別の DiskGroup リソースと Volume リソースが必要になります。

図 B-6 共有ディスク上のゾーンにある単一の Oracle インスタンス



クラスタ内の共有ディスクグループとボリュームは、それぞれ DiskGroup タイプおよび Volume タイプのリソースとして設定します。これらのボリュームは、Mount エージェントによりマウントされます。Solaris ゾーンは、ゾーンリソースにより監視されます。このゾーンリソースは、Mount リソースおよび NIC リソースに依存します。これらの各リソースがオンラインにされた後で、Oracle サーバーを起動できます。

Veritas Volume Manager を使う設定になっていない場合は、DiskGroup および Volume リソースタイプの代わりに、DiskReservation リソースタイプを使って共有ストレージを設定します。

設定例

リソースタイプ **Zone** は、**main.cf** で次のように設定できます。

```
Zone ORA_zone (
    ZoneName = z-ora
)
```

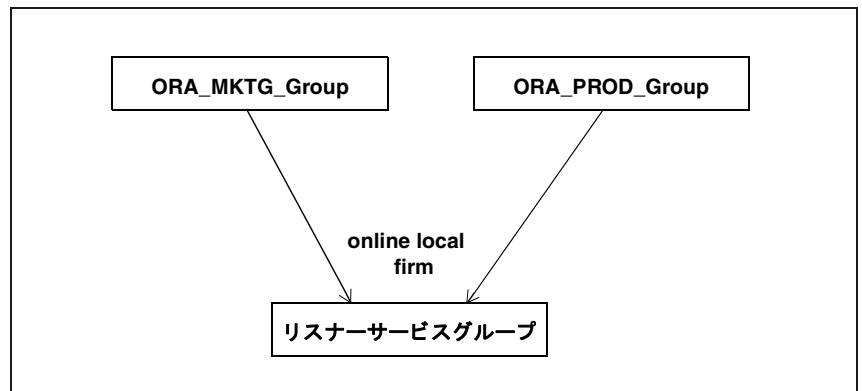
このリソースの追加に伴って、**NIC** リソースおよび **Mount** リソースが **Zone** リソースに依存するように、サービスグループ設定も変更されます。

複数の Oracle インスタンス（単ーリスナー）の設定例

図 B-7 は、1 つのリスナーを共有する 2 つの Oracle インスタンスを監視するための典型的な VCS 設定を示しています。この設定の場合、それぞれの Oracle インスタンスに対して個別にサービスグループを作成します。リスナーについても、リスナーを制御するための専用のサービスグループを作成します。

102 ページの「Solaris ゾーンに設定された複数の Oracle インスタンス（単ーリスナー）」を参照してください。

図 B-7 リスナーを共有する 2 つの Oracle インスタンス



リスナーサービスグループが、Oracle サービスグループに対して依存するようにします。このとき、グループ間の依存関係の種類として、**Online Local Firm** を設定します。

図 B-8 は、VCS 設定における Oracle インスタンスのいずれか 1 つの依存関係グラフを示しています。次の Oracle サービスグループでは、クラスタ内の共有ディスクグループとボリュームは、それぞれ **DiskGroup** タイプおよび **Volume** タイプのリソースとして設定されています。これらのボリュームは、**Mount** エージェントによりマウントされます。

図 B-8 Oracle インスタンスのいずれか 1 つの依存関係グラフ

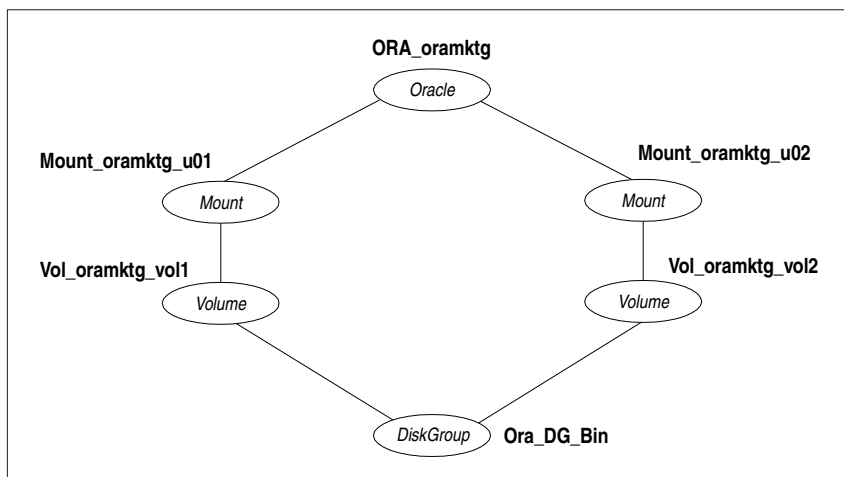
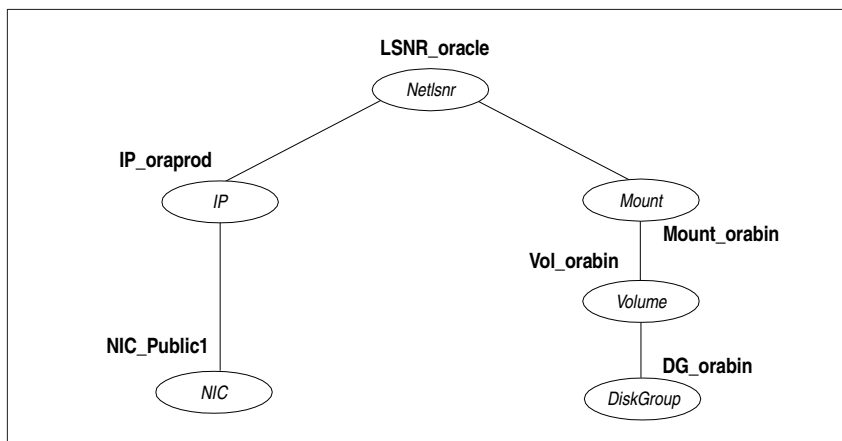


図 B-9 は、VCS 設定で 2 つの Oracle インスタンスが共有するリスナーの依存関係グラフを示しています。リスナーサービスグループでは、IP および NIC タイプのリソースを使って、仮想 IP アドレスの設定を行います。IP および NIC リソースがオンラインにされた後で、リスナーを起動できます。

