

# Cluster Server 8.0 Agent for Oracle インストールおよび設 定ガイド - Solaris

最終更新: 2022-01-25

## 法的通知と登録商標

Copyright © 2022 Veritas Technologies LLC. All rights reserved.

Veritas、Veritas ロゴは、Veritas Technologies LLC または関連会社の米国およびその他の国における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。

この製品には、サードパーティへの著作権を示す必要のあるサードパーティのソフトウェアが含まれる場合があります(「サードパーティプログラム」)。一部のサードパーティプログラムは、オープンソースまたはフリーソフトウェアライセンスの下で利用できます。本ソフトウェアに含まれる本使用許諾契約は、オープンソースまたはフリーソフトウェアライセンスでお客様が有する権利または義務を変更しないものとします。このVeritas製品に付属のサードパーティ法的通知文書を参照するか、<https://www.veritas.com/about/legal/license-agreements> をご覧ください。

本書に記載されている製品は、その使用、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ます。Veritas Technologies LLC 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。

本書は、現状のまま提供されるものであり、その商品性、特定目的への適合性、または不侵害の暗黙的な保証を含む、明示的あるいは暗黙的な条件、表明、および保証はすべて免責されるものとします。ただし、これらの免責が法的に無効であるとされる場合を除きます。Veritas Technologies LLC は、この文書の供給、履行、または使用に関連して付随的または間接的に起こる損害に対して責任を負いません。本書に記載の情報は、予告なく変更される場合があります。

ライセンスソフトウェアおよび文書は、FAR 12.212 に定義される商用コンピュータソフトウェアと見なされ、Veritasがオンプレミスサービスまたはホストサービスとして提供するかを問わず、必要に応じて FAR 52.227-19「商用コンピュータソフトウェア - 制限される権利 (Commercial Computer Software - Restricted Rights)」、DFARS 227.7202「商用コンピュータソフトウェアおよび商用コンピュータソフトウェア文書 (Commercial Computer Software and Commercial Computer Software Documentation)」、およびそれらの後継の規制に定める制限される権利の対象となります。米国政府によるライセンス対象ソフトウェアおよび資料の使用、修正、複製のリリース、実演、表示または開示は、本使用許諾契約の条項に従ってのみ行われるものとします。

Veritas Technologies LLC  
2625 Augustine Drive  
Santa Clara, CA 95054  
<http://www.veritas.com>

## テクニカルサポート

テクニカルサポートはグローバルにサポートセンターを管理しています。すべてのサポートサービスは、サポート契約と現在のエンタープライズテクニカルサポートポリシーに応じて提供されます。サポート内容およびテクニカルサポートの利用方法に関する情報については、次の Web サイトにアクセスしてください。

<https://www.veritas.com/support>

Veritas Account 情報は、次の URL で管理できます。  
<https://my.veritas.com>

現在のサポート契約についてご不明な点がある場合は、次に示すお住まいの地域のサポート契約管理チームに電子メールでお問い合わせください。

世界中 (日本以外)

[CustomerCare@veritas.com](mailto:CustomerCare@veritas.com)

日本

[CustomerCare\\_Japan@veritas.com](mailto:CustomerCare_Japan@veritas.com)

## マニュアル

マニュアルの最新バージョンがあることを確認してください。各マニュアルには、2 ページ目に最終更新日が記載されています。最新のマニュアルは、ベリタスの Web サイトで入手できます。

<https://sort.veritas.com/documents>

## マニュアルに関するご意見やご感想

ご意見、ご感想をお待ちしています。改善すべき点や、マニュアル上の誤記、欠落がありましたらお寄せください。お送りいただく際は、マニュアルの題名とバージョン、章のタイトル、セクションのタイトルを明記してください。フィードバックの送信先:

[infoscaledocs@veritas.com](mailto:infoscaledocs@veritas.com)

ベリタスのコミュニティサイトで、マニュアルに関する情報を確認したり、質問を投稿することもできます。

<http://www.veritas.com/community/>

## Veritas Services and Operations Readiness Tools (SORT)

Veritas SORT (Services and Operations Readiness Tools) は、時間のかかる特定の管理タスクを自動化および単純化するための情報とツールを提供する Web サイトです。製品に応じて、SORT はインストールとアップグレードの準備、データセンターのリスクの識別、効率性の改善に役立ちます。使用している製品に対して SORT が提供しているサービスおよびツールについては、次のデータシートを参照してください。

[https://sort.veritas.com/data/support/SORT\\_Data\\_Sheet.pdf](https://sort.veritas.com/data/support/SORT_Data_Sheet.pdf)

<b>第 1 章</b>	<b>Cluster Server agent for Oracle の導入</b> .....	9
	Cluster Server agent for Oracle について .....	9
	Oracle ASM のエージェントについて .....	10
	Oracle の VCS エージェントのサポート対象ソフトウェア .....	10
	エージェントが Oracle の高可用性を保証するしくみ .....	10
	Oracle エージェントと Netlsnr エージェントがインテリジェントなリソース監視をサポートするしくみ .....	11
	エージェントによる、Solaris ゾーンで実行される Oracle インスタンスの監視 .....	12
	エージェントが Oracle ASM インスタンスの高可用性を確保する方法 .....	12
	エージェントの Oracle コンテナデータベース (CDB) とプラグ可能データベース (PDB) のサポート方法 .....	13
	Oracle 用 Cluster Server エージェントの機能について .....	13
	Oracle エージェント関数 .....	14
	Netlsnr エージェント関数 .....	28
	ASMinst エージェントの機能 .....	29
	ASMDG エージェントの機能 .....	32
	VCS clusterにおける標準的な Oracle 設定 .....	33
	VCS clusterでの Oracle 設定について .....	34
<b>第 2 章</b>	<b>Oracle のインストールと設定</b> .....	36
	VCS 環境での Oracle のインストール .....	36
	VCS 環境に Oracle をインストールする前に .....	37
	Oracle のインストールのための VCS の必要条件について .....	37
	\$ORACLE_HOME の場所 .....	41
	エージェント属性内の \$ORACLE_HOME の長いパス名の置き換え .....	42
	ホットバックアップ中に VCS nodeに障害が発生した後の Oracle のフェールオーバー .....	43
	VCS のための Oracle インストールタスクについて .....	44
	共有ディスクで行う \$ORACLE_HOME のインストールタスク .....	45
	ローカルディスクで行う \$ORACLE_HOME のインストールタスク .....	46
	ASM を使用する Oracle のインストールタスク .....	46
	VCS 環境での ASM バイナリのインストール .....	48

	Oracle Restart 機能をサポートするための VCS 設定 .....	50
	クラスタの最初のノードでの Oracle ASM の設定 .....	51
	ASM で管理されたデータベースのためのクラスタデーモンの有効化 .....	52
	ASM の自動起動の無効化 .....	52
	残りのノードでの ASM の構成と起動 .....	53
	クラスタの最初のノードへの Oracle バイナリのインストール .....	54
	Oracle クラスタデーモンを無効にします .....	54
	Oracle データベースの設定 .....	55
	\$ORACLE_BASE/admin/SID ディレクトリのコピー .....	56
	Oracle ASM の初期化パラメータファイルのコピー .....	56
	Oracle データベースへのアクセスの確認 .....	56
<b>第 3 章</b>	<b>Oracle 用エージェントのインストールと削除 .....</b>	<b>59</b>
	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 .....	59
	VCS agent for Oracle ソフトウェアのインストール .....	59
	VCS agent for Oracle のアップグレード .....	62
	VCS agent for Oracle のアップグレード .....	62
	VCS 8.0 で使うための VCS 3.5 agent for Oracle のアップグレード (Solaris SPARC) .....	64
	Cluster Server agent for Oracle の無効化 .....	65
	Cluster Server agent for Oracle の削除 .....	66
<b>第 4 章</b>	<b>Oracle 用 VCS サービスグループの設定 .....</b>	<b>67</b>
	Oracle 用サービスグループの設定について .....	67
	VCS における Oracle インスタンスの設定 .....	68
	VCS における単一の Oracle インスタンスの設定 .....	68
	VCS における複数の Oracle インスタンス(単一リスナー)の設定 .....	70
	VCS における複数の Oracle インスタンス(複数リスナー)の設定 .....	72
	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 .....	74
	Oracle の VCS サービスグループを設定する前に .....	76
	Cluster Server agent for Oracle のタイプ定義ファイルのインポート .....	77
	Oracle のための VCS サービスグループの設定 .....	79
	Cluster Manager (Java コンソール)を使った Oracle 用 VCS サービス グループの設定 .....	79
	コマンドラインを使った Oracle 用 VCS サービスグループの設定 .....	82
	VCS agent for Oracle の詳細監視の設定 .....	85
	Oracle に対する詳細監視の設定 .....	85
	Netlsnr に対する詳細監視の設定 .....	89

	エージェントに対するインテリジェントなリソース監視の手動による有効化と無効化 .....	90
	AMF カーネルドライバの管理 .....	92
<b>第 5 章</b>	<b>Oracle 用 VCS サービスグループの管理 .....</b>	<b>95</b>
	VCS サービスグループの管理について .....	95
	サービスグループのオンライン化 .....	96
	サービスグループのオフライン化 .....	96
	サービスグループの切り替え .....	96
	サービスグループ設定の修正 .....	97
<b>第 6 章</b>	<b>プラグ可能データベース (PDB) の移行 .....</b>	<b>98</b>
	コンテナデータベース (CDB) 間でのプラグ可能データベース (PDB) の移行 .....	98
<b>第 7 章</b>	<b>Cluster Server agent for Oracle のトラブルシューティング .....</b>	<b>102</b>
	Cluster Server agent for Oracle のトラブルシューティングについて .....	102
	Oracle エージェントと Netlsnr エージェントに共通のエラーメッセージ .....	103
	Oracle エージェント固有のエラーメッセージ .....	105
	Netlsnr エージェント固有のエラーメッセージ .....	109
	ASMinst エージェント固有のエラーメッセージ .....	110
	ASMDG エージェント固有のエラーメッセージ .....	110
	VCS 環境における Oracle 固有の問題のトラブルシューティング .....	111
	Oracle 診断バイナリと Oracle のインスタンスの意図的なオフライン化の検証 .....	111
	VCS Oracle エージェントの意図的なオフライン動作の検証 .....	114
	PDB リソースの IMF の無効化 .....	115
<b>付録 A</b>	<b>リソースタイプの定義 .....</b>	<b>117</b>
	リソースタイプの定義と属性の定義について .....	117
	Oracle エージェントのリソースタイプ定義 .....	117
	Oracle エージェントの属性の定義 .....	118
	ポリシー管理データベースの Sid 属性について .....	127
	Netlsnr エージェントのリソースタイプ定義 .....	128
	Netlsnr エージェントの属性定義 .....	129
	ASMinst エージェントのリソースタイプ定義 .....	133
	ASMinst エージェントの属性定義 .....	133
	ASMDG エージェントのリソースタイプ定義 .....	137
	ASMDG エージェントの属性定義 .....	137

## 付録 B

<b>設定例</b> .....	140
Oracle enterprise agent の設定例について .....	140
Oracle 単一インスタンスの設定例 .....	141
単一 Oracle インスタンスの VCS 設定ファイルの例 .....	141
複数の Oracle インスタンス(単一リスナー)の設定例 .....	144
複数の Oracle インスタンス(単一リスナー)の VCS 設定ファイルの例 .....	146
複数のインスタンス(複数リスナー)の設定例 .....	151
複数の Oracle インスタンス(複数リスナー)の VCS 設定ファイルの例 .....	151
共有サーバーサポートの Oracle の設定例 .....	155
共有サーバーサポートで設定された Oracle インスタンスの VCS 設 定ファイルの例 .....	156
Solaris ゾーンの Oracle インスタンスの設定例 .....	159
単一 Oracle インスタンスにおける、ローカルディスク上のゾーンルー ト .....	159
単一 Oracle インスタンスにおける、共有ディスク上のゾーンルート .....	160
複数 Oracle インスタンスにおける、ローカルディスク上のゾーンルー ト .....	161
複数 Oracle インスタンスにおける、共有ディスク上のゾーンルート .....	164
共有サーバーサポートの Oracle インスタンスにおける、ローカルディ スク上のゾーンルート .....	166
共有サーバーサポートの Oracle インスタンスにおける、共有ディス ク上のゾーンルート .....	167
プロジェクト環境の Oracle インスタンス .....	168
ローカルディスクまたは共有ディスク上のゾーンルートの VCS 設定ファ イルの例 .....	169
Oracle ASM の設定例 .....	170
RAW ディスクとしての ASM ディスクの設定例 .....	170
VxVM ボリュームとしての ASM ディスクの設定例 .....	174
CVM ボリュームとしての ASM ディスクの設定例 .....	178
main.cf の Oracle プラグ可能なデータベース(PDB)リソースの設定例 .....	183
main.cf の移行可能な Oracle プラグ可能データベース(PDB)リソースの 設定例 .....	186
systemD でサポートされる Oracle の設定例 .....	192
systemD でサポートされる ASMInst の設定例 .....	194

付録 C	ベストプラクティス .....	196
	VCS 環境で複数の Oracle インスタンスを設定する場合のベストプラクティ ス .....	196
付録 D	Oracle の VCS cluster での SPFILE の使用 .....	198
	Oracle 初期化パラメータファイルについて .....	198
	Oracle インスタンスの起動 .....	198
	VCS cluster での SPFILE の使用 .....	199
付録 E	単一インスタンスのデータベース環境での OHASD .....	201
	単一インスタンスのデータベース環境での OHASD について .....	201
	OHASD の高可用性を確保するための Application エージェントの設定 .....	201

# Cluster Server agent for Oracle の導入

この章では以下の項目について説明しています。

- [Cluster Server agent for Oracle](#) について
- [Oracle ASM](#) のエージェントについて
- [Oracle](#) の [VCS](#) エージェントのサポート対象ソフトウェア
- エージェントが [Oracle](#) の高可用性を保証するしくみ
- [Oracle](#) 用 [Cluster Server](#) エージェントの機能について
- [VCS cluster](#)における標準的な [Oracle](#) 設定
- [VCS cluster](#)での [Oracle](#) 設定について

## Cluster Server agent for Oracle について

[Cluster Server agent for Oracle](#) は、[Oracle](#) データベースソフトウェアの高可用性を実現します。

[Veritas](#) 高可用性エージェントは次の処理を行います。

- 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の監視。
- これらのリソースの状態判断。
- 外部イベントに従ったリソースの起動または停止。

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の実行ファイルが含まれています。[Oracle](#) のエージェントは、[Oracle](#) とリスナープロセスの監視、オンライン化、オフライン化を行います。

Oracle 用のエージェントpackageには、Oracle の高可用性を実現するために連携して働く次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。

## Oracle ASM のエージェントについて

Oracle 用の Cluster Server エージェントのスイートは、自動ストレージ管理 (ASM) を使う Oracle データベースに高可用性のエージェントを提供します。Oracle ASM のためのエージェントは Oracle ASM のインスタンスと ASM ディスクグループを監視します。

VCS Enterprise Agent パッケージには、Oracle ASM 用の次のエージェントも含まれています。

- ASMInst エージェントは ASM インスタンスを監視します。
- ASMDG エージェントは Oracle ASM ディスクグループを監視します。

## Oracle の VCS エージェントのサポート対象ソフトウェア

Oracle の VCS 8.0 エージェントによってサポートされるソフトウェアバージョンについては、『Veritas InfoScale リリースノート』を参照してください。

次の場所にあるサポート表を参照してください。

[https://www.veritas.com/support/en\\_US/doc/112632971-112632974-1](https://www.veritas.com/support/en_US/doc/112632971-112632974-1)

## エージェントが Oracle の高可用性を保証するしくみ

Cluster Server 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、正常に機能することを確認します。

p.13 の「Oracle 用 Cluster Server エージェントの機能について」を参照してください。

このエージェントには、次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  
このモードにはプロセスチェックと診断監視オプションがあります。デフォルトのプロセスチェックのオプションの場合は、エージェントは Oracle とリスナープロセスがプロセスステータブルに存在することを確認します。プロセスチェックでは、プロセスのハングアップまたは停止状態を検知できません。  
Oracle のエージェントには、Oracle のリソースが意図的にオフラインになったかどうかを検出する機能があります。エージェントは、Oracle データベースの正常なシャットダウンを検出します。管理者が Oracle を正常に停止させたときは、Oracle が停止していてもエージェントはリソース障害をトリガしません。タイプレベルの

**IntentionalOffline** 属性と、リソースレベルの **MonitorOption** 属性の値は **1** に設定し、Oracle エージェントの意図的なオフライン機能を有効にする必要があります。たとえば、意図的なオフライン機能では、インスタンスの異常終了が発生した場合、エージェントは Oracle リソースの障害と判断します。また、shutdown、shutdown immediate、shutdown abort、shutdown transactional などのコマンドを使用して Oracle を正常に停止した場合、エージェントは Oracle リソースをオフラインとして報告します。

- セカンダリまたは詳細監視  
このモードでは、Perl スクリプトを実行し、データベースやリスナーに対してコマンドを実行してその状態を確認します。
- Oracle データベースが READ WRITE モードの場合、エージェントは UPDATE クエリーを実行します。READ ONLY モードの場合はデータベースに対して SELECT クエリーを実行します。

Oracle エージェントは、基本監視のプロセスチェックモードで IMF (Intelligent Monitoring Framework) もサポートします。IMF はインテリジェントなリソース監視を可能にします。Oracle エージェントは IMF 対応であり、非同期監視フレームワーク AMF (Asynchronous Monitoring Framework) カーネルドライバを使って、リソースの状態変化の通知を受信します。

p.11 の「[Oracle エージェントと Netlsnr エージェントがインテリジェントなリソース監視をサポートするしくみ](#)」を参照してください。

Oracle またはリスナープロセスが正しく機能していないことを監視ルーチンが報告した場合には、エージェントはアプリケーションのエラーを検出します。このアプリケーションのエラーが発生したときには、Oracle サービスグループは cluster 内の別の node にフェールオーバーします。このようにして、エージェントは Oracle のサービスとデータベースの高可用性を実現します。

Solaris コンテナ環境で実行されるアプリケーションに VCS が高可用性を提供します。Solaris ゾーンおよび Solaris プロジェクト環境で実行されるこれらのリソースを監視するように、Cluster Server agent for Oracle を設定できます。

p.12 の「[エージェントによる、Solaris ゾーンで実行される Oracle インスタンスの監視](#)」を参照してください。

## Oracle エージェントと Netlsnr エージェントがインテリジェントなリソース監視をサポートするしくみ

VCS では、ポーリングベースの監視に加えて、Intelligent Monitoring Framework (IMF) によるインテリジェントなリソース監視がサポートされます。ポーリングベースの監視がリソースを定期的にポーリングする一方で、インテリジェントな監視は非同期的な監視を実行します。Oracle および Netlsnr エージェントのインテリジェントなリソース監視機能を実効または無効にできます。

p.90 の「エージェントに対するインテリジェントなリソース監視の手動による有効化と無効化」を参照してください。

IMF に対応したエージェントが起動すると、エージェントは非同期監視フレームワーク (AMF) カーネルドライバを初期化します。リソースが安定状態になった後に、エージェントは、リソースを監視するために必要な詳しい情報を AMF カーネルドライバに登録します。たとえば、Oracle エージェントは Oracle プロセスの PID を AMF カーネルドライバに登録します。エージェントの *imf\_getnotification* 関数は、リソースのあらゆる状態変化を待機します。AMF カーネルドライバモジュールがリソースの状態変化について *imf\_getnotification* 関数を通知すると、エージェントフレームワークが監視エージェントの関数を実行し、リソースの状態を確認します。エージェントは状態変化を VCS に通知し、VCS は適切な処理を実行します。

詳しくは、『Cluster Server Administrator’s Guide』を参照してください。

## エージェントによる、Solaris ゾーンで実行される Oracle インスタンスの監視

Solaris 11 は、オペレーティングシステムのサービスを仮想化する方法を提供します。これを使えば、1 つ以上のプロセスをシステム上の他の活動から分離して実行できます。この「サンドボックス」を「非グローバルゾーン」と呼びます。各ゾーンは、機能が豊富でカスタマイズされたサービス一式を提供できます。「グローバルゾーン」で実行されるプロセスには、現在の Solaris システムで使用できるものと同じ権限一式があります。

VCS は、フェールオーバー機能をゾーンにまで拡張することで、非グローバルゾーンで実行されるアプリケーションに対して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されるアプリケーションについて、エージェントはゾーン内のスクリプトエントリポイントを実行します。VCS 制御下で設定されたゾーンに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。

『Cluster Server Administrator’s Guide』を参照してください。

Cluster Server 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る Oracle インスタンスを監視できます。

## エージェントが Oracle ASM インスタンスの高可用性を確保する方法

Oracle には、データファイル、コントロールファイル、ログファイルなど、Oracle データを格納、管理するための ASM 機能が搭載されています。ASM ディスクグループに格納されるこれらの Oracle データファイルは、ASM ファイルと呼ばれます。

ASM で管理されたデータベースでは、データベースインスタンスを起動する前に ASM インスタンスを起動する必要があります。ASM インスタンスは ASM ディスクグループをマウントし、ASM ファイルがデータベースインスタンスから使用できるようにします。

ASM は、クラスタの同期サービスが ASM インスタンスとデータベースインスタンス間の同期を有効にすることを要求します。

Oracle のマニュアルを参照してください。

Cluster Server agent for Oracle には、Oracle ASM インスタンスと ASM ディスクグループの高可用性を確保するために、次のエージェントがあります。

- ASMInst エージェント  
p.29 の「[ASMInst エージェントの機能](#)」を参照してください。
- ASMDG エージェント  
p.32 の「[ASMDG エージェントの機能](#)」を参照してください。

Veritas High Availability Agent for Oracle は継続的に ASM インスタンスと ASM ディスクグループを監視し、正しく機能していることを確認します。

Oracle の Flex ASM 機能は、InfoScale ASMDG agent for Oracle を介してサポートされます。

エージェントは、`sysasm` ロールを使用して ASM インスタンスに接続します。

## エージェントの Oracle コンテナデータベース (CDB) とプラグ可能データベース (PDB) のサポート方法

Oracle の VCS エージェントは、コンテナデータベース (CDB) とプラグ可能データベース (PDB) を管理できます。

CDB または重要な PDB に障害が発生している場合、VCS は特定のサーバーから別のサーバーに設定済みの PDB とともに CDB 全体をフェールオーバーします。CDB の一部である PDB は、別の PDB リソースを使って管理します。

---

メモ: Oracle エージェントは、IMF、診断監視、意図的なオフライン、または PDB をサポートしていません。

---

p.115 の「[PDB リソースの IMF の無効化](#)」を参照してください。

p.98 の「[コンテナデータベース \(CDB\) 間でのプラグ可能データベース \(PDB\) の移行](#)」を参照してください。

## Oracle 用 Cluster Server エージェントの機能について

エージェントが実行する機能はエン트리ポイントと呼ばれます。Oracle 用の Cluster Server エージェントスイートの一部である次のエージェントの機能を見直します。

- Oracle エージェントの機能

- p.14 の「[Oracle エージェント関数](#)」を参照してください。
- Netlsnr エージェントの機能  
 p.28 の「[Netlsnr エージェント関数](#)」を参照してください。
  - ASMInst エージェントの機能  
 p.29 の「[ASMInst エージェントの機能](#)」を参照してください。
  - ASMDG エージェントの機能  
 p.32 の「[ASMDG エージェントの機能](#)」を参照してください。

## Oracle エージェント関数

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 エージェント関数を一覧で示します。

表 1-1 Oracle エージェント関数

エージェントの操作	説明
Online	<p>次の 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 の異なる起動オプションを使ってデータベースを開始するようにエージェントを設定することもできます。</p> <p>p.15 の「<a href="#">Oracle エージェントの起動オプションと停止オプション</a>」を参照してください。</p>
Offline	<p>次の sqlplus コマンドに指定のオプションを加えて Oracle データベースを停止させます。</p> <pre>shutdown immediate</pre> <p>IMMEDIATE は、デフォルトのシャットダウンオプションです。Oracle の異なるシャットダウンオプションを使ってデータベースを停止するようにエージェントを設定することもできます。</p> <p>p.15 の「<a href="#">Oracle エージェントの起動オプションと停止オプション</a>」を参照してください。</p>
Monitor	<p>Oracle プロセスの状態を確認します。Oracle エージェントでは、基本と詳細の 2 段階の監視レベルを設定できます。</p> <p>p.21 の「<a href="#">従来のデータベースおよびコンテナデータベース内の Oracle エージェントに対する監視オプション</a>」を参照してください。</p>

エージェントの操作	説明
oracle_imf_init	Oracle エージェント用の IMF 通知モジュールである AMF カーネルドライバとインターフェースをとるためにエージェントを初期化します。この機能は、エージェントが起動すると実行されます。
oracle_imf_getnotification	リソースの状態の変更についての通知を取得します。この機能は、AMF カーネルモジュールと連動するようにエージェントが初期化された後に実行されます。この機能は連続的に通知を待機し、通知を受けた時点でリソースに対する処理を実行します。
oracle_imf_register	リソースエンティティを AMF カーネルモジュールに登録または登録解除します。この機能は、リソースが安定した状態(オンラインまたはオフライン)に入った後に、リソースごとに実行されます。
Clean	次の 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。  shutdown abort  プロセスが shutdown コマンドに応答しない場合、エージェントは次のことを実行します。  <ul style="list-style-type: none"> <li>■ 設定されたインスタンスと関連付けられているプロセスについて、プロセステーブルをスキャンします。</li> <li>■ 設定されたインスタンスと関連付けられているプロセスを強制終了します。</li> </ul>
Info	データベースの状態について静的または動的な情報を表示します。  <b>メモ:</b> この属性は PDB リソースには適用されません。  p.23 の「 <a href="#">Cluster Server agent for Oracle の Info エントリポイント</a> 」を参照してください。
Action	リソースで事前定義済みの処理を実行します。  p.24 の「 <a href="#">Cluster Server agent for Oracle の Action エントリポイント</a> 」を参照してください。

## Oracle エージェントの起動オプションと停止オプション

すでに設定が行われた Oracle インスタンスに起動オプションや停止オプションを指定できます。

メモ:

CDB または従来型データベースタイプの場合、SRVCTLSTART または SRVCTLSTART\_RO 起動オプションは UseSystemD 属性と併用できません。

PDB データベースタイプの場合、UseSystemD 属性は使用できません。

どちらの場合も、UseSystemD 属性が設定されると、エージェントはリソースを UNKOWN として報告し、対応するメッセージをログに記録します。

表 1-2は、エージェントが従来のデータベースおよびコンテナデータベースに対してサポートする起動オプションを一覧表示します。

表 1-2 従来のデータベースおよびコンテナデータベースの起動オプション

オプション	説明
STARTUP_FORCE (デフォルト)	<p>Pfile が設定されている場合、startup force pfile='location_of_pfile' コマンドを実行します。</p> <p>Pfile が構成されていない場合、エージェントは startup force を実行します。この起動オプションでは、デフォルトの場所からデフォルトのパラメータファイルを取得します。</p>
STARTUP	<p>Pfile が設定されている場合、startup pfile='pfile_name_with_absolute_path' コマンドを実行します。</p> <p>次に例を示します。</p> <pre>startup pfile = '&lt;/dbs/pfilename&gt;'</pre> <p>pfile がまだ設定されていない場合、エージェントは、pfile なしで startup を実行します。</p>
RESTRICTED	データベースを 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のリカバリを実行します。

オプション	説明
CUSTOM	<p>事前定義された SQL スクリプト (<code>start_custom_\$\$SID.sql</code>) を使用して、カスタム起動オプションを実行します。このスクリプトは、<code>/opt/VRTSagents/ha/bin/Oracle</code> ディレクトリに格納し、Oracle Owner OS ユーザーに対するアクセス権を割り当てる必要があります。このスクリプト自体が見つからない場合は、エージェントでエラーメッセージがログ出力されます。</p> <p>カスタムのスクリプトを使って、エージェントで次の処理を実行します。</p> <pre>sqlplus /nolog &lt;&lt;! connect / as sysdba; @start_custom_\$\$SID.sql exit; !</pre>
SRVCTLSTART	<p><code>srvctl</code> 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>RAC クラスタの場合、デフォルトの起動オプションを <code>SRVCTLSTART</code> として手動で設定する必要があります。</p> <p><b>メモ:</b> これは Oracle Grid Infrastructure が設定されるときにだけ適用されます。</p>
SRVCTLSTART_RO	<p><code>srvctl</code> ユーティリティを使って <code>-o 'read only'</code> オプションを指定してデータベースのインスタンスを起動します。</p> <p><b>メモ:</b> これは Oracle Grid Infrastructure が設定されるときにだけ適用されます。</p>

表 1-3は、エージェントが従来のデータベースおよびコンテナデータベースに対してサポートするシャットダウンオプションをリストします。

表 1-3 従来のデータベースおよびコンテナデータベースのシャットダウンオプション

オプション	説明
IMMEDIATE (デフォルト)	<p><code>shutdown immediate</code> を実行して、Oracle インスタンスを停止します。</p>

オプション	説明
TRANSACTIONAL	shutdown transactional コマンドを実行します。このオプションは、このオプションをサポートするバージョンのデータベースでのみ有効となります。
CUSTOM	事前定義された SQL スクリプト (shut_custom_\$\$SID.sql) を使用して、カスタムシャットダウンオプションを実行します。このスクリプトは、/opt/VRTSagents/ha/bin/Oracle ディレクトリに格納し、Oracle Owner OS ユーザーに対するアクセス権を割り当てる必要があります。このスクリプト自体が見つからない場合は、エージェントは、デフォルトのオプションで自身を停止させます。
SRVCTLSTOP	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。  RAC クラスタの場合、デフォルトのオプションを SRVCTLSTOP として手動で設定する必要があります。  <b>メモ:</b> これは Oracle Grid Infrastructure が設定されるときにだけ適用されます。
SRVCTLSTOP_TRANSACT	srvctl ユーティリティを使って -o transactional オプションを指定してデータベースのインスタンスを停止します。  <b>メモ:</b> これは Oracle Grid Infrastructure が設定されるときにだけ適用されます。
SRVCTLSTOP_ABORT	srvctl ユーティリティを使って -o abort オプションを指定してデータベースのインスタンスを停止します。  <b>メモ:</b> これは Oracle Grid Infrastructure が設定されるときにだけ適用されます。
SRVCTLSTOP_IMMEDIATE	srvctl ユーティリティを使って -o immediate オプションを指定してデータベースのインスタンスを停止します。

## PDB の起動オプションと停止オプション

すでに設定が行われたプラグ可能データベース(PDB)リソースに起動オプションや停止オプションを指定できます。

表 1-4 に、PDB でエージェントがサポートする起動オプションを一覧で示します。

表 1-4 PDB の起動オプション

オプション	説明
STARTUP	<code>pdbname</code> が設定されている場合は、 <code>STARTUP PLUGGABLE DATABASE &lt;pdbname&gt;</code> ; コマンドを実行します。
STARTUP_FORCE	<code>pdbname</code> が設定されている場合は、 <code>STARTUP PLUGGABLE DATABASE &lt;pdbname&gt; FORCE</code> ; コマンドを実行します。
RESTRICTED	<code>pdbname</code> が設定されている場合は、 <code>STARTUP PLUGGABLE DATABASE &lt;pdbname&gt; RESTRICT</code> ; コマンドを実行します。
OPEN_RO	<code>pdbname</code> が設定されている場合は、 <code>STARTUP PLUGGABLE DATABASE &lt;pdbname&gt; OPEN READ ONLY</code> ; コマンドを実行します。

オフライン中に、`shutdown immediate` を実行して個々の PDB の停止が試みられません。

p.19 の「[コンテナデータベース\(CDB\)の起動モードに基づくプラグ可能なデータベース\(PDB\)の推奨起動モード](#)」を参照してください。

p.98 の「[コンテナデータベース\(CDB\)間でのプラグ可能データベース\(PDB\)の移行](#)」を参照してください。

## コンテナデータベース(CDB)の起動モードに基づくプラグ可能なデータベース(PDB)の推奨起動モード

CDB で設定された値に基づいて PDB の起動オプションを指定できます。

表 1-5 に、エージェントがサポートする停止オプションを一覧で示します。

表 1-5 CDB 起動オプション

CDB 起動モード	対応する PDB 起動モード
SRVCTLSTART	CDB に起動値 <code>SRVCTLSTART</code> を設定している場合は、PDB に以下のいずれかの値を割り当てることができます。 <ul style="list-style-type: none"> <li>■ STARTUP</li> <li>■ STARTUP_FORCE</li> <li>■ RESTRICTED</li> <li>■ OPEN_RO</li> </ul>

CDB 起動モード	対応する PDB 起動モード
SRVCTLSTART_RO	CDB に SRVCTLSTART_RO 起動値を割り当てている場合、PDB は OPEN_RO である必要があります。
STARTUP	CDB に起動値 STARTUP を設定している場合は、PDB に以下のいずれかの値を割り当てることができます。 <ul style="list-style-type: none"> <li>■ STARTUP</li> <li>■ STARTUP_FORCE</li> <li>■ RESTRICTED</li> <li>■ OPEN_RO</li> </ul>
STARTUP_FORCE	CDB に起動値 STARTUP_FORCE を設定している場合は、PDB に次のいずれかの値を割り当てることができます。 <ul style="list-style-type: none"> <li>■ STARTUP</li> <li>■ STARTUP_FORCE</li> <li>■ RESTRICTED</li> <li>■ OPEN_RO</li> </ul>
RESTRICTED	CDB に起動値 RESTRICTED を割り当てている場合、PDB は RESTRICTED である必要があります。
RECOVERDB	CDB に起動値 RECOVERDB を設定している場合は、PDB に以下のいずれかの値を割り当てることができます。 <ul style="list-style-type: none"> <li>■ STARTUP</li> <li>■ STARTUP_FORCE</li> <li>■ RESTRICTED</li> <li>■ OPEN_RO</li> </ul>
CUSTOM	カスタムスクリプトは、PDB を CUSTOM として起動します。

p.183 の「[main.cf の Oracle プラグ可能なデータベース \(PDB\) リソースの設定例](#)」を参照してください。

p.98 の「[コンテナデータベース \(CDB\) 間でのプラグ可能データベース \(PDB\) の移行](#)」を参照してください。

## 従来のデータベースおよびコンテナデータベース内の Oracle エージェントに対する監視オプション

Oracle エージェントでは、基本と詳細の 2 段階の監視レベルを設定できます。デフォルトでは、エージェントは基本監視を実行します。

基本監視モードでは、次のオプションが用意されています。

- プロセスチェック
- 診断

Oracle リソースの MonitorOption 属性は、エージェントがプロセスチェックモードまたは診断モードで、基本監視を実行する必要があるかどうかを決定します。

表 1-6 には、基本監視オプションについて説明されています。

**表 1-6**                      **基本監視オプション**

オプション	説明
0 (デフォルト)	<p>プロセスチェック</p> <p>エージェントでは、ora_dbw0、ora_smon、ora_pmon、ora_lmon、ora_lgwr の各プロセスについてのプロセステーブルをスキャンして、Oracle が稼働していることを確認します。</p> <p>このモードでは、エージェントはインテリジェントなリソース監視もサポートします。</p> <p>p.11 の「<a href="#">Oracle エージェントと Netlsnr エージェントがインテリジェントなリソース監視をサポートするしくみ</a>」を参照してください。</p>
1	<p>診断</p> <p>エージェントは Oracle からの診断用 API を使って、SGA を監視し、インスタンスに関する情報を取得します。</p> <p>Oracle エージェントの意図的なオフライン機能を使う場合は、診断監視を有効にする必要があります。</p> <p>p.10 の「<a href="#">エージェントが Oracle の高可用性を保証するしくみ</a>」を参照してください。</p> <p>このモードでは、エージェントはインテリジェントなリソース監視をサポートしません。</p> <p>p.26 の「<a href="#">Oracle エージェントによる診断監視のサポートのしくみ</a>」を参照してください。</p>

詳細監視モードでは、エージェントはデータベースのテストテーブルでトランザクションを実行し、Oracle データベースが正常に機能していることを確認します。エージェントはこのテストテーブルを内部の目的で使用します。テストテーブルではその他のトランザクションを実行しないことをお勧めします。

p.22 の「[詳細監視での Oracle エラーコードの操作](#)」を参照してください。

p.85 の「[VCS agent for Oracle の詳細監視の設定](#)」を参照してください。

## プラグ可能なデータベースの監視

監視中に、Oracle エージェントは CDB に接続し、CDB の一部であるすべての PDB の状態を取得します。

CDB と PDB リソースに `IntentionalOffline` が設定されていて、VCS の外部で CDB が閉じている場合、PDB リソースは `FAULTED` でなく、`Offline` として報告します。この機能を有効にするには、PDB リソースの `IntentionalOffline` 属性を 1 に設定します。

---

**メモ:** プラグ可能データベースはコンテナデータベースとともにフェールオーバーされます。VCS を使ってプラグ可能データベースを個別にフェールオーバーすることはできません。

---

## 詳細監視での Oracle エラーコードの操作

Cluster Server 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の一覧が格納されています。

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
```

---

**メモ:** `oraerror.dat` ファイルに 2 つ以上のエラーが格納されている場合、Cluster Server agent for Oracle がエラーを処理する順序は `FAILOVER/NOFAILOVER`、`UNKNOWN`、`IGNORE/WARN` のようになります。

---

表 1-7 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 1-7 Oracle エラーの事前定義済みエージェントのアクション

Action	説明
IGNORE	エラーを無視します。  Oracle のための Veritas エージェントでエラーが発生したときは、エージェントは <code>oraerror.dat</code> のファイルのエラーコードと比較します。ファイルに一致するエラーコードがなければ、エージェントは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 通知について詳しくは、『Cluster Server Administrator's Guide』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。これはデフォルトの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。  <code>oraerror.dat</code> 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。

## Cluster Server agent for Oracle の Info エントリポイント

Info エントリポイントをサポートする Cluster Server agent for Oracle で、データベースの状態に関する静的または動的な情報を表示します。

Info 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 -value resource ResourceInfo [system]¥
[-clus cluster | -localclus]
```

上記のコマンドを正常に実行するには、次のコマンドを実行する必要があります:

- 次のコマンドを使って、リソースの `refreshinfo` スクリプトを実行します。
 

```
# hares -refreshinfo <res-name> -sys <sys-name>
```

このエントリポイントでは、次の静的な情報を取得します。

- バージョン
- インスタンス番号 (InstanceNo)
- インスタンス名 (InstanceName)
- データベース名 (DatabaseName)
- ホスト名 (HostName)
- 開始時間 (StartupTime)
- パラレル (Parallel)
- スレッド (Thread)
- インスタンスの役割 (InstanceRole)

このエントリポイントでは、次の動的な情報を取得します。

- インスタンスの状態 (InstanceStatus)
- ログイン情報 (Logins)
- オープンモード (OpenMode)
- ログモード (LogMode)
- シャットダウンの待機 (ShutdownPending)
- データベースの状態 (DatabaseStatus)
- 共有プールの空き割合 (Shared Pool Percent free)
- バッファ使用率 (Buffer Hits Percent)

SQL 文を `/opt/VRTSagents/ha/bin/Oracle/resinfo.sql` に追加して、属性を加えることができます。次に例を示します。

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

選択したレコードの型は、次のようになります。

```
attribute_type:userkey_name:userkey_value
```

変数 `attribute_type` は、静的または動的のいずれかか、両方の性質を持った値にできます。

## Cluster Server agent for Oracle の Action エントリポイント

Cluster Server agent for Oracle は Action エントリポイントをサポートし、リソース上で事前定義済みの処理を実行できるようにします。

リソースで処理を実行するには、次のコマンドを入力します。

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

エージェントのカスタム処理を追加することもできます。

詳しくは『Cluster Server Agent Developer’s Guide』を参照してください。

p.25 の表 1-9 を参照してください。に、エージェントの事前定義済み仮想ファイアドリル処理の説明を示します。

表 1-8 に、エージェントの事前定義済み処理の説明を示します。

表 1-8 事前定義済みのエージェント処理

アクション	説明
VRTS_GetInstanceName	すでに設定済みのインスタンスからそのインスタンス名を取得します。このオプションは、Oracle リソースと Netlsnr リソースに使うことができます。
VRTS_GetRunningServices	エージェントが監視する処理の一覧を取得します。このオプションは、Oracle リソースと Netlsnr リソースに使うことができます。
DBRestrict	データベースのセッションを変更して、RESTRICTED モードを有効にします。
DBUndoRestrict	データベースのセッションを変更して、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	一時停止していたデータベースを再開させます。
DBTbspBackup	表領域のバックアップを実行します。バックアップを作成する表領域名は、actionargs に含まれています。

メモ: VRTS\_GetRunningServices、DBRestrict、DBResume、DBSuspend、DBTbspBackup および DBUndoRestrict の action エントリポイントは Oracle PDB リソースではサポートされません。

表 1-9 に、インフラチェックを実行し特定のエラーを修正できるようにする、Cluster Server agent for Oracle の仮想ファイアドリル処理の一覧を示します。

表 1-9 事前定義済みの仮想ファイアドリル処理

仮想ファイアドリル処理	説明
getid (Oracle エージェント)	Oracle Owner がnode上に存在することを確認します。

仮想ファイアドリル処理	説明
home.vfd (Oracle エージェント)	以下を確認します。 <ul style="list-style-type: none"> <li>■ ORACLE_HOME がnode上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は、Action エントリポイントは、いずれかの他のリソースがすでに ORACLE_HOME をマウントしているかどうかをチェックします。</li> <li>■ Pfile が提供されており、それがnode上に存在すること。</li> <li>■ \$ORACLE_HOME/dbs/orapw[SID]のパスワードファイルが存在すること。</li> </ul>
owner.vfd (Oracle エージェント)	Oracle Owner 属性の uid と gid を確認します。Owner 属性の uid と gid が、Oracle リソースが現在オンラインであるnode上で同じであるかどうかをチェックします。
pfile.vfd (Oracle エージェント)	ローカルディスクに pfile または spfile が存在するかどうかを調べます。pfile と spfile の両方が存在しない場合、エージェントの機能は終了します。Oracle リソースがクラスター内でオンラインである場合、エージェント機能は、Oracle リソースがオンラインなので spfile が共有ストレージ上に存在する必要があるというメッセージをログに記録します。
tnsadmin.vfd (Netlsnr エージェント)	listener.ora ファイルが存在するかどうかチェックします。listener.ora ファイルが存在しない場合は、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。

## Oracle エージェントによる診断監視のサポートのしくみ

Cluster Server agent for Oracle は、Oracle 診断 API を使って、ノード上の Oracle インスタンスが適切にシャットダウンしたか、中止されたかを判別します。Oracle インスタンスが VCS の制御外で適切にシャットダウンした場合、エージェントはその操作を意図的なオフライン化として認識します。

build\_oraapi.sh スクリプトを使って、Oracle 診断監視に必要なライブラリをリンクできます。Veritas は、build\_oraapi.sh スクリプトを、/opt/VRTSagents/ha/bin/Oracle ディレクトリの VCS agent for Oracle とともに出荷しています。

## Oracle 診断監視に必要な実行可能ファイルの生成

build\_oraapi.sh スクリプトを使う前に注意事項を確認してください。

### build\_oraapi.sh を実行するための注意事項

- Oracle 診断監視を使うクラスタの各ノードで build\_oraapi.sh スクリプトを実行します。
- Oracle 診断監視を使う各ローカルコンテナ (Solaris ゾーン) 内部で build\_oraapi.sh スクリプトを実行します。
- Oracle データベースバージョンをノードでアップグレードするときは、常に build\_oraapi.sh スクリプトを実行して Oracle ライブラリを再リンクする必要があります。
- Oracle エージェントディレクトリで生成した Oracle 診断バイナリは、ORACLE\_HOME ディレクトリと、build\_oraapi.sh スクリプトの実行時に指定した Oracle データベースバージョンに対応します。ORACLE\_HOME は環境変数として設定するか、コマンドラインオプションとして指定できます。Oracle データベースバージョンは build\_oraapi.sh スクリプトによりコマンドラインオプションとして指定できます。
- build\_oraapi.sh スクリプトは Veritas に付属していますが、事前に構築された Oracle 診断バイナリ (oraapi\_32、oraapi\_3211g、oraapi\_64、oraapi\_6411g) は付属していません。
- Oracle エージェントで診断サポートを有効にするには、build\_oraapi.sh スクリプトを実行して、Oracle バージョンに基づき Oracle 診断バイナリを構築する必要があります。

### build\_oraapi.sh スクリプトを使った Oracle エージェントの実行可能ファイルの構築

build\_oraapi.sh スクリプトを使って Oracle エージェントの実行可能ファイルを構築するには

- 1 root ユーザーとしてノードにログインします。
- 2 Oracle ホームディレクトリパスを設定します。

```
# export ORACLE_HOME=/u01/oraHome
```

ここで、/u01/oraHome はファイルを格納する Oracle ホームディレクトリです。
- 3 Oracle エージェントのディレクトリにアクセスします。

```
# cd /opt/VRTSagents/ha/bin/Oracle
```

- 4 Oracle エージェントディレクトリから次のコマンドを実行します。

```
# ./build_oraapi.sh
```

- 5 `ORACLE_HOME` 変数を指定しない場合、`oracle_home` パラメータをコマンドラインオプションとして設定することもできます。

```
#./build_oraapi.sh -oracle_home /u01/oraHome
```

ここで、`/u01/oraHome` は `ORACLE_HOME` の変数値として設定する Oracle ホームディレクトリのパスです。

このスクリプトにより、`/opt/VRTSagents/ha/bin/Oracle` というエージェントディレクトリにバイナリが生成されます。1 つのノード上で生成される Oracle 診断バイナリは、そのノードで利用可能なシステムアーキテクチャ (32 ビットまたは 64 ビット) と Oracle の完全なバージョンに基づきます。

たとえば、64 ビットシステム上の Oracle バージョン 12 の場合、`oraapi_64_121010` というバイナリが作成されます。

p.111 の「Oracle 診断バイナリと Oracle のインスタンスの意図的なオフライン化の検証」を参照してください。

## Netlsnr エージェント関数

リスナーとは、新規のクライアント接続要求を待機し、データベースに対する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切り替え、状態を監視し、オフラインに切り替えます。

Netlsnr エージェントは IMF 対応です。

p.11 の「Oracle エージェントと Netlsnr エージェントがインテリジェントなリソース監視をサポートするしくみ」を参照してください。

表 1-10 に、Netlsnr エージェント関数を一覧で示します。

表 1-10 Netlsnr エージェント関数

エージェントの操作	説明
Online	次のコマンドを使ってリスナーの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のプロセスを停止します。  <code>lsnrctl stop \$LISTENER</code>  パスワードを使ってリスナーが設定されている場合、エージェントは、そのパスワードを使ってそのリスナーを停止します。

エージェントの操作	説明
Monitor	<p>リスナープロセスの状態を確認します。</p> <p>Netlsnr エージェントでは、基本と詳細の 2 段階の監視レベルを設定できます。</p> <ul style="list-style-type: none"> <li>■ 基本監視モードでは、エージェントは tnslnsr プロセスのプロセステーブルをスキャンして、リスナーのプロセスが実行中であることを確認します。</li> <li>■ 詳細監視モードでは、エージェントは lsnrctl status \$LISTENER コマンドを使ってリスナープロセスの状態を確認します。(デフォルト)</li> </ul>
netlsnr_imf_init	<p>Netlsnr エージェント用の IMF 通知モジュールである AMF カーネルドライバとインターフェースをとるためにエージェントを初期化します。この機能は、エージェントが起動すると実行されます。</p>
netlsnr_imf_getnotification	<p>リソースの状態の変更についての通知を取得します。この機能は、AMF カーネルモジュールと連動するようにエージェントが登録された後に実行されます。この機能は連続的に通知を待機し、通知を受けた時点でリソースに対する処理を実行します。</p>
netlsnr_imf_register	<p>リソースエンティティを AMF カーネルモジュールに登録または登録解除します。この機能は、リソースが安定した状態(オンラインまたはオフライン)に入った後に、リソースごとに実行されます。</p>
Clean	<p>tnslnsr \$LISTENER のプロセステーブルをスキャンし、そのテーブルを破棄します。</p>
Action	<p>リソースで事前定義済みの処理を実行します。</p> <p>p.24 の「Cluster Server agent for Oracle の Action エントリポイント」を参照してください。</p>

## ASMIInst エージェントの機能

ASMIInst エージェントは ASM インスタンスのプロセスを監視します。

表 1-11 は ASMIInst エージェント操作を一覧表示します。

表 1-11 ASMIInst エージェントの操作

エージェントの操作	説明
ONLINE	<p>次の SQL のコマンドを使って Oracle ASM インスタンスを起動します。</p> <pre>startup nomount</pre> <p>Online 操作は、どのディスクグループもマウントしないで ASM インスタンスを起動します。</p>
オフライン (Offline)	<p>次の SQL のコマンドを使って Oracle ASM インスタンスを停止します。</p> <pre>shutdown immediate</pre>
監視	<p>ASM インスタンスの状態を確認します。</p> <p>ASMIInst エージェントは、プロセスチェック監視を使用して、ASM インスタンスを監視します。</p> <p>プロセスチェック監視モードでは、エージェントはプロセステーブルで <code>asm_pmon</code>、<code>asm_smon</code>、<code>asm_lgwr</code>、<code>asm_rbal</code>、<code>sm_dbw0</code> の各プロセスをスキャンして、ASM インスタンスが実行中であることを確認します。</p> <p>エージェントは <code>occssd.bin</code> プロセスが実行中かどうかを確認します。エージェントは次の条件でオフラインに戻ります。</p> <ul style="list-style-type: none"> <li>■ プロセスが実行中でない。</li> <li>■ プロセスが再起動された。</li> </ul>
Clean	<p>次の SQL コマンドを使って Oracle ASM インスタンスを強制的に停止させます。</p> <pre>shutdown abort</pre> <p>プロセスが <code>shutdown</code> コマンドに応答しない場合、エージェントは、<code>SIGTERM</code> コマンドまたは <code>SIGKILL</code> コマンドを使ってプロセスを強制終了します。</p>

## ASMIInst エージェントの起動オプション

すでに設定されている Oracle ASM インスタンスに起動オプションを指定できます。以下にエージェントがサポートする起動オプションを一覧で示します。

表 1-12 起動オプション

オプション	説明
STARTUP	ASM インスタンスリソースのデフォルトの起動オプション。sqlplus を使って、startup nomount 起動オプションを指定して ASM インスタンスを起動します。このオプションを使うと、どの ASM ディスクグループもマウントされません。
STARTUP_MOUNT	sqlplus を使って、startup mount 起動オプションを指定して ASM インスタンスを起動します。このオプションを使うと、すべての ASM ディスクグループがマウントされます。
STARTUP_OPEN	sqlplus を使って、startup open 起動オプションを指定して ASM インスタンスを起動します。このオプションを使うと、すべての ASM ディスクグループがマウントされ、ボリュームが有効化されます。
SRVCTLSTART	srvctl を使って、nomount 起動オプションを指定して ASM インスタンスを起動します。このオプションを使うと、どの ASM ディスクグループもマウントされません。
SRVCTLSTART_MOUNT	srvctl を使って、-o mount 起動オプションを指定して ASM インスタンスを起動します。このオプションを使うと、すべてのディスクグループがマウントされます。
SRVCTLSTART_OPEN	srvctl を使って、-o open 起動オプションを指定して ASM インスタンスを起動します。このオプションを使うと、すべての ASM ディスクグループがマウントされ、ボリュームが有効化されます。

メモ: Veritas ASMIInst リソースには、STARTUP オプションまたは SRVCTLSTART オプションを使うことをお勧めします。

## ASMIInst エージェントのシャットダウンオプション

すでに設定が行われた Oracle ASM インスタンスに停止オプションを指定できます。

表 1-13 停止オプション

オプション	説明
SRVCTLSTOP	srvctl ユーティリティを使って ASM インスタンスを停止します。

デフォルトでは、エージェントは `sqlplus` コマンドを使って Oracle ASM インスタンスを停止します。`srvctl` ユーティリティを使って ASM インスタンスを停止するには、オプションを `SRVCTLSTOP` に設定します。

p.133 の「[ASMinst エージェントの属性定義](#)」を参照してください。

## ASMDG エージェントの機能

ASMDG エージェントは、Oracle データベースが使う ASM ディスクグループをマウントし、状態を監視して、ASM ディスクグループのマウント解除を行います。

ASMDG エージェントの `DiskGroups` 属性で、ディスクグループ名を指定しておく必要があります。

表 1-14 に、ASMDG エージェントの操作を一覧で示します。

表 1-14 ASMDG エージェントの操作

エージェントの操作	説明
Online	次の SQL のコマンドを使って、指定された Oracle ASM ディスクグループを ASM インスタンスにマウントします。  <code>alter diskgroup dg_name1, dg_name2 mount</code>
オフライン (Offline)	次の SQL のコマンドを使って、指定された Oracle ASM ディスクグループを ASM インスタンスからマウント解除します。  <code>alter diskgroup dg_name1, dg_name2 dismount</code>  <b>メモ:</b> 次の Oracle メッセージは、マウントされた ASM ディスクグループがない ASM インスタンスが停止されるときに、VCS ログに記録されます。  ORA-15100: invalid or missing diskgroup name

エージェントの操作	説明
Monitor	<p>指定された ASM ディスクグループの状態を確認します。                      ディスクグループは次の状態のいずれかになっています。</p> <ul style="list-style-type: none"> <li>■ マウントされている</li> <li>■ マウント解除されている</li> <li>■ 不明</li> <li>■ 壊れている</li> <li>■ 接続されている</li> </ul> <p>1つのリソースに複数のASM ディスクグループが設定されている場合、ASMDG エージェントは指定されたすべてのASM ディスクグループの状態を考慮してリソースの状態を返します。</p> <p><b>メモ:</b> Flex ASM の場合、ASMDG エージェントはASM インスタンスに設定しているカーディナリティを優先します。特定のノードでASM インスタンスが停止した場合、ASM インスタンスの数がカーディナルの値と同じであればオフラインと報告され、ASM インスタンスの数がカーディナルの値を下回ると障害と報告されます。</p>
Clean	<p>次の SQL コマンドを使って Oracle ASM ディスクグループを強制的にマウント解除します。</p> <pre>alter diskgroup dg_name1, dg_name2 dismount force</pre>

## VCS clusterにおける標準的な Oracle 設定

VCS clusterにおける標準的な 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nodes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。

Oracle ASM の機能を使う場合、次の特徴があります。

- Oracle のバイナリは両方のnodesでローカルにインストールされます。
- データベースファイルは ASM ディスクグループにインストールされます。  
 ASM で管理されたデータベースのための標準的なクラスタの設定例を見直します。  
 p.34 の [図 1-2](#) を参照してください。
- Cluster Server agent for Oracle は両方のnodesにインストールされます。  
 ASM の場合は、ASMinstとASMDGのエージェントは両方のnodes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

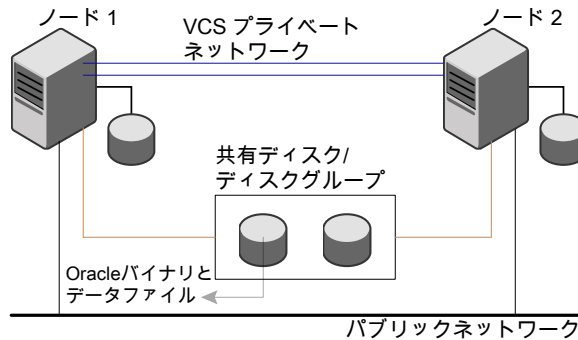
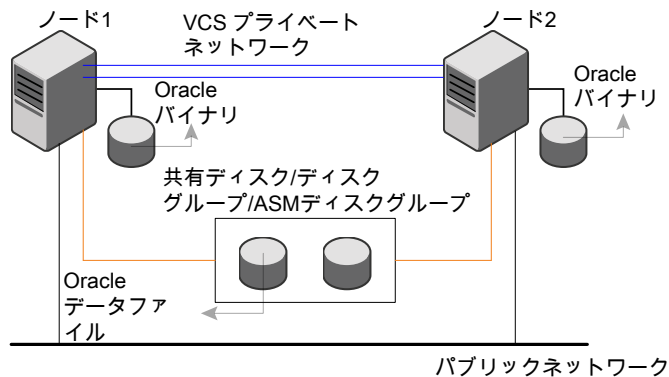


図 1-2 は、Oracle バイナリが各 node でローカルにインストールされ、Oracle データが共有ディスクにインストールされる設定を示しています。

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



## VCS clusterでの Oracle 設定について

VCS 環境での Oracle 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『Cluster Server Installation Guide』を参照してください。
- Oracle のインストールと設定

p.36 の「VCS 環境での Oracle のインストール」を参照してください。

- Cluster Server agent for Oracle のインストール  
p.59 の「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」を参照してください。
- Oracle のための VCS サービスグループの設定  
p.67 の「Oracle 用サービスグループの設定について」を参照してください。

# Oracle のインストールと設定

この章では以下の項目について説明しています。

- VCS 環境での Oracle のインストール
- VCS 環境に Oracle をインストールする前に
- Oracle のインストールのための VCS の必要条件について
- VCS のための Oracle インストールタスクについて
- VCS 環境での ASM バイナリのインストール
- クラスタの最初のノードでの Oracle ASM の設定
- 残りのノードでの ASM の構成と起動
- クラスタの最初のノードへの Oracle バイナリのインストール
- Oracle データベースの設定
- \$ORACLE\_BASE/admin/SID ディレクトリのコピー
- Oracle ASM の初期化パラメータファイルのコピー
- Oracle データベースへのアクセスの確認

## VCS 環境での Oracle のインストール

VCS clusterに Oracle をインストールするための戦略は、cluster内のすべてのnodes上でインストールが均一であることを目標としています。

Solarisの Oracle 社のマニュアルを参照してください。

VCS 環境では、次の方法で Oracle をインストールできます。

共有ディスク上の \$ORACLE_HOME	Oracle バイナリと Oracle データは共有ディスク上にインストールされます。
ローカルディスク上の \$ORACLE_HOME	Oracle バイナリは各 node 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。

---

**メモ:** Oracle データベースストレージ用に ASM を使うには、\$ORACLE\_HOME を cluster 内のすべての nodes のローカルディスクにインストールする必要があります。

---

p.44 の「[VCS のための Oracle インストールタスクについて](#)」を参照してください。

Oracle データにはデータファイル、コントロールファイル、REDO ログファイル、アーカイブログファイルが含まれていること注意してください。

Oracle をインストールするときは、Oracle 所有者の login\_id、id\_name、group\_id、group\_name がすべての nodes 上で同じであることを確認します。ユーザー oracle とグループ dba は、Network Information Service (NIS と 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。

## VCS 環境に Oracle をインストールする前に

次の前提条件を満たしていることを確認してください。

- cluster 内のすべての nodes に VCS がインストールされていることを確認します。
- cluster 内のすべての nodes に Oracle と VCS を実行するのに十分なリソースがあることを確認します。
- 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ます。
- 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ます。

## Oracle のインストールのための VCS の必要条件について

VCS cluster に Oracle をインストールするには、次の必要条件を満たしている必要があります。

カーネルパラメータの設定 Oracle をインストールする各 node が、Oracle の設定に関する次の必要条件を満たしている必要があります。

- ディスクパーティション
- 共有メモリ
- スワップサイズ
- セマフォ
- ファイルハンドル

個々の必要条件について詳しくは、対応するオペレーティングシステムに関する Oracle のマニュアルを参照してください。

`$ORACLE_HOME` の場所 環境にもよりますが、Oracle ホームディレクトリ (`$ORACLE_HOME`) は次のどちらかの方法で配置できます。

- 各サーバーのディスクのローカルに配置する
- 共有ストレージに配置する

Oracle ASM を使う場合は、それぞれの node のローカルディスク上のみ Oracle のホームディレクトリを配置する必要があります。

それぞれの方法の利点を見直して、どちらかに決めます。

p.41 の「`$ORACLE_HOME` の場所」を参照してください。

複数の Oracle インスタンス (SID) の設定

シングル cluster の設定で定義される複数の Oracle インスタンスを設定できます。このような場合、各インスタンスのパラメータファイルはサービスグループの `SystemList` 属性で指定されたすべての nodes でアクセス可能である必要があります。

**メモ:** 同じシステムに Oracle の複数のバージョンをインストールした場合は、SID に重複がないことを確認します。

Oracle データベース表領域の場所

表領域を通常の (UFS または VxFS) ファイル上に作成する場合、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 node 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。

Oracle の表領域に共有ディスク上の RAW デバイスを使う場合は、次の必要条件を満たしている必要があります。

- 所有者が Oracle dba ユーザーであること。
- Oracle のデータを格納する RAW デバイスでの権限またはアクセスモードが 660 であること。

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 と 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。

異常終了した Oracle プロセスの  
 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 エージェントスクリプトまたは Oracle バイナリを実行する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 バイナリに、`/opt/VRTSagents/ha/bin/Oracle` への書き込み権限はありません。したがって、プロセスが異常終了したときに Oracle バイナリによって生成された「コア」ファイルが失われます。Veritas では、Solaris `coreadm (1M)` コマンドを使用して、このようなコアファイルの名前と場所を指定することをお勧めします。

リスナーの透過的フェールオーバー

`node` の切り替え後でも再設定することなく、Oracle サーバーのクライアントを再接続可能にできます。そのような再接続のためには、Oracle リソースのサービスグループに少なくとも 1 つの IP リソースを含める必要があります。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 プロトコルを使う場合は、ファイル `/etc/services` に Oracle Net Service のサービス名が記載されていることを確認します。サービスグループの `SystemList` 属性に定義されている各 `node` で、このファイルを確認する必要があります。

VCS 環境のリスナー認証

Netlsnr エージェントは、リスナープロセスのためのパスワード認証と同様、OS の認証をサポートします。Veritas では、OS 認証を設定することをお勧めします。リスナーのパスワードを設定する場合は、パスワードが正しく設定されたかを確認してください。パスワードが不適切に設定された場合には、リスナーにエラーが生じる場合があります。

p.84 の「Oracle データベースのユーザーパスワードとリスナーパスワードの暗号化」を参照してください。

リスナー認証の設定について詳しくは、Oracle のマニュアルを参照してください。

`$ORACLE_HOME` の長いパス名の制限

Solaris プロセステーブルでは、プロセスのパス名は 79 文字までに制限されます。

`$ORACLE_HOME` では、プロセスの完全パス名は 80 文字以上になる可能性があります。この場合、`$ORACLE_HOME` ディレクトリへのソフトリンクを作成できます。その後、`main.cf` ファイルの `Home` 属性でこのソフトリンクを長いファイル名の代わりに使用できます。

p.42 の「エージェント属性内の `$ORACLE_HOME` の長いパス名の置き換え」を参照してください。

Oracle NLS 情報

NLS 情報は、次のいずれかの方法で定義できます。

- Oracle パラメータファイルに適切なパラメータを定義する。
- エージェントの `EnvFile` 属性に適切な環境変数を定義する。  
 p.117 の「リソースタイプの定義と属性の定義について」を参照してください。

Oracle パラメータファイルにパラメータを定義すると、Oracle サーバーの NLS 設定が影響を受けます。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 入力と出力が影響を受けます。

VCS 環境での Oracle データベースのホットバックアップ

VCS 環境では、Oracle データベースのホットバックアップはデフォルトで有効に設定されています。

Oracle データベースのホットバックアップを実行しているときに、**node** で障害が発生することがあります。障害の発生時に、VCS が別の **node** へのフェールオーバーを実行できるのは、次の必要条件が満たされている場合だけです。

- **AutoEndBkup** の属性値はデフォルトの 1 に設定されている必要があります。
- Oracle エージェントの起動オプションは、**STARTUP**、**STARTUP\_FORCE**、または **CUSTOM** でなければなりません。  
 p.15 の「[Oracle エージェントの起動オプションと停止オプション](#)」を参照してください。

これらの VCS の必要条件を満たしていない場合は、手動でホットバックアップを終了してから、別の **node** に Oracle をフェールオーバーする必要があります。

**メモ:** コンテナデータベースまたはプラグ可能データベースのホットバックアップ中にノードに障害が発生した場合は、対応する CDB リソースの **AutoEndBkup** 属性を 1 に設定する必要があります。CDB の **AutoEndBkup** が 1 に設定されている場合、オンライン時に CDB と PDB の両方のバックアップも終了します。

p.43 の「[ホットバックアップ中に VCS node に障害が発生した後の Oracle のフェールオーバー](#)」を参照してください。

**メモ:** **AutoEndBkup** 属性値を 0 に設定した場合は、予期しない VCS の動作を回避するために詳細監視を有効にする必要があります。

p.85 の「[VCS agent for Oracle の詳細監視の設定](#)」を参照してください。

VCS で Oracle ASM を設定するためのストレージデバイス

Oracle ASM では次のストレージデバイスの 1 つを選択できます。

- **RAW** ディスクとして ASM のディスク  
 RAW ディスクを使う場合は、ディスクに **ASM \$ORACLE\_HOME** に指定された永続的な権限があることを確認します。
- **Veritas Volume Manager** ボリュームとしての ASM のディスク  
 VxVM ボリュームを使う場合は、ディスクに再ブートを実行可能な永続的な権限があることを確認します。権限は **ASM \$ORACLE\_HOME** のそれと同じでなければなりません。  
**Veritas Volume Manager** のマニュアルを参照してください。
- **Cluster Volume Manager** ボリュームとして ASM のディスク  
 CVM ボリュームを使う場合は、ディスクに再ブートを実行可能な永続的な権限があることを確認します。権限は **ASM \$ORACLE\_HOME** のそれと同じでなければなりません。  
**Veritas Volume Manager** のマニュアルを参照してください。

VxVM または CVM ボリュームを使う ASM のディスクに対してミラー化を設定する場合は、VxVM のミラー化は設定する必要がありますが、ASM のミラー化は設定しません。

p.170 の「[Oracle ASM の設定例](#)」を参照してください。

**ASMinst** エージェントは ASM ディスクグループの ASM インスタンスの **pfile** または **spfile** をサポートしません。**Veritas** では、このファイルを ASM ディスクグループからローカルファイルシステムにコピーすることを推奨します。

Solaris ゾーン環境の VxVM または CVM ボリュームに設定された ASM インスタンスに対して次の操作を行う必要があります。

- グローバルゾーンで新しい Oracle ユーザーを作成します。  
新しい Oracle ユーザーのユーザー名、グループ名、ユーザー ID、グループ ID は、ASM を設定したローカルゾーンで使われているものと同じである必要があります。
- RAW ボリュームで I/O を実行できるようにするには、グローバルゾーンから権限を次のように設定します。
  - 所有者が Oracle dba ユーザーであること。
  - Oracle のデータを格納する RAW デバイスでの権限またはアクセスモードが 660 であること。

次に例を示します。

```
# 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 と 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。

- ローカルゾーン内の VxVM ボリュームに該当するデバイスの権限を変更します。次に例を示します。

```
localzone# chown oracle:dba  
/dev/vx/rdisk/ASMDG/asmvol  
localzone# chmod 660  
/dev/vx/rdisk/ASMDG/asmvol
```

## \$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) は共有ディスクにインストールできます。cluster 内の各 node には、共有ファイルシステムに対して同じマウントポイントディレクトリがある必要があります。Oracle バイナリを共有ストレージにインストールすると、インスタンスを実行するための cluster 内の任意の node のセットアップが簡略化されます。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、cluster 内でストレージを共有している新しい node に移動することも可能です。

たとえば、4 つの nodes を持つ cluster 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 nodes 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

Oracle データベースサーバー (\$ORACLE\_HOME) はローカルディスクにインストールできます。利点は、データベースサーバーが別の node で実行中に、オフラインの node 上の Oracle データベースのバイナリをアップグレードできることです。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 node 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 nodes を使った場合、さまざまな Oracle のインストール環境の保守が困難になることです。

Oracle ASM を使う場合は、それぞれの node のローカルディスク上のみ Oracle のホームディレクトリを配置する必要があります。Oracle データベースと同じ Oracle のホームに Oracle ASM をインストールできます。または、別の Oracle のホームディレクトリに Oracle ASM をインストールすることもできます。

Oracle のマニュアルを参照してください。

## エージェント属性内の \$ORACLE\_HOME の長いパス名の置き換え

cluster 内の各 nodes で \$ORACLE\_HOME のソフトリンクを作成してから、エージェント属性の値を変更する必要があります。

長いパス名を置き換えるには

- 1 \$ORACLE\_HOME の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 /etc/VRTSvcs/conf/config/main.cf ファイルで、Oracle と Netlsnr のリソースタイプ  
の Home 属性と Pfile 属性を変更します。

ASM については、ASMinst と ASMDG のリソースタイプのこれらの属性も変更しま  
す。

p.140 の「[Oracle enterprise agent の設定例について](#)」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"
```

## ホットバックアップ中に VCS node に障害が発生した後の Oracle のフェー ルオーバー

ホットバックアップ中に node に障害が発生したときに、VCS が node のフェールオーバー  
を実行できるのは、Oracle のホットバックアップに関する VCS の必要条件が満たされて  
いる場合だけです。

p.37 の「[Oracle のインストールのための VCS の必要条件について](#)」を参照してくださ  
い。

VCS の必要条件が満たされていない場合、エージェントはフェールオーバー node にあ  
るデータベースをバックアップモードで開くことができません。したがって、VCS はフェー  
ルオーバー node の Oracle グループをオンラインにできません。この状況では、次の  
Oracle エラーが発生する可能性があります。

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

または

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

これらのエラーが発生した場合は、手動でホットバックアップを終了してから、別の node  
に Oracle をフェールオーバーする必要があります。

ホットバックアップ中に **VCS node** の障害が発生した後に **Oracle** を手動でフェールオーバーするには

- 1 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。
- 2 データベースを停止します。
- 3 フェールオーバーノードの **Oracle** サービスグループをオンラインにします。  
これで、エージェントは **Oracle** インスタンスを起動し、データベースをマウントし、データファイルを開くことができます。