図 B-9 単一のリスナーの依存関係グラフ



リスナーサービスグループと Oracle サービスグループのリソースがオンラインにされた後で、Oracle サーバーを起動できます。

Veritas Volume Manager を使う設定になっていない場合は、DiskGroup および Volume リソースタイプの代わりに、DiskReservation リソースタイプを使って共有ストレージを設定します。

メモ: この場合、複数のデータベースをクラスタシステムで稼働させることができるように、/etc/system、/etc/passwd、/etc/group および /etc/shadow などのシステムファイルで適切な設定が行われていることを確認してください。特に、複数のインスタンスを実行するために、物理的なメモリ容量や、共有メモリのセグメントについてのカーネルパラメータの設定には、十分注意してください。さらに、システムの故障などで、クラスタ内の 1 台のシステム上ですべてのサービスグループがオンラインとなったとしても、そのシステムが複数のインスタンスの負荷に対処できるように設定することが必要です。

複数の Oracle インスタンス（単ーリスナー）の設定ファイルの例

```
include "types.cf"
include "OracleTypes.cf"

cluster vcs (
)

system vcssun48 (
)

system vcssun49 (
)

group ORA_MKTG_Group (
  SystemList = { vcssun48 = 0, vcssun49 = 1 }
  AutoStartList = { vcssun48 }
)

  DiskGroup DG_oramktg (
    DiskGroup = ora_mktg_dg
    StartVolumes = 0
    StopVolumes = 0
  )

  Mount Mount_oramktg_u01 (
    MountPoint = "/mktg/u01"
    BlockDevice = "/dev/vx/dsk/ora_mktg_dg/u01-vol1"
    FSType = vxfs
    FsckOpt = "-n"
  )

  Mount Mount_oramktg_u02 (
    MountPoint = "/mktg/u02"
    BlockDevice = "/dev/vx/dsk/ora_mktg_dg/u02-vol1"
    FSType = vxfs
  )
)
```

複数の Oracle インスタンス (単ーリスナー) の設定例

```
        FsckOpt = "-n"
    )

    Oracle ORA_oramktg (
        Sid = MKTG
        Owner = oramktg
        Home = "/orahome/Oracle"
        EnvFile = "/tmp/env.sh"
        DetailMonitor = 1
        MonScript = "./bin/Oracle/SqlTest.pl"
        User = thor
        Pword = hvlTptWvj
        Table = thor
        MonitorOption = 0
    )

    Volume Vol_oramktg_vol1 (
        Volume = u01-vol
        DiskGroup = ora_mktg_dg
    )

    Volume Vol_oramktg_vol2 (
        Volume = u02-vol
        DiskGroup = ora_mktg_dg
    )

    requires group Common_Service online firm
    Mount_oramktg_u01 requires Vol_oramktg_vol1
    Mount_oramktg_u02 requires Vol_oramktg_vol2
    ORA_oramktg requires Mount_oramktg_u01
    ORA_oramktg requires Mount_oramktg_u02
    Vol_oramktg_vol1 requires DG_oramktg
    Vol_oramktg_vol2 requires DG_oramktg

group ORA_PROD_Group (
    SystemList = { vcssun48 = 0, vcssun49 = 1 }
    AutoStartList = { vcssun48 }
)

    DiskGroup DG_oraprod (
        DiskGroup = ora_prod_dg
        StartVolumes = 0
        StopVolumes = 0
    )

    Mount Mount_oraprod_u01 (
        MountPoint = "/prod/u01"
        BlockDevice = "/dev/vx/dsk/ora_prod_dg/u01-vol"
        FSType = vxfs
        FsckOpt = "-n"
    )
```

```

Mount Mount_oraprod_u02 (
  MountPoint = "/prod/u02"
  BlockDevice = "/dev/vx/dsk/ora_prod_dg/u02-vol"
  FSType = vxfs
  FsckOpt = "-n"
)

Oracle ORA_oraprod (
  Sid = PROD
  Owner = oraprod
  Home = "/orahome/Oracle"
  EnvFile = "/tmp/env.sh"
  DetailMonitor = 1
  MonScript = "./bin/Oracle/SqlTest.pl"
  User = thor
  Pword = hvlTptWvj
  Table = thor
  MonitorOption = 0
)

Volume Vol_oraprod_vol1 (
  Volume = u01-vol
  DiskGroup = ora_prod_dg
)

Volume Vol_oraprod_vol2 (
  Volume = u02-vol
  DiskGroup = ora_prod_dg
)

requires group Common_Service online firm
Mount_oraprod_u01 requires Vol_oraprod_vol1
Mount_oraprod_u02 requires Vol_oraprod_vol2
ORA_oraprod requires Mount_oraprod_u01
ORA_oraprod requires Mount_oraprod_u02
Vol_oraprod_vol1 requires DG_oraprod
Vol_oraprod_vol2 requires DG_oraprod

group Common_Service (
  SystemList = { vcssun48 = 0, vcssun49 = 1 }
  AutoStartList = { vcssun48 }
)

DiskGroup DG_orabin (
  DiskGroup = ora_bin_dg
  StartVolumes = 0
  StopVolumes = 0
)

IP IP_oraprod (

```

複数の Oracle インスタンス (単ーリスナー) の設定例

```
Device = hme0
Address = "10.212.102.13"
)

Mount Mount_orabin (
  MountPoint = "/orahome/Oracle"
  BlockDevice = "/dev/vx/dsk/ora_bin_dg/u01-vol"
  FSType = vxfs
  FsckOpt = "-n"
)

NIC NIC_Public1 (
  Device = hme0
)

Netlsnr LSNR_oracle (
  Owner = oracle
  Home = "/orahome/Oracle"
  TnsAdmin = "/orahome/Oracle/network/admin"
  Listener = LISTENER_ORACLE
  MonScript = "./bin/Netlsnr/LsnrTest.pl"
  LsnrPwd = cqfOdoOolOo
)

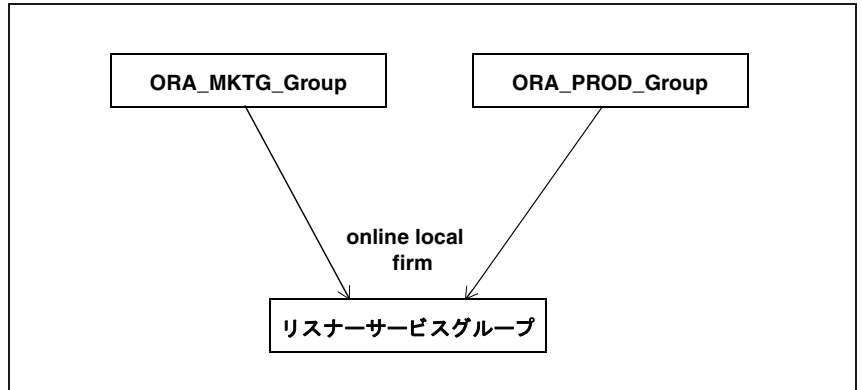
Volume Vol_orabin (
  Volume = u01-vol
  DiskGroup = ora_bin_dg
)

IP_oraprod requires NIC_Public1
LSNR_oracle requires IP_oraprod
LSNR_oracle requires Mount_orabin
Mount_orabin requires Vol_orabin
Vol_orabin requires DG_orabin
```

Solaris ゾーンに設定された複数の Oracle インスタンス (単ーリスナー)

次の例は、Solaris ゾーンに設定された複数の Oracle インスタンスの状態を監視するように設定された典型的なサービスグループを示しています。

図 B-10 Solaris ゾーンでリスナーを共有する2つの Oracle インスタンス



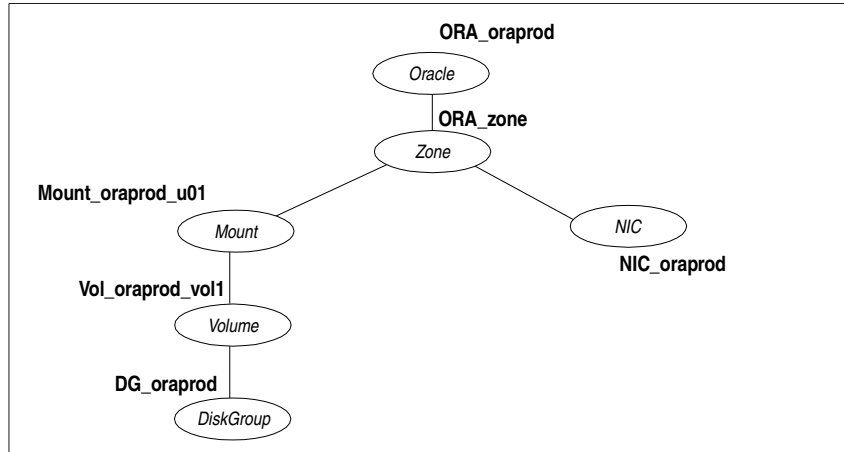
リスナーサービスグループが、Oracle サービスグループに対して依存するようにします。このとき、グループ間の依存関係の種類として、**Online Local Firm**を設定します。

ローカルディスク上のゾーンルート

ゾーンのルートファイルシステムが、各ノードのローカルディスク上にある場合、システムの起動時に、このファイルシステムがマウントされます。したがって、サービスグループは、ゾーン用に個別の **DiskGroup** リソースおよび **Volume** リソースを必要としません。

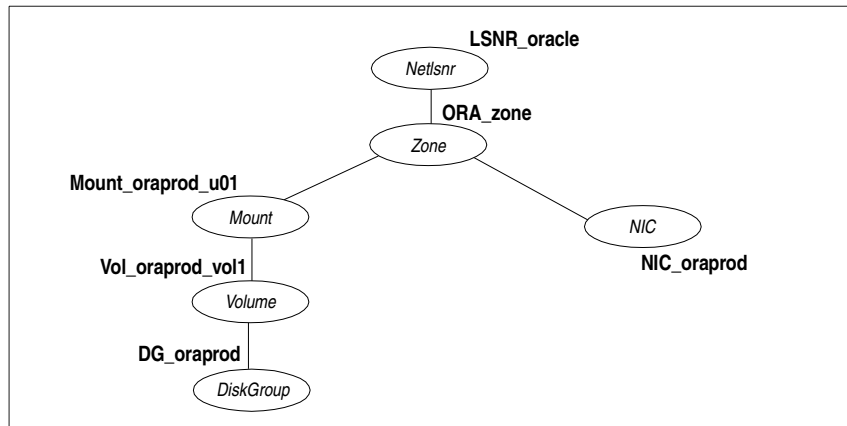
次の Oracle サービスグループでは、クラスタ内の共有ディスクグループとボリュームは、それぞれ **DiskGroup** タイプおよび **Volume** タイプのリソースとして設定されています。これらのボリュームは、**Mount** エージェントによりマウントされます。**Solaris** ゾーンは、ゾーンリソースにより監視されます。このゾーンリソースは、**Mount** リソースおよび **NIC** リソースに依存します。

図 B-11 ローカルディスク上のゾーンにある Oracle インスタンスの依存関係



リスナーサービスグループでは、**Listener** リソースはゾーンリソースに依存しています。ゾーンリソースがオンラインにされた後で、リスナーを起動できます。

図 B-12 ローカルディスク上のゾーンにある単一のリスナーの依存関係



リスナーサービスグループと Oracle サービスグループのリソースがオンラインにされた後で、**Oracle** サーバーを起動できます。

Veritas Volume Manager を使う設定になっていない場合は、**DiskGroup** および **Volume** リソースタイプの代わりに、**DiskReservation** リソースタイプを使って共有ストレージを設定します。

メモ: この場合、複数のデータベースをクラスタシステムで稼働させることができるように、`/etc/system`、`/etc/passwd`、`/etc/group` および `/etc/shadow` などのシステムファイルで適切な設定が行われていることを確認してください。特に、複数のインスタンスを実行するために、物理的なメモリ容量や、共有メモリのセグメントについてのカーネルパラメータの設定には、十分注意してください。さらに、システムの故障などで、クラスタ内の 1 台のシステム上ですべてのサービスグループがオンラインとなったとしても、そのシステムが複数のインスタンスの負荷に対処できるように設定することが必要です。

設定例

リソースタイプ **Zone** は、`main.cf` で次のように設定できます。

```
Zone ORA_zone (  
    ZoneName = z-ora  
)
```