## VCS のための Oracle インストールタスクについて

VCS cluster 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。

p.45 の「[共有ディスクで行う \\$ORACLE\\_HOME のインストールタスク](#)」を参照してください。

p.46 の「[ローカルディスクで行う \\$ORACLE\\_HOME のインストールタスク](#)」を参照してください。

ASM では、**\$ORACLE\_HOME** をローカルディスクにインストールし、**Oracle ASM** を設定する必要があります。

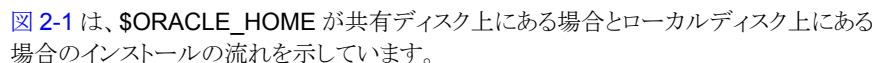
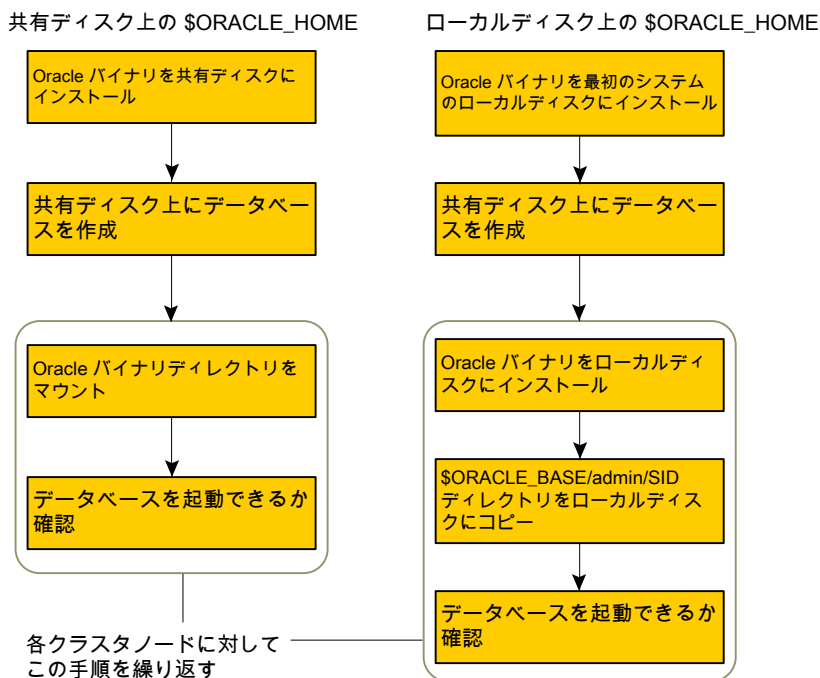
 **図 2-1** は、**\$ORACLE\_HOME** が共有ディスク上にある場合とローカルディスク上にある場合のインストールの流れを示しています。

図 2-1 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

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 nodes 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 2-1には、\$ORACLE\_HOME が共有ディスク上にあるようにするために Oracle をインストールするタスクが一覧表示されます。

表 2-1 共有ディスクで行う \$ORACLE\_HOME のインストールタスク

タスク	参照先
cluster 内の任意の node から、共有ディスクに Oracle バイナリをインストールします。	p.54 の「 <a href="#">クラスタの最初のノードへの Oracle バイナリのインストール</a> 」を参照してください。
Oracle クラスタデーモンを無効にします。	p.54 の「 <a href="#">Oracle クラスタデーモンを無効にします</a> 」を参照してください。
Oracle をインストールした node から、共有ディスクにデータベースを作成します。	p.55 の「 <a href="#">Oracle データベースの設定</a> 」を参照してください。

タスク	参照先
Oracle cluster の一部になる各 node 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。	p.56 の「Oracle データベースへのアクセスの確認」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

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 nodes 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。この方法を使うと、すべての Oracle のインストールが同一となり、データベースへのアクセスが共有ディスク上の同じ場所から行われます。

表 2-2 には、\$ORACLE\_HOME がローカルディスク上にあるようにするために Oracle をインストールするタスクが一覧表示されます。

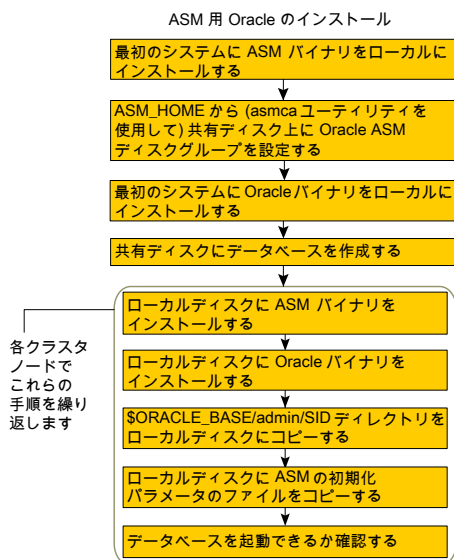
表 2-2 ローカルディスクで行う \$ORACLE\_HOME のインストールタスク

タスク	参照先
cluster 内の最初の node 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。	p.54 の「クラスタの最初のノードへの 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 node から、共有ディスクにデータベースを作成します。	p.55 の「Oracle データベースの設定」を参照してください。
cluster 内の残りのノードで、次のタスクを実行します。 <ul style="list-style-type: none"> <li>■ ローカルディスクに Oracle バイナリをインストールする。</li> <li>■ \$ORACLE_BASE/admin/SID ディレクトリを最初のノードから残りのノードにコピーする。ローカルディスクに \$ORACLE_HOME/dbs/*SID* をコピーする。</li> <li>■ データベースを起動できるか確認する。</li> </ul>	<ul style="list-style-type: none"> <li>■ p.54 の「クラスタの最初のノードへの Oracle バイナリのインストール」を参照してください。</li> <li>■ p.56 の「\$ORACLE_BASE/admin/SID ディレクトリのコピー」を参照してください。</li> <li>■ p.56 の「Oracle データベースへのアクセスの確認」を参照してください。</li> </ul>

## ASM を使用する Oracle のインストールタスク

図 2-2 では、Oracle データベースに ASM を使用する際のインストールフローを説明します。

図 2-2 Oracle ASM のインストールの流れ



次のように実行します。

- Oracle バイナリと ASM バイナリをローカルディスクにインストールし、ASM を設定し、最初のインストールの間に ASM のディスクグループ上に Oracle データベースを作成します。
- 次に、他の nodes のローカルディスクに Oracle バイナリと ASM バイナリをインストールします。

この方法を使うと、すべての Oracle のインストールが同一となり、データベースへのアクセスが同じ場所から行えます。

次の表に、\$ORACLE\_HOME がローカルディスク上に配置されるように Oracle と ASM をインストールするためのタスクの一覧を示します。

表 2-3 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	参照先
cluster 内の最初の node で、ローカルディスクに ASM バイナリをインストールします。	p.48 の「VCS 環境での ASM バイナリのインストール」を参照してください。
ASM をインストールした最初の node から、(asmca ユーティリティを使って) 共有ディスク上に Oracle ASM ディスクグループを設定します。	p.51 の「クラスタの最初のノードでの Oracle ASM の設定」を参照してください。

タスク	参照先
cluster 内の最初の node 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。  Oracle クラスターデーモンを有効にします。	p.54 の「 <a href="#">クラスターの最初のノードへの Oracle バイナリのインストール</a> 」を参照してください。  p.52 の「 <a href="#">ASM で管理されたデータベースのためのクラスターデーモンの有効化</a> 」を参照してください。
Oracle をインストールした最初の node から、ASM ディスクグループ上にデータベースを作成します。	p.55 の「 <a href="#">Oracle データベースの設定</a> 」を参照してください。
cluster 内の各 node で、次のタスクを実行します。 <ul style="list-style-type: none"><li>ローカルディスクに ASM バイナリをインストールする。</li><li>ローカルディスクに Oracle バイナリをインストールする。</li><li>\$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。</li><li>ASM の初期化パラメータのファイルをローカルディスクにコピーする。</li><li>データベースを起動できるか確認する。</li></ul>	p.54 の「 <a href="#">クラスターの最初のノードへの Oracle バイナリのインストール</a> 」を参照してください。  p.56 の「 <a href="#">\$ORACLE_BASE/admin/SID ディレクトリのコピー</a> 」を参照してください。  p.56 の「 <a href="#">Oracle ASM の初期化パラメータファイルのコピー</a> 」を参照してください。  p.56 の「 <a href="#">Oracle データベースへのアクセスの確認</a> 」を参照してください。

## VCS 環境での ASM バイナリのインストール

高可用性のために Oracle ASM インスタンスを設定する場合、次の手順を実行します。

---

メモ: Oracle ASM 機能を使う場合には、クラスター同期サービスを無効にしないでください。

---

### VCS 用に Oracle の ASM バイナリをインストールするには

- Oracle ユーザーとして、Oracle インストーラを実行します。
- [インストールオプションの選択 (Select Installation Option)]ダイアログボックスで、[Grid Infrastructure ソフトウェアのみのインストール (Install Grid Infrastructure Software Only)]オプションを選択して[次へ (Next)]をクリックします。
- 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
- [インストール場所の指定 (Specify Installation Location)]ダイアログボックスで、Oracle ベースとソフトウェアの場所の両方について、ローカルディスク上の場所を指定します。

- 5 ウィザードの指示に従い、他のオプションを選択します
- 6 [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。変更しない場合は、[完了 (Finish)]をクリックします。
- 7 インストーラから、インスタンス化するソフトウェアがスタンドアロンバージョンかクラスタバージョンかを尋ねるメッセージが表示されます。インストールするバージョンを選択します。
  - スタンドアロンバージョンをインスタンス化 (Instantiate a standalone version)

---

**メモ:** このオプションを選択する場合は、`roothas.pl` を実行する必要があります。このオプションでは、**Oracle Restart** という名前の設定が作成されます。これは、**RAC** クラスタではなく、スタンドアロンサーバーで使われる設定です。したがって、**VCS** クラスタ内にある、**Oracle Grid Infrastructure** と **Oracle ASM** のフェールオーバー先として使われるすべてのノードでこのインストールを実行する必要があります。

このオプションでは、各ノードで **Oracle Grid Infrastructure** ソフトウェアをローカルにインストールする必要があります。複数のノードで同じバイナリのセットを使用することはできません。これは、ホスト名が異なるノードで同じ **CSSD** デーモン (特定の設定で使用) を使うことができないためです。フェールオーバーのたびに、または別のノードでこのデーモンが使用されるたびに、デーモンの再設定が必要です。

---

**Oracle Restart** 設定では、新しく作成されたデータベースインスタンスと **netlshr** リソースが **Oracle Grid Infrastructure** に自動的に登録されます。**VCS** では、これらのリソースはフェールオーバーサービスグループの一部として設定されます。

データベースと **netlshr** リソースが誤ってオフラインになった場合、**Oracle Grid Infrastructure** はリソースの再起動を試みます。ただし、**VCS** では、フェールオーバーサービスグループ内の別のノード上でこれらのリソースのオンライン化が試みられます。この動作により、同時性違反が発生します。

同時性違反を回避するには、次のいずれかを実行します。

- **Oracle Grid Infrastructure** で、各データベースと **netlistner** リソースの `auto_start` 属性を `0` に設定します。
- **VCS** で、各データベースと **netlistner** リソースの `RestartLimit` 属性の値を、**Oracle Grid Infrastructure** の `RESTART_ATTEMPTS` 属性でこれらのリソースに対応する値に設定します。**Oracle Restart** 機能のサポートに関する **VCS** 設定について詳しくは、p.50 の「**Oracle Restart 機能をサポートするための VCS 設定**」を参照してください。を参照してください。

- スタンドアロンバージョンをインスタンス化しない (Don't instantiate a standalone version)  
このオプションは、ソフトウェアのクラスタバージョンをインスタンス化するために選択します。これは RAC データベースに必須です。
- 8 インストールが完了したら、[次へ (Next)]をクリックします。
  - 9 [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。
  - 10 次の手順では、asmca ユーティリティを使って共有ディスク上の ASM ディスクグループを設定します。  
  
p.51 の「クラスタの最初のノードでの Oracle ASM の設定」を参照してください。

## Oracle Restart 機能をサポートするための VCS 設定

VCS で Oracle Restart 機能がサポートされるようにするには、Oracle グリッドインフラストラクチャが同じノード上でリソースの再起動を試みた時にフェールオーバー処理を試行しないように VCS パラメータを調整する必要があります。VCS で適切な判断が下されるようにするには、ToleranceLimit リソース属性を適切に調整する必要があります。

Oracle Restart 機能では、リソースを利用できない場合、CHECK\_INTERVAL 属性と RESTART\_ATTEMPTS 属性を使って、監視間隔と再起動の試行を判断します。

たとえば、ASM インスタンスが 1 つのノードで利用できなくなると、グリッドインフラストラクチャは ASM リソースの再起動を試みます。しかし、VCS はそのアプリケーションが利用不能であることを検出し、リソースエラーを報告し、そのサービスグループを別のクラスタノードにフェールオーバーしようとしています。このシナリオを回避するには、Oracle がその処理を完了するまで VCS ポリシーが有効にならないようにする必要があります。

VCS リソースの ToleranceLimit 属性を修正して、ToleranceLimit \* MonitorInterval が (CHECK\_INTERVAL \* RESTART\_ATTEMPTS) + MonitorInterval よりも大きくなるようにします。

Oracle の ASM リソース用の属性値が CHECK\_INTERVAL = 1 (秒) と RESTART\_ATTEMPTS = 5 (再起動試行数) で、ASMInst リソース用の VCS 設定値で MonitorInterval が 60 に設定されている場合は、次の計算に基づいて、ToleranceLimit を 2 に設定します。

$$2 \text{ (ToleranceLimit)} * 60 \text{ (MonitorInterval)} = 120$$
$$1 \text{ (CHECK\_INTERVAL)} * 5 \text{ (RESTART\_ATTEMPTS)} + 60 \text{ (MonitorInterval)} = 65$$

次のコマンドは、属性 ToleranceLimit を設定するのに使うことができます。

- # hares -override <リソース名> ToleranceLimit
- # hares -modify <リソース名> ToleranceLimit 2

---

メモ: Oracle Restart 機能と VCS の両方を使って設定されている、ASM、ASM Diskgroup、Oracle Database、Oracle netlistner などの Oracle アプリケーション用に ToleranceLimit 値を調整します。

---

## クラスタの最初のノードでの Oracle ASM の設定

Oracle の ASM 機能を使用する場合は、ASM Configuration Assistant を使用して Oracle ASM を設定します。

ASM の設定は、最初に Oracle をインストールした node 上で、一度だけ行います。

**Oracle データベース用に ASM を設定するには**

---

メモ: Oracle ASM Configuration Assistant を使って、VCS クラスタ内の 1 つのサーバーのみでディスクグループを作成します。これらのディスクグループは、VCS クラスタがそのディスクグループのデータベースをホストしているため、同じクラスタ内の残りのサーバーでも利用できます。

---

- 1 **DISPLAY** 変数を設定して Oracle ユーザーで Oracle ASM Configuration Assistant を起動します。

```
# asmca
```

- 2 [ASM インスタンスの作成 (Create ASM Instance)]ダイアログボックスで、ASM 管理者のパスワードを指定し、[ディスクグループを指定 (Specify Disk Group)]ボタンをクリックします。
- 3 [ディスクグループの作成 (Create Disk Group)]ダイアログボックスで、新しいディスクグループを作成するか、既存のディスクグループにディスクを追加することを選択します。
- 4 [ディスク検出パスの変更 (Change Disk Discovery Path)]ダイアログボックスで、パス名を入力します。

ASM で使用するディスクデバイスに応じて、ディスクのディスクカバリパスを入力します。

- 5 [ディスクグループの作成 (Create Disk Group)]ダイアログボックスで、次の操作を行います。
  - 冗長性オプションの 1 つを選択します。
  - [メンバーディスクの選択 (Select Member Disks)]領域で、[候補の表示 (Show Candidates)]のオプションを選択します。
  - [ディスクパス (Disk Path)]に対してリストからディスクを選択し、ASM ディスクグループを作成します。

- [OK]をクリックします。
- 6 ウィザードの指示に従って Oracle ASM の設定を完了します。

---

**メモ:** Oracle ASM Configuration Assistant が正常に実行されたら、`$GRID_HOME/bin/crsctl stat res -t -init` コマンドを実行して、新しい Oracle リソース `ora.<DGname>.dg` が作成されたことを確認します。

---

- 7 ASM インスタンスの PFILE または SPFILE を、ASM ディスクグループからローカルファイルシステムにコピーします。
- ASMInst エージェント用の VCS エージェントは、ASM ディスクグループ上の ASM インスタンスの PFILE または SPFILE をサポートしません。

## ASM で管理されたデータベースのためのクラスタデーモンの有効化

データベースが ASM で管理されている場合、Oracle クラスタデーモンが cluster の各 node で有効になっていることを確認する必要があります。CSSD が有効になっていない場合は、次の手順を実行します。

**ASM で管理されたデータベースのためのクラスタデーモンを有効化するには**

- 1 スーパーユーザーとしてログインします。
- 2 次のコマンドを実行して、クラスタデーモンの設定と起動を行います。

```
# $GRID_HOME/bin/crsctl start has
```
- 3 node で実行されている Oracle CSSD プロセスは 1 つだけであることを確認します。

## ASM の自動起動の無効化

`ocssd.bin` プロセスが起動すると、ASM インスタンスが自動的に起動します。これは、ASMInst リソースがフェールオーバーサービスグループの一部として設定されている場合に、同時性違反の問題を引き起こす可能性があります。たとえば、あるノードが何らかの理由で停止または再起動し、このノード上のすべてのリソースが 2 番目のノードでオンラインになるとします。再起動後、`ohasd` プロセスは `ocssd.bin` プロセスを自動的に起動します。その結果、最初のノードで ASM インスタンスが自動的に起動し、これが同時性違反につながります。この問題を解決するため、Veritas は ASM の自動起動を無効にすることをお勧めします。

クラスタのノードごとに、次のコマンドを実行します。

```
# $GRID_HOME/bin/srvctl disable asm
```

## 残りのノードでの ASM の構成と起動

この手順は、Oracle Grid Infrastructure がすべてのノードにインストールされ、ASM が最初のノードで設定されているときに適用できます。p.51 の「[クラスタの最初のノードでの Oracle ASM の設定](#)」を参照してください。

次の手順に従って、残りのノードを設定します。

### ASM を作成し、残りのノードで ASM を起動するには

- 1 リモートノードで使う ASM ディスクから、ASM の spfile をコピーします。

次に例を示します。

```
ASMCMD> spget +DATA/asm/asmparameterfile/<registry_file>
ASMCMD> spcopy +DATA/asm/asmparameterfile/<registry_file>
/u01/oraHome/dbs/spfileASM.ora
```

- 2 最初のノードのデータベースと ASM ディスクグループを停止します。
- 3 spfile を最初のノードからリモートノードにコピーします。
- 4 \$ORACLE\_BASE/admin/SID\* を最初のノードからリモートノードにコピーします。
- 5 リモートノードの asm インスタンスを追加します。

```
例: $/u01/product/oracleVersion/grid/bin/srvctl add asm -p
/u01/oraHome/dbs/spfileASM.ora
```

- 6 srvctl コマンドを使って asm インスタンスを起動します。
- 7 リモートノードの VxVM ディスクグループをインポートします。

### ASM ディスクグループを使う残りのノードでデータベースを設定するには

- 1 srvctl add database コマンドを実行して、ノード上で実行中の Oracle データベースを登録します。

---

**メモ:** Oracle ソフトウェアの所有者のクレデンシャルを使ってデータベースを登録します。パラメータの完全なリストについては、Oracle のマニュアルを参照してください。

---

- 2 リモートノードで実行中の Oracle ASM インスタンスにログインします。
- 3 SQL> ALTER disk group <DGname> mount コマンドを実行して Oracle ASM ディスクグループをマウントします。
- 4 手順 3 を繰り返して、必要なディスクグループすべてをマウントします。

- 5 \$GRID\_HOME/bin/crsctl stat res -t -init コマンドを実行して、ディスクグループが OHASD に自動登録されるかを確認します。  
出力には、登録されたディスクグループの `ora.<DGname>.dg` が表示されます。
- 6 \$GRID\_HOME/bin/srvctl modify database -a <diskgroup\_list> コマンドを実行して、データベースへの依存関係として Oracle ASM ディスクグループを追加します。

## クラスタの最初のノードへの Oracle バイナリのインストール

Cluster Server 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。

---

**メモ:** Oracle ASM 機能を使う場合には、クラスタ同期サービスを無効にしないでください。

---

Oracle バイナリをインストールするには、Oracle のマニュアルに記載されている手順に従ってください。

Oracle クラスタデーモンを無効にします。

p.54 の「[Oracle クラスタデーモンを無効にします](#)」を参照してください。

ASM を使う場合は、この手順をスキップし、Oracle ASM の設定に進みます。

p.51 の「[クラスタの最初のノードでの Oracle ASM の設定](#)」を参照してください。

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。

p.55 の「[Oracle データベースの設定](#)」を参照してください。

### Oracle クラスタデーモンを無効にします

Oracle バイナリを共有ディスクにインストールした場合、Oracle クラスタリングデーモンを無効にする必要があります。

---

**メモ:** ASM 機能を使う場合には、Oracle クラスタデーモンを無効にしないでください。

---

Oracle には、Oracle Cluster Synchronization Service Daemon (CSSD) と呼ばれるクラスタデーモンが搭載されます。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 cluster ではこのデーモンを必要としないため、Veritas はデーモンを無効にすることを推奨します。

### Solaris 11 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 **node** で次の処理を行います。
  - **cssd** デーモンの障害管理リソース識別子 (**FMRI**) を確認します。コマンドプロンプトで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** がデーモンを制御しないように、サービス設定リポジトリを変更します。コマンドプロンプトで次のように入力します。

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

**FMRI** は、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 **node** 上で、一度だけ行います。

Oracle データベースのストレージ用に **ASM** を設定した場合は、**ASM** ディスクグループ上で Oracle データベースを設定します。

### Oracle データベースを設定するには

- 1 **DISPLAY** 変数を設定し、Oracle ユーザーとして Oracle Database Configuration Assistant を起動します。

```
# dbca
```
- 2 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
- 3 [ストレージオプション (Storage Options)]ダイアログボックスで、[ファイルシステム (File System)]または[RAW デバイス (Raw Devices)]オプションを選択し、[次へ (Next)]をクリックします。

ASM を設定する場合は、[ASM (Automatic Storage Management)]を選択し、[次へ (Next)]をクリックします。
- 4 ASM を設定する場合は、[ASM ディスクグループ (ASM Disk Groups)]ダイアログボックスで、Oracle データベースのストレージ用の ASM ディスクグループを選択し、[次へ (Next)]をクリックします。

- 5 [データベースファイルの場所 (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。  
ASM を設定したら、[Oracle Managed Files の使用 (Use Oracle-Managed Files)] を選択し、データベース領域を指定し、続いて [次へ (Next)] をクリックします。
- 6 [リカバリ設定 (Recovery Configuration)] ダイアログボックスで、共有ディスク上の [フラッシュリカバリ領域 (Flash Recovery Area)] を指定し、[次へ (Next)] をクリックします。  
ASM を設定したら、[フラッシュリカバリ領域 (Flash Recovery Area)] の適切な場所を指定し、[次へ (Next)] をクリックします。
- 7 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、cluster 内の node 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**

- ◆ データベースインスタンスを最初に設定した node の \$ORACLE\_BASE/admin/SID を、cluster 内にある別の nodes のローカルディスク (\$ORACLE\_BASE/admin) にリモートコピーします。

変数 *SID* は、データベースインスタンスを表します。

OFA の構造について詳しくは、Oracle のマニュアルを参照してください。

## Oracle ASM の初期化パラメータファイルのコピー

Oracle が ASM で管理されている場合は、cluster 内のそれぞれの node で手順を実行します。

**ASM の初期化パラメータのファイルをコピーするには**

- ◆ 初めて Oracle ASM をインストールした node 上の \$GRID\_HOME/dbs ディレクトリからクラスタの各 node に *initSID.ora* ファイルをコピーします。

ここで、*SID* は、ASM のインスタンスの識別子です。

## Oracle データベースへのアクセスの確認

SQL クエリを発行すると、データベースへのアクセス状態を確認できます。  
\$ORACLE\_HOME の場所に応じて、次のどちらかの手順を実行します。

完了したら、Oracle データベースは最後にこの手順を実行したcluster内のnodeから起動されます。

### Oracle データベースへのアクセスを確認するには

- ◆ \$ORACLE\_HOME の場所に応じて、次のどちらかの手順を実行します。

\$ORACLE\_HOME が共有ディスク上にある場合は、Oracle データベースを起動します。各node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 clusterの一部であるそれぞれのnodeから次の手順を実行します。

#### Oracle データベースを起動するには (\$ORACLE\_HOME が共有ディスク上にある場合)

- 1 Oracle バイナリが他のnode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node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は、Oracle データベースを起動します。Oracle clusterの一部となる各nodeについて、データベースにそれぞれアクセスできることを確認します。

#### Oracle データベースを起動するには (\$ORACLE\_HOME がローカルディスク上にある場合)

- 1 node上に Oracle バイナリがインストールされているかを確認します。
- 2 共有ディスクのデータファイルが任意のnode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。

ASM では、\$ORACLE\_HOME Oracle clusterの一部となる各nodeについて、データベースはローカルディスクにあり、データベースはASM ディスクグループ上で設定されます。データベースにそれぞれアクセスできることを確認します。

ASM で管理された Oracle データベースを起動します。

**Oracle データベースを起動するには (\$ORACLE\_HOME がローカルディスク上にある場合)**

- 1 node上に Oracle バイナリがインストールされているかを確認します。
- 2 ASM のディスクとして VxVM ディスクを使う場合は、次のように処理します。
  - ASM のディスクグループがどれかのnodeでマウントされている場合は、次の操作を行います。
    - ASM インスタンスからASM ディスクグループをマウント解除します。
    - ASM インスタンスを停止します。
    - VxVM ボリュームを停止します。
    - VxVM ディスクグループをデポートします。
  - VxVM ディスクグループをこのnodeにインポートし、ボリュームを起動します。
  - ASM インスタンスを起動します。
  - ASM ディスクグループをマウントします。
- 3 ASM に CVM のボリュームを使用する場合は、次の操作を行います。
  - ディスクグループがマウントされるノードのASM インスタンスからASM ディスクグループをマウント解除します。
  - 他のノードのASM ディスクグループをマウントします。
- 4 ASM のディスクとしてRAW ディスクを使う場合は、次のように操作します。
  - ASM のディスクグループがどれかのnodeでマウントされている場合は、それをマウント解除します。
  - ASM インスタンスがこのノードで実行されていない場合は、ASM インスタンスを起動します。
  - ASM ディスクグループをマウントします。
- 5 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と削除

この章では以下の項目について説明しています。

- [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](#)
- [VCS agent for Oracle ソフトウェアのインストール](#)
- [VCS agent for Oracle のアップグレード](#)
- [Cluster Server agent for Oracle の無効化](#)
- [Cluster Server agent for Oracle の削除](#)

## 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

Cluster Server agent for Oracle をインストールまたはアップグレードするために次の前提条件を満たす必要があります。

- VCS がclusterにインストールされていることを確認します。
- 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。  
Oracle のマニュアルを参照してください。  
p.36 の「[VCS 環境での Oracle のインストール](#)」を参照してください。

## VCS agent for Oracle ソフトウェアのインストール

Cluster Server agent for Oracle のエージェントバイナリは VRTSvcsea package に含まれています。VCS のインストール時にすべての packages または推奨 packages のイ

インストールを選択した場合は、VRTSvcsea package はすでにインストールされています。VCS のインストール中に最小限の packages をインストールした場合は、VRTSvcsea package を手動でインストールする必要があります。

製品ディスクから Cluster Server agent for Oracle をインストールできます。Oracle サービスグループをホストにするすべての nodes に Oracle enterprise agent をインストールする必要があります。

### Solaris SPARC ノード 上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 エージェントがまだインストールされていないことを確認します。`/opt/VRTSagents/ha/bin` に移動し、ディレクトリの内容を一覧表示します。  
表示された一覧に VCS agent for Oracle が含まれている場合は、この手順を省略できます。
- 3 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
```

- 4 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、ディスクは `/cdrom/cdrom0` として自動的にマウントされます。次のコマンドを入力して目的の場所に移動します。

```
# 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、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
```

この例の `/dev/dsk/c0t6d0s2` は、CD ドライブのデフォルトです。ディスクをマウントしたら、パッケージファイルが含まれている場所に移動します。

```
# cd /cdrom
```

- 5 `pkgs` ディレクトリからパッケージをインストールします。

```
# pkgadd -d VRTSvcsea.pkg
```

VRTSvcsea パッケージには、Oracle、Netlsnr、ASMinst、ASMDG の各エージェント用のエージェントバイナリが含まれています。このパッケージでは、DB2 用と Sybase 用の VCS エージェントもインストールされます。

**Solaris SPARC ノード に日本語言語パックをインストールするには**

- 1 エージェントのインストール後に、システムに接続されているドライブに「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、ディスクは `/cdrom/cdrom0` として自動的にマウントされます。次のコマンドを入力して目的の場所に移動します。

```
# 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、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
```

この例の `/dev/dsk/c0t6d0s2` は、CD ドライブのデフォルトです。  
ディスクをマウントしたら、パッケージファイルが含まれている場所に移動します。

```
# cd /cdrom
```

- 3 日本語パッケージをインストールします。

```
# pkgadd -d VRTSjacse.pkg
```

**Solaris x64 ノード 上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 エージェントがまだインストールされていないことを確認します。`/opt/VRTSagents/ha/bin` に移動し、ディレクトリの内容を一覧表示します。  
表示された一覧に **VCS agent for Oracle** が含まれている場合は、この手順を省略できます。
- 3 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
```

- 4 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、ディスクは `/cdrom/cdrom0` として自動的にマウントされます。次のコマンドを入力して目的の場所に移動します。

```
# 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、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
```

この例の `/dev/dsk/c0t6d0s2` は、CD ドライブのデフォルトです。  
ディスクをマウントしたら、パッケージファイルが含まれている場所に移動します。

```
# cd /cdrom
```

- 5 パッケージをインストールします。

```
# pkgadd -d VRTSvcsea.pkg
```

VRTSvcsea パッケージには、Oracle、Netlsnr、ASMinst、ASMDG の各エージェント用のエージェントバイナリが含まれています。このパッケージでは、DB2 用と Sybase 用の VCS エージェントもインストールされます。

## VCS agent for Oracle のアップグレード

Cluster Server agent for Oracle のエージェントバイナリは VRTSvcsea package に含まれています。以前のバージョンの Cluster Server agent for Oracle がインストールされているクラスタで VCS 8.0 にアップグレードした場合は、VRTSvcsea package はすでにアップグレード済みです。

エージェントをアップグレードする前に、Cluster Server agent for Oracle をアップグレードするための前提条件を満たしているかを確認します。

p.59 の「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」を参照してください。

---

**警告:** エージェント package VRTSvcsea には、Oracle 用、Sybase 用、DB2 用の VCS エージェントが含まれています。そのため、VCS agent for Oracle を削除する手順で、他のエージェントもすべて削除されます。VRTSvcsea package に含まれる他のエージェントについても、アップグレード前提条件を満たしていることを確認してください。

---

## VCS agent 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 8.0 で使うために VCS agent for Oracle を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 agent for Oracle の以前のバージョンを削除します。

- VCS 5.1 またはそれ以降の場合:

```
# pkgrm VRTSvcsea
```

- 5.1 による VCS 5.0MP3 の場合 (5.1 を除いて):

```
# pkgrm VRTSvcSOR
```

```
# pkgrm VRTScsow
```

- 3 VCS 8.0 agent for Oracle をインストールします。

p.59 の「[VCS agent for Oracle 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 ファイルが /etc/VRTSagents/ha/conf/Oracle ディレクトリから /etc/VRTSvcS/ha/conf/config ディレクトリにコピーされていることを確認します。

古い OracleTypes.cf ファイルに対して行ったタイプレベルの変更のすべてが、新しくコピーされた OracleTypes.cf ファイルにも適用されている必要があります。

たとえば、MonitorInterval 属性の値をデフォルトの 60 秒から 120 秒に変更した場合、OracleTypes.cf ファイルは更新されています。それらのタイプレベルの変更を、新しくコピーした OracleTypes.cf ファイルに手動で適用する必要があります。

- 5 データベースストレージに ASM を使っている場合

は、/etc/VRTSagents/ha/conf/OracleASM ディレクトリから

/etc/VRTSvcS/ha/conf/config ディレクトリに OracleASMTTypes.cf ファイルがコピーされていることを確認します。

古い OracleASMTTypes.cf ファイルに対して行ったタイプレベルの変更のすべてが、新しくコピーされた OracleASMTTypes.cf ファイルにも適用されている必要があります。

- 6 以前のバージョンで Oracle エージェントに対する詳細監視を有効にしている場合は、LevelTwoMonitorFreq 属性の値を以前のバージョンの DetailMonitor 属性と同じ値に設定します。

---

メモ: DetailMonitor 属性は VCS 6.0 以降では推奨されていません。

---

- 7 VCS エンジンを再起動します。

```
# hastart
```

## VCS 8.0 で使うための VCS 3.5 agent for Oracle のアップグレード (Solaris SPARC)

Veritas Cluster Server agent for Oracle では、VCS 3.5 と VCS 8.0 間でリソースタイプが変更されています。したがって、Sqlnet タイプのリソースを Netlsnr タイプのリソースに置き換える必要があります。

**VCS 8.0 で使うために VCS 3.5 agent for Oracle をアップグレードするには**

- 1 VCS 設定を保存して、VCS エンジンを停止します。

```
# haconf -dump -makero  
# hastop -all -force
```

- 2 Oracle エージェントをインストールしたすべてのノードで、次の処理を実行します。

- Oracle エージェントを削除します。

```
# pkgrm VRTSvcsor
```

- /etc/VRTSvcs/conf/config/OracleTypes.cf ファイルを削除します。

- VCS 8.0 agent for Oracle をインストールします。

p.59 の「[VCS agent for Oracle 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 ファイルが /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されていることを確認します。

古い OracleTypes.cf ファイルに対して行ったタイプレベルの変更のすべてが、新しくコピーされた OracleTypes.cf ファイルにも適用されている必要があります。

たとえば、MonitorInterval 属性の値をデフォルトの 60 秒から 120 秒に変更した場合、OracleTypes.cf ファイルは更新されています。それらのタイプレベルの変更を、新しくコピーした OracleTypes.cf ファイルに手動で適用する必要があります。

- 4 Oracle 10g 以降で、データベースストレージに ASM を使っている場合は、`/etc/VRTSagents/ha/conf/OracleASM` ディレクトリから `/etc/VRTSvcs/conf/config` ディレクトリに `OracleASMTypes.cf` ファイルがコピーされていることを確認します。
- 5 以前のバージョンで Oracle エージェントに対する詳細監視を有効にしている場合は、`LevelTwoMonitorFreq` 属性の値を設定します。
- 6 クラスタ内のノードから、設定ファイル `/etc/VRTSvcs/conf/config/main.cf` を編集します。  
  
Sqlnet タイプのすべてのリソースを Netlsnr タイプのリソースに置き換えます。  
  
p.117 の「リソースタイプの定義と属性の定義について」を参照してください。
- 7 設定を確認します。  
  

```
# hacf -verify config
```
- 8 まずローカルノードで VCS を起動します。
- 9 他のノードで VCS を起動します。

## Cluster Server agent for Oracle の無効化

Cluster Server 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。この際、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できます。

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、サービスグループがオンラインかどうかを確認します。  
  

```
# hagrps -state service_group -sys system_name
```
- 2 サービスグループがオンラインである場合、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nodeのサービスグループをオフラインにして別のnodeでオンラインにするには、`-switch` オプションを使うことができます。  
  

```
# hagrps -switch service_group -to system_name
```
  - cluster内の別のnodeでオンラインにしないでサービスグループをオフラインにするには、次を入力します。  
  

```
# hagrps -offline service_group -sys system_name
```

- 3 node上で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/VRTSvcsl/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。

- 5 すべてのnodesのエージェントを無効にした後に、サービスグループカリゾースタイプ、またはその両方を VCS 設定から削除できます。

詳しくは、『Cluster Server Administrator's Guide』を参照してください。

## Cluster Server agent for Oracle の削除

サービスグループカリゾースタイプ、またはその両方を VCS 設定から削除する前に、すべてのnodesのエージェントを無効にしたことを確認します。

---

**警告:** エージェントpackage VRTSvcsea には、Oracle 用、Sybase 用、DB2 用の VCS エージェントが含まれています。そのため、次の手順で VCS agent for Oracle を削除すると、他のエージェントもすべて削除されます。

---

次の手順を実行して、クラスタの各ノードのエージェントを削除します。

ノード上のエージェントを削除するには

- 1 すべてのノードのすべてのリソースを停止します。

```
# hastop -local
```

- 2 VRTSvcsea package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 VRTSvcsea
```

**Solaris SPARC ノードで日本語言語パックを削除するには**

- ◆ 日本語パッケージを削除します。

```
# pkgrm VRTSjacse
```

# Oracle 用 VCS サービスグループの設定

この章では以下の項目について説明しています。

- Oracle 用サービスグループの設定について
- VCS における Oracle インスタンスの設定
- Oracle の VCS サービスグループを設定する前に
- Oracle のための VCS サービスグループの設定
- VCS agent for Oracle の詳細監視の設定
- エージェントに対するインテリジェントなリソース監視の手動による有効化と無効化

## 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と設定には、Administrator 権限が必要です。

次のいずれかを使用して、Oracle 用の VCS サービスグループを設定できます。

- Veritas InfoScale Operations Manager
- Cluster Manager (Java コンソール)
- コマンドライン

---

**メモ:** Oracle ASM の機能を使う場合は、ASMinst リソースと ASMDG リソースも設定して、ASM インスタンスがいつも利用できるようにする必要があります。これらのリソースの設定を行うには、コマンドラインを使う必要があります。

---

## VCS における Oracle インスタンスの設定

VCS 環境では、さまざまな方法で Oracle を設定できます。VCS 用に Oracle を設定するときは、Oracle ファイル 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。

非グローバルゾーンで実行する Oracle インスタンスのサービスグループを設定する場合、サービスグループに含めるすべてのシステムにゾーンが作成されていることを確認する必要があります。

表 4-1 VCS での Oracle 設定

設定のタイプ (Configuration Type)	参照先
単一の Oracle インスタンスの設定	p.68 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	p.70 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	p.72 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	p.74 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。

p.196 の「VCS 環境で複数の 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。

**メモ:** Oracle ファイル \$TNS\_ADMIN/listener.ora と \$TNS\_ADMIN/tnsnames.ora がクラスタノード間で同一であることを確認します。ファイルが同一でない場合、ローカルノードからクラスタ内の他のノードにファイルをリモートコピーします。

グリッドインフラストラクチャが設定されている場合は、データベースリスナーにデフォルト名を使用しないようにしてください。そうしないと、グリッドインフラストラクチャがデフォルト名のリスナー (LISTENER) を起動し、さまざまな問題が発生する可能性があります。

## 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。

p.141 の「Oracle 単一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。

p.117 の「リソースタイプの定義と属性の定義について」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 Oracle の VCS サービスグループを作成します。  
p.79 の「[Oracle のための VCS 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。  
p.96 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。

p.144 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(単ーリスナー)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。  
p.117 の「リソースタイプの定義と属性の定義について」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)
    )
  )
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。

`$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 Oracle の VCS サービスグループを作成します。  
p.79 の「[Oracle のための VCS 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。  
p.96 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。

- p.151 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。  
p.117 の「リソースタイプの定義と属性の定義について」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 Oracle の VCS サービスグループを作成します。  
p.79 の「[Oracle のための VCS 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。  
p.96 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。

p.155 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。

p.117 の「リソースタイプの定義と属性の定義について」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 Oracle の VCS サービスグループを作成します。  
p.79 の「[Oracle のための VCS サービスグループの設定](#)」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。  
p.96 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。

## Oracle の VCS 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 cluster 内のすべての nodes 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『[Cluster Server Installation Guide](#)』を参照してください。

- cluster内のすべてのnodes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。  
p.36 の「VCS 環境での Oracle のインストール」を参照してください。
- cluster内のすべてのnodesに Cluster Server agent for Oracle がインストールされていることを確認します。  
p.59 の「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」を参照してください。
- Cluster Server 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。  
p.77 の「Cluster Server agent for Oracle のタイプ定義ファイルのインポート」を参照してください。
- Oracle データベースが ASM で管理されている場合は、Oracle クラスタデーモンが有効になっていることを確認します。他に、OCSSD のデーモンは無効にしておく必要があります。

## Cluster Server agent for Oracle のタイプ定義ファイルのインポート

VRTSvcsea パッケージを手動でインストールした場合に、この手順を実行します。VCS のインストール時にすべてのパッケージまたは推奨パッケージを選択した場合は、インストーラプログラムによってタイプ定義ファイルがインポートされます。

Cluster Server 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。

ASM エージェントを設定する場合は、OracleASMTypes.cf ファイルをインポートする必要があります。コマンドラインを使用して OracleASMTypes.cf ファイルをインポートします。

### Cluster Manager を使ってインポートするには

- 1 Cluster Manager を起動して、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システムにログインします。
- 2 次のコマンドを使って、既存の設定に対する変更を確実に保存します。そして、**main.cf** を変更している間、各システムの状態、アプリケーションやシステムの設定などを変更しないようにします。

```
# 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
```

**ASM** エージェントを設定する場合は、次の操作を行います。

```
# cp /etc/VRTSagents/ha/conf/OracleASM/OracleASMTypes.cf ¥  
/etc/VRTSvcs/conf/config/OracleASMTypes.cf
```

- 6 **main.cf** ファイルを編集し、**OracleTypes.cf** ファイルをインクルードします。

```
include OracleTypes.cf
```

**ASM** エージェントを設定する場合は、**OracleASMTypes.cf** ファイルも含める必要があります。

```
include "OracleASMTypes.cf"
```

- 7 設定を確認します。

```
# cd /etc/VRTSvcs/conf/config  
# hacf -verify .
```

- 8 **VCS** エンジンを起動します。

```
# hastart
```

## Oracle のための VCS 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。

p.68 の「[VCS における Oracle インスタンスの設定](#)」を参照してください。

表 4-2 は、Oracle を設定してエージェントをインストールした後に、VCS agent for Oracle を設定するために利用できるインターフェースの一覧です。

表 4-2 VCS agent for Oracle を設定するためのインターフェース

設定方法	参照
Cluster Manager (Java コンソール)	p.79 の「 <a href="#">Cluster Manager (Java コンソール)を使った Oracle 用 VCS サービスグループの設定</a> 」を参照してください。
コマンドライン	p.82 の「 <a href="#">コマンドラインを使った Oracle 用 VCS サービスグループの設定</a> 」を参照してください。
Veritas InfoScale Operations Manager	詳しくは、Veritas InfoScale Operations Manager のマニュアル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  
 p.140 の「[Oracle enterprise agent の設定例について](#)」を参照してください。
- リソースタイプと、Oracle エージェントや Netlshr エージェントの属性定義  
 ASM については、ASMInst と ASMDG のエージェントのリソースタイプと属性定義。  
 p.117 の「[リソースタイプの定義と属性の定義について](#)」を参照してください。

### Cluster Manager (Java コンソール) を使った Oracle 用 VCS 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も、Solaris の非グローバルゾーンで実行される Oracle インスタンスのサービスグループを設定できません。このようなサービスグループを設定するには、設定ウィザードを使います。

---

### 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。次のいずれかを実行します。
  - 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)]画面の情報を読み、[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。
  - サービスグループの名前を入力します。
  - [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。
  - 右矢印ボタンをクリックして、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。

p.117 の「リソースタイプの定義と属性の定義について」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 Oracle 用 VCS サービスグループの設定

Cluster Server 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属しています。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。

また、Cluster Server agent for Oracle には VCS 環境で ASM を設定するための設定ファイルの例も含まれています。

p.140 の「[Oracle enterprise agent の設定例について](#)」を参照してください。

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 Oracle が動作しているシステム上で、スーパーユーザーとしてログインします。
- 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、次のコマンドを実行し、すべてのシステム上で VCS エンジン を停止して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 `main.cf` ファイルを編集します。

設定の参考として、`/etc/VRTSagents/ha/conf/Oracle/sample_main.cf` ファイルを利用できます。

次の処理を行います。

- Oracle と Netlsnr のリソースを作成します。  
ASM インスタンスおよびディスクグループ用に、`ASMinst` リソースおよび `ASMDG` リソースを作成します。
- 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。  
ゾーンの環境内で `ASMinst` エージェントを設定した場合、リソースタイプ属性 `OnlineRetryLimit` の値を 1 に設定します。  
[p.117 の「リソースタイプの定義と属性の定義について」](#)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。  
[p.140 の「Oracle enterprise agent の設定例について」](#)を参照してください。

依存関係の割り当てについて詳しくは、『Cluster Server Administrator's Guide』を参照してください。

6 ファイルを保存して閉じます。

7 `/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 .
```

8 VCS エンジンを起動します。

```
# hastart
```

9 すべての Oracle サービスグループのリソースがオンラインであることを確認します。

```
# hagr -state
```

10 サービスグループをオフラインにし、すべてのリソースが停止していることを確認します。

```
# hagr -offline service_group -sys system_name  
# hagr -state
```

11 サービスグループを再びオンラインにし、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
```

12 VCS エンジンをクラスタ内の残りのすべてのノード上で起動します。

```
# hastart
```

13 (省略可能) Oracle サービスグループがフェールオーバー用に適切に設定されていること検証するため、サービスグループをクラスタノードのいずれかに切り替えます。

```
# hagr -switch service_group -to any
```

すべての Oracle サービスグループリソースが、システム上でオンライン状態であることを確認します。

```
# hagr -state
```

14 すべての **nodes** について、次のログファイルでエラーや状態について調べてください。

```
/var/VRTSvcs/log/engine_A.log  
/var/VRTSvcs/log/Oracle_A.log  
/var/VRTSvcs/log/Netlsnr_A.log  
  
/var/VRTSvcs/log/ASMinst_A.log  
/var/VRTSvcs/log/ASMDG_A.log
```

## Oracle データベースのユーザーパスワードとリスナー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。

`vcscrypt` のマニュアルページを参照してください。

`vcscrypt` ユーティリティを使っても、セキュリティキーを使ったエージェントパスワードの暗号化ができます。セキュリティキーでは、より安全なエージェントパスワードを作成する AES (Advanced Encryption Standard) 暗号化がサポートされています。

詳しくは、『Cluster Server Administrator’s Guide』を参照してください。

---

**メモ:** VCS Cluster Manager (Java コンソール) を使って属性を設定している場合は、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。

Oracle `lsnrctl` ユーティリティを使ってリスナーパスワードを暗号化した場合は、使用する Oracle のバージョンに応じて次の操作を行います。

- クリアテキストのパスワードを `vcscrypt` ユーティリティに渡します。  
たとえば、リスナーパスワードが「`oracle`」の場合は、次のコマンドを使って、このクリアテキストパスワードを `vcscrypt` に渡します。

```
# vcscrypt -agent oracle
```

`vcscrypt` ユーティリティに暗号化したパスワードが表示されます。たとえば、「`bnqPdnCneNnnG`」のように表示されます。

vcsencrypt ユーティリティを使っても、セキュリティキーを使ったエージェントパスワードの暗号化ができます。セキュリティキーでは、安全なエージェントパスワードを作成する AES (Advanced Encryption Standard) 暗号化がサポートされています。詳しくは、『Cluster Server 管理者ガイド』を参照してください。

## VCS agent for Oracle の詳細監視の設定

Cluster Server 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。

---

**メモ:** Oracle ASM のためのエージェントは ASM のインスタンスのための基本的な監視のみ行います。

---

- 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。

p.21 の「[従来のデータベースおよびコンテナデータベース内の Oracle エージェントに対する監視オプション](#)」を参照してください。

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:** データベースの保守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、データベース内のテーブル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を確認します。この更新アクションは、Cluster Server 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、テーブル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テーブル(タイムスタンプ付きの)を作成する必要があります。エージェントは、このテストテーブルを内部的に使用します。テストテーブルではその他のトランザクションを実行しないことをお勧めします。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、詳細監視で使うテーブルの作成とテストの方法と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 と ORACLE\_SID の環境変数を設定します。

```
# export ORACLE_HOME=<Home>
```

```
# export ORACLE_SID=<Sid>
```

- 5 sqlplus ユーティリティを起動してデータベーステーブルを設定します。

```
sqlplus /nolog
```

- 6 データベース管理者として、sqlplusプロンプトに次の文を挿入してテストテーブルを作成します。

従来のデータベースおよびコンテナデータベースの場合:

```
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);
```

---

**メモ:** CDB リソースの場合、ユーザー名の先頭に **C##** を付加する必要があります。

---

PDB の詳細監視を有効にするには、sqlplus プロンプトに次の文を挿入してテストテーブルを作成します。

```
$_ORACLE_HOME/bin/sqlplus /nolog
connect / as sysdba
```

CDB の共通ユーザーを作成してない場合は作成します。

```
create user <User>
identified by <Pword>
default tablespace USERS
temporary tablespace TEMP
quota 100K on USERS;
grant connect to <User> container=all;
alter session set container=PDBName
create table <User>.<Table> ( tstamp date );
insert into <User>.<Table> (tstamp) values (SYSDATE);
```

---

**メモ:** PDB の詳細監視を行わない場合でも、VCS Agent for Oracle には CDB と PDB 両方の詳細監視の共通ユーザーが必要です。

---

- 7 データベーステーブルが使えることをテストするには、CDBまたは従来のデータベースで次の操作をします。

```
$ORACLE_HOME/bin/sqlplus /nolog
connect / as sysdba
connect <User>/<Pword>
update <User>.<Table> set (tstamp)=SYSDATE;
select TO_CHAR(tstamp, 'MON DD, YYYY HH:MI:SS AM')
from <User>.<Table>;
```

#### PDB の場合

```
alter session set container=PDBName;
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 -override OracleResource LevelTwoMonitorFreq
# hares -modify OracleResource LevelTwoMonitorFreq 1

# haconf -dump -makero

# hagrps -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 リソースに対する詳細監視の有効化と無効化

手順を見直して、詳細な監視の有効と無効を切り替えます。

詳細監視を有効にするには

- ◆ LevelTwoMonitorFreq 属性の値を設定します。

```
# hares -override OracleResource LevelTwoMonitorFreq
# hares -modify OracleResource LevelTwoMonitorFreq 1
```

詳細監視を無効にするには

- ◆ `LevelTwoMonitorFreq` 属性の値を 0 に設定します。

```
# hares -modify OracleResource LevelTwoMonitorFreq 0
```

## Netlsnr に対する詳細監視の設定

`Netlsnr` エージェントに対しては、リスナープロセスを監視するための詳細な監視はデフォルトで有効に設定されていません。

`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` の詳細な監視を無効にするには

- ◆ 詳細監視を無効にするには、`MonScript` 属性に空の文字列を設定します。

```
# haconf -makerw
# hagr -freeze service_group
# hares -modify LsnrResource MonScript ""

# hares -override LsnrResource LevelTwoMonitorFreq
# hares -modify LsnrResource LevelTwoMonitorFreq 0

# haconf -dump -makero
# hagr -unfreeze
```

### 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。

```
# hagrps -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"
```

```
# hares -override LsnrResource LevelTwoMonitorFreq
```

```
# hares -modify LsnrResource LevelTwoMonitorFreq 1
```

```
# haconf -dump -makero
```

```
# hagrps -unfreeze service_group
```

## エージェントに対するインテリジェントなリソース監視の手動による有効化と無効化

インテリジェントなリソース監視を手動で有効または無効にするための次の手順を確認します。インテリジェントなリソース監視機能は、デフォルトで有効になっています。IMF リソースタイプ属性により、IMF 対応のエージェントがインテリジェントなリソース監視を実行する必要があるかどうかが決まります。

p.11 の「[Oracle エージェントと Netlsnr エージェントがインテリジェントなリソース監視をサポートするしくみ](#)」を参照してください。

### インテリジェントなリソース監視を有効にするには

- 1 VCS 設定を書き込み可能にします。

```
# haconf -makerw
```

- 2 インテリジェントなリソース監視を有効にするには、次のコマンドを実行します。

- オフラインのリソースのインテリジェントな監視を有効にするには:

```
# hatype -modify resource_type IMF -update Mode 1
```

- オンラインのリソースのインテリジェントな監視を有効にするには:

```
# hatype -modify resource_type IMF -update Mode 2
```

- オンラインおよびオフライン両方のリソースのインテリジェントな監視を有効にするには:

```
# hatype -modify resource_type IMF -update Mode 3
```

- 3 必要に応じて、IMF 属性の **MonitorFreq** キーと **RegisterRetryLimit** キーの値を変更します。

p.118 の「[Oracle エージェントの属性の定義](#)」を参照してください。

p.129 の「[Netlsnr エージェントの属性定義](#)」を参照してください。

- 4 VCS 設定を保存します。

```
# haconf -dump -makero
```

- 5 クラスタのすべてのノードで **AMF** カーネルドライバが設定されていることを確認します。

```
/lib/svc/method/amf status
```

**AMF** のカーネルドライバが設定されている場合、出力は次のようになります。

```
AMF: Module loaded and configured
```

コマンドにより、**AMF** ドライバがロードされていないか設定されていないことを示す出力が返された場合は、**AMF** ドライバを設定します。

p.92 の「[AMF カーネルドライバの管理](#)」を参照してください。

- 6 エージェントを再起動します。各ノードで次のコマンドを実行します。

```
# haagent -stop agent_name -force -sys sys_name
```

```
# haagent -start agent_name -sys sys_name
```

インテリジェントなリソースの監視を無効にするには

- 1 VCS 設定を書き込み可能にします。

```
# haconf -makerw
```

- 2 特定のタイプのすべてのリソースに対するインテリジェントなリソースの監視を無効にするには、次のコマンドを実行します。

```
# hatype -modify resource_type IMF -update Mode 0
```

- 3 特定のリソースに対するインテリジェントなリソースの監視を無効にするには、次のコマンドを実行します。

```
# hares -override resource_name IMF
```

```
# hares -modify resource_name IMF -update Mode 0
```

- 4 VCS 設定を保存します。

```
# haconf -dump -makero
```

---

**メモ:** VCS はエージェントの IMF 機能の有効と無効を切り替えるために `haimfconfig` スクリプトを提供します。実行または停止状態の VCS のスクリプトを使うことができます。スクリプトを使って、IMF 対応の付属エージェント、エンタープライズエージェント、およびカスタムエージェントの IMF を有効または無効にします。

---

## AMF カーネルドライバの管理

AMF カーネルドライバを起動、停止、アンロードするための次の手順を確認します。

**AMF カーネルドライバを起動するには**

- 1 `AMF_START` 変数の値が 1 でない場合、次のファイルでその値を 1 に設定します。

```
# /etc/default/amf
```

- 2 AMF カーネルドライバを起動します。次のコマンドを実行します。

```
# svcadm enable amf
```

### AMF カーネルドライバを停止するには

- 1 AMF\_STOP 変数の値が 1 でない場合、次のファイルでその値を 1 に設定します。

```
# /etc/default/amf
```

- 2 AMF カーネルドライバを停止します。次のコマンドを実行します。

```
# svcadm disable amf
```

### AMF カーネルドライバをアンロードするには

- 1 エージェントの停止時間が問題でなければ、次の手順に従って AMF カーネルドライバをアンロードします。

- AMF カーネルドライバに登録されているエージェントを停止します。  
 amfstat コマンドの出力の **Registered Reapers** セクションに、AMF に登録されているエージェントのリストがあります。  
 amfstatのマニュアルページを参照してください。

- Solaris 11 で、AMF\_DISABLE 変数の値を次のファイルで 1 に設定します (AMF カーネルドライバを停止する前に)。

```
# /etc/default/amf
```

---

**メモ:** AMF カーネルドライバを起動する前に、`/etc/default/amf` ファイルの `AMF_DISABLE` 変数をリセットします。

---

- AMF カーネルドライバを停止します。  
 p.93 の「[AMF カーネルドライバを停止するには](#)」を参照してください。
- エージェントを起動します。

- 2 エージェントの停止時間を最小限に抑える必要がある場合は、次の手順に従って AMF カーネルドライバをアンロードします。

- 次のコマンドを実行して、エージェントが登録されている状態のまま AMF ドライバを無効にします。

```
# amfconfig -Uof
```

- Solaris 11 で、AMF\_DISABLE 変数の値を次のファイルで 1 に設定します (AMF カーネルドライバを停止する前に)。

```
# /etc/default/amf
```

---

メモ: AMF カーネルドライバを起動する前に、`/etc/default/amf` ファイルの `AMF_DISABLE` 変数をリセットします。

---

- AMF カーネルドライバを停止します。  
p.93 の「[AMF カーネルドライバを停止するには](#)」を参照してください。

# Oracle 用 VCS サービスグループの管理

この章では以下の項目について説明しています。

- [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化](#)
- [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](#)

## VCS サービスグループの管理について

次のいずれかのインターフェースを使って、Cluster Server でサービスグループを管理できます。

- Veritas InfoScale Operations Manager  
詳しくは、Veritas InfoScale Operations Manager のマニュアルを参照してください。
- Cluster Manager (Java コンソール)  
p.96 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。  
p.96 の「[サービスグループのオフライン化](#)」を参照してください。  
p.96 の「[サービスグループの切り替え](#)」を参照してください。  
p.97 の「[サービスグループ設定の修正](#)」を参照してください。
- コマンドライン  
詳しくは、『Cluster Server 管理者ガイド』を参照してください。

## 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状態にするには

- 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)] を選択してサービスグループのすべてのリソースを有効にします。
- 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 Configuration)] の順に選択)。
- 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化

サービスグループをオフライン化するには、次の手順を実行します。

サービスグループをオフライン化するには

- 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

VCS を管理するためのサポート対象インターフェースのいずれかを使って、Cluster Server agent for Oracle エージェントを動的に修正できます。

# プラグ可能データベース (PDB) の移行

この章では以下の項目について説明しています。

- コンテナデータベース (CDB) 間でのプラグ可能データベース (PDB) の移行

## コンテナデータベース (CDB) 間でのプラグ可能データベース (PDB) の移行

`hapdbmigrate` ユーティリティを使って、同じデータベースバージョンのコンテナ全体でプラグ可能データベースの計画済みエンドツーエンド移行を実行できます。移行先コンテナデータベースは同じノードに存在しても別のノードに存在してもかまいません。PDBには独自の独立したストレージが必要です。

PDBに親グループがある場合、移行中は親グループのリンクは解除され、ユーティリティが凍結します。移行が正常に終了すると、親グループは再びリンクされ、ユーティリティが解凍します。

設定が次の必要条件を満たしている必要があります。

- PDB と CDB のデータベースが Veritas CFS (Cluster File System) または VxFS (Veritas File system) に存在する
- PDB リソースの親リソースが存在しない
- PDB がソース CDB にプラグされている(PDB リソースがオフラインかどうかは任意)
- 移行元の CDB リソースと移行先の CDB リソースが異なる
- 移行先 CDB のバージョンが移行元 CDB のバージョンと同じ
- 移行する PDB を独立ファイルシステムにマウントしている (CDB と PDB のデータファイルのマウントポイントが異なる)

- CDB と PDB のリソースを同じサービスグループで設定している  
 ユーティリティは移行時に次の処理を実行します。
- 移行する PDB リソースを設定するソース CDB グループに応じて、親グループのリンクを解除して凍結します (該当する場合)。
- PDB リソースをオフラインにします。
- CDB から PDB をプラグ解除して、ユーザーが指定した XML ディレクトリに XML ファイル `<pdb_res_name>_<dest_cdb_res_name>.xml` を作成します。
- データファイルを維持したまま CDB から PDB を切断します。
- PDB のすべての子リソースをオフラインにします。ソース CDB リソースから PDB リソースのリンクを解除して PDB リソースを削除します。PDB の子リソースすべてのリンクを解除して削除します。
- 移行先 CDB サービスグループで PDB リソースとそのすべての子を元の依存関係で再作成します。
- 移行先 CDB サービスグループがオンラインのすべてのノードで PDB 子リソースをオンラインにします。
- 移行先 CDB で PDB をプラグします。
- 正常にプラグしたら PDB リソースをオンラインにします。親グループを凍結解除してソース CDB グループにリンクします。

hapdbmigrate ユーティリティは、移行する前にクラスタで PDB 移行の準備ができていることを確認するために、特定の前提条件チェックを実行します。ユーティリティで問題が発生した場合は手動で問題を解決する必要があります。

ユーティリティは `$VCSHOME/bin` ディレクトリにあります。

移行のログファイルは `$VCSLOG/log/hapdbmigrate.log` にあります。ファイルが 5 MB を超えるとログはローテーションして `.gz` 形式で `hapdbmigrate.log[1..7].gz` に保存されます。

---

**メモ:** hapdbmigrate ユーティリティで一度に実行できるインスタンスは 1 つだけです。

---

## コンテナデータベース (CDB) 間でプラグ可能データベース (PDB) を移行するには

- 1 VCS 設定ファイル `/etc/VRTSvcs/conf/config/main.cf` のバックアップを作成します。

```
# cp /etc/VRTSvcs/conf/config/main.cf ¥  
/etc/VRTSvcs/conf/config/main.cf.save
```

- 2 クラスタのすべてのノードで高可用性デーモン (**had**) を実行していることを確認します。

```
# hastatus -summary
```

- 3 障害状態または不明な状態のリソースが存在しないことを確認します。

```
# hares -state|grep FAULTED  
# hares -state|grep UNKNOWN
```

- 4 `main.cf` 設定ファイルの正しい値を設定した `PDBName` 属性が **PDB** リソースに存在することを確認します。

- 5 **CDB** の移行元リソースと移行先リソースがオンラインであることを確認します。

```
# hares -state resname
```

- 6 移行先 **CDB** で次のことを確認します。

- 移行先 **CDB** が `suspended` モードではないことを確認します。  
移行先 **CDB** のインスタンスがマウント状態ではないことを確認します。  
詳しくは、**Oracle** のマニュアルを参照してください。
- 移行先 **CDB** のインスタンスが `restricted` 状態でない場合は、移行する **PDB** リソースの `StartUpOpt` 属性を `restricted` に設定していることを確認します。

```
# haconf -makerw  
# hares -modify pdb1 StartUpOpt ¥  
RESTRICTED  
# haconf -dump -makero
```

- 7 既存の依存関係が移行プロセスと競合しないことを確認します。

**PDB** 子リソースは **CDB** リソースまたはその子リソースに依存することはできません。

**PDB** 子リソースは **PDB** 子階層に含まれない親を持つことはできません。

- 8 **XML** データディレクトリに、「**oracle**」ユーザーの読み取り権限と書き込み権限があることを確認します。**XML** データディレクトリは、**PDB** マウントまたは移行元と移行先両方の **CDB** にアクセスできる場所に配置する必要があります。

- 9 ルートユーザーとして hapdbmigrate ユーティリティを実行します。

---

**メモ:** 移行元 CDB グループに依存する親グループがある場合は `-ignoreparentgrp` オプションを指定します。

---

```
# $VCS_HOME/bin/hapdbmigrate -pdbres pdb_resname -cdbres
cdb_resname ¥
-XMLdirectory xml_directory [-ignoreparentgrp]
[-prechecks]-pdbres: Name of the PDB resource, which needs to be
migrated
-cdbres: Name of the CDB resource, where the PDB needs to migrate

-XMLdirectory: XML directory location for the unplugged PDB
-ignoreparentgrp: Utility proceeds even
if the PDB group has parent groups
-prechecks: Performs prechecks and validation
-help|h: Prints usage
```

移行ログファイルは `$VCSLOG/log/hapdbmigrate.log` にあります。

- 10 移行先 CDB で PDB リソースがオンラインになっていることを確認します。

```
# hares -state pdb_resname
```

- 11 移行元 CDB グループの親サービスグループが移行した PDB に依存している場合は移行先 CDB グループに手動で再度リンク付けします。

```
# haconf -makerw
# hagr -dep parent_sg
#Parent          Child          Relationship
parent_sg        source_CDB      online local firm
# hagr -offline parent_sg -any
# hagr -unlink parent_sg source_CDB
# hagr -link parent_sg dest_CDB online local firm
# hagr -dep parent_sg
#Parent          Child          Relationship
parent_sg        dest_CDB        online local firm
# haconf -dump -makero
```

p.186 の「[main.cf の移行可能な Oracle プラグ可能データベース \(PDB\) リソースの設定例](#)」を参照してください。

p.117 の「[Oracle エージェントのリソースタイプ定義](#)」を参照してください。

p.128 の「[Netlsnr エージェントのリソースタイプ定義](#)」を参照してください。

# Cluster Server agent for Oracle のトラブルシューティング

この章では以下の項目について説明しています。

- [Cluster Server agent for Oracle](#) のトラブルシューティングについて
- [Oracle](#) エージェントと [Netlsnr](#) エージェントに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
- [ASMIInst](#) エージェント固有のエラーメッセージ
- [ASMDG](#) エージェント固有のエラーメッセージ
- [VCS](#) 環境における [Oracle](#) 固有の問題のトラブルシューティング
- [Oracle](#) 診断バイナリと [Oracle](#) のインスタンスの意図的なオフライン化の検証
- [PDB](#) リソースの [IMF](#) の無効化

## Cluster Server agent for Oracle のトラブルシューティングについて

アクセスが必要なエラーログの情報を見直します。

- [Oracle](#) のインストールのエラーログを調べるには、次のファイルにアクセスする必要があります。

```
$ORACLE_BASE/oraInventory/logs/installActionsdate_time.log
```

このファイルにはインストールの間に起きたエラーが記載されています。その記載内容はエラーの性質を明らかにするとともに、インストールの間に生じたことを正確に指し示しています。インストールに問題がある場合は、問題をデバッグするためにテクニカルサポートにこのファイルを送る必要があります。

- ログファイルを調べるには、次のファイルにアクセスする必要があります。

```
/var/VRTSvcs/log/engine_A.log
/var/VRTSvcs/log/Oracle_A.log
/var/VRTSvcs/log/Netlsnr_A.log

/var/VRTSvcs/log/ASMinst_A.log
/var/VRTSvcs/log/ASMDG_A.log
```

これらのファイルには、VCS エンジンや他の Oracle のエージェントが実行したすべてのアクションが含まれています。

次に示すエージェントについてのエラーメッセージ内容と、実施可能な解決策を見直します。

- Oracle エージェント
- Netlsnr エージェント
- ASMinst エージェント
- ASMDG エージェント

## Oracle エージェントと Netlsnr エージェントに共通のエラーメッセージ

表 7-1 に、Cluster Server agent for Oracle 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 7-1 Cluster Server agent for Oracle 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。

メッセージ	説明と解決方法
Oracle ホームディレクトリ <Oracle_Home> が存在しません (Oracle home directory <Oracle_Home> 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。
ファイル <env_file> は有効なテキストファイルではありません (File <env_file> 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か、読み込めないか、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 <Process_name> を再起動しました (Process <Process_name> 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 <monscript> から <output> が返されました (Monitor procedure <monscript> returned <output>)	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 <monscript> は終了しませんでした (Monitor procedure <monscript> did not exit)	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 <owner_name> が、Oracle の所有者として定義されています (Invalid owner <owner_name> for Oracle executables was specified)	Owner 属性が指定するオペレーティングシステムの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。

メッセージ	説明と解決方法
<p>Monscript &lt;monscript&gt; 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 &lt;monscript&gt; denied. Detail Monitoring will not be enabled!! Please specify a valid file.)</p>	<p>MonScript の属性が指定するファイルは、アクセスできないか、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>
<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>	<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: vcsencrypt 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 7-2 に、Oracle の VCS 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 7-2 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
<p>SID が定義されていません (No SID specified)</p>	<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 属性の値に正しいデータベースインスタンスを設定します。</p>
<p>sqlplus が &lt;Oracle_Home&gt;/bin で見つかりません (sqlplus not found in &lt;Oracle_Home&gt;/bin)</p>	<p>クライアントユーティリティ sqlplus が \$ORACLE_HOME/bin 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>

メッセージ	説明と解決方法
srvctl が <Oracle_Home>/bin で見つかりません (srvctl not found in <Oracle_Home>/bin)	クライアントユーティリティ srvctl が \$ORACLE_HOME/bin 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。
Oracle <SID> の停止に失敗しました (Oracle failed to stop)	次のコマンドが <b>clean</b> または <b>offline</b> エントリーポイントで Oracle インスタンスを閉じる処理に失敗したことを示す警告メッセージ。 <ul style="list-style-type: none"> <li>■ Shutdown immediate</li> <li>■ Shutdown abort</li> </ul>
Oracle データベース <SID> は起動していません (Oracle database not running)	clean または offline エントリー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 (<SID>) の強制終了の用語 <PID1, PID2, ...> (Oracle (<SID>) kill TERM <PID1, PID2, ...>)	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 (<SID>) の強制終了の強制終了 <PID1, PID2, ...> (Oracle (<SID>) kill KILL <PID1, PID2, ...>)	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 状態または QUIESCED 状態であることを示す警告メッセージ。

メッセージ	説明と解決方法
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。
リソース <Res_name> - 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 - 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 属性の値に応じて、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:  <b>FaultOnMonitorTimeouts</b> 属性の値を 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 パラメータの値を TRUE に設定します。</p>
VCS ERROR Resource <Res_name> - オンラインプロシージャが、設定した時間内に完了しませんでした。(VCS ERROR Resource <Res_name> - online procedure did not complete within the expected time.)	解決方法: Oracle エージェントの OnlineTimeout 間隔を増加します。

メッセージ	説明と解決方法
<p>カスタムスクリプト /opt/VRTSagents/ha/bin/Oracle/start_custom_&lt;SID&gt;.sql が存在しません。(Custom script /opt/VRTSagents/ha/bin/Oracle/start_custom_&lt;SID&gt;.sql does not exist.) データベースが起動できなくなります。</p>	<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>
<p>カスタムスクリプト /opt/VRTSagents/ha/bin/Oracle/shut_custom_&lt;SID&gt;.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_&lt;SID&gt;.sql does not exist. Using default shutdown option.)</p>	<p>データベースを停止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>
<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>	<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>
<p>監視オプションが正しくありません (Incorrect Monitor Option)</p>	<p>MonitorOption の値が、0 未満または 1 より大きくなっています。</p> <p>解決方法: MonitorOption 属性の値を 0 または 1 に設定します。</p>
<p>診断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)</p>	<p>内部エラーです。</p> <p>解決方法: テクニカルサポートまでご連絡ください。</p>
<p>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)</p>	<p>内部エラーです。</p> <p>解決方法: テクニカルサポートまでご連絡ください。</p>