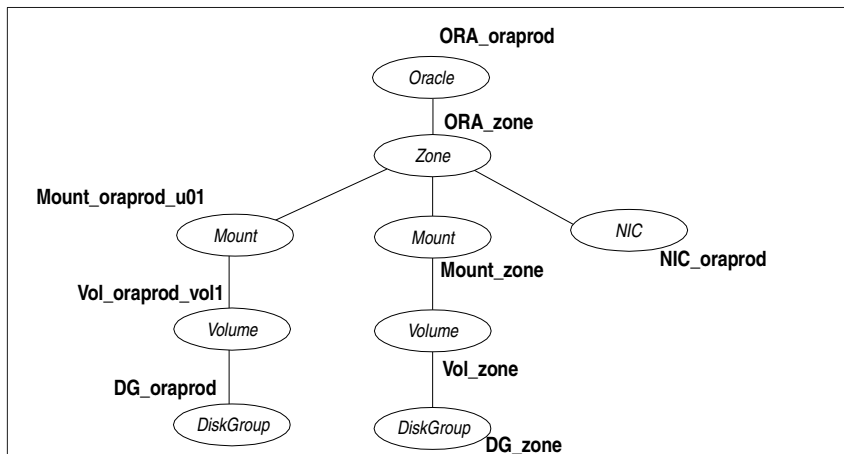
このリソースの追加に伴って、**NIC** リソースおよび **Mount** リソースが **Zone** リソースに依存するように、サービスグループ設定も変更されます。

共有ディスク上のゾーンルート

ゾーンのルートファイルシステムが共有ディスク上に置かれている場合、このファイルシステムは、**VCS** によりマウントされる必要があります。したがって、ゾーン用に個別の **DiskGroup** リソースと **Volume** リソースが必要になります。

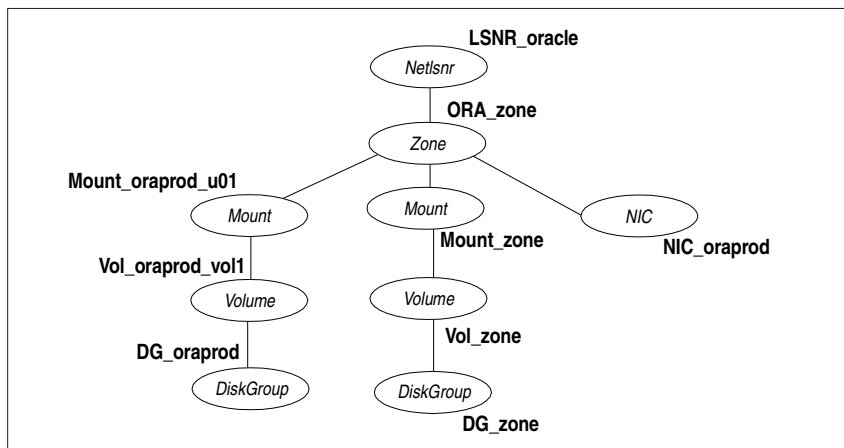
次の **Oracle** サービスグループでは、クラスタ内の共有ディスクグループとボリュームは、それぞれ **DiskGroup** タイプおよび **Volume** タイプのリソースとして設定されています。これらのボリュームは、**Mount** エージェントによりマウントされます。**Solaris** ゾーンは、ゾーンリソースにより監視されます。このゾーンリソースは、**Mount** リソースおよび **NIC** リソースに依存します。

図 B-13 共有ディスク上のゾーンにある Oracle インスタンスの依存関係



リスナーサービスグループでは、Listener リソースはゾーンリソースに依存しています。ゾーンリソースがオンラインにされた後で、リスナーを起動できます。

図 B-14 共有ディスク上のゾーンにある単一のリスナーの依存関係



リスナーサービスグループと Oracle サービスグループのリソースがオンラインにされた後で、Oracle サーバーを起動できます。

Veritas Volume Manager を使う設定になっていない場合は、DiskGroup および Volume リソースタイプの代わりに、DiskReservation リソースタイプを使って共有ストレージを設定します。

メモ: この場合、複数のデータベースをクラスタシステムで稼働させることができるように、`/etc/system`、`/etc/passwd`、`/etc/group` および `/etc/shadow` などのシステムファイルで適切な設定が行われていることを確認してください。特に、複数のインスタンスを実行するために、物理的なメモリ容量や、共有メモリのセグメントについてのカーネルパラメータの設定には、十分注意してください。さらに、システムの故障などで、クラスタ内の 1 台のシステム上ですべてのサービスグループがオンラインとなったとしても、そのシステムが複数のインスタンスの負荷に対処できるように設定することが必要です。

設定例

リソースタイプ `Zone` は、`main.cf` で次のように設定できます。

```
Zone ORA_zone (
    ZoneName = z-ora
)
```

このリソースの追加に伴って、`NIC` リソースおよび `Mount` リソースが `Zone` リソースに依存するように、サービスグループ設定も変更されます。

複数のインスタンス（複数リスナー）の設定例

単一の `Oracle` インスタンスとリスナーで設定されたサービスグループが、複数存在する設定となります。この場合、各サービスグループごとに `Oracle` インスタンスとリスナーの設定を行います。リソースの依存関係については、91 ページの「[単一の Oracle インスタンスの設定例](#)」の説明と同様です。

設定ファイルの例

```
include "types.cf"
include "OracleTypes.cf"

cluster vcs (
)

system vcssun48 (
)

system vcssun49 (
)

group ORA_MKTG_Group (
    SystemList = { vcssun48 = 0, vcssun49 = 1 }
    AutoStartList = { vcssun48 }
)

DiskGroup DG_oramktg (
    DiskGroup = ora_mktg_dg
```

複数のインスタンス（複数リスナー）の設定例

```
        StartVolumes = 0
        StopVolumes = 0
    )

IP IP_oramktg (
    Device = hme0
    Address = "10.212.102.14"
)

Mount Mount_oramktg_u01 (
    MountPoint = "/mktg/u01"
    BlockDevice = "/dev/vx/dsk/ora_mktg_dg/u01-vol"
    FSType = vxfs
    FsckOpt = "-n"
)

Mount Mount_oramktg_u02 (
    MountPoint = "/mktg/u02"
    BlockDevice = "/dev/vx/dsk/ora_mktg_dg/u02-vol"
    FSType = vxfs
    FsckOpt = "-n"
)

Netlsnr LSNR_oramktg_lsnr (
    Owner = oramktg
    Home = "/orahome/Oracle"
    TnsAdmin = "/orahome/Oracle/network/admin"
    Listener = LISTENER_MKTG
    MonScript = "./bin/Netlsnr/LsnrTest.pl"
    LsnrPwd = cqf0do0o10o
)

Oracle ORA_oramktg (
    Sid = MKTG
    Owner = oramktg
    Home = "/orahome/Oracle"
    EnvFile = "/tmp/env.sh"
    DetailMonitor = 1
    MonScript = "./bin/Oracle/SqlTest.pl"
    User = thor
    Pword = hvlTptWvj
    Table = thor
    MonitorOption = 0
)

Proxy NICProxy_oramktg (
    TargetResName = NIC_Public1
)

Volume Vol_oramktg_vol1 (
    Volume = u01-vol
    DiskGroup = ora_mktg_dg
```

```

)

Volume Vol_oramktg_vol2 (
  Volume = u02-vol
  DiskGroup = ora_mktg_dg
)

IP_oramktg requires NICProxy_oramktg
LSNR_oramktg_lsnr requires IP_oramktg
LSNR_oramktg_lsnr requires ORA_oramktg
Mount_oramktg_u01 requires Vol_oramktg_vol1
Mount_oramktg_u02 requires Vol_oramktg_vol2
ORA_oramktg requires Mount_oramktg_u01
ORA_oramktg requires Mount_oramktg_u02
Vol_oramktg_vol1 requires DG_oramktg
Vol_oramktg_vol2 requires DG_oramktg

group ORA_PROD_Group (
  SystemList = { vcssun48 = 0, vcssun49 = 1 }
  AutoStartList = { vcssun48 }
)

DiskGroup DG_oraprod (
  DiskGroup = ora_prod_dg
  StartVolumes = 0
  StopVolumes = 0
)

IP IP_oraprod (
  Device = hme0
  Address = "10.212.102.13"
)

Mount Mount_oraprod_u01 (
  MountPoint = "/prod/u01"
  BlockDevice = "/dev/vx/dsk/ora_prod_dg/u01-vol"
  FSType = vxfs
  FsckOpt = "-n"
)

Mount Mount_oraprod_u02 (
  MountPoint = "/prod/u02"
  BlockDevice = "/dev/vx/dsk/ora_prod_dg/u02-vol"
  FSType = vxfs
  FsckOpt = "-n"
)

Netlsnr LSNR_oraprod_lsnr (
  Owner = oraprod
  Home = "/orahome/Oracle"
  TnsAdmin = "/orahome/Oracle/network/admin"

```

複数のインスタンス（複数リスナー）の設定例

```
Listener = LISTENER_PROD
MonScript = "./bin/Netlsnr/LsnrTest.pl"
LsnrPwd = cqfOdoOo1Oo
)

Oracle ORA_oraprod (
  Sid = PROD
  Owner = oraprod
  Home = "/orahome/Oracle"
  EnvFile = "/tmp/env.sh"
  DetailMonitor = 1
  MonScript = "./bin/Oracle/SqlTest.pl"
  User = thor
  Pword = hvlTptWvj
  Table = thor
  MonitorOption = 0
)

Proxy NICProxy_oraprod (
  TargetResName = NIC_Public1
)

Volume Vol_oraprod_vol1 (
  Volume = u01-vol
  DiskGroup = ora_prod_dg
)

Volume Vol_oraprod_vol2 (
  Volume = u02-vol
  DiskGroup = ora_prod_dg
)

IP_oraprod requires NICProxy_oraprod
LSNR_oraprod_lsnr requires IP_oraprod
LSNR_oraprod_lsnr requires ORA_oraprod
Mount_oraprod_u01 requires Vol_oraprod_vol1
Mount_oraprod_u02 requires Vol_oraprod_vol2
ORA_oraprod requires Mount_oraprod_u01
ORA_oraprod requires Mount_oraprod_u02
Vol_oraprod_vol1 requires DG_oraprod
Vol_oraprod_vol2 requires DG_oraprod

group Parallel_Service (
  SystemList = { vcssun48 = 0, vcssun49 = 1 }
  Parallel = 1
  AutoStartList = { vcssun48 }
)

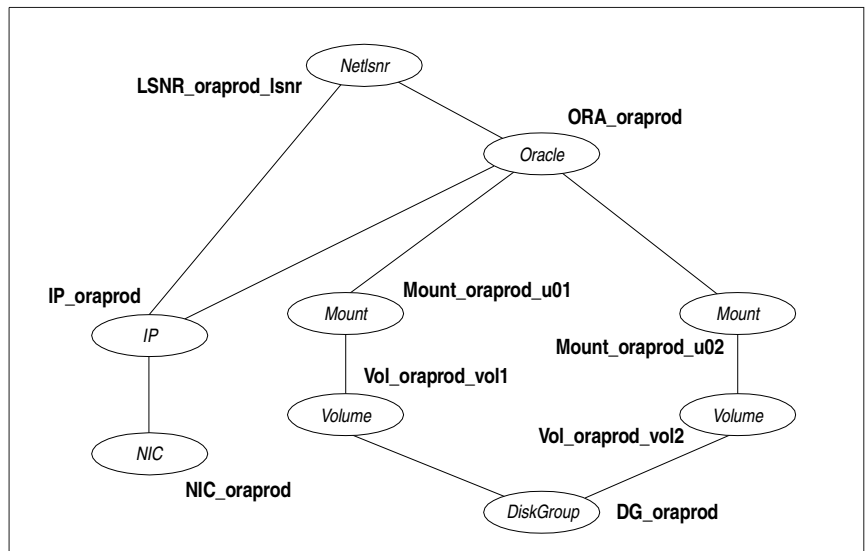
NIC NIC_Public1 (
  Device = hme0
)
```

共有サーバーサポートの Oracle の設定例

図 B-15 は、共有サーバーサポートの Oracle を監視するように設定された典型的なサービスグループを示しています。

Solaris ゾーンで設定される Oracle インスタンスについては、113 ページの「Solaris ゾーンに設定された共有サーバーサポートの Oracle インスタンス」を参照してください。