メッセージ	説明と解決方法
属性 User:Pword:Table:MonScript の 1 つ以上が正しく設定されていません。(One or more of the attributes User:Pword:Table:MonScript are not set correctly.) 詳細監視が有効になりません。	<p>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。</p> <p>解決方法: 詳細監視の必須属性の値を正しく設定します。</p>

## Netlsnr エージェント固有のエラーメッセージ

表 7-3 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 7-3** 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.)	<p>特定の監視サイクルで、/proc エントリを処理できませんでした。</p> <p>解決方法: 解決する必要はありません。</p>
リスナープロセス <Listener> が、動作していません (Listener process <Listener> not running)	<p>clean または offline 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。</p> <p>解決方法: 解決する必要はありません。</p>
Listener <Listener> を TERM <PID1, PID2, ...> で強制終了します (Listener <Listener> kill TERM <PID1, PID2, ...>)	<p>リスナープロセスに SIGTERM が発行されたことを示す警告メッセージ。</p> <p>解決方法: 解決する必要はありません。</p>
Listener <Listener> を KILL <PID1, PID2, ...> で強制終了します (Listener <Listener> kill KILL <PID1, PID2, ...>)	<p>リスナープロセスに SIGKILL が発行されたことを示す警告メッセージ。</p> <p>解決方法: 解決する必要はありません。</p>
<Oracle_Home>/bin で lsnrctl が見つかりません。(Lsnrctl not found in <Oracle_Home>/bin)	<p>クライアントユーティリティ lsnrctl が \$ORACLE_HOME/bin 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>

メッセージ	説明と解決方法
lsnrctl がタイムアウトしました。(lsnrctl operation timed out)	tnsnmr プロセスが応答しません。 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。

## ASMinst エージェント固有のエラーメッセージ

表 7-4 に、ASMinst 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 7-4 ASMinst 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
Home 属性には複数のトークンを設定できません。(Attribute Home cannot have multiple tokens.)	ASMinst エージェントの Home 属性に複数のトークンがあります。 解決方法: Home 属性の値を修正します。
クラスタ同期サービスプロセスは動作していません。(Cluster Synchronization Service process is not running.)	Oracle CSSD プロセスが実行されていません。 解決方法: CSSD プロセスを有効にします。 p.52 の「 <a href="#">ASM で管理されたデータベースのためのクラスタデーモンの有効化</a> 」を参照してください。
クラスタ同期サービスが停止したか、再起動されます。(Cluster Synchronization Service died or is restarted.)	Oracle CSSD プロセスは停止しました。 解決方法: CSSD プロセスを有効にします。 p.52 の「 <a href="#">ASM で管理されたデータベースのためのクラスタデーモンの有効化</a> 」を参照してください。

## ASMDG エージェント固有のエラーメッセージ

表 7-5 に、ASMDG 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 7-5 ASMDG 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
ASM ディスクグループの名前が指定されていないか、指定された文字列がヌルです。(No ASM Diskgroup name specified, or is null.)	ASMDG エージェントの DiskGroups 属性に値が指定されていません。 解決方法: Diskgroup 属性の値を指定します。

メッセージ	説明と解決方法
エージェントが ASMDG リソースの状態を識別できません。ASM の初期化パラメータファイルで <code>asm_diskstring</code> パラメータが正しく設定されていません。(Agent unable to identify state of the ASMDG resource. The <code>asm_diskstring</code> parameter is not set correctly in the ASM initialization parameter file.)	ASM ディスクグループを検索するための ASM インスタンスに対する正しい値が ASM の初期化パラメータファイルにありません。  解決方法: ASM の初期化パラメータファイルの中で <code>asm_diskstring</code> パラメータに対して正しい値を指定します。
エージェントがリソースの状態を識別できません。(Agent unable to identify state of the resource.)	ASMDG エージェントがリソースの状態を識別できません。  解決方法: テクニカルサポートまでご連絡ください。

## VCS 環境における Oracle 固有の問題のトラブルシューティング

表 7-6 に、VCS 環境で発生する可能性がある Oracle の問題の一覧を示します。

表 7-6 Oracle 共通の問題

メッセージ	説明と解決方法
Solaris x64 の場合  Oracle データベースインスタンスが一定の短い間隔で終了する。	解決方法: Oracle は、データベース初期化パラメータの <code>db_cache_size</code> と <code>java_pool_size</code> の値を最適な値に設定することを推奨します。
ORA-15097: RDBMS インスタンスが接続されている ASM インスタンスは停止できません。(ORA-15097: Cannot SHUTDOWN ASM instance with connected RDBMS instance.)	このメッセージは、ASM で管理されたデータベースに関して、Oracle サービスグループをオンラインにした 2 ~ 3 分以内に、今度はオフラインにしようとする则表示されます。  詳しくは、Oracle のバグ 5045309 を参照してください。

## Oracle 診断バイナリと Oracle のインスタンスの意図的なオフライン化の検証

ここでは、Oracle インスタンスの状態、Oracle 診断バイナリ、Oracle エージェントの意図的なオフライン化の動作を検証する手順について説明します。

**メモ:** 次の表に示す手順は、Owner 属性で指定されたオペレーティングシステムユーザーが実行します。

表 7-7 では、Oracle 診断バイナリで検証可能なチェック項目の一覧を示します。

表 7-7 Oracle 診断オプション

検証内容	ソリューション
ORACLE_HOME 変数が設定されている場合	次のコマンドを実行して、 <b>\$ORACLE_HOME</b> 変数が正しく設定されていることを検証します。  <pre># echo \$ORACLE_HOME</pre> ORACLE_HOME 環境変数が設定されていない場合は設定します。たとえば、次のコマンドを実行して、ORACLE_HOME 変数を設定します。  <pre># export ORACLE_HOME=/u01/oraHome</pre> ここで、/u01/oraHome は Oracle ホームディレクトリパスです。
Oracle インスタンスの SID が設定されている場合	次のコマンドを実行して、 <b>\$ORACLE_SID</b> 変数が正しく設定されていることを検証します。  <pre># echo \$ORACLE_SID</pre> 必要な場合は、# export ORACLE_SID=db コマンドを実行して <b>\$ORACLE_SID</b> 変数を設定します。
ライブラリパスが設定されている場合	次のコマンドを実行して、 <b>\$LD_LIBRARY_PATH</b> 変数が正しく設定されていることを検証します。  <pre># echo \$LD_LIBRARY_PATH</pre> 必要な場合は、# export LD_LIBRARY_PATH=\$ORACLE_HOME/lib:\$LD_LIBRARY_PATH コマンドを実行して <b>\$LD_LIBRARY_PATH</b> 変数を設定します。

検証内容	ソリューション
<p>Oracle インスタンスがオンラインの場合</p>	<p>次のコマンドを実行して、Oracle 診断バイナリが状態を正しく報告することを検証します。</p> <pre># ./oraapi_&lt;Arch&gt;_&lt;Oracle_FullVersion&gt;</pre> <p>&lt;Arch&gt;_&lt;Oracle_FullVersion&gt; はシステムアーキテクチャと Oracle の完全なバージョンです。</p> <p>たとえば、# ./oraapi_64_121010 の oraapi_64_121010 は、64 ビットシステムの Oracle バージョン 12 のバイナリビルドです。</p> <p>システムによって次のメッセージが表示されます。</p> <pre>Instance is online</pre> <p># echo \$? を実行します。</p> <p>システムにより 110 が表示された場合、インスタンスはオンラインでアクティブです。</p>
<p>Oracle インスタンスがオフラインの場合</p>	<p>次のコマンドを実行して、Oracle 診断バイナリが状態を正しく報告することを検証します。</p> <pre># ./oraapi_&lt;Arch&gt;_&lt;Oracle_FullVersion&gt;</pre> <p>&lt;Arch&gt;_&lt;Oracle_FullVersion&gt; はシステムアーキテクチャと Oracle の完全なバージョンです。</p> <p>たとえば、# ./oraapi_64_121010 の oraapi_64_121010 は、64 ビットシステムの Oracle バージョン 12 のバイナリビルドです。</p> <p>システムによって次のメッセージが表示されます。</p> <pre>Failure: Instance Shutdown          class: SOFT</pre> <p>Proper Shutdown</p> <p># echo \$? を実行します。</p> <p>システムにより 100 が表示された場合、インスタンスはオフラインです。</p>

検証内容	ソリューション
Oracle インスタンスが異常終了した場合	重要な Oracle プロセス(ora_pmon_<sid>、ora_smon_<sid> など)のいずれかが強制終了した場合、Oracle インスタンスが異常終了します。このような場合、Oracle 診断バイナリを実行すると、システムによって次のメッセージが表示されます。  たとえば、# ./oraapi_64_121010 を実行した場合  <pre>Failure: Abnormal Termination class: SOFT Abnormal termination  # echo \$? を実行します。</pre> システムにより 98 が表示された場合、Oracle の終了が処理されます。Oracle の終了が異常で意図的ではないため、VCS リソースがフェールオーバーします。

## VCS Oracle エージェントの意図的なオフライン動作の検証

次の手順を実行して、VCS Oracle エージェントがリソースの段階的な終了を意図的なオフライン化の動作として判別するかどうかを検証します。

- 1 リソースをオンラインにします。  

```
# hares -online ORA_oraprod -sys system
```

 ここで、**system** は Oracle データベースがインストールされているシステムの名前です。
- 2 IntentionalOffline と MonitorOption 属性が 1 に設定されていることを確認します。

- 3 VCS の制御外で Oracle インスタンスを適切に停止します。

```
# su - oracle
# bash
# bash-3.00$ sqlplus "/ as sysdba"

The system displays the following message:
SQL> shutdown immediate
Database closed.
Database dismounted.
ORA-03113: end-of-file on communication channel
```

- 4 ログファイルを検証して、シャットダウンが適切に行われたことを確認できます。

```
Sample log message:
VCS INFO V-16-1-13470 Resource ORA_oraprod
(Owner: Unspecified, Group: ORA_PROD_Group) is offline on system.

(Intentional But NOT initiated by VCS)
```

Oracle エージェントによってリソースの意図的なオフライン化が認識されます。

## PDB リソースの IMF の無効化

PDB リソースの IMF を無効にするには次の手順を実行します。

- 1 特定の PDB リソース IMF 属性を無効にします。

```
# hares -override <pdb_resname> IMF
```

- 2 次のコマンドを使用して IMF 属性が無効にされたか確認します。

```
# hares -value pdb1 IMF
Mode      3      MonitorFreq      5      RegisterRetryLimit
3
```

**3** モード値を 0 に修正します

```
# hares -modify pdb1 IMF -update Mode 0
```

**4** 次のコマンドを使用してモード値を確認します。

```
# hares -value pdb1 IMF
Mode      0      MonitorFreq      5      RegisterRetryLimit
3
```

# リソースタイプの定義

この付録では以下の項目について説明しています。

- [リソースタイプの定義と属性の定義について](#)
- [Oracle エージェントのリソースタイプ定義](#)
- [Netlsnr エージェントのリソースタイプ定義](#)
- [ASMinst エージェントのリソースタイプ定義](#)
- [ASMDG エージェントのリソースタイプ定義](#)

## 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 エージェントのリソースタイプ定義

Cluster Server agent for Oracle の Oracle エージェントは、VCS では Oracle リソースタイプとして表されます。

```
type Oracle (  
    static str AgentDirectory = "/opt/VRTSagents/ha/bin/Oracle"  
    static keylist SupportedActions = { VRTS_GetInstanceName,  
        VRTS_GetRunningServices, DBRestrict, DBUndoRestrict,  
        DBResume, DBSuspend, DBTbspBackup, PDBMigrate_PreCheck,  
        PlugPDB, "home.vfd", "owner.vfd", "getid", "pfile.vfd" }  
    static str ArgList[] = { Sid, Owner, Home, Pfile, StartUpOpt,
```

```
    ShutDownOpt, DBAUser, DBAPword, EnvFile, AutoEndBkup,  
    User, Pword, Table, MonScript, Encoding, MonitorOption,  
    DBName, ManagedBy, PDBName }  
static int IMF{} = { Mode=3, MonitorFreq=5, RegisterRetryLimit=3 }  
static str IMFRegList[] = { Home, Owner, Sid, MonitorOption }  
str Sid  
str Owner  
str Home  
str Pfile  
str StartUpOpt = STARTUP_FORCE  
str ShutDownOpt = IMMEDIATE  
str DBName  
str ManagedBy = "ADMIN"  
str DBAUser  
str DBAPword  
str EnvFile  
boolean AutoEndBkup = 1  
str MonScript = "./bin/Oracle/SqlTest.pl"  
str User  
str Pword  
str Table  
str Encoding  
int MonitorOption = 0  
static boolean IntentionalOffline = 0  
static boolean AEPTIMEOUT = 1  
static int OnlineWaitLimit = 5  
str PDBName  
)
```

## Oracle エージェントの属性の定義

Oracle エージェントの属性は、必須、オプション、内部に分類されます。

次の必須属性には、値を割り当てる必要があります。

**表 A-1 Oracle エージェントの必須属性**

必須属性	データ形式と値のタイプ	定義
SID	文字列 - スカラー	<p>Oracle インスタンスを表す \$ORACLE_SID 変数。Oracle エージェントと Oracle データベースサーバーでは Sid の大文字小文字が区別されます。</p> <p>ポリシー管理データベースでは、Sid の属性は Sid プレフィックスに設定する必要があります。p.127 の「<a href="#">ポリシー管理データベースの Sid 属性について</a>」を参照してください。</p>
Owner	文字列 - スカラー	<p>データベースインスタンスを開始または停止する権限がある Oracle ユーザー。エージェントでは、Oracle ユーザーとして LDAP ユーザーもサポートされます。</p>
Home	文字列 - スカラー	<p>Oracle バイナリファイルと設定ファイルへの \$ORACLE_HOME パス。たとえば、パスとして /opt/ora_home を指定できます。</p> <p><b>メモ:</b> パスの最後にスラッシュ (/) は付けしないでください。</p>

次のオプションの属性は、設定に必要な場合に設定できます。

**表 A-2 Oracle エージェントのオプションの属性**

オプション属性	データ形式と値のタイプ	定義
ContainerOpts	静的 - 関連 - 整数	<p>このリソースタイプレベルの属性は、Solaris コンテナ (ゾーンまたはプロジェクト) 環境で実行される Oracle インスタンスのコンテナオプションを指定します。</p> <p>以下のキーの値は、ContainerInfo サービスグループ属性を設定している場合にのみ有効です。Veritas では、以下の値は変更しないことをお勧めします。</p> <ul style="list-style-type: none"> <li>■ RunInContainer (RIC)                     <ul style="list-style-type: none"> <li>1 - Oracle エージェントはローカルコンテナ内部で動作する Oracle インスタンスを監視します。</li> <li>0 - Oracle エージェントはローカルコンテナ外部 (グローバル環境) で動作する Oracle インスタンスを監視します。</li> </ul>                     デフォルトは 1 です。                 </li> <li>■ PassCInfo (PCI)                     <ul style="list-style-type: none"> <li>Oracle エージェントは、サービスグループの ContainerInfo 属性で定義されたコンテナ情報を受け取ります。</li> </ul>                     デフォルトは 0 です。                 </li> </ul> <p>『Cluster Server 管理者ガイド』と『Veritas InfoScale 8.0 仮想化ガイド』を参照してください。</p>

オプション属性	データ形式と値のタイプ	定義
DBAUser	文字列 - スカラー	データベースを開始または停止する <b>sysdba</b> 権限があるデータベースユーザー。
DBAPword	文字列 - スカラー	<p>DBAUser 用の暗号化されたパスワード。</p> <p>コマンドラインで入力する場合にのみパスワードを暗号化します。パスワードは、<b>VCS Encrypt</b> ユーティリティ (<code>/opt/VRTSvcs/bin/vcsencrypt</code>) を使って暗号化する必要があります。</p> <p>p.84 の「<b>Oracle</b> データベースのユーザーパスワードとリスナーパスワードの暗号化」を参照してください。</p>
StartUpOpt	文字列 - スカラー	<p><b>Oracle</b> インスタンスの起動オプション。この属性は従来のデータベースおよびコンテナデータベースに対しては次の値をとります。</p> <ul style="list-style-type: none"> <li>■ STARTUP</li> <li>■ STARTUP_FORCE</li> <li>■ RESTRICTED</li> <li>■ RECOVERDB</li> <li>■ SRVCTLSTART</li> <li>■ CUSTOM</li> <li>■ SRVCTLSTART_FORCE</li> </ul> <p>デフォルトは <b>STARTUP_FORCE</b> です。</p> <p>p.15 の「<b>Oracle</b> エージェントの起動オプションと停止オプション」を参照してください。</p> <p>プラグ可能データベース内の <b>Oracle</b> 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。</p> <ul style="list-style-type: none"> <li>■ STARTUP</li> <li>■ STARTUP_FORCE</li> <li>■ RESTRICTED</li> <li>■ OPEN_RO</li> </ul> <p>p.18 の「<b>PDB</b> の起動オプションと停止オプション」を参照してください。</p>

オプション属性	データ形式と値のタイプ	定義
ShutDownOpt	文字列 - スカラー	<p><b>Oracle</b> インスタンスの停止オプション。この属性は従来のデータベースおよびコンテナデータベースに対しては次の値をとります。</p> <ul style="list-style-type: none"> <li>■ IMMEDIATE</li> <li>■ TRANSACTIONAL</li> <li>■ SRVCTLSTOP</li> <li>■ CUSTOM</li> <li>■ SRVCTLSTOP_TRANSACT</li> <li>■ SRVCTLSTOP_ABORT</li> <li>■ SRVCTLSTOP_IMMEDIATE</li> </ul> <p>デフォルトは IMMEDIATE です。</p> <p>p.15 の「<a href="#">Oracle エージェントの起動オプションと停止オプション</a>」を参照してください。</p> <p>プラグ可能データベース内の <b>Oracle</b> インスタンスのシャットダウンオプションは IMMEDIATE です。</p> <p>p.18 の「<a href="#">PDB の起動オプションと停止オプション</a>」を参照してください。</p>
EnvFile	文字列 - スカラー	<p>エントリポイントスクリプトによって提供されるファイルの絶対パス名。このファイルには、<b>Oracle</b> データベースサーバー環境の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>ファイル内容の構文は、<b>Owner</b> のログインシェルによって異なります。ファイルは <b>Owner</b> が読み取ることができなければなりません。ファイルにユーザー入力を求めるプロンプトが含まれてはいけません。</p>
Pfile	文字列 - スカラー	<p>起動プロファイルの絶対パスを指定した初期化パラメータの名前。</p> <p>また、サーバーパラメータファイルを使用することもできます。SPFILE パラメータのみを含むオンラインのテキスト初期化パラメータファイルを作成します。詳しくは、<b>Oracle</b> のマニュアルを参照してください。</p> <p><b>メモ:</b> この属性は従来のデータベースおよびコンテナデータベースにのみ適用されます。</p> <p>p.199 の「<a href="#">VCS clusterでの SPFILE の使用</a>」を参照してください。</p>

オプション属性	データ形式と値のタイプ	定義
AutoEndBkup	ブール - スカラー	<p><b>AutoEndBkup</b> 属性をゼロ以外の値に設定すると、オンライン時に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ます。</p> <p>デフォルト: 1</p> <p><b>メモ:</b> コンテナデータベースまたはプラグ可能データベースのホットバックアップ中にノードに障害が発生した場合は、対応する CDB リソースの <b>AutoEndBkup</b> 属性を 1 に設定する必要があります。CDB の <b>AutoEndBkup</b> が 1 に設定されている場合、オンライン時に CDB と PDB の両方のバックアップも終了します。</p> <p>p.37 の「<a href="#">Oracle のインストールのための VCS の必要条件について</a>」を参照してください。</p> <p>p.43 の「<a href="#">ホットバックアップ中に VCS nodeに障害が発生した後の Oracle のフェールオーバー</a>」を参照してください。</p>
MonitorOption	整数 - スカラー	<p><b>Oracle</b> インスタンスの監視オプション。この属性には 0 または 1 を指定できます。</p> <p>従来のデータベースおよびコンテナデータベースの場合:</p> <ul style="list-style-type: none"> <li>■ 0 - プロセスチェック監視 (推奨)</li> <li>■ 1 - 診断監視</li> </ul> <p>プラグ可能データベースの場合:</p> <ul style="list-style-type: none"> <li>■ 0 - 基本監視</li> </ul> <p>エージェントの意図的なオフライン機能を使うには、この属性の値を 1 に設定する必要があります。</p> <p>デフォルト: 0</p> <p>この属性値を 0 に設定した場合にのみ、エージェントは従来の DB および CDB に対するインテリジェントなリソース監視をサポートします。</p> <p>p.21 の「<a href="#">従来のデータベースおよびコンテナデータベース内の Oracle エージェントに対する監視オプション</a>」を参照してください。</p> <p>p.22 の「<a href="#">プラグ可能なデータベースの監視</a>」を参照してください。</p>

オプション属性	データ形式と値のタイプ	定義
IMF	整数 - 関連付け	

オプション属性	データ形式と値のタイプ	定義
		<p>このリソースタイプレベルの属性は、<b>Oracle</b> エージェントがインテリジェントなリソース監視を実行する必要があるかどうかを決定します。リソースレベルでこの属性の値を上書きすることもできます。</p> <p>この属性は次のキーを含んでいます。</p> <ul style="list-style-type: none"> <li>■ <b>Mode:</b> この属性を定義して、インテリジェントなリソースの監視を有効または無効にします。</li> </ul> <p>次に、有効な値を示します。</p> <ul style="list-style-type: none"> <li>■ <b>0</b> - インテリジェントなリソースの監視を実行しない</li> <li>■ <b>1</b> - オフラインリソースについてはインテリジェントなリソースの監視を実行し、オンラインリソースについてはポーリングベースの監視を実行する</li> <li>■ <b>2</b> - オンラインリソースについてはインテリジェントなリソースの監視を実行し、オフラインリソースについてはポーリングベースの監視を実行する</li> <li>■ <b>3</b> - オンラインリソースとオフラインリソースの両方についてインテリジェントなリソースの監視を実行する</li> </ul> <p>デフォルト: <b>3</b></p> <ul style="list-style-type: none"> <li>■ <b>MonitorFreq:</b> このキー値は、エージェントが監視エージェント機能呼び出す頻度を指定します。このキーの値は整数です。</li> </ul> <p>デフォルト: <b>5</b></p> <p>エージェントがポーリングベースの監視とインテリジェントなリソースの監視の両方を実行する必要がある場合は、このキーをゼロ以外の値に設定できます。値が<b>0</b>の場合、エージェントはポーリングベースのプロセスチェック監視を実行しません。</p> <p>リソースが<b>AMF</b> カーネルドライバに登録されると、エージェントは次のように監視エージェント機能呼び出します。</p> <ul style="list-style-type: none"> <li>■ オンラインリソースの場合は <b>(MonitorFreq x MonitorInterval)</b> 秒おき</li> <li>■ オフラインリソースの場合は <b>(MonitorFreq x OfflineMonitorInterval)</b> 秒おき</li> </ul> <ul style="list-style-type: none"> <li>■ <b>RegisterRetryLimit:</b> インテリジェントなリソース監視を有効にすると、エージェントは <b>oracle_imf_register</b> エージェント関数を呼び出して、リソースを <b>AMF</b> カーネルドライバに登録します。</li> </ul> <p><b>RegisterRetryLimit</b> キーの値によって、エージェントがリソースの登録を再試行する必要がある回数が決まります。エージェントが、指定された制限内にリソースを登録できない場合、リソースの状態が変化するか、<b>Mode</b> キーの値が変更されるまで、インテリジェントな監視は無効になります。</p> <p>デフォルト: <b>3</b></p> <p><b>メモ:</b> <b>IMF</b> は従来のデータベースおよびコンテナデータベースでのみ</p>

オプション属性	データ形式と値のタイプ	定義
		<p>サポートされます。</p> <p>p.90 の「エージェントに対するインテリジェントなリソース監視の手動による有効化と無効化」を参照してください。</p>
MonScript	文字列 - スカラー	<p>詳細な監視のために提供されているスクリプトへのパス名。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。</p> <p><b>メモ:</b> MonScript 属性の値が無効であったり、空の文字列を設定している場合、詳細監視は無効になります。</p> <p>提供されている詳細監視スクリプトへのパス名は <code>/opt/VRTSagents/ha/bin/Oracle/SqlTest.pl</code> です。</p> <p>また、MonScript には <code>/opt/VRTSagents/ha</code> への相対パス名も指定できます。相対パス名は、パス <code>./bin/Oracle/SqlTest.pl</code> のように「./」で始まる必要があります。</p>
User	文字列 - スカラー	<p>内部データベースユーザー。詳細監視のためにデータベースに接続します。</p>
LevelTwoMonitorFreq	整数 - スカラー	<p>このリソースタイプのエージェントが第 2 レベル、つまり詳細な監視を実行する必要がある頻度を指定します。リソースレベルでこの属性の値を上書きすることもできます。</p> <p>この値は、エージェントが <b>Oracle</b> を詳細に監視する監視サイクルの数を示します。たとえば、値 5 は、エージェントが 5 つのオンライン監視間隔ごとに <b>Apache</b> を詳細に監視することを意味します。</p> <p>VCS 8.0 エージェントに手動でアップグレードし、以前のバージョンの詳細監視を有効にしている場合は、LevelTwoMonitorFreq 属性の値を適切に設定します。</p> <p><b>メモ:</b> AutoEndBkup 属性の値を 0 に設定した場合は、LevelTwoMonitorFreq 属性の値が詳細な監視用の 1 になっていることを確認します。</p> <p>デフォルト: 0</p>

オプション属性	データ形式と値のタイプ	定義
Pword	文字列 - スカラー	<p>内部データベースユーザー認証のための暗号化されたパスワード。</p> <p>コマンドラインで入力する場合にのみパスワードを暗号化します。パスワードは、<b>VCS Encrypt</b> ユーティリティ (/opt/VRTSvcs/bin/vcscrypt) を使って暗号化する必要があります。</p> <p><b>メモ:</b> この属性は PDB リソースには適用されません。</p> <p>p.84 の「<b>Oracle</b> データベースのユーザーパスワードとリスナーパスワードの暗号化」を参照してください。</p>
Table	文字列 - スカラー	<p>User/Pword による更新用の表。</p>
Encoding	文字列 - スカラー	<p>表示される <b>Oracle</b> の出力の <b>Oracle</b> 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。</p> <p>たとえば、<b>Oracle</b> の出力が「JAPANESE_JAPAN.JA16EUC」の場合、<b>Encoding</b> の <b>Solaris</b> 値は「eucJP」になります。各エンコーディング値については、<b>Oracle</b> と <b>Solaris</b> のマニュアルを参照してください。</p> <p>デフォルトは "" です。</p> <p><b>メモ:</b> この属性は PDB リソースには適用されません。</p>
IntentionalOffline		<p>このリソースタイプレベルの属性は、<b>Oracle</b> が <b>VCS</b> 制御の外側で意図的に停止されている場合に、<b>VCS</b> が反応する方法を定義します。</p> <p><b>VCS</b> の制御下でない <b>Oracle</b> の従来型データベースおよびコンテナデータベースを停止すると、エージェントの動作は次のようになります。</p> <ul style="list-style-type: none"> <li>■ 0 - <b>Oracle</b> エージェントは障害を登録して、サービスグループのフェールオーバーを開始します。</li> <li>■ 1 - 診断監視が有効になっている場合、<b>Oracle</b> エージェントは <b>Oracle</b> リソースをオフラインで取得します。          診断監視が有効になっていない場合、エージェントは障害を登録して、サービスグループのフェールオーバーを開始します。</li> </ul> <p><b>メモ:</b> エージェントの意図的なオフラインの機能を使う場合は、<b>MonitorOption</b> 属性の値に 1 を設定して、<b>CDB</b> で診断監視を有効にする必要があります。</p> <p><b>メモ:</b> <b>CDB</b> リソースが <b>IntentionalOffline</b> によって設定されていて、この <b>CDB</b> が <b>VCS</b> の制御外で停止した場合、<b>PDB</b> リソースもオフラインとして報告されます。この機能を有効にするには、<b>PDB</b> リソースの <b>IntentionalOffline</b> 属性を 1 に設定します。</p> <p>『Cluster Server Administrator's Guide』を参照してください。</p>

オプション属性	データ形式と値のタイプ	定義
DBName	文字列 - スカラー	この属性は、データベースがポリシー管理の RAC データベースの場合にのみ設定します。この属性の値は一意のデータベース名に設定する必要があります。
ManagedBy	文字列 - スカラー	この属性のデフォルト値は ADMIN です。ポリシー管理の RAC データベースでは、この属性は POLICY に設定する必要があります。
PDBName	文字列 - スカラー	PDB にこの属性を設定する必要があります。PDB データベース名に値を設定する必要があります。従来のデータベースおよび CDB データベースの場合、この属性を設定しないでください。

表 A-3は、Oracle エージェントの内部属性をリストします。この属性は内部使用のみを目的としています。Veritas は、この属性の値を変更しないよう推奨します。

**表 A-3** Oracle エージェントの内部属性

オプション属性	データ形式と値のタイプ	定義
AgentDirectory	静的 - 文字列	Oracle エージェントに関連付けられているバイナリやスクリプトなど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。

## ポリシー管理データベースの Sid 属性について

SID 属性は必須属性です。ここでは、ポリシー管理データベースの SID 属性を定義するための情報を提供します。

SID プレフィックスは、一意のデータベース名の最初の 8 文字の英数字で構成されません。a - z の英字(大文字と小文字)と、0 - 9 の数字の組み合わせになります。

SID プレフィックスには、オペレーティングシステムの特許文字は使えません。このため、一意のデータベース名の最初の 8 文字には、特許文字を使わないようにします。最初の 8 文字に特許文字が使われている場合、それらの文字は省略されます。各データベースには、単一の SID プレフィックスがあります。データベースの SID プレフィックスはクラスター内で一意である必要があります。

Oracle RAC データベースでは、各インスタンスに一意の識別子 ORACLE\_SID があります。これは、SID プレフィックスとインスタンス番号で構成されます。Oracle RAC データベースの ORACLE\_SID の生成方法は、データベースの管理方法をどのように選択するかによって異なります。ポリシー管理データベースを選択した場合、Oracle は name\_# の形式で SID を生成します。この name は DB\_UNIQUE\_NAME の最初の 8 文字の英数字で、# はインスタンス番号を表します。管理者管理データベースを選択した場合、

DBCA はインスタンス名に対する SID を先に生成します。この SID は `name#` の形式です。

SID プレフィックス名を確認するには、次のコマンドを実行します。

```
# ${GRID_HOME}/bin/crsctl status resource ora.${DBName}.db -f | grep  
GEN_USR_ORA_INST_NAME@ | tail -1 | sed 's/.*/' | sed 's/_[0-9]$/'
```

`GRID_HOME` はグリッドのホームへのパスで、`DBName` は一意のデータベース名です。

---

**メモ:** ポリシー管理データベースが作成されると、インストール手順の確認ページに SID プレフィックスが表示されます。

---

p.118 の「[Oracle エージェントの属性の定義](#)」を参照してください。

## Netlsnr エージェントのリソースタイプ定義

Cluster Server agent for Oracle の Netlsnr エージェントは、VCS では Netlsnr リソースタイプとして表されます。

```
type Netlsnr (  
    static str AgentDirectory = "/opt/VRTSagents/ha/bin/Netlsnr"  
    static keylist SupportedActions = { VRTS_GetInstanceName,  
        VRTS_GetRunningServices, "tnsadmin.vfd" }  
    static str ArgList[] = { Owner, Home, TnsAdmin, Listener,  
        EnvFile, MonScript, LsnrPwd, Encoding }  
    static int IMF{} = { Mode=3, MonitorFreq=5, RegisterRetryLimit=3 }  
    static str IMFRegList[] = { Home, Owner, Listener }  
    str Owner  
    str Home  
    str TnsAdmin  
    str Listener = "LISTENER"  
    str EnvFile  
    str MonScript = "./bin/Netlsnr/LsnrTest.pl"  
    str LsnrPwd  
    str Encoding  
    static boolean IntentionalOffline = 0  
)
```

## Netlsnr エージェントの属性定義

Netlsnr エージェント属性の説明を確認します。エージェント属性は、必須、オプション、内部に分類されます。

表 A-4 には、Netlsnr エージェントの必須属性がリストされています。必須属性には必ず値を割り当てる必要があります。

表 A-4 Netlsnr エージェントの必須属性

必須属性	データ形式と値のタイプ	定義
Owner	文字列 - スカラー	リスナープロセスを開始または停止する権限がある <b>Oracle</b> ユーザー。 エージェントでは、 <b>Oracle</b> ユーザーとして LDAP ユーザーもサポートされます。
Home	文字列 - スカラー	<b>Oracle</b> バイナリファイルと設定ファイルへの \$ORACLE_HOME パス。たとえば、パスとして /opt/ora_home を指定できます。 パスの最後にスラッシュ (/) は付けしないでください。

表 A-5 には、Netlsnr エージェントの必須属性がリストされています。オプションの属性は必要に応じて設定できます。

表 A-5 Netlsnr エージェントのオプションの属性

オプション属性	データ形式と値のタイプ	定義
ContainerOpts	静的 - 関連 - 整数	このリソースタイプレベルの属性は、 <b>Solaris</b> コンテナ (ゾーンまたはプロジェクト) 環境で実行される <b>Netlsnr</b> インスタンスのコンテナオプションを指定します。 これらの値はサービスグループの <b>ContainerInfo</b> 属性が設定されている場合にのみ有効です。 <b>Veritas</b> では、以下の値は変更しないことをお勧めします。 <ul style="list-style-type: none"> <li>■ <b>RunInContainer (RIC)</b>  <b>1</b> - Netlsnr エージェントはローカルコンテナ内部で動作するリスナープロセスを監視します。  <b>0</b> - Netlsnr エージェントはローカルコンテナ外部 (グローバル環境) で動作するリスナープロセスを監視します。                      デフォルトは <b>1</b> です。</li> <li>■ <b>PassCInfo (PCI)</b>                      Netlsnr エージェントは、サービスグループの <b>ContainerInfo</b> 属性で定義されたコンテナ情報を受け取ります。                      デフォルトは <b>0</b> です。</li> </ul> 『Cluster Server 管理者ガイド』と『Veritas InfoScale 8.0 Solutions 仮想化ガイド』を参照してください。