図 B-15 共有サーバーサポートの Oracle の設定例



クラスタ内の共有ディスクグループとボリュームは、それぞれ **DiskGroup** タイプおよび **Volume** タイプのリソースとして設定します。これらのボリュームは、**Mount** エージェントによりマウントされます。サービスグループ用の仮想 IP アドレスは、**IP** および **NIC** リソースタイプのリソースにより設定されます。これらの各リソースがオンラインにされた後で、**Oracle** サーバーを起動できます。

Veritas Volume Manager を使う設定になっていない場合は、**DiskGroup** および **Volume** リソースタイプの代わりに、**DiskReservation** リソースタイプを使って共有ストレージを設定します。

共有サーバーサポート で設定された Oracle インスタンスの設定ファイルの例

```
include "types.cf"
include "OracleTypes.cf"

cluster vcs (
)

system vcssun48 (
)

system vcssun49 (
)

group ORA_PROD_Group (
    SystemList = { vcssun48 = 0, vcssun49 = 1 }
    AutoStartList = { vcssun48 }
)

DiskGroup DG_oraprod (
    DiskGroup = ora_prod_dg
    StartVolumes = 0
    StopVolumes = 0
)

IP IP_oraprod (
    Device = hme0
    Address = "10.212.102.13"
)

Mount Mount_oraprod_u01 (
    MountPoint = "/prod/u01"
    BlockDevice = "/dev/vx/dsk/ora_prod_dg/u01-vol"
    FSType = vxfs
    FsckOpt = "-n"
)

Mount Mount_oraprod_u02 (
    MountPoint = "/prod/u02"
    BlockDevice = "/dev/vx/dsk/ora_prod_dg/u02-vol"
    FSType = vxfs
    FsckOpt = "-n"
)

NIC NIC_ORAPROD (
    Device = hme0
)

Netlsnr LSNR_oraprod_lsnr (
    Owner = oraprod
    Home = "/orahome/Oracle"
```



```

TnsAdmin = "/orahome/Oracle/network/admin"
Listener = LISTENER_PROD
MonScript = "./bin/Netlsnr/LsnrTest.pl"
LsnrPwd = cqfOdoOolOo
)
Oracle ORA_oraprod (
  Sid = PROD
  Owner = oraprod
  Home = "/orahome/Oracle"
  EnvFile = "/tmp/env.sh"
  DetailMonitor = 1
  MonScript = "./bin/Oracle/SqlTest.pl"
  User = thor
  Pword = hvlTptWvj
  Table = thor
  MonitorOption = 0
)

Volume Vol_oraprod_vol1 (
  Volume = u01-vol
  DiskGroup = ora_prod_dg
)

Volume Vol_oraprod_vol2 (
  Volume = u02-vol
  DiskGroup = ora_prod_dg
)

IP_oraprod requires NIC_ORAPROD
LSNR_oraprod_lsnr requires IP_oraprod
Mount_oraprod_u01 requires Vol_oraprod_vol1
Mount_oraprod_u02 requires Vol_oraprod_vol2
ORA_oraprod requires IP_oraprod
ORA_oraprod requires Mount_oraprod_u01
ORA_oraprod requires Mount_oraprod_u02
Vol_oraprod_vol1 requires DG_oraprod
Vol_oraprod_vol2 requires DG_oraprod

```

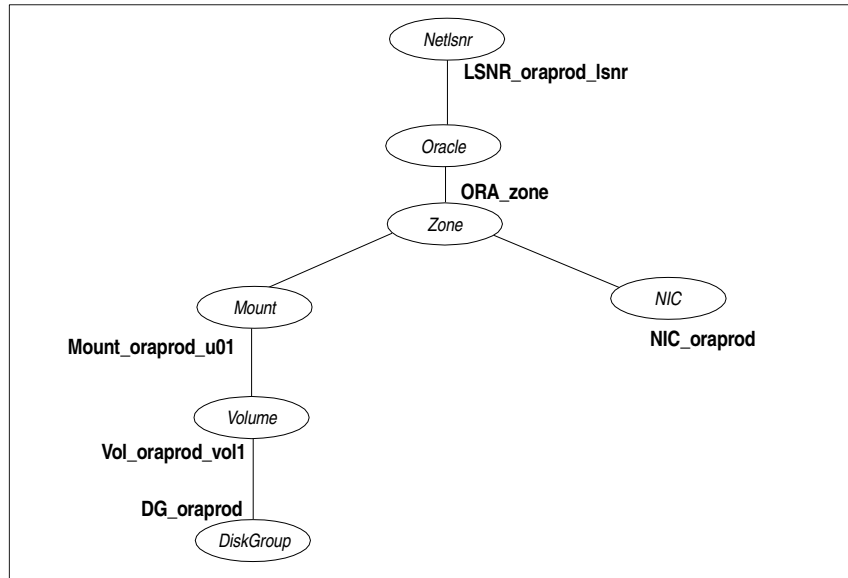
Solaris ゾーンに設定された共有サーバーサポートの Oracle インスタンス

次の例は、Solaris ゾーンに設定された共有サーバーサポートの Oracle インスタンスを監視するように設定された典型的なサービスグループを示しています。

ローカルディスク上のゾーンルート

ゾーンのルートファイルシステムが、各ノードのローカルディスク上にある場合、システムの起動時に、このファイルシステムがマウントされます。したがって、サービスグループは、ゾーン用に個別の **DiskGroup** リソースおよび **Volume** リソースを必要としません。

図 B-16 ローカルディスク上のゾーンにある共有サーバーサポートの Oracle インスタンス



クラスタ内の共有ディスクグループとボリュームは、それぞれ **DiskGroup** タイプおよび **Volume** タイプのリソースとして設定します。これらのボリュームは、**Mount** エージェントによりマウントされます。**Solaris** ゾーンは、ゾーンリソースにより監視されます。このゾーンリソースは、**Mount** リソースおよび **NIC** リソースに依存します。これらの各リソースがオンラインにされた後で、**Oracle** サーバーを起動できます。

Veritas Volume Manager を使う設定になっていない場合は、**DiskGroup** および **Volume** リソースタイプの代わりに、**DiskReservation** リソースタイプを使って共有ストレージを設定します。

設定例

リソースタイプ **Zone** は、**main.cf** で次のように設定できます。

```

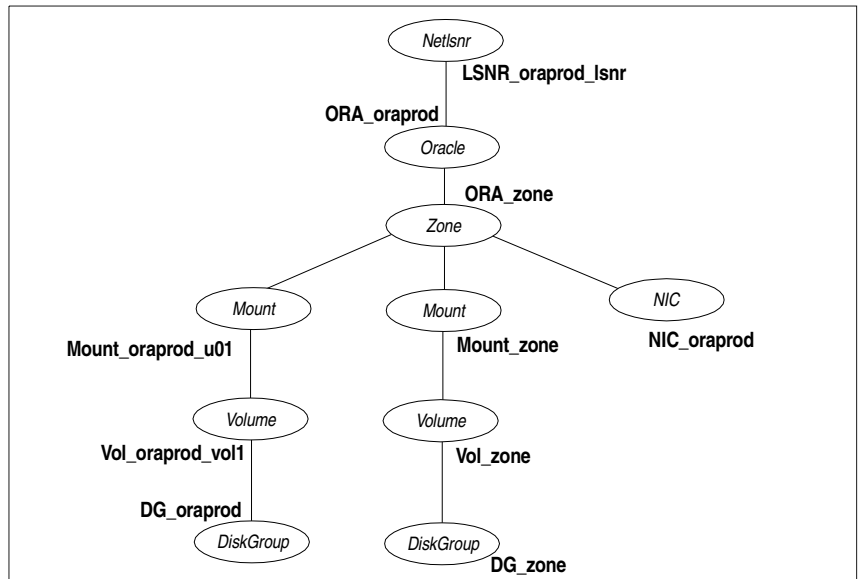
Zone ORA_zone (
    ZoneName = z-ora
)
  
```

このリソースの追加に伴って、**NIC** リソースおよび **Mount** リソースが **Zone** リソースに依存するように、サービスグループ設定も変更されます。

共有ディスク上のゾーンルート

ゾーンのルートファイルシステムが共有ディスク上に置かれている場合、このファイルシステムは、VCS によりマウントされる必要があります。したがって、ゾーン用に個別の **DiskGroup** リソースと **Volume** リソースが必要になります。

図 B-17 共有ディスク上のゾーンにある共有サーバーサポートの Oracle インスタンス



クラスタ内の共有ディスクグループとボリュームは、それぞれ **DiskGroup** タイプおよび **Volume** タイプのリソースとして設定します。これらのボリュームは、**Mount** エージェントによりマウントされます。**Solaris** ゾーンは、ゾーンリソースにより監視されます。このゾーンリソースは、**Mount** リソースおよび **NIC** リソースに依存します。これらの各リソースがオンラインにされた後で、**Oracle** サーバーを起動できます。

Veritas Volume Manager を使う設定になっていない場合は、**DiskGroup** および **Volume** リソースタイプの代わりに、**DiskReservation** リソースタイプを使って共有ストレージを設定します。

設定例

リソースタイプ **Zone** は、**main.cf** で次のように設定できます。

```

Zone ORA_zone (
    ZoneName = z-ora
)
    
```

このリソースの追加に伴って、**NIC** リソースおよび **Mount** リソースが **Zone** リソースに依存するように、サービスグループ設定も変更されます。

ベストプラクティス

この付録では、次の内容について説明します。

- 複数の Oracle インスタンスを設定する場合のベストプラクティス

複数の Oracle インスタンスを設定する場合の ベストプラクティス

VCS 環境で複数の Oracle インスタンスを使うためのベストプラクティスのいくつかを見直してください。

- 設定する各 SID に対して、DBA 権限を持つ UNIX アカウントを作成する。
- 各 Oracle インスタンスに個別のディスクグループがあり、各インスタンスが個別のサービスグループとして設定されていることを確認する。
- セマフォおよび共有メモリの割り当てがすべてのシステムで適切に行われるように、システムパラメータを定義する。
- 各インスタンスが同一バージョンの Oracle を使っている場合でも、それぞれの Oracle インスタンスに対して専用のバイナリセットを使う。
- すべてのインスタンスが同一バージョンの Oracle を使う設定の場合は、ルートディスクまたは、可能であればセカンダリディスクにそのバージョンをインストールする。pfile をデフォルトの場所に配置し、複数のリスナープロセスを定義して、フェールオーバーをわかりやすくする。
- バージョンの異なる Oracle を使う設定の場合は、各バージョンの Oracle に対し、\$ORACLE_HOME を個別に作成する。
- Optimal Flexible Architecture (OFA) の標準 (/u`xxx`/`<SID>`) に従う。クラスタの設定で、この標準をアプリケーションに合わせてカスタマイズする。たとえば、/app/u`xxx`/`<SID>` のように設定する。
- バージョンの異なる Oracle-に付属するリスナーには下位互換性がない場合がある。単一の listener.ora ファイルを作成する場合は、リスナーがクラス

タ内の他のバージョンの **Oracle** をサポートしていることを確認する。各バージョンの **Oracle** に対しては、個別の **Envfile** を作成する。

- リスナーがそれぞれ異なる仮想アドレスをリスニングすることを確認する。さらに、同じポートをリスニングすることのないよう、リスナーに異なる名前を割り当てる。
- **pfile** をシステム間で連携させる。データベースの同一インスタンスに関しては、参照する **pfile** をノード間で一致させる。

Oracle 用 VCS クラスタにおける SPFILE の使用

この付録では、次の内容について説明します。

- Oracle 初期化パラメータファイルについて
- Oracle インスタンスの起動
- VCS クラスタでの SPFILE の使用

Oracle 初期化パラメータファイルについて

Oracle9i 以前のバージョンの Oracle では、テキストファイルの `initSID.ora` を初期化パラメータファイルとして使い、データベースインスタンスを起動していました。このファイルには、データベースセッション中に変更されたインスタンスパラメータは保存されないため、変更内容を手動で初期化パラメータファイルに反映させる必要がありました。

Oracle9i では、バイナリファイルとしてデータベースサーバーに格納される SPFILE が追加されました。この機能により、インスタンスパラメータへの変更を、すべての起動および停止のプロシージャ間で統一することができます。