オプション属性	データ形式と値のタイプ	定義
TnsAdmin	文字列 - スカラー	<p>Listener 設定ファイルが存在するディレクトリへの \$TNS_ADMIN パス (listener.ora)。</p> <p>デフォルトは /var/opt/oracle です。</p>
Listener	文字列 - スカラー	<p>Listener の名前。Netlsnr エージェントと Oracle データベースサーバーでは、Listener の名前の大文字小文字は区別されません。</p> <p>デフォルトは LISTENER です。</p> <p><b>メモ:</b> グリッドインフラストラクチャが設定されている場合は、データベースリスナーにデフォルト名を使用しないようにしてください。そうしないと、グリッドインフラストラクチャがデフォルト名のリスナー (LISTENER) を起動し、さまざまな問題が発生する可能性があります。</p>
LsnrPwd	文字列 - スカラー	<p>リスナーの停止と監視に使用される VCS の暗号化されたパスワード。このパスワードは Listener 設定ファイルで設定されます。</p> <p>コマンドラインで入力する場合にのみパスワードを暗号化します。パスワードは VCS Encrypt ユーティリティを使用して暗号化する必要があります。</p> <p>p.84 の「Oracle データベースのユーザーパスワードとリスナーパスワードの暗号化」を参照してください。</p>
EnvFile	文字列 - スカラー	<p>エントリポイントスクリプトによって提供されるファイルの絶対パス名を指定します。このファイルには、Oracle リスナー環境の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。</p> <p>ファイル内容の構文は、Owner のログインシェルによって異なります。このファイルは Owner が読み取ることができる必要があります。ファイルにユーザー入力を求めるプロンプトが含まれてはいけません。</p>

オプション属性	データ形式と値のタイプ	定義
IMF	整数 - 関連付け	<p>このリソースタイプレベルの属性は、<b>Netlsnr</b> エージェントがインテリジェントなリソース監視を実行する必要があるかどうかを決定します。リソースレベルでこの属性の値を上書きすることもできます。</p> <p>この属性は次のキーを含んでいます。</p> <ul style="list-style-type: none"> <li>■ <b>Mode:</b> この属性を定義して、インテリジェントなリソースの監視を有効または無効にします。</li> </ul> <p>次に、有効な値を示します。</p> <ul style="list-style-type: none"> <li>■ <b>0</b> - インテリジェントなリソースの監視を実行しない</li> <li>■ <b>1</b> - オフラインリソースについてはインテリジェントなリソースの監視を実行し、オンラインリソースについてはポーリングベースの監視を実行する</li> <li>■ <b>2</b> - オンラインリソースについてはインテリジェントなリソースの監視を実行し、オフラインリソースについてはポーリングベースの監視を実行する</li> <li>■ <b>3</b> - オンラインリソースとオフラインリソースの両方についてインテリジェントなリソースの監視を実行する</li> </ul> <p>デフォルト: <b>3</b></p> <ul style="list-style-type: none"> <li>■ <b>MonitorFreq:</b> このキー値は、エージェントが監視エージェント機能呼び出す頻度を指定します。このキーの値は整数です。</li> </ul> <p>デフォルト: <b>5</b></p> <p>インテリジェントなリソース監視に加えてポーリングベースのリソース監視をエージェントが実行する必要がある場合、この属性をゼロ以外の値に設定できます。</p> <p>リソースが <b>AMF</b> カーネルドライバに登録されると、エージェントは次のように監視エージェント機能呼び出します。</p> <ul style="list-style-type: none"> <li>■ オンラインリソースの場合は (<b>MonitorFreq x MonitorInterval</b>) 秒おき</li> <li>■ オフラインリソースの場合は (<b>MonitorFreq x OfflineMonitorInterval</b>) 秒おき</li> </ul> <ul style="list-style-type: none"> <li>■ <b>RegisterRetryLimit:</b> インテリジェントなリソースの監視を有効にした場合、エージェントは <b>inetsnr_imf_register</b> エージェント機能呼び出し、リソースを <b>AMF</b> カーネルドライバに登録します。<b>RegisterRetryLimit</b> キーの値によって、エージェントがリソースの登録を再試行する必要がある回数が決まります。エージェントが、指定された制限内にリソースを登録できない場合、リソースの状態が変化するか、<b>Mode</b> キーの値が変更されるまで、インテリジェントな監視は無効になります。</li> </ul> <p>デフォルト: <b>3</b></p> <p>p.90 の「エージェントに対するインテリジェントなリソース監視の手動による有効化と無効化」を参照してください。</p>

オプション属性	データ形式と値のタイプ	定義
MonScript	文字列 - スカラー	<p>詳細な監視のために提供されているスクリプトへのパス名。デフォルトでは、リスナープロセスを監視する詳細監視は有効になっています。</p> <p><b>メモ:</b> 属性 <b>MonScript</b> の値が空の文字列に設定されている場合、エージェントは詳細監視を無効にします。</p> <p>提供されている詳細監視スクリプトへのパス名は <code>/opt/VRTSagents/ha/bin/Netlsnr/LsnrTest.pl</code> です。</p> <p>また、<b>MonScript</b> には <code>/opt/VRTSagents/ha</code> への相対パス名も指定できます。相対パス名は、パス <code>./bin/Netlsnr/LsnrTest.pl</code> のように「./」で始まる必要があります。</p>
LevelTwoMonitorFreq	整数 - スカラー	<p>このリソースタイプのエージェントが第 2 レベル、つまり詳細な監視を実行する必要がある頻度を指定します。</p> <p>詳細監視を有効にした場合、<b>LevelTwoMonitorFreq</b> 属性の値を設定してください。</p> <p>デフォルト: 0</p>
Encoding	文字列 - スカラー	<p>表示される <b>Oracle</b> の出力の <b>Oracle</b> 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。</p> <p>たとえば、<b>Oracle</b> の出力が「JAPANESE_JAPAN.JA16EUC」の場合、<b>Encoding</b> の <b>Solaris</b> 値は「eucJP」になります。各エンコーディング値については、<b>Oracle</b> と <b>Solaris</b> のマニュアルを参照してください。</p> <p>デフォルトは "" です。</p>
IntentionalOffline		<p>将来使うための属性です。</p> <p>この属性の値は変更しないでください。</p> <p>デフォルト: 0</p>

表 A-6 には、**Netlsnr** エージェントの内部属性がリストされています。この属性は内部使用のみを目的としています。**Veritas** は、この属性の値を変更しないよう推奨します。

表 A-6 Netlsnr エージェントの内部属性

オプション属性	データ形式と値のタイプ	定義
AgentDirectory	静的 - 文字列	<p><b>Netlsnr</b> エージェントに関連付けられているバイナリやスクリプトなどのファイルの場所を指定します。</p> <p>デフォルトは <code>/opt/VRTSagents/ha/bin/Netlsnr</code> です。</p>

## ASMinst エージェントのリソースタイプ定義

Cluster Server agent for Oracle の ASMinst エージェントは、VCS では ASMinst リソースタイプとして表されます。

```
type ASMinst (
    static str AgentDirectory = "/opt/VRTSagents/ha/bin/ASMinst"
    static str ArgList[] = { Sid, Owner, Home, DBAUser,
        DBAPword, Pfile, StartUpOpt, ShutDownOpt,
        EnvFile, Encoding, MonitorOption }
    static int ContainerOpts{} = { RunInContainer=1, PassCInfo=0 }
    str Sid
    str Owner
    str Home
    str DBAUser
    str DBAPword
    str Pfile
    str StartUpOpt
    str ShutDownOpt
    str EnvFile
    str Encoding
    int MonitorOption = 0
)
```

## ASMinst エージェントの属性定義

ASMinst エージェントの属性の説明を確認します。エージェント属性は、必須、オプション、内部に分類されます。

表 A-7 は、必要な属性をリストします。必須属性には必ず値を割り当てる必要があります。

表 A-7 ASMinst エージェントの必須属性

必須属性	データ形式と値のタイプ	定義
Sid	文字列 - スカラー	ASM インスタンスを表す \$ORACLE_SID 変数。ASMinst エージェントでは、Sid の大文字小文字が区別されます。
Owner	文字列 - スカラー	ASM インスタンスを開始または停止する権限がある Oracle ユーザー。エージェントでは、Oracle ユーザーとして LDAP ユーザーもサポートされます。

必須属性	データ形式と値のタイプ	定義
Home	文字列 - スカラー	Oracle ASM バイナリファイルと設定ファイルへの \$ORACLE_HOME パス。たとえば、パスとして /opt/ora_home を指定できます。 <b>メモ:</b> パスの最後にスラッシュ (/) は付けないでください。

表 A-8は、ASMinst エージェントのオプションの属性をリストします。オプションの属性は必要に応じて設定できます。

**表 A-8** ASMinst エージェントのオプションの属性

オプション属性	データ形式と値のタイプ	定義
ContainerOpts	静的 - 関連 - 整数	このリソースタイプレベルの属性は、Solaris コンテナ (ゾーン) 環境で実行される Oracle ASM インスタンスのコンテナオプションを指定します。 これらの値はサービスグループの ContainerInfo 属性が設定されている場合にのみ有効です。Veritas では、以下の値は変更しないことをお勧めします。 <ul style="list-style-type: none"><li>■ RunInContainer (RIC) 1 - ASMinst エージェントはローカルコンテナ内部で動作する Oracle ASM インスタンスを監視します。 0 - ASMinst エージェントはローカルコンテナ外部 (グローバル環境) で動作する Oracle ASM インスタンスを監視します。 デフォルトは 1 です。</li><li>■ PassCInfo (PCI) ASMinst エージェントは、サービスグループの ContainerInfo 属性で定義されたコンテナ情報を受け取ります。 デフォルトは 0 です。</li></ul> 『Cluster Server 管理者ガイド』と『Veritas InfoScale 8.0 仮想化ガイド』を参照してください。

オプション属性	データ形式と値のタイプ	定義
EnvFile	文字列 - スカラー	<p>エントリポイントスクリプトによって提供されるファイルの絶対パス名。このファイルには、<b>Oracle</b> データベースサーバー環境のユーザーが設定する <b>LD_LIBRARY_PATH</b>、<b>NLS_DATE_FORMAT</b> などの環境変数が含まれます。</p> <p>ファイル内容の構文は、<b>Owner</b> のログインシェルによって異なります。ファイルは <b>Owner</b> が読み取ることができなければなりません。ファイルにユーザー入力を求めるプロンプトが含まれてはいけません。</p>
Pfile	文字列 - スカラー	<p>起動プロファイルの絶対パスを指定した <b>ASM</b> インスタンスの初期化パラメータファイルの名前。</p> <p>また、サーバーパラメータファイルを使用することもできます。 <b>SPFILE</b> パラメータのみを含むオンラインのテキスト初期化パラメータファイルを作成します。詳しくは、<b>Oracle</b> のマニュアルを参照してください。</p> <p><a href="#">p.199 の「VCS clusterでの SPFILEの使用」</a> を参照してください。</p>
DBAUser	文字列 - スカラー	<p><b>ASM</b> インスタンスを開始または停止する <b>sysasm</b> 権限がある <b>ASM</b> ユーザー。</p>
DBAPword	文字列 - スカラー	<p><b>DBAUser</b> 用の暗号化されたパスワード。</p> <p>コマンドラインで入力する場合にのみパスワードを暗号化します。パスワードは <b>VCS Encrypt</b> ユーティリティを使用して暗号化する必要があります。</p> <p><a href="#">p.84 の「Oracle データベースのユーザーパスワードとリスナーパスワードの暗号化」</a> を参照してください。</p>

オプション属性	データ形式と値のタイプ	定義
Encoding	文字列 - スカラー	表示される Oracle の出力の Oracle 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。デフォルトは "" です。
StartUpOpt	文字列 - スカラー	<p>Oracle ASM インスタンスの起動オプション。エージェントは <code>sqlplus</code> コマンドを使って Oracle ASM インスタンスを起動します。</p> <p>この属性は次の値をとることができます。</p> <ul style="list-style-type: none"> <li>■ STARTUP</li> <li>■ STARTUP_MOUNT</li> <li>■ STARTUP_OPEN</li> <li>■ SRVCTLSTART</li> <li>■ SRVCTLSTART_MOUNT</li> <li>■ SRVCTLSTART_OPEN</li> </ul> <p>デフォルトは「STARTUP」です。</p> <p><b>メモ:</b> Veritas ASMinst リソースには、STARTUP オプションまたは SRVCTLSTART オプションを使うことをお勧めします。</p>
ShutDownOpt	文字列 - スカラー	<p>Oracle ASM インスタンスの停止オプション。デフォルトでは、エージェントは <code>sqlplus</code> コマンドを使って Oracle ASM インスタンスを停止します。 <code>srvctl</code> ユーティリティを使って ASM インスタンスを停止するには、オプションを <code>SRVCTLSTOP</code> に設定します。</p> <p>デフォルトは "" です。</p>

表 A-9は、ASMinst エージェントの内部属性をリストします。この属性は内部使用のみを目的としています。Veritas は、この属性の値を変更しないよう推奨します。

**表 A-9** ASMInst エージェントの内部属性

オプション属性	データ形式と値のタイプ	定義
AgentDirectory	静的 - 文字列	ASMInst エージェントに関連付けられているバイナリやスクリプトなどのファイルの場所を指定します。  デフォルトは /opt/VRTSagents/ha/bin/ASMInst です。

## ASMDG エージェントのリソースタイプ定義

ASMDG エージェントは、VCS の ASMDG リソースタイプとして表されます。次の OracleASMTypes.cf ファイルの抜粋は、ASMDG リソースタイプの定義を示します。

```
type ASMDG (
  static str AgentDirectory = "/opt/VRTSagents/ha/bin/ASMDG"
  static str ArgList[] = { Sid, Owner, Home, DBAUser,
    DBAPword, DiskGroups, EnvFile, Encoding }
  static int ContainerOpts{} = { RunInContainer=1, PassCInfo=0 }
  static boolean IntentionalOffline = 1
  str Sid
  str Owner
  str Home
  str DBAUser
  str DBAPword
  keylist DiskGroups
  str EnvFile
  str Encoding
)
```

## ASMDG エージェントの属性定義

ASMDG エージェントの属性は、必須、オプション、内部に分類されます。

次の必須属性には、値を割り当てる必要があります。

**表 A-10** ASMDG エージェントの必須属性

必須属性	データ形式と値のタイプ	定義
DiskGroups	キーリスト	Oracle データベースファイルを格納する ASM ディスクグループ。

必須属性	データ形式と値のタイプ	定義
SID	文字列 - スカラー	ASM インスタンスを表す \$ORACLE_SID 変数。  ASMIInst エージェントでは、Sid の大文字小文字が区別されます。
Owner	文字列 - スカラー	ASM ディスクグループをマウントまたはマウント解除する権限がある Oracle ユーザー。  エージェントでは、Oracle ユーザーとして LDAP ユーザーもサポートされません。
Home	文字列 - スカラー	Oracle ASM バイナリファイルと設定ファイルへの \$ORACLE_HOME パス。たとえば、パスとして /opt/ora_home を指定できます。  <b>メモ:</b> パスの最後にスラッシュ (/) は付けないでください。

次のオプションの属性は、設定に必要な場合に設定できます。

**表 A-11** ASMDG エージェントのオプションの属性

オプション属性	データ形式と値のタイプ	定義
DBAUser	文字列 - スカラー	ASM インスタンスを開始または停止する sysasm 権限がある ASM ユーザー。
DBAPword	文字列 - スカラー	DBAUser 用の暗号化されたパスワード。  コマンドラインで入力する場合にのみパスワードを暗号化します。 vcsencrypt ユーティリティを使用してパスワードを暗号化する必要があります。  p.84 の「Oracle データベースのユーザーパスワードとリスナーパスワードの暗号化」を参照してください。

オプション属性	データ形式と値のタイプ	定義
EnvFile	文字列 - スカラー	<p>エントリポイントスクリプトによって提供されるファイルの絶対パス名。このファイルには、<b>Oracle</b> データベースサーバー環境のユーザーが設定する <b>LD_LIBRARY_PATH</b>、<b>NLS_DATE_FORMAT</b> などの環境変数が含まれます。</p> <p>ファイル内容の構文は、<b>Owner</b> のログインシェルによって異なります。ファイルは <b>Owner</b> が読み取ることができなければなりません。ファイルにユーザー入力を求めるプロンプトが含まれてはいけません。</p>
Encoding	文字列 - スカラー	<p>表示される <b>Oracle</b> の出力の <b>Oracle</b> 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。デフォルトは "" です。</p>

次の属性は内部使用のみを目的としています。**Veritas** では、この属性の値を変更しないことをお勧めします。

**表 A-12** ASMDG エージェントの内部属性

オプション属性	データ形式と値のタイプ	定義
AgentDirectory	静的 - 文字列	<p>ASMDG エージェントに関連付けられているバイナリ、スクリプトなどのファイルの場所を指定します。</p> <p>デフォルトは <code>/opt/VRTSagents/ha/bin/ASMDG</code> です。</p>

# 設定例

この付録では以下の項目について説明しています。

- [Oracle enterprise agent](#) の設定例について
- [Oracle](#) 単一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
- [Solaris](#) ゾーンの [Oracle](#) インスタンスの設定例
- [Oracle ASM](#) の設定例
- [main.cf](#) の [Oracle](#) プラグ可能なデータベース(PDB)リソースの設定例
- [main.cf](#) の移行可能な [Oracle](#) プラグ可能データベース (PDB) リソースの設定例
- [systemD](#) でサポートされる [Oracle](#) の設定例
- [systemD](#) でサポートされる [ASMIInst](#) の設定例

## Oracle enterprise agent の設定例について

設定例には、VCS cluster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。

p.68 の「[VCS における Oracle インスタンスの設定](#)」を参照してください。

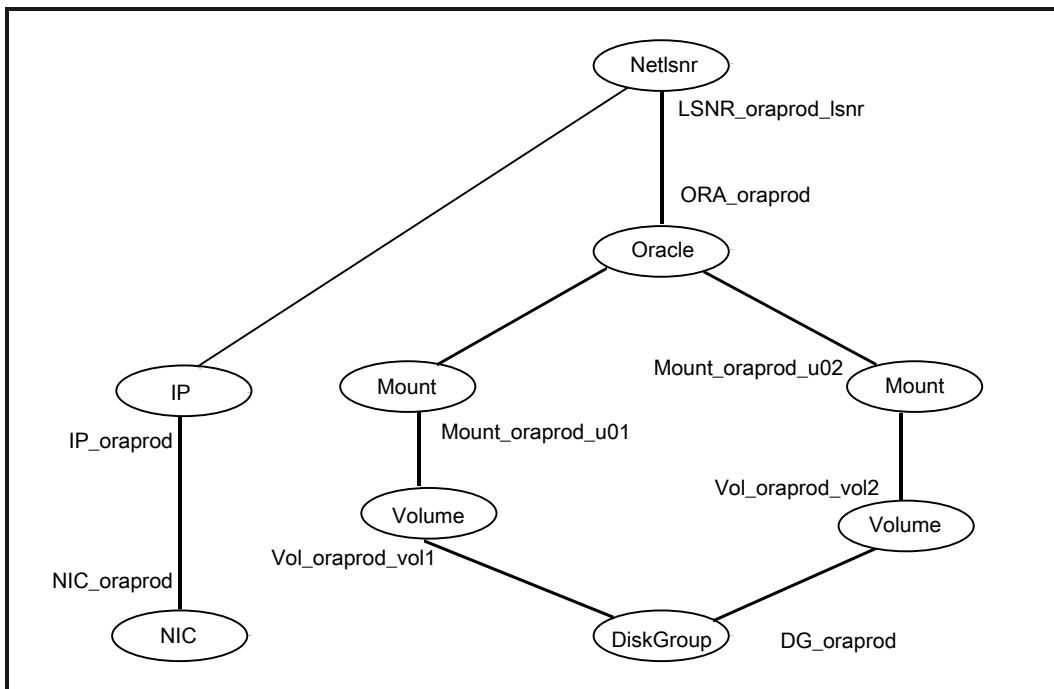
依存関係のグラフのサンプルは、サービスグループ間におけるリソースタイプ、リソース、リソースの依存関係を示しています。参照用として、設定ファイル ([main.cf](#)) の例も記載されています。

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 のリソースタイプについて詳しくは、『Cluster Server Bundled Agents Reference Guide』を参照してください。

## Oracle 単一インスタンスの設定例

図 B-1 に、VCS cluster環境で実行される 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ます。

図 B-1 Oracle 単一インスタンスの依存関係グラフ



cluster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** と **NIC** のリソースタイプにより設定されます。これらの各リソースがオンライン状態にされた後で、**Oracle** サーバーを起動できます。

### 単一 Oracle インスタンスの VCS 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。

---

**メモ:** 次のサンプル `main.cf` では、NIC デバイスに Solaris SPARC の例が使用されています。たとえば、Solaris x64 の場合は NIC デバイス例の `hme0` を `bge0` に置き換えてください。

---

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

cluster vcs (
)

system sys1 (
)

system sys2 (
)

group ORA_PROD_Group (
    SystemList = { sys1 = 0, sys2 = 1 }
    AutoStartList = { sys1 }
)

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

IP IP_oraprod (
    Device = hme0
    Address = "192.168.1.22"
    Netmask = "255.255.255.0"
)

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"
    )

NIC NIC_oraprod (
    Device = hme0
    NetworkHosts = {"192.168.1.1"}
)

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"
    MonScript = "./bin/Oracle/SqlTest.pl"
    User = thor
    Pword = hvlTptWvj

    DBAUser = "orauser"
    DBAPword = "DPSrFPeIrGPiRF"

    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
)

```

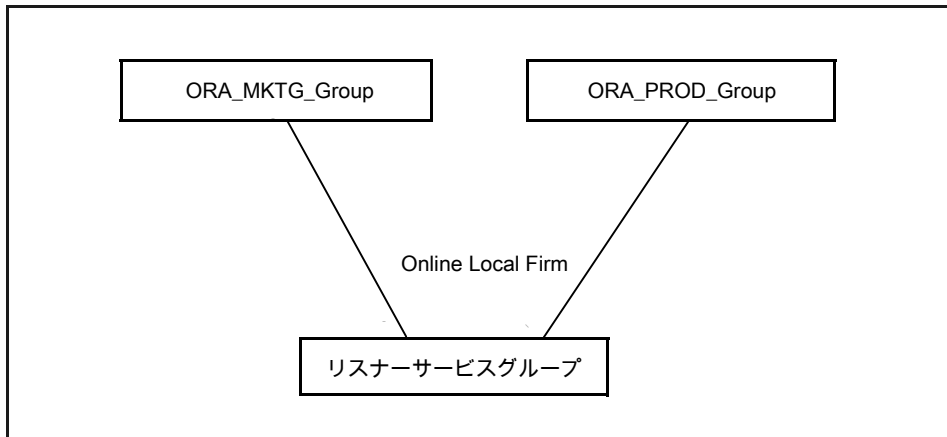
```

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
    
```

## 複数の Oracle インスタンス(単一リスナー)の設定例

図 B-2 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。

図 B-2 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タイプとして、**Online Local Firm** を設定します。

図 B-3 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、cluster 内の共有ディスクグループとボリュームは、それぞれ **DiskGroup** タイプと **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-3 Oracle インスタンスのいずれか 1 つの依存関係グラフ

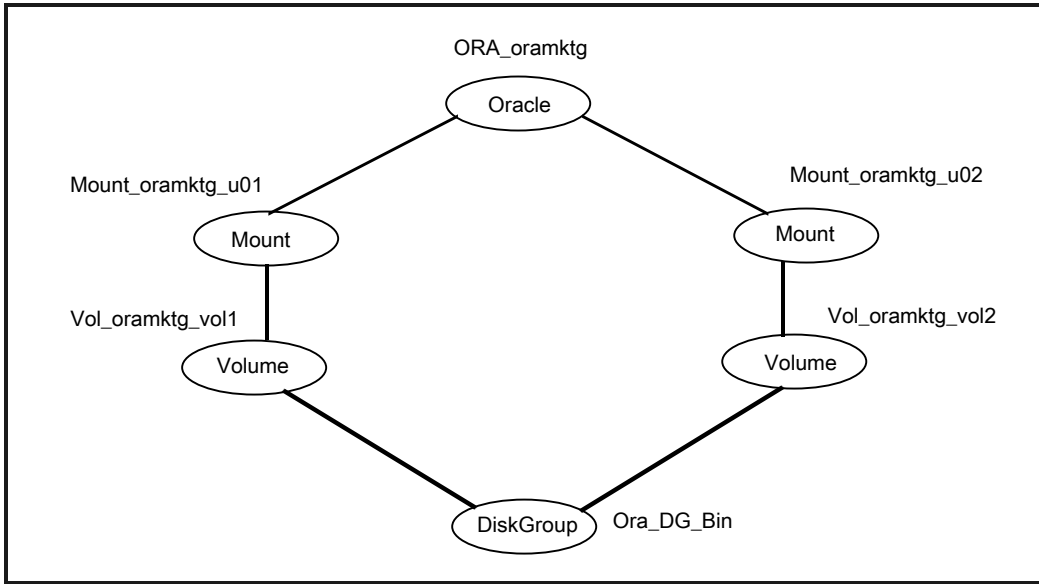
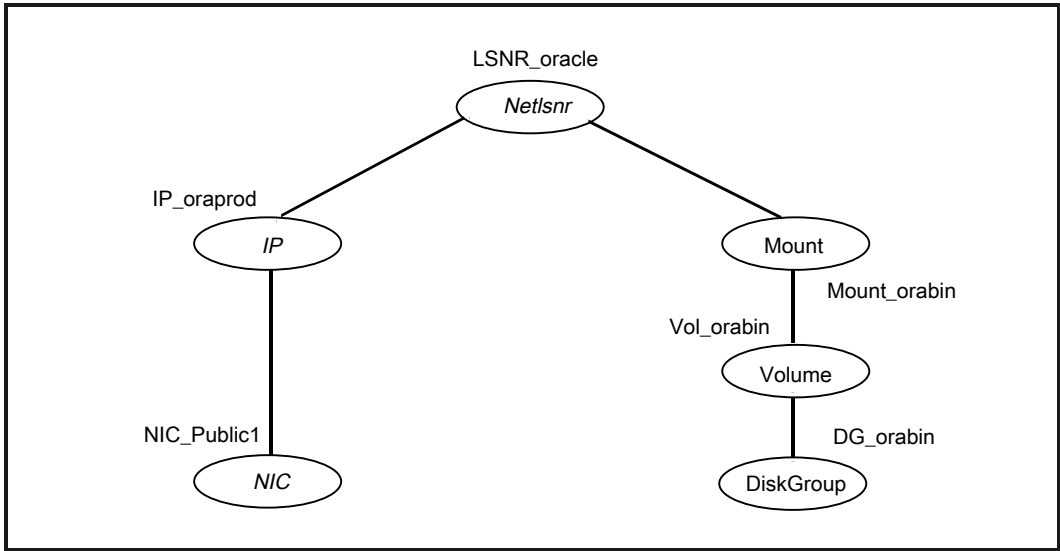


図 B-4 は、VCS 設定で 2 つの Oracle インスタンスが共有するリスナーの依存関係グラフを示しています。リスナーサービスグループでは、IP と NIC のリソースタイプを使って、仮想 IP アドレスの設定を行います。IP リソースと NIC リソースがオンライン状態にされた後で、リスナーを起動できます。

図 B-4 単ー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状態にされた後で、Oracle サーバーを起動できます。

**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`、`/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 VCS 設定ファイルの例

複数の Oracle インスタンスの設定ファイルの例を見直します。

**メモ:** 次のサンプル `main.cf` では、NIC デバイスに Solaris SPARC の例が使用されています。たとえば、Solaris x64 の場合は NIC デバイス例の `hme0` を `bge0` に置き換えてください。

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

```

cluster vcs (
)
system sys1 (
)

system sys2 (
)

group ORA_MKTG_Group (
    SystemList = { sys1 = 0, sys2 = 1 }
    AutoStartList = { sys1 }
)

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-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"
)

Oracle ORA_oramktg (
    Sid = MKTG
    Owner = oramktg
    Home = "/orahome/Oracle"
    EnvFile = "/tmp/env.sh"
    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 local 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 = { sys1 = 0, sys2 = 1 }
    AutoStartList = { sys1 }
)

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"
        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 local 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 = { sys1 = 0, sys2 = 1 }
        AutoStartList = { sys1 }
    )

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

```

```

)

IP IP_oraprod (
    Device = hme0
    Address = "192.168.1.22"
    Netmask = "255.255.255.0"
)

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

NIC NIC_Public1 (
    Device = hme0
    NetworkHosts = {"192.168.1.1"}
)

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

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
    
```

## 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は単一の Oracle インスタンスの設定に類似しています。

p.141 の「[Oracle 単一インスタンスの設定例](#)」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)の VCS 設定ファイルの例

複数のリスナーがある複数の Oracle インスタンスの設定ファイルの例を見直します。

---

**メモ:** 次のサンプル `main.cf` では、NIC デバイスに Solaris SPARC の例が使用されています。たとえば、Solaris x64 の場合は NIC デバイス例の `hme0` を `bge0` に置き換えてください。

---

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

cluster vcs (
)

system sys1 (
)

system sys2 (
)

group ORA_MKTG_Group (
    SystemList = { sys1 = 0, sys2 = 1 }
    AutoStartList = { sys1 }
)

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

IP IP_oramktg (
    Device = hme0
    Address = "192.168.1.22"
```

```
Netmask = "255.255.255.0"
)

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 = cqfOdoOolOo
)

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

Proxy NICProxy_oramktg (
  TargetResName = NIC_Public1
)

Volume Vol_oramktg_voll (
```

```

        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 = { sys1 = 0, sys2 = 1 }
    AutoStartList = { sys1 }
)

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

    IP IP_oraprod (
        Device = hme0
        Address = "192.168.1.21"
        Netmask = "255.255.255.0"
    )

    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"
)
```

```
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"
    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 = { sys1 = 0, sys2 = 1 }
    Parallel = 1
    AutoStartList = { sys1 }
)

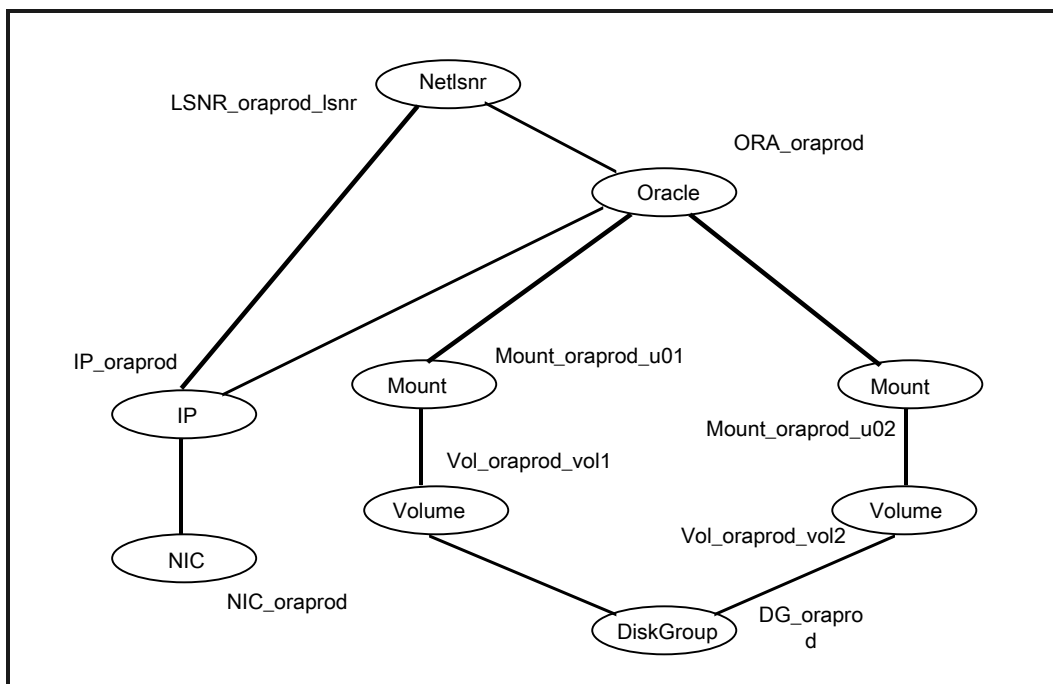
NIC NIC_Public1 (
    Device = hme0
    NetworkHosts = {"192.168.1.1"}
)
```

## 共有サーバーサポートの Oracle の設定例

図 B-5 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。

共有サーバーサポートの Oracle インスタンスは Solaris ゾーンに設定することもできます。

図 B-5 共有サーバーサポートで設定された Oracle の依存関係



cluster内の共有ディスクグループとボリュームは、それぞれ DiskGroup タイプと 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 と NIC のリソースタイプにより設定されます。これらの各リソースがオンライン状態にされた後で、Oracle サーバーを起動できます。