Oracle インスタンスの起動

Oracle9i 以上のバージョンでは、Oracle インスタンスを次の方法で起動できます。

- デフォルトの SPFILE である `spfileSID.ora` を使う
- デフォルトの `init.ora` ファイルである `initSID.ora` を使う
- 初期化パラメータファイル `init.ora` を指定する
- 初期化パラメータファイル `init.ora` で SPFILE を指定する

SPFILE 文を記述せずに `startup` コマンドを実行した場合、Oracle は SPFILE から初期化パラメータファイルを読み込みます。Solaris プラットフォームの場合、SPFILE または PFILE のデフォルトの場所は `$ORACLE_HOME/dbs` です。

Oracle は、次に示す順にファイル名を確認し、初期化パラメータファイルを見つけます。

- SPFILESID.ora
- SPFILE.ora
- initSID.ora

VCS クラスタでの SPFILE の使用

Veritas High Availability Agent for Oracle の使用時には、PFILE を指定してデータベースインスタンスを起動できます。PFILE を指定しない場合、データベースインスタンスはデフォルトの SPFILE を使って起動されます。

エージェント属性 `Pfile` は、PFILE の場所を指定します。SPFILE を使う設定の場合は、PFILE から作成される SPFILE の場所を PFILE 内のパラメータに指定する必要があります。

メモ: SPFILE 内のセッションパラメータに対する変更をインスタンスのフェールオーバー間で統一する場合は、SPFILE を共有ストレージ上に保存することを推奨します。

PFILE における SPFILE の位置指定

PFILE で SPFILE の場所を指定するには、PFILE を作成し、PFILE 内に次のエントリを指定します。

```
SPFILE = spfile_location
```

変数 `spfile_location` は、SPFILE の絶対パスです。次に例を示します。

```
SPFILE = /database/startup/spfileora1.ora
```

この場合、データベースを起動するには次のコマンドを使います。

```
startup pfile=location_of_pfile
```

PFILE からの SPFILE 作成

SPFILE は、PFILE から作成する必要があります。SPFILE の作成には、`sysdba` または `sysoper` システム権限が必要です。

次のコマンドを使って SPFILE を作成できます。

```
CREATE SPFILE [= spfile_name] FROM PFILE [= pfile_name ];
```

SPFILE の絶対パスを指定しない場合は、このコマンドによって SPFILE がデフォルトの場所 (Solaris では `$ORACLE_HOME/dbs`) に作成されます。

索引

記号

- \$ORACLE_HOME
 - 共有ストレージ 24
 - 制限 26
 - 場所 24
 - ローカルディスク 24

C

- Cluster Manager 60
- CSSD デーモン 32
- CSSD デーモン、無効化
 - Solaris 10 の場合 32
 - Solaris 9 以前の場合 32

N

- Netlsnr エージェント
 - 属性の定義 87
 - 動作 17
 - リソースタイプ 87
- Netlsnr エージェントの属性
 - AgentDebug 89
 - ContainerName 89
 - ContainerType 88
 - Encoding 89
 - EnvFile 88
 - Home 87
 - Listener 88
 - LsnrPwd 88
 - MonScript 89
 - Owner 87
 - TnsAdmin 88
- NLS 情報、定義 26

O

- Oracle
 - インストール 23
 - エラー操作 19
 - 起動オプション 12
 - サポートされる設定 42

- 停止オプション 13
- データベース表領域 25
- 複数のインスタンス 24

- Oracle エージェント
 - 属性の定義 82
 - 動作 11
 - リソースタイプ 81
- Oracle エージェントの属性
 - AgentDebug 86
 - AutoEndBkup 84
 - ContainerName 86
 - ContainerType 83
 - DetailMonitor 85
 - Encoding 86
 - EnvFile 84
 - Home 82
 - MonitorOption 84
 - MonScript 85
 - Owner 82
 - Pfile 84
 - Pword 85
 - ShutdownOpt 83
 - Sid 82
 - StartupOpt 83
 - Table 85
 - User 85

- Oracle データベース、ホットバックアップ 26

- Oracle の設定
 - 単一インスタンス 91
 - 複数のインスタンス (単一リスナー) 97
 - 複数のインスタンス (複数リスナー) 45
- Oracle の設定例 20

S

- Solaris 10 ズーン 18
- SPFILE
 - VCS での使用 119
 - 作成 120
 - 場所 120

V

vcsencrypt ユーティリティ 64

う

ウィザード 50

え

エージェントの削除 69

エージェントの動作

Netlsnr エージェント 17

Oracle エージェント 11

エラーメッセージ 73

エンタープライズエージェント

Cluster Manager を使った設定 60

アップグレード 39

インストール 38

ウィザードを使った設定 50

エラーメッセージ 73

概要 9

コマンドラインを使った設定 62

削除 72

設定の変更 71

トラブルシューティング 73

標準設定 20

無効化 71

か

仮想ファイアドリル 16

監視

基本 14

詳細 14

監視オプション 18

き

起動オプション 12

基本監視 14

プロセス 14

ヘルスチェック 14

こ

コアファイル、場所 25

さ

サービスグループ

オフライン 70

オンライン化 69

切り替え 70

サービスグループのオフライン化 70

サービスグループのオンライン化 69

サービスグループの切り替え 70

し

詳細監視 14

Netlsnr に対する無効化 68

Netlsnr に対する有効化 68

Oracle に対する無効化 68

Oracle に対する有効化 68

概要 65

設定 65

せ

設定

標準設定 20

変更 71

設定ウィザード 50

設定ファイル 91

設定ファイルの例 91

そ

属性

変更 61

編集 61

属性の定義

Netlsnr エージェント 87

Oracle エージェント 82

属性の編集 61

ゾーン、Solaris 10 を使用 18

て

停止オプション 13

データベース表領域 25

と

動作

Netlsnr エージェント 17

Oracle エージェント 11

トラブルシューティング 73

は

パスワードの暗号化 64

パスワード、暗号化 64

ふ

フェールオーバー 18
プロセス監視 14

へ

ヘルスチェック API 14
ヘルスチェック監視 14

ほ

ホットバックアップ 26

め

メッセージログ 73

り

リソース属性、編集 61
リソースタイプ
 Netlsnr 87
 Oracle 81