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

共有サーバーのサポートが設定された Oracle インスタンスの設定ファイルを見直します。

**メモ:** 次のサンプル main.cf では、NIC デバイスに Solaris SPARC の例が使用されています。たとえば、Solaris x64 の場合は NIC デバイス例の hme0 を bge0 に置き換えてください。

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

```
cluster vcs (  
)  
  
system sys1 (  
)  
  
system sys2 (  
)  
  
group ORA_PROD_Group (  
    SystemList = { sys1 = 0, sys2 = 1 }  
    AutoStartList = { sys1 }  
)  
  
    DiskGroup DG_oraprod (  
        DiskGroup = ora_prod_dg  
        StartVolumes = 0  
        StopVolumes = 0  
    )  
  
    IP IP_oraprod (  
        Device = hme0  
        Address = "192.168.1.22"  
        Netmask = "255.255.255.0"  
    )  
  
    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"  
    )  
  
    NIC NIC_ORAPROD (  
        Device = hme0  
        NetworkHosts = {"192.168.1.1"}  
    )  
)
```

```
)

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"
    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 ゾーンでの Oracle インスタンスの設定例

ローカルディスクまたは共有ディスク上にゾーンルートを持つ Solaris ゾーンでの Oracle インスタンスを設定できます。

以下の依存関係図で、VCS がサポートする Solaris ゾーンでのさまざまな Oracle 設定を見直します。

- 単一の Oracle インスタンスの設定
  - p.159 の「[単一 Oracle インスタンスにおける、ローカルディスク上のゾーンルート](#)」を参照してください。
  - p.160 の「[単一 Oracle インスタンスにおける、共有ディスク上のゾーンルート](#)」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
  - p.161 の「[複数 Oracle インスタンスにおける、ローカルディスク上のゾーンルート](#)」を参照してください。
  - p.164 の「[複数 Oracle インスタンスにおける、共有ディスク上のゾーンルート](#)」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
- 共有サーバーサポートの Oracle の設定
  - p.166 の「[共有サーバーサポートの Oracle インスタンスにおける、ローカルディスク上のゾーンルート](#)」を参照してください。
  - p.167 の「[共有サーバーサポートの Oracle インスタンスにおける、共有ディスク上のゾーンルート](#)」を参照してください。

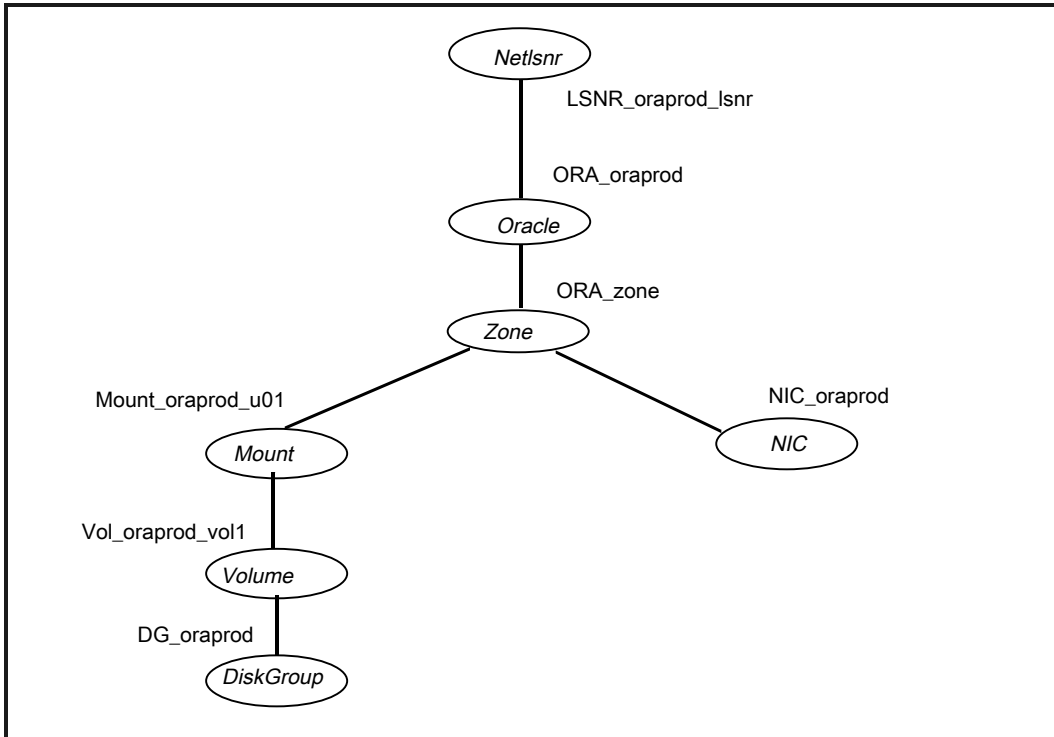
直接マウントポイントを使った依存関係図の例については、『Veritas InfoScale8.0 仮想化ガイド』を参照してください。

Solaris ゾーンでの Oracle インスタンス用 VCS 設定ファイルの例を見直します。

### 単一 Oracle インスタンスにおける、ローカルディスク上のゾーンルート

**図 B-6** は、ローカルディスク上に配置された Solaris ゾーンでの一般的なサービスグループを示します。ゾーンのルートファイルシステムが各 node のローカルディスク上に配置されている場合、そのファイルシステムはシステムのブート時にマウントされます。したがって、サービスグループがゾーン用に個別の DiskGroup リソースと Volume リソースを持つ必要はありません。

図 B-6 ローカルディスク上のゾーンでの単一 Oracle インスタンス

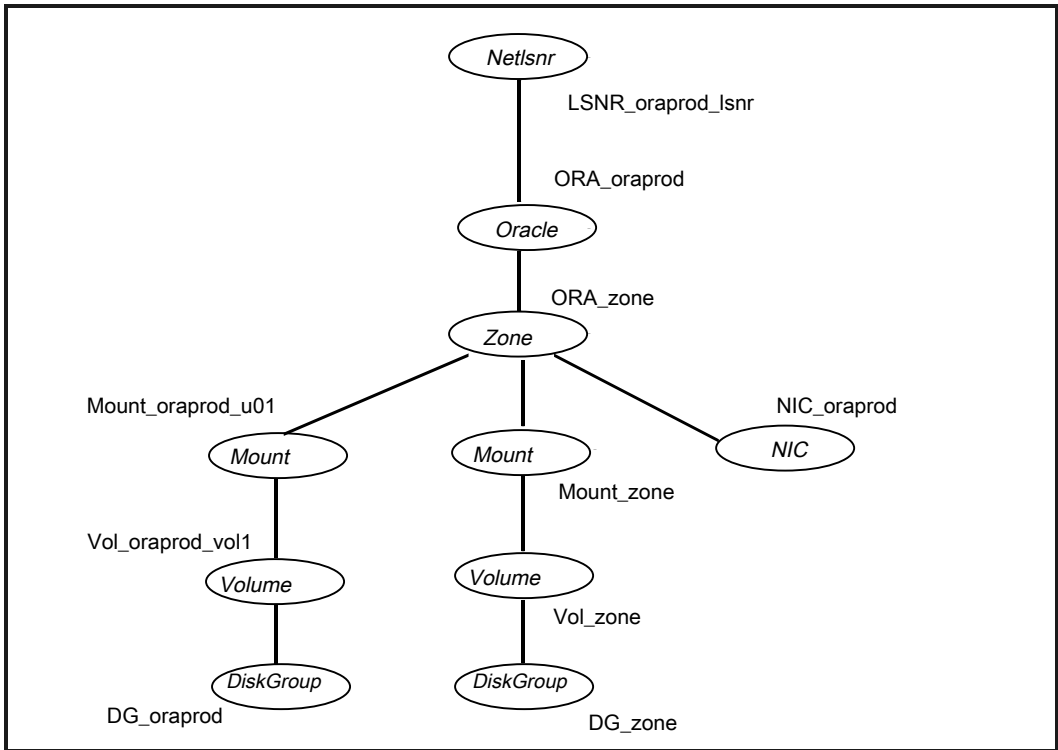


この設定例では、Oracle はゾーン内にローカルにインストールされ、Oracle インスタンスのデータファイルは共有ディスクに格納されています。cluster内の共有ディスクグループとボリュームは、それぞれ DiskGroup タイプと Volume タイプのリソースとして設定します。これらのボリュームは、Mount エージェントによりマウントされます。Mount エージェントがマウントするファイルシステムは、ゾーンの設定により、ゾーン内に LOFS マウントとしてマウントされます。Solaris ゾーンは、Mount リソースと NIC リソースに依存するゾーンリソースを使って監視されます。これらの各リソースがオンライン状態にされた後で、Oracle サーバーを起動できます。

## 単一 Oracle インスタンスにおける、共有ディスク上のゾーンルート

図 B-7 は、共有ディスク上に配置された Solaris ゾーンでの一般的なサービスグループを示します。ゾーンのルートファイルシステムが共有ディスク上に配置されている場合、そのファイルシステムは VCS によって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 リソースが必要です。

図 B-7 共有ディスク上のゾーンでの単一 Oracle インスタンス



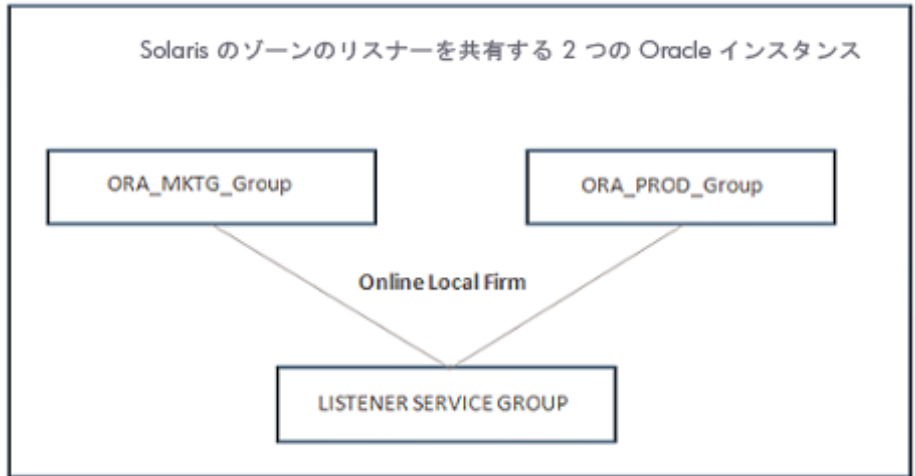
8

この設定例では、Oracle ホームはゾーン内にローカルにインストールされ、Oracle インスタンスのデータファイルは共有ディスクに格納されています。cluster内の共有ディスクグループとボリュームは、それぞれ DiskGroup タイプと 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。Mount エージェントがマウントするファイルシステムは、ゾーンの設定により、ゾーン内に LOFS マウントとしてマウントされます。Solaris ゾーンは、Mount リソースと NIC リソースに依存するゾーンリソースを使って監視されます。これらの各リソースがオンライン状態にされた後で、Oracle サーバーを起動できます。

## 複数 Oracle インスタンスにおける、ローカルディスク上のゾーンルート

この設定では、Netlsnr サービスグループに対して Oracle サービスグループが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タイプとして、Online Local Firm を設定します。ゾーンのリソースは、リスナーサービスグループで設定されます。Oracle サービスグループは、ゾーンの ContainerInfo 属性を設定しますが、ゾーンのリソースは含んでいません。

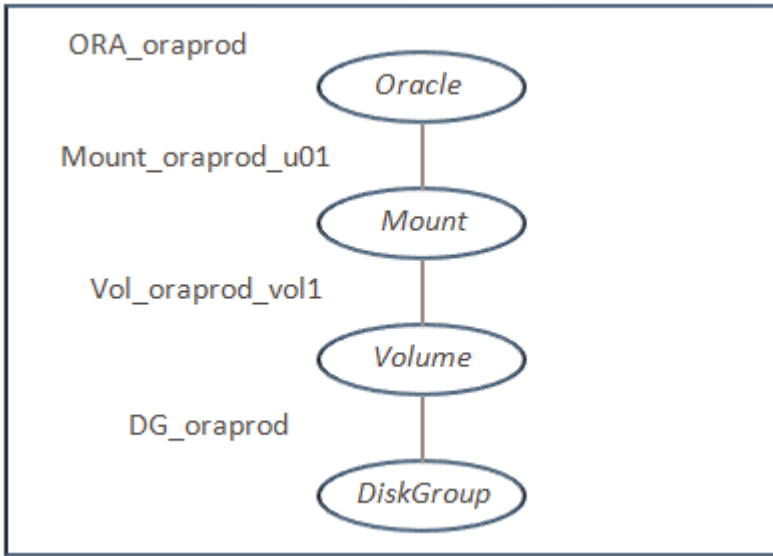
図 B-8 Solaris ゾーンでリスナーを共有する 2 つの Oracle インスタンス



ゾーンのルートファイルシステムが各nodeのローカルディスク上に配置されている場合、そのファイルシステムはシステムのブート時にマウントされます。したがって、サービスグループがゾーン用に個別の **DiskGroup** リソースと **Volume** リソースを持つ必要はありません。

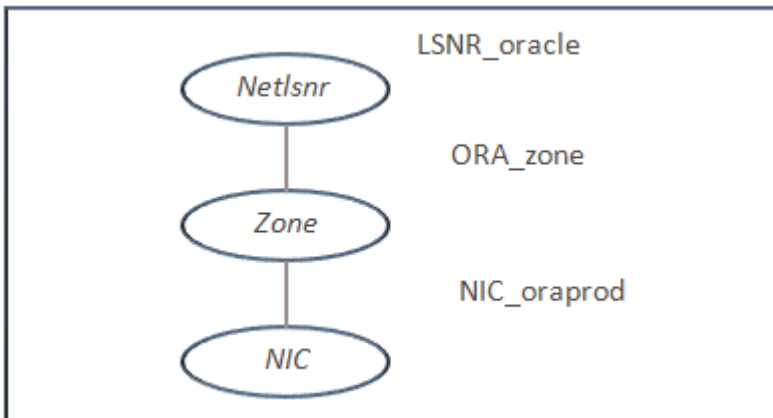
Oracle ホームはゾーン内にローカルにインストールされ、Oracle インスタンスのデータファイルは共有ディスクに格納されています。次の Oracle サービスグループでは、cluster 内の共有ディスクグループとボリュームは、それぞれ **DiskGroup** タイプと **Volume** タイプのリソースとして設定されています。これらのボリュームは、Mount エージェントによりマウントされます。Solaris ゾーンは、Mount リソースと NIC リソースに依存するゾーンリソースを使って監視されます。

図 B-9 Oracle サービスグループの依存関係



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

図 B-10 Netlsnr サービスグループの依存関係



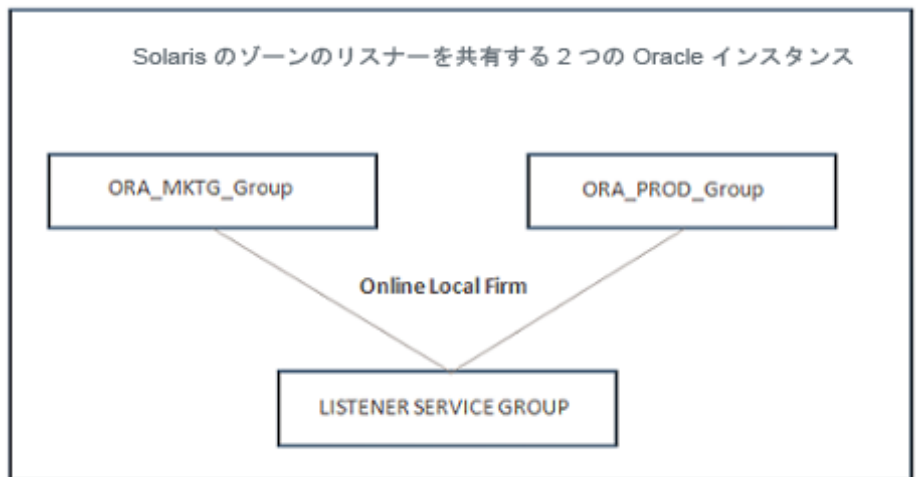
リスナーサービスグループと Oracle サービスグループのリソースがオンライン状態にされた後で、Oracle サーバーを起動できます。

**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`、`/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 Oracle インスタンスにおける、共有ディスク上のゾーンルート

この設定では、`Netlsnr` サービスグループに対して Oracle サービスグループが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タイプとして、`Online Local Firm`を設定します。ゾーンのリソースは、リスナーサービスグループで設定されます。Oracle サービスグループは、ゾーンの `ContainerInfo` 属性を設定しますが、ゾーンのリソースは含んでいません。

図 B-11 Solaris ゾーンでリスナーを共有する 2 つの Oracle インスタンス

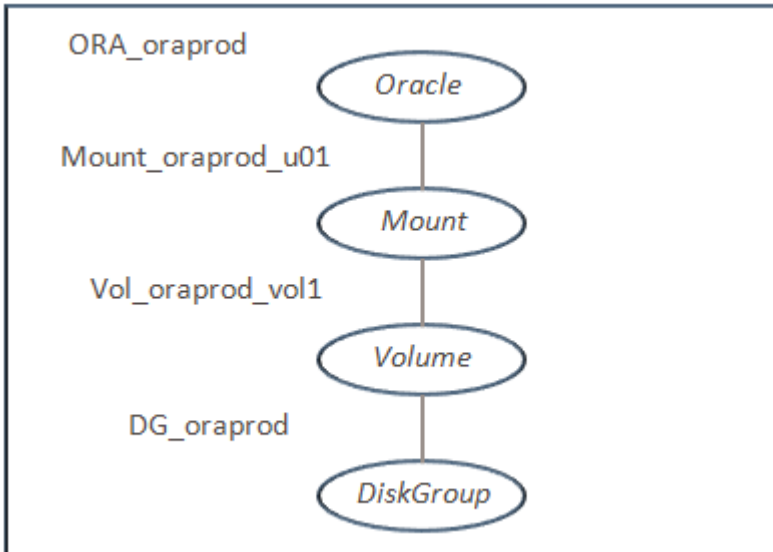


ゾーンのルートファイルシステムが共有ディスク上に配置されている場合、そのファイルシステムは `VCS` によってマウントされる必要があります。したがって、ゾーン用に個別の `DiskGroup` リソースと `Volume` リソースが必要です。

Oracle ホームはゾーン内にローカルにインストールされ、Oracle インスタンスのデータファイルは共有ディスクに格納されています。次の Oracle サービスグループでは、cluster 内の共有ディスクグループとボリュームは、それぞれ `DiskGroup` タイプと `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

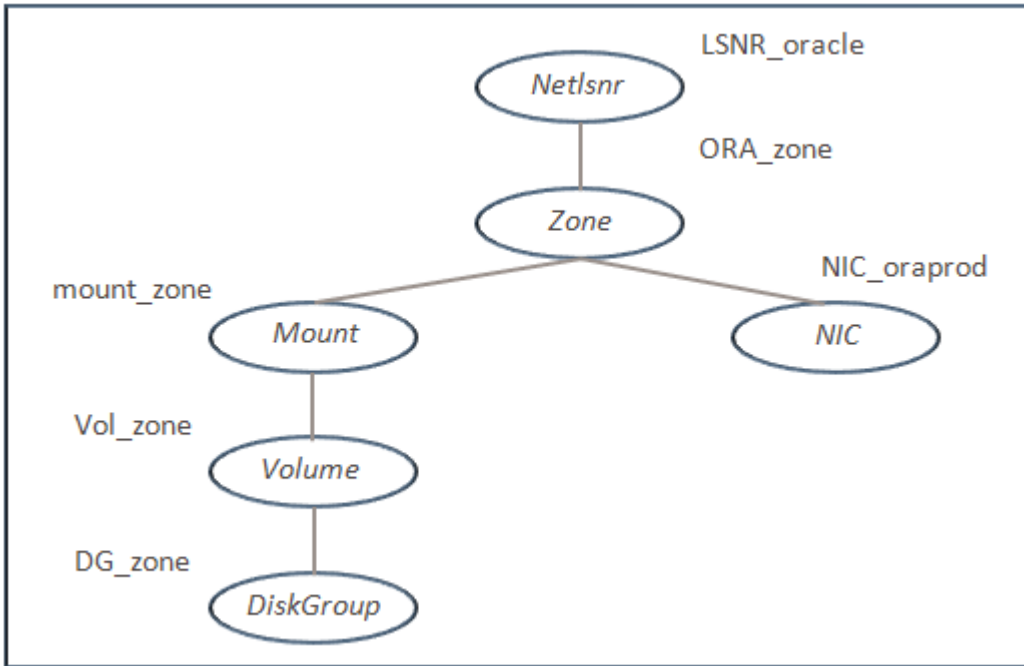
ントされます。**Solaris** ゾーンは、**Mount** リソースと **NIC** リソースに依存するゾーンリソースを使って監視されます。

図 B-12 Oracle サービスグループの依存関係



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

図 B-13 Netlsnr サービスグループの依存関係



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状態にされた後で、Oracle サーバーを起動できます。

---

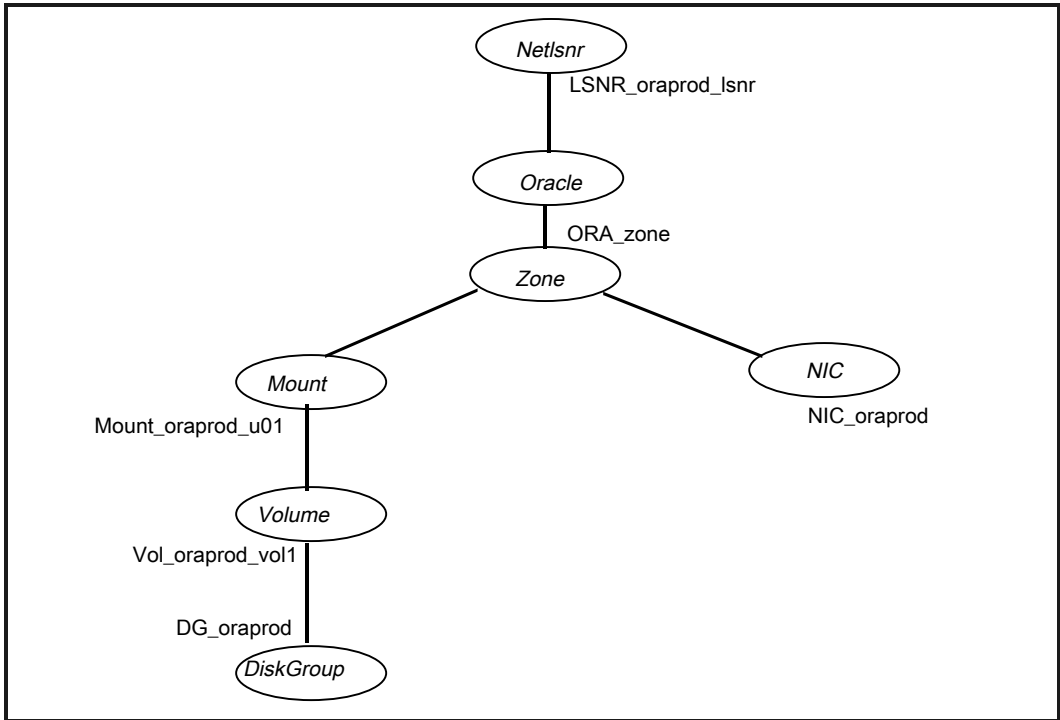
**メモ:** この場合、複数のデータベースをクラスタシステムで稼働させることができるように、`/etc/system`、`/etc/passwd`、`/etc/group`、`/etc/shadow` などのシステムファイルで適切な設定が行われていることを確認してください。特に、複数のインスタンスを実行するために、物理的なメモリ容量や、共有メモリのセグメントについてのカーネルパラメータの設定には、十分注意してください。さらに、システムの故障などで、クラスタ内の 1 台のシステム上ですべてのサービスグループがオンラインとなったとしても、そのシステムが複数のインスタンスの負荷に対処できるように設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

---

## 共有サーバーサポートの Oracle インスタンスにおける、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 node のローカルディスク上に配置されている場合、そのファイルシステムはシステムのブート時にマウントされます。したがって、サービスグループがゾーン用に個別の `DiskGroup` リソースと `Volume` リソースを持つ必要はありません。

図 B-14 ローカルディスク上のゾーンでの共有サーバーサポートの Oracle インスタンス

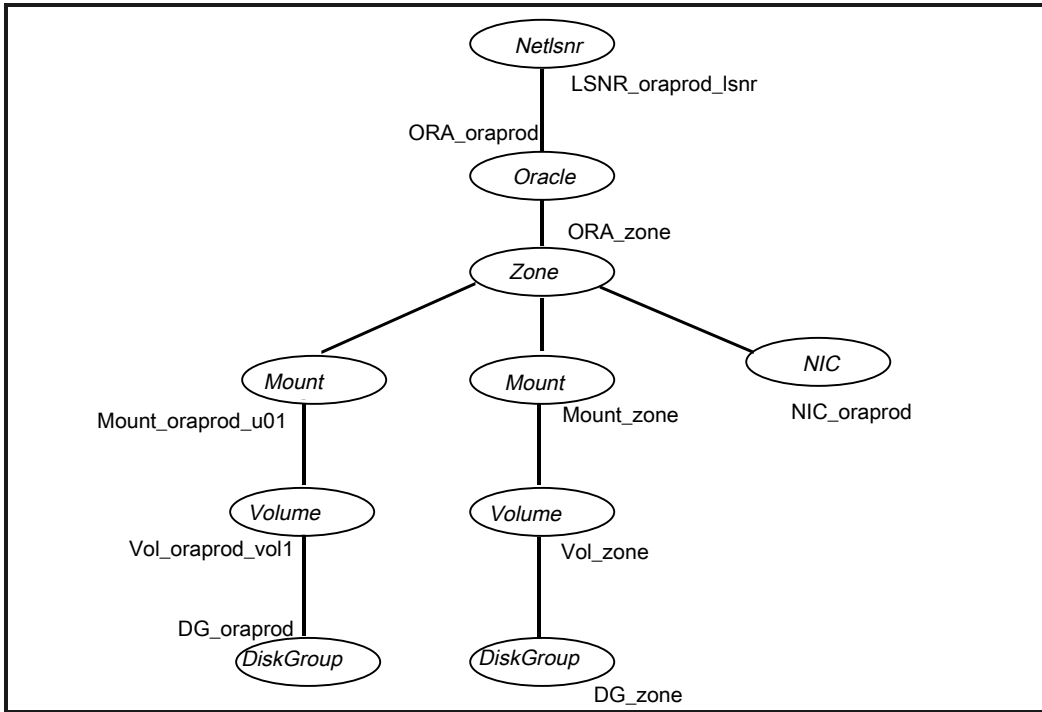


この設定例では、Oracle ホームはゾーン内にローカルにインストールされ、Oracle インスタンスのデータファイルは共有ディスクに格納されています。cluster内の共有ディスクグループとボリュームは、それぞれ DiskGroup タイプと Volume タイプのリソースとして設定します。これらのボリュームは、Mount エージェントによりマウントされます。Mount エージェントがマウントするファイルシステムは、ゾーンの設定により、ゾーン内に LOFS マウントとしてマウントされます。Solaris ゾーンは、Mount リソースと NIC リソースに依存するゾーンリソースを使って監視されます。これらの各リソースがオンライン状態にされた後で、Oracle サーバーを起動できます。

## 共有サーバーサポートの Oracle インスタンスにおける、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配置されている場合、そのファイルシステムは VCS によって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 リソースが必要です。

図 B-15 共有ディスク上のゾーンでの共有サーバーサポートの Oracle インスタンス

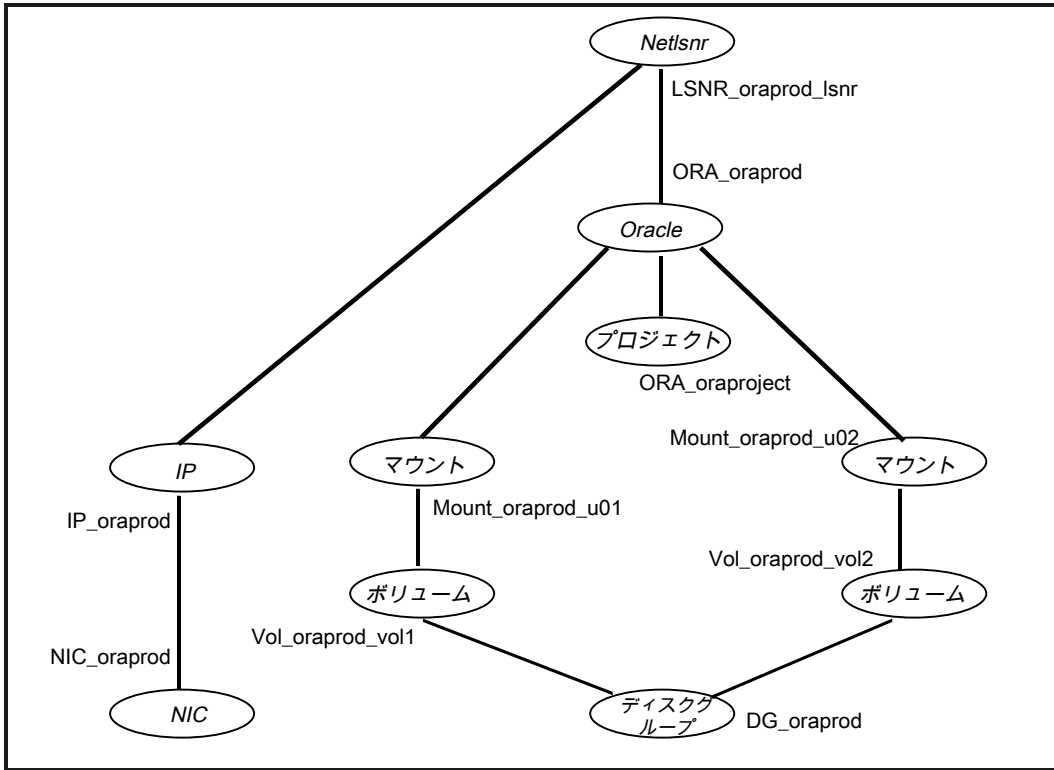


cluster内の共有ディスクグループとボリュームは、それぞれ DiskGroup タイプと Volume タイプのリソースとして設定します。これらのボリュームは、Mount エージェントによりマウントされます。Solaris ゾーンは、Mount リソースと NIC リソースに依存するゾーンリソースを使って監視されます。これらの各リソースがオンライン状態にされた後で、Oracle サーバーを起動できます。

## プロジェクト環境の Oracle インスタンス

図 B-16 に、Solaris プロジェクト環境で実行される 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ます。

図 B-16 Solaris プロジェクト環境で実行される単一の Oracle インスタンス



Solaris プロジェクトはプロジェクトリソースを介して監視されます。

## ローカルディスクまたは共有ディスク上のゾーンルートの VCS 設定ファイルの例

次のように、Zone タイプのリソースを main.cf 内に設定できます。

```
Zone ORA_zone (
)
```

このリソースの追加により、サービスグループ設定も Zone リソースに応じて NIC および Mount リソースを持つように変更されます。ゾーン設定を反映するようにリソースの依存関係を変更するには、リソース依存関係図を参照します。

p.159 の「Solaris ゾーン の Oracle インスタンス の設定例」を参照してください。

グループレベルの属性 ContainerInfo の値が次のキーに対して正しく定義されていることを確認します。

- **Name:** 非グローバルゾーンの名前を定義します。
- **Type:** この値は **Zone** に設定します。
- **Enabled:** リソースの管理方法に応じて、この値を **0** または **1** に設定します。

『Cluster Server 管理者ガイド』と『Veritas InfoScale 8.0 仮想化ガイド』を参照してください。

## Oracle ASM の設定例

VCS clusterで ASM により管理される Oracle インスタンスの状態を監視するために設定されたサービスグループに関する依存関係グラフを見直します。VCS 環境では、次のように Oracle ASM を設定できます。

- RAW ディスクとして ASM のディスク
- Veritas Volume Manager ボリュームとして ASM のディスク
- Veritas Cluster Volume Manager ボリュームとして ASM のディスク

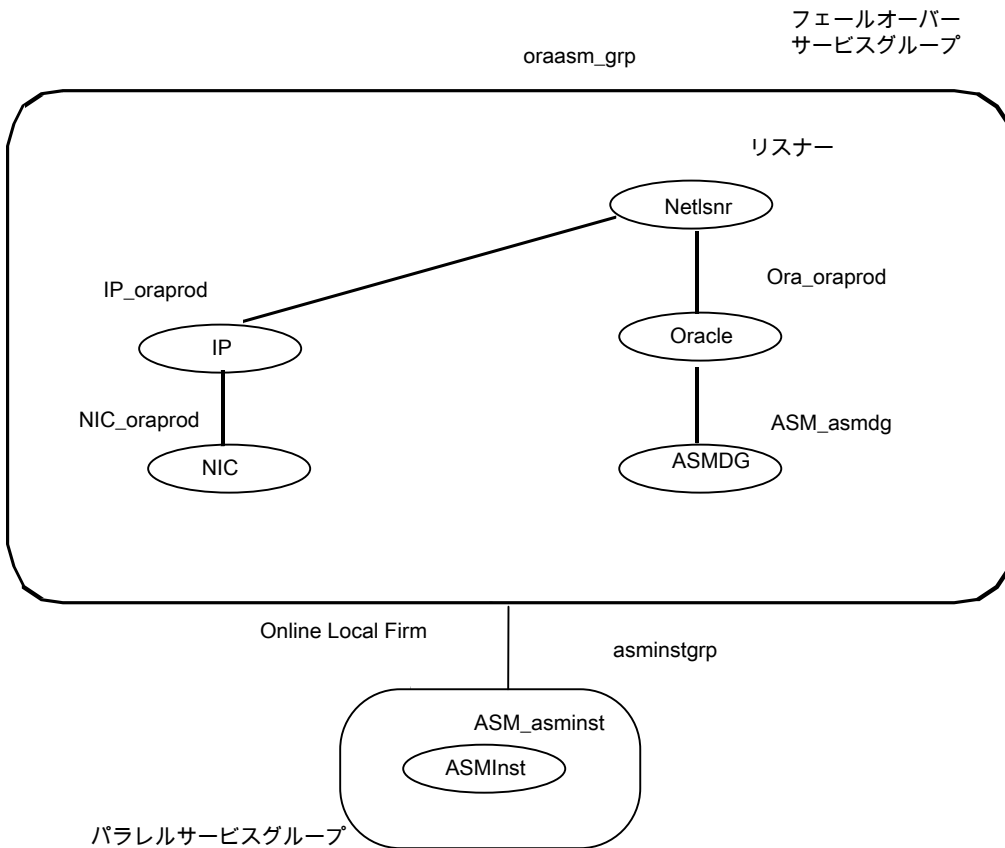
## RAW ディスクとしての ASM ディスクの設定例

ASM ディスクとして RAW デバイスを使う場合は、サービスグループを次の方法で設定できます。

- Oracle と ASMDG リソースを親のフェールオーバーサービスグループとして、ASMIInst リソースをパラレルサービスグループとして設定。
- サービスグループの依存は、Online Local Firm の依存関係を使用して設定されます。

図 B-17 は、ASMIInst リソースをパラレルサービスグループとして持つ典型的なサービスグループを表しています。

**図 B-17** ASMInst リソースがパラレルサービスグループとなっている依存関係グラフ



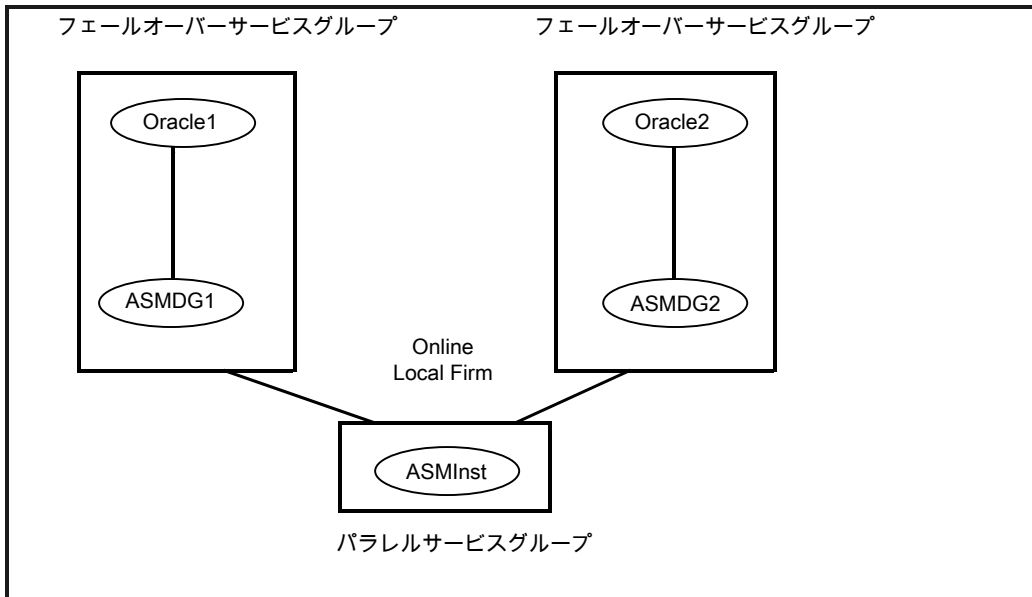
cluster内の Oracle ASM インスタンスと ASM ディスクグループは、それぞれ ASMInst と ASMDG のタイプのリソースとして設定されます。ASMInst エージェントは、パラレルサービスグループ **asminstgrp** として設定されます。

サービスグループ用の仮想 IP アドレスは、IP と NIC のリソースタイプにより設定されます。Oracle と ASMDG のリソースは、フェールオーバーサービスグループ **oraasm\_grp** として設定されます。これらの各リソースがオンライン状態にされた後で、Oracle サーバーを起動できます。

**oraasm\_grp** は、**Online Local Firm** の依存関係を使用して **asminstgrp** に依存した設定になります。

図 B-18 は、ASMInst リソースを共有している複数の Oracle インスタンスを持つ、パラレルサービスグループとして設定される典型的なサービスグループを表します。

図 B-18 ノードに複数の Oracle インスタンスを持つ Oracle ASM の依存関係グラフ



複数の Oracle インスタンスがある場合、フェールオーバーサービスグループは各 Oracle インスタンスに対して設定されます。Oracle サービスグループは、パラレルサービスグループとして設定される単一の ASM インスタンスを共有します。ASMInst サービスグループに対して Oracle サービスグループが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タイプとして、Online Local Firm を設定します。ただし、Cluster Server agent for Oracle がディスクグループをフェールオーバーできるようにするため、各データベースは排他的に ASM ディスクグループを使う必要があります。

## ASM で管理された Oracle インスタンスの VCS 設定ファイルの例

ASM で管理された Oracle インスタンスの設定例を見直します。サンプルファイルにはパラレルサービスグループの一部として ASMInst リソースの記述があります。

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

cluster vcs (
)
```

```
system symnode01 (
)

system symnode02 (
)

group asminstgrp (
  SystemList = { symnode01 = 0, symnode02 = 1 }
  Parallel = 1
)

ASMInst ASM_asminst (
  Sid = "+ASM"
  Owner = "oraprod"
  Home = "/orahome/Oracle"
)

group oraasm_grp (
  SystemList = { symnode01 = 0, symnode02 = 1 }
  AutoStartList = { symnode01 }
)

ASMDG ASM-asmdg (
  Sid = "+ASM"
  Owner = "oraprod"
  Home = "/orahome/Oracle"
  DiskGroups = { asmhighdg }
)

IP IP_oraprod (
  Device = hme0

  Address = "192.168.1.22"
  NetMask = "255.255.240.0"
)

NIC NIC_oraprod (
  Device = hme0

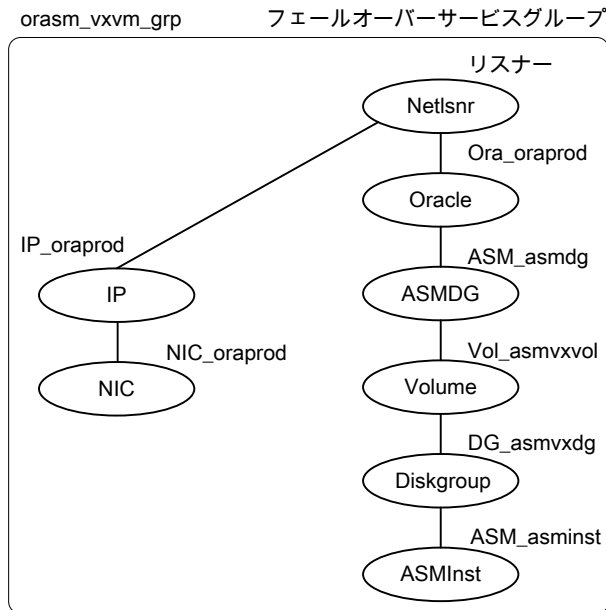
  NetworkHosts = { "192.168.1.1" }
)
```

```
Netlsnr LSNR_oraprod_lsnr (  
    Owner = "oraprod"  
    Home = "/orahome/Oracle"  
    TnsAdmin = "/orahome/Oracle/network/admin"  
    Listener = LISTENER_PROD  
    MonScript = "./bin/Netlsnr/LsnrTest.pl"  
)  
  
Oracle ORA_oraprod (  
    Sid = PROD  
    Owner = "oraprod"  
    Home = "/orahome/Oracle"  
    EnvFile = "/tmp/env.sh"  
    MonScript = "./bin/Oracle/SqlTest.pl"  
    User = thor  
    Pword = hvlTptWvj  
    Table = thor  
    MonitorOption = 0  
)  
  
requires group asminstgrp online local firm  
IP_oraprod requires NIC_oraprod  
LSNR_oraprod_lsnr requires IP_oraprod  
LSNR_oraprod_lsnr requires ORA_oraprod  
ORA_oraprod requires ASM_asmdg
```

## VxVM ボリュームとしての ASM ディスクの設定例

図 B-19 は、Oracle ASM として選択された VxVM ディスクを持つ典型的なサービスグループを表しています。

図 B-19 VxVM ディスク上に ASM を設定した場合の依存関係グラフ

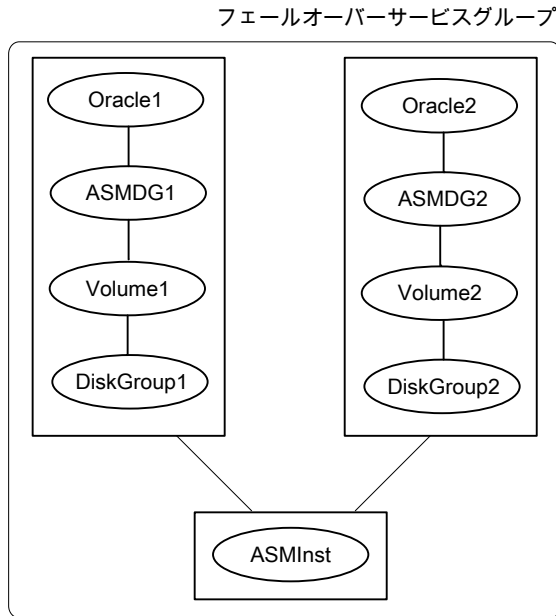


この設定には単一のフェールオーバーサービスグループがあります。cluster内の Oracle ASM インスタンスと ASM ディスクグループは、それぞれ ASMIInst と ASMDG のタイプのリソースとして設定されます。

クラスタ内の VxVM ディスクグループとボリュームは、それぞれ DiskGroup タイプと Volume タイプのリソースとして設定します。サービスグループ用の仮想 IP アドレスは、IP と NIC のリソースタイプにより設定されます。これらの各リソースがオンライン状態にされた後で、Oracle サーバーを起動できます。

図 B-20 は、単一のフェールオーバーサービスグループの一部である ASMIInst リソースを共有している複数の Oracle インスタンスを持つ典型的なサービスグループを表します。

**図 B-20** ノードに複数の Oracle インスタンスを持つ Oracle ASM の依存関係グラフ



## VxVM ディスクとしての ASM ディスクの VCS 設定ファイル例

ASM で管理された Oracle インスタンスの設定例を見直します。

```

include "types.cf"
include "OracleTypes.cf"
include "OracleASMTypes.cf"

cluster vcs (
)

system symnode01 (
)

system symnode02 (
)

group orasm_vxvm_grp (
    SystemList = { symnode01 = 0, symnode02 = 1 }
    AutoStartList = { symnode01 }
)
  
```

```
ASMDG ASM-asmdg (  
  Sid = "+ASM"  
  Owner = oraprod  
  Home = "/orahome/Oracle"  
  DiskGroups = { asmhhdg }  
  
  DBAUser = "orauser"  
  DBAPword = "DPSrFFeRiRGPiRF"  
  
  )  
  
ASMinst ASM_asminst (  
  Sid = "+ASM"  
  Owner = oraprod  
  Home = "/orahome/Oracle"  
  
  DBAUser = "orauser"  
  DBAPword = "DPSrFFeRiRGPiRF"  
  
  )  
  
DiskGroup DG_asmvxdg (  
  DiskGroup = asmvxdg  
  
  )  
  
IP IP_oraprod (  
  
  Device = hme0  
  
  Address = "192.168.1.22"  
  NetMask = "255.255.240.0"  
  
  )  
  
NIC NIC_oraprod (  
  
  Device = hme0  
  
  )  
  
NetworkHosts = { "192.168.1.1" }  
  
  )  
  
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"
  MonScript = "./bin/Oracle/SqlTest.pl"
  User = thor
  Pword = hv1TptWvj
  Table = thor
  MonitorOption = 0
)

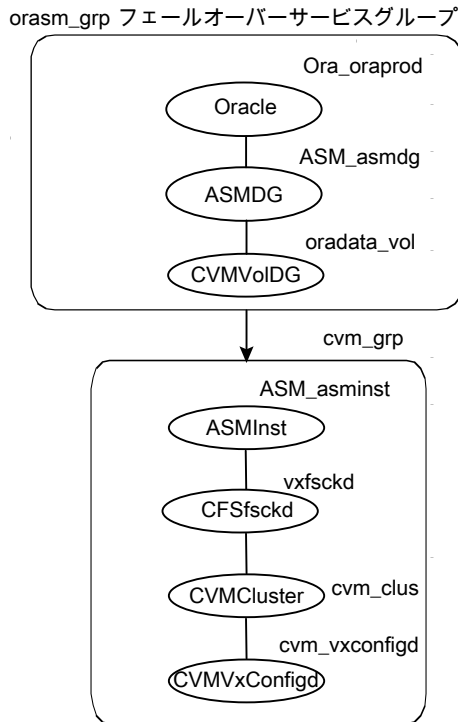
Volume Vol_asmvxxvol (
  Volume = asmvxxvol
  DiskGroup = asmvxxdg
)

IP_oraprod requires NIC_oraprod
LSNR_oraprod_lsnr requires IP_oraprod
LSNR_oraprod_lsnr requires ORA_oraprod
ASM_asmdg requires VOL_asmvxxvol
ORA_oraprod requires ASM_asmdg
VOL_asmvxxvol requires DG_asmvxxdg
DG_asmvxxdg requires ASM_asminst
```

## CVM ボリュームとしての ASM ディスクの設定例

図 B-21 は、Oracle ASM として選択された CVM ボリュームを持つ典型的なサービスグループを表しています。

**図 B-21** CVM ボリューム上の ASM の依存関係グラフ

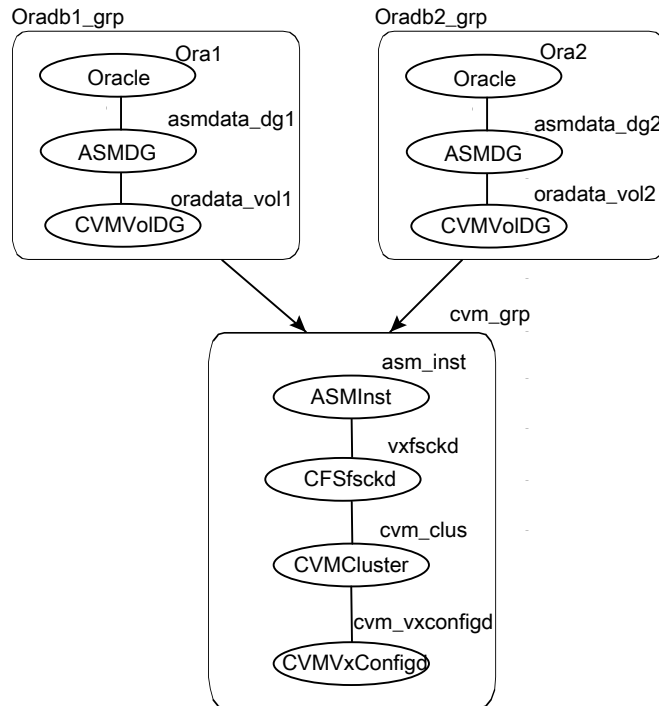


この設定には2つのサービスグループがあります。Oracle、ASMDG、およびCVMVolDGリソースは、親のフェールオーバーサービスグループ `oraasm_grp` の一部です。ASMinstリソースは、パラレルサービスグループであるCVM サービスグループの `cvm_grp` に属します。サービスグループは `online local firm` の依存関係でリンクされます。

データベースが存在するCVM ボリュームがオンライン状態になった後で、ASMDG エージェントはデータベースが必要とするASM ディスクグループをマウントします。サービスグループ用の仮想IPアドレスは、IPとNICのリソースタイプにより設定されます。これらの各リソースがオンライン状態にされた後で、Oracleサーバーを起動できます。

図B-22は、CVM パラレルサービスグループの一部であるASMinstリソースを共有している複数のOracleインスタンスを持つ典型的なサービスグループを表します。

図 B-22 ノードに複数の Oracle インスタンスを持つ Oracle ASM の依存関係グラフ



複数の Oracle インスタンスがある場合、フェールオーバーサービスグループは各 Oracle インスタンスに対して設定されます。Oracle サービスグループは、CVM パラレルサービスグループの一部として設定される単一の ASM インスタンスを共有します。CVM サービスグループに対して Oracle サービスグループが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タイプとして、Online Local Firm を設定します。ただし、Cluster Server agent for Oracle がディスクグループをフェールオーバーできるようにするため、各データベースは排他的に ASM ディスクグループを使う必要があります。

### CVM ボリュームを使う ASM の VCS 設定ファイルの例

ASM 用に CVM ボリュームを使う Oracle インスタンスの設定例を見直します。

```

include "types.cf"
include "CFSTypes.cf"
include "CVMTypes.cf"
include "OracleTypes.cf"
include "OracleASMTypes.cf"
  
```

```
cluster vcclus_asm (
  UserNames = { admin = abcdef }
  Administrator = { admin }
  UseFence = SCSI3
  HacliUserLevel = COMMANDROOT
)

system symnode01 (
)

system symnode02 (
)

group cvm (
  SystemList = { symnode01 = 0, symnode02 = 1 }
  AutoFailOver = 0
  Parallel = 1
  AutoStartList = { symnode01, symnode02 }
)

CFSfsckd vxfsckd (
)

CVMCluster cvm_clus (
  CVMClustName = vcclus_asm
  CVMNodeId = { symnode01 = 0, symnode02 = 1 }
  CVMTransport = gab
  CVMTimeout = 200
)

CVMVxconfigd cvm_vxconfigd (
  Critical = 0
  CVMVxconfigdArgs = { syslog }
)

ASMInst ASM_asminst (
  Sid = "+ASM"
  Owner = oracle
  Home = "/orahome/Oracle"
)

ASM_asminst requires vxfsckd
```

```
vxfsckd requires cvm_clus
cvm_clus requires cvm_vxconfig

group oraasm_grp (
  SystemList = { symnode01 = 0, symnode02 = 1 }
  AutoFailOver = 1
  AutoStartList = { symnode01, symnode02 }
)

CVMVolDg oradata_vol (
  CVMDiskGroup = ora_dg
  CVMVolume = { oradatavol }
  CVMActivation = sw
)

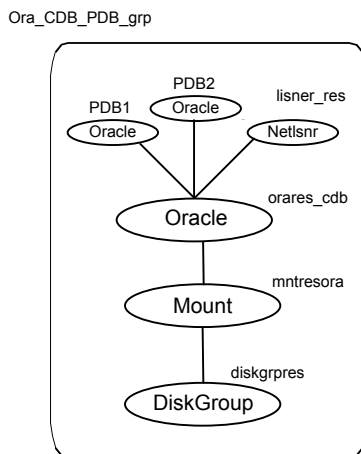
ASMDG ASM_asmdg (
  Sid = "+ASM"
  Owner = oracle
  Home = "/orahome/Oracle"
  DiskGroups = { ASM_DG }
)

Oracle Ora_oraprod (
  Owner = oracle
  Sid = vrts
  Home = "/orahome/Oracle"
)

requires group cvm_grp online local firm
Ora_oraprod requires ASM_asmdg
ASM_asmdg requires oradata_vol
```

# main.cf の Oracle プラグ可能なデータベース (PDB) リソースの設定例

図 B-23 Oracle CDB-PDB の設定例



main.cf ファイルで設定されている 2 つの PDB リソースの設定例を確認してください。

```

group CDB (
    SystemList = { sys1 = 1, sys2 = 1 }
    AutoStartList = { sys2 }
)

DiskGroup diskgrpres (
    DiskGroup = cdb_dg
)

Mount mntresora (
    MountPoint = "/db01"
    BlockDevice = "/dev/vx/dsk/cdb_dg/oravol"
)
    
```

```
IP IP_oraprod (  
    Device = eth0  
    Address = "192.168.1.22"  
    NetMask = "255.255.240.0"  
)  
  
NIC NIC_oraprod (  
    Device = eth0  
    NetworkHosts = {"192.168.1.1"}  
)  
  
Netlsnr lisner_res(  
    Owner = oracle  
    Home = "/u01/orahome"  
    TnsAdmin = "/u01/orahome"  
    Listener = LISTENER_PROD  
    MonScript = "./bin/Netlsnr/LsnrTest.pl"  
    LsnrPwd = cqfOdoOolOo  
)  
  
Oracle orares_cdb (  
    Critical = 0  
    Sid = cdb  
    Owner = oracle  
    Home = "/u01/oraHome"  
    StartUpOpt = STARTUP  
    DBName = cdb  
    User = "c##vcsuser"  
    Pword = hvnTkvKvmVktMvj  
    Table = vcs  
    LevelTwoMonitorFreq = 5  
)  
  
Oracle pdb1 (  
    Critical = 0  
    Sid = cdb  
    Owner = oracle  
    Home = "/u01/oraHome"  
    StartUpOpt = STARTUP  
    PDBName = pdb1  
    User = "c##vcsuser"
```

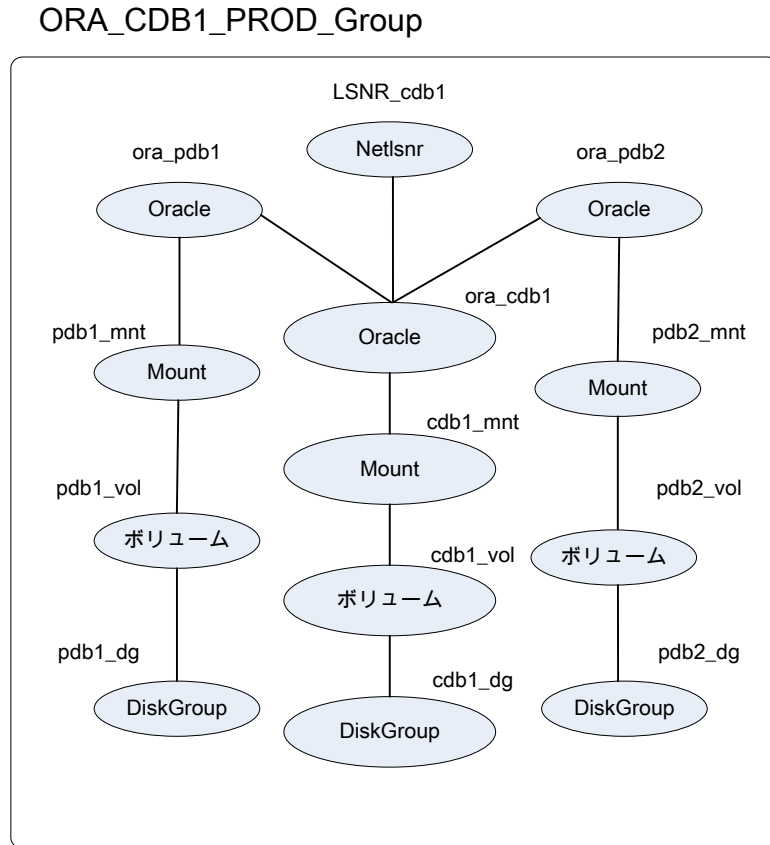
```
Pword = hvnTkvKvmVktMvj
Table = vcs
LevelTwoMonitorFreq = 5
)

Oracle pdb2 (
  Critical = 1
  Sid = cdb
  Owner = oracle
  Home = "/u01/oraHome"
  PDBName = PDB2
  User = "c##vcsuser"
  Pword = hvnTkvKvmVktMvj
  Table = vcs
  LevelTwoMonitorFreq = 5
)

mntresora requires diskgrpres
oraes_cdb requires mntresora
pdb1 requires oraes_cdb
pdb2 requires oraes_cdb
lisner_res requires oraes_cdb
IP_oraprod requires NIC_oraprod
lisner_res requires IP_oraprod
```

# main.cf の移行可能な Oracle プラグ可能データベース (PDB) リソースの設定例

図 B-24 移行可能な Oracle CDB-PDB の設定例



```

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

cluster vcs (
)

system SysA (
)

system SysB(

```

**main.cf** の移行可能な Oracle プラグ可能データベース (PDB) リソースの設定例

```
)

group ORA_CDB1_PROD_Group(
  SystemList = { SysA = 0, SysB = 1 }
)

    DiskGroup cdb1_dg (
        DiskGroup = cdb_dg1
    )

Volume cdb1_vol (
  Volume = cdbvol1
  DiskGroup = cdb_dg1
)

Mount cdb1_mnt (
  MountPoint = "/cdb1"
  BlockDevice = "/dev/vx/dsk/cdb_dg1/cdbvol1"
  FSType = vxfs
  FsckOpt = "-y"
)

DiskGroup pdb1_dg (
    DiskGroup = pdb_dg1
)

    Volume pdb1_vol (
        Volume = pdbvol1
        DiskGroup = pdb_dg1
    )

    Mount pdb1_mnt (
        MountPoint = "/pdb1"
        BlockDevice = "/dev/vx/dsk/pdb_dg1/pdbvol1"
        FSType = vxfs
        FsckOpt = "-y"
    )

DiskGroup pdb2_dg (
    DiskGroup = pdb_dg2
)

    Volume pdb2_vol (
```

## main.cf の移行可能な Oracle プラグ可能データベース (PDB) リソースの設定例

```
Volume = pdbvol2
DiskGroup = pdb_dg2
)

Mount pdb2_mnt (
    MountPoint = "/pdb2"
    BlockDevice = "/dev/vx/dsk/pdb_dg2/pdbvol2"
    FSType = vxfs
    FsckOpt = "-y"
)

IP IP_cdb1 (
    Device = hme0
    Address = "10.212.102.13"
    NetMask = "255.255.252.0"
)

NIC NIC_cdb1 (
    Device = hme0
    NetworkHosts = { "10.212.102.1" }
)

Netlsnr LSNR_cdb1 (
    Owner = oracle
    Home = "/u01/oraHome1"
)

Oracle ora_cdb1 (
    Sid = cdb1
    Owner = oracle
    Home = "/u01/oraHome1"
    StartUpOpt = STARTUP
    DBName = cdb1
)

Oracle ora_pdb1 (
    Sid = cdb1
    Owner = oracle
    Home = "/u01/oraHome1"
    PDBName = PDB1
)

Oracle ora_pdb2 (
```

## main.cf の移行可能な Oracle プラグ可能データベース (PDB) リソースの設定例

```
Critical = 0
Sid = cdb1
Owner = oracle
Home = "/u01/oraHome1"
PDBName = PDB2
)

IP_cdb1 requires NIC_cdb1
LSNR_cdb1 requires IP_cdb1
LSNR_cdb1 requires ora_cdb1
cdb1_mnt requires cdb1_vol
cdb1_vol requires cdb1_dg
pdb1_mnt requires pdb1_vol
pdb1_vol requires pdb1_dg
pdb2_mnt requires pdb2_vol
pdb2_vol requires pdb2_dg
ora_cdb1 requires cdb1_mnt
ora_pdb1 requires ora_cdb1
ora_pdb1 requires pdb1_mnt
ora_pdb2 requires ora_cdb1
ora_pdb2 requires pdb2_mnt

group ORA_CDB2_PROD_Group (
  SystemList = { SysA = 0, SysB = 1 }
)

  DiskGroup cdb2_dg (
    DiskGroup = cdb_dg2
  )

  Volume cdb2_vol (
    Volume = cdbvol2
    DiskGroup = cdb_dg2
  )

  Mount cdb2_mnt (
    MountPoint = "/cdb2"
    BlockDevice = "/dev/vx/dsk/cdb_dg2/cdbvol2"
    FSType = vxfs
    FsckOpt = "-y"
  )

  DiskGroup pdb3_dg (
```

**main.cf** の移行可能な Oracle プラグ可能データベース (PDB) リソースの設定例

```
DiskGroup pdb3_dg (
    DiskGroup = pdb_dg3
)

Volume pdb3_vol (
    Volume = pdbvol3
    DiskGroup = pdb_dg3
)

Mount pdb3_mnt (
    MountPoint = "/pdb3"
    BlockDevice = "/dev/vx/dsk/pdb_dg3/pdbvol3"
    FSType = vxfs
    FsckOpt = "-y"
)

DiskGroup pdb4_dg (
    DiskGroup = pdb_dg4
)

Volume pdb4_vol (
    Volume = pdbvol4
    DiskGroup = pdb_dg4
)

Mount pdb4_mnt (
    MountPoint = "/pdb4"
    BlockDevice = "/dev/vx/dsk/pdb_dg4/pdbvol4"
    FSType = vxfs
    FsckOpt = "-y"
)

IP IP_cdb2 (
    Device = hme0
    Address = "10.212.102.14"
    NetMask = "255.255.252.0"
)

NIC NIC_cdb2 (
    Device = hme0
    NetworkHosts = { "10.212.102.1" }
)

Netlsnr LSNR_cdb2 (
```

**main.cf** の移行可能な Oracle プラグ可能データベース (PDB) リソースの設定例

```
        Owner = oracle
        Home = "/u01/oraHome2"
    )

Oracle ora_cdb2 (
    Sid = cdb2
    Owner = oracle
    Home = "/u01/oraHome2"
    StartUpOpt = STARTUP
    DBName = cdb2
)

Oracle ora_pdb3 (
    Sid = cdb2
    Owner = oracle
    Home = "/u01/oraHome2"
    PDBName = PDB3
)

Oracle ora_pdb4 (
    Sid = cdb2
    Owner = oracle
    Home = "/u01/oraHome2"
    PDBName = PDB4
)

IP_cdb2 requires NIC_cdb2
LSNR_cdb2 requires IP_cdb2
LSNR_cdb2 requires ora_cdb2
cdb2_mnt requires cdb2_vol
cdb2_vol requires cdb2_dg
pdb3_mnt requires pdb3_vol
pdb3_vol requires pdb3_dg
pdb4_mnt requires pdb4_vol
pdb4_vol requires pdb4_dg
ora_cdb2 requires cdb2_mnt
ora_pdb3 requires ora_cdb2
ora_pdb3 requires pdb3_mnt
ora_pdb4 requires ora_cdb2
ora_pdb4 requires pdb4_mnt
```

## systemD でサポートされる Oracle の設定例

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

cluster vcs (
)

system sys1 (
)

system sys2 (
)

group orasg1 (
    SystemList = { sys1 = 0, sys2 = 1 }
)

DiskGroup DG_oraprod (
    Critical = 0
    DiskGroup = oradb_dg
)

IP IP_oraprod (
    Critical = 0
    Device @sys1 = eth0
    Device @sys2 = eno16777728
    Address = "123.123.145.42"
    NetMask = "255.255.255.0"
)

Mount Mount_oraprod (
    Critical = 0
    MountPoint = "/home/oracle/oradata"
    BlockDevice = "/dev/vx/dsk/oradb_dg/oradb_vol"
    FSType = vxfs
    MountOpt = rw
    FsckOpt = "-y"
)

NIC NIC_Public1 (
    Critical = 0
    Device @sys1 = eth0
    Device @sys2 = eno16777728
```

```

NetworkHosts = { "123.123.123.1" }
                )

Netlsnr LSNR_oracle (
Critical = 0
Owner = oracle
Home = "/home/oracle/app/oracle/product/dbhome_1"
TnsAdmin = "/home/oracle/app/oracle/product/dbhome_1/network/admin"
EnvFile = "/home/oracle/.bashrc"
MonScript = "/opt/VRTSagents/ha/bin/Netlsnr/LsnrTest.pl"
                )

Oracle ORA_oraprod (
SystemDAttrList = { LimitLOCKS = 44432 }
Sid = oradb
Owner = ORA_oraprod
Home = "/home/oracle/app/oracle/product/dbhome_1"
Pfile = "/home/oracle/app/oracle/product/dbhome_1/dbs/initoradb.ora"
StartUpOpt = STARTUP
EnvFile = "/tmp/a"
UseSystemD = 1
MonScript = "/opt/VRTSagents/ha/bin/Oracle/SqlTest.pl"
User = vcsuser
Pword = aogMdoDofOdmFoc
Table = vcs
LevelTwoMonitorFreq = 0
                )

Volume Vol_oraprod (
Critical = 0
DiskGroup = oradb_dg
Volume = oradb_vol
                )

IP_oraprod requires NIC_Public1
LSNR_oracle requires IP_oraprod
Mount_oraprod requires Vol_oraprod
ORA_oraprod requires LSNR_oracle
ORA_oraprod requires Mount_oraprod
Vol_oraprod requires DG_oraprod

```

## systemD でサポートされる ASMinst の設定例

```
ASMinst Sample main.cf
include "OracleASMTypes.cf"
include "types.cf"
include "OracleTypes.cf"

cluster vcs (
    UserNames = { admin = Hk, admini = gmnLmjMj }
    Administrators = { admini }
)

system sys1 (
)

system sys2 (
)

group asminstgrp (
    SystemList = { sys1 = 0, sys2 = 1 }
)

ASMinst ASM_asminst (
    SystemDAttrList = { LimitNOFILE = 55588 }
    Sid = "+ASM"
    Owner = oracle
    Home = "/u01/app"
    StartUpOpt = STARTUP_MOUNT
    UseSystemD = 1
)

group ora_asmgrp (
    SystemList = { sys1 = 0, sys2 = 1 }
)

ASMDG ASM_asmdg(
    Sid = "+ASM"
    Owner = oracle
    Home = "/u01/app"
    DiskGroups = { DBDG }
)
```

```

Oracle Ora_oraprod (
    Sid = oradb
    Owner = oracle
    Home = "/home/oracle/oracle"
    Pfile = "/home/oracle/oracle/dbs/initrddb.ora"
    StartUpOpt = SRVCTLSTART
    ShutDownOpt = SRVCTLSTOP
    MonScript = "/opt/VRTSagents/ha/bin/Oracle/SqlTest.pl"

)

requires group asminstgrp online local firm
Ora_oraprod requires ASM_asmdg

```

# ベストプラクティス

この付録では以下の項目について説明しています。

- [VCS 環境で複数の 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。
- セマフォと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`uxx`/`<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`uxx`/`<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 cluster 内の他のバージョン

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 nodes 間で一致させる。

# Oracle の VCS cluster での SPFILE の使用

この付録では以下の項目について説明しています。

- [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 clusterでの SPFILE の使用](#)

## Oracle 初期化パラメータファイルについて

Oracle9i では、データベースサーバーに格納されるバイナリファイルである 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、すべての起動と停止のプロシージャ間で統一できます。

## Oracle インスタンスの起動

Oracle インスタンスは、次のように開始できます。

- 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

PFILE 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 clusterでの SPFILE の使用

Oracle9i より前のバージョンの 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。データベースのセッション中にインスタンスパラメータに対して適用された変更はファイルに保存されませんでした。初期化ファイルには手動でその変更を加えなければなりませんでした。

Cluster Server 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` 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` から `SPFILE` を作成するには

- ◆ `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。

**PFILE で SPFILE の場所を指定するには**

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

```
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
```

# 単一インスタンスのデータベース環境での OHASD

この付録では以下の項目について説明しています。

- [単一インスタンスのデータベース環境での OHASD について](#)
- [OHASD の高可用性を確保するための Application エージェントの設定](#)

## 単一インスタンスのデータベース環境での OHASD について

Oracle Grid Infrastructure ソフトウェアは、Oracle Automatic Restart と Oracle ASM 機能を含む Oracle データベースのサポートを提供します。

単一インスタンスデータベース環境内の Oracle Grid Infrastructure がインストールされたすべてのノードで、Oracle High Availability Services デーモン (OHASD) プロセスが Oracle リソースより早くオンラインになることを確認します。

これらのノードでの Oracle リソースの管理は OHASD プロセスに依存しています。したがって、リソースを確実にオンラインにするために、このプロセスが最初にオンラインになる必要があります。

## OHASD の高可用性を確保するための Application エージェントの設定

VCS クラスタ内で OHASD プロセスの高可用性を確保するように、Application エージェントを設定できます。

OHASD プロセス用に Application エージェントを設定して、サービスのオンライン状態を確保するには

1. Application エージェントを設定します。

```
Application ohasd_res (  
  StartProgram = "<GRID_HOME>/bin/crsctl start has"  
  
  StopProgram = "<GRID_HOME>/bin/crsctl stop has"  
  PidFiles @node1 = {  
    "<GRID_HOME>/ohasd/init/node1.pid" }  
  PidFiles @node2 = {  
    "<GRID_HOME>/ohasd/init/node2.pid" }  
)
```

2. リソースが次のように設定されている場合、OHASD デーモンでリソースの依存関係を設定します。

- Oracle 再起動モードで実行されているデータベース向け
- ASMInst リソース向け

3. 次の依存関係ツリーでは、ohasd リソースはフェールオーバーサービスグループまたはパラレルサービスグループで設定されます。

VCS クラスタのすべてのノードで ohasd の自動起動を無効にするために、#  
\$GRID\_HOME/bin/srvctl disable has コマンドを実行します。

図 E-1      平行サービスグループで設定された OHASD リソースと、フェールオーバーサービスグループ内の ASMInst リソースを示します。

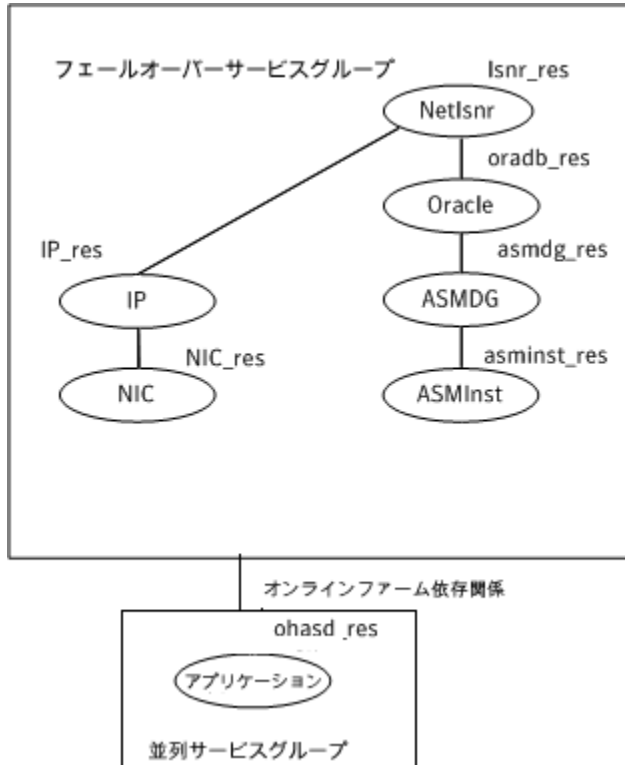


図 E-2 フェールオーバーサービスグループで設定された OHASD リソースと、ASMIInst リソースを示します。

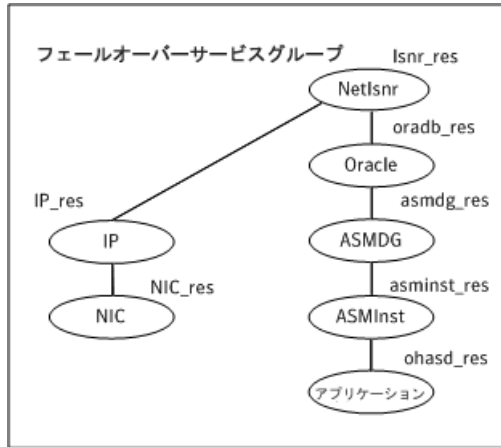


図 E-3 Oracle 再起動モードで実行されているデータベースの依存関係ツリーを示します。

