

Veritas™ Cluster Server Agent for DB2 インストールおよび設 定ガイド

Linux

6.0.1

Veritas Cluster Server Agent for DB2 インストールおよび設定ガイド

このマニュアルで説明するソフトウェアは、使用許諾契約に基づいて提供され、その内容に同意する場合にのみ使用することができます。

製品バージョン: VCS 6.0.1

マニュアルバージョン: 6.0.1 Rev 0

著作権について

Copyright © 2012 Symantec Corporation. All rights reserved.

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

本書に記載の製品は、ライセンスに基づいて配布され、使用、コピー、配布、逆コンパイル、リバースエンジニアリングはそのライセンスによって制限されます。本書のいかなる部分も、Symantec Corporation とそのライセンサーの書面による事前の許可なく、いかなる形式、方法であっても複製することはできません。

本書は「現状有姿のまま」提供され、商品性、特定目的への適合性、不侵害の黙示的な保証を含む、すべての明示的または黙示的な条件、表明、保証は、この免責が法的に無効であるとみなされない限り、免責されるものとします。Symantec Corporation は、本書の供給、性能、使用に関する付随的または間接的損害に対して責任を負わないものとします。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。

ライセンス対象ソフトウェアと関連書類は、FAR 12.212 の規定によって商用コンピュータソフトウェアとみなされ、場合に応じて、FAR 52.227-19「Commercial Computer Software - Restricted Rights」、DFARS 227.7202「Rights in Commercial Computer Software or Commercial Computer Software Documentation」、その後継規制の規定により制限された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と関連書類の使用、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。

Symantec Corporation
350 Ellis Street
Mountain View, CA 94043
<http://www.symantec.com>

第 1 章	Veritas™ Cluster Server Agent for DB2 の概要	9
	Veritas™ Cluster Server Agent for DB2 について	9
	Veritas™ Cluster Server Agent for DB2 が DB2 の高可用性を実現する しくみ	10
	DB2 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	10
	DB2 の VCS エージェントの のサポート対象ソフトウェア	11
	VCS Agent for DB2 のエージェント関数について	11
	VCS agent for DB2 のオンラインエージェント関数について	11
	VCS agent for DB2 のオフラインエージェント関数について	11
	DB2 の VCS エージェントの監視エージェント機能について	12
	VCS Agent for DB2 クリーンエージェント機能について	12
	VCS agent for DB2 用情報エージェント機能について	13
	アクションエージェント機能について VCS Agent for DB2	13
	VCS Agent for DB2 の IMF の統合機能について	14
	VCS Agent for DB2 データベース情報を取得するための情報エージェ ント機能の実行について	14
	VCS クラスタの標準的な DB2 設定	15
	VCS 環境に DB2 UDB を設定するためのロードマップ	16
第 2 章	DB2 のインストールと設定	17
	DB2 をインストールするための VCS の必要条件	17
	非 MPP 構成 DB2 UDB をインストールするための準備	17
	MPP 構成 DB2 UDB をインストールするための準備	18
	DB2 の非 MPP インスタンス用のファイルシステムの作成	19
	DB2 の MPP インスタンス用の共有ファイルシステムの作成	19
	DB2 用ユーザーアカウントとグループアカウントの定義	20
	VCS 環境での DB2 のインストール	22
	DB2 の設定	23
	DB2 用 /etc/services の確認	23
	すべてのクラスタードの DB2 ユーザーに対するパスワードなし通信 の有効化	24
	DB2 の \$DB2InstHome/sqlib/db2nodes.cfg ファイルの変更	24
	DB2 インストールの確認	27

第 3 章	Veritas™ Cluster Server Agent for DB2 のインストールと削除	29
	VCS Agent for DB2 をインストールまたはアップグレードする前に	29
	DB2 ソフトウェアの VCS エージェントのインストール	30
	VCS Agent for DB2 のアップグレード	31
	VCS Agent for DB2 の無効化	32
	VCS Agent for DB2 の削除	33
第 4 章	DB2 のための VCS サービスグループの設定	35
	DB2 用サービスグループの設定について	35
	VCS の DB2 の設定について	36
	DB2 用非 MPP 構成サービスグループ	36
	DB2 用 MPP 構成サービスグループ	37
	DB2 サービスグループを設定する前に	38
	Db2udbTypes.cf ファイルのインポート	38
	Cluster Manager (Java コンソール) からの VCS Agent for DB2 の設定	40
	Cluster Manager (Java コンソール) による DB2 MPP のサービスグループの追加	40
	Cluster Manager (Java コンソール) による DB2 非 MPP のサービスグループの追加	42
	main.cf ファイルの編集による VCS Agent for DB2 の設定	44
	MPP 構成 DB2 を使うためのエージェントの設定	45
	非 MPP 構成 DB2 を使うためのエージェントの設定	46
	VCS agent for DB2 の設定の検証	47
	DB2 インスタンスの詳細監視の設定	48
	DB2 インスタンスの詳細監視の有効化	49
	DB2 のコマンドラインからの詳細監視の有効化	49
	Db2udb リソースに対するインテリジェントなリソースの監視の有効化と無効化	50
	詳細監視中の DB2 エラーコードの処理	51
	DB2 用詳細監視の無効化	52
第 5 章	DB2 のための VCS サービスグループの管理	53
	DB2 のサービスグループの管理について	53
	DB2 サービスグループのオンライン化	53
	DB2 サービスグループのオフライン化	54
	DB2 サービスグループの切り替え	54
	DB2 用エージェントの無効化	54

付録 A	Veritas™ Cluster Server Agent for DB2 のトラブル シューティング	57
	DB2 用環境変数の db2profile の作成	57
	VCS agent for DB2 用 RestartLimit 属性の設定	58
	MPP モードの DB2 V9.5 と DB2 V9.7 のパーティションまたはノードの切り 替えシナリオ	58
付録 B	Veritas™ Cluster Server Agent for DB2 のリソース タイプ情報	59
	DB2 のリソースタイプ定義について	59
	Linux での DB2 UDB タイプの定義ファイル: Db2udbTypes.cf	59
付録 C	DB2 のリソースタイプ属性	61
	Db2udb リソースタイプと属性の定義について	61
付録 D	設定例	67
	Linux 上の VCS agent for DB2 用設定ファイルの例	67
	Linux での Veritas Cluster Server Agent for DB2 用非 MPP main.cf の設定	67
	Linux での VCS agent for DB2 の MPP main.cf の設定	70
索引	77

Veritas™ Cluster Server Agent for DB2 の概要

この章では以下の項目について説明しています。

- Veritas™ Cluster Server Agent for DB2 について
- Veritas™ Cluster Server Agent for DB2 が DB2 の高可用性を実現するしくみ
- DB2 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
- DB2 の VCS エージェントの のサポート対象ソフトウェア
- VCS Agent for DB2 のエージェント関数について
- VCS クラスタの標準的な DB2 設定
- VCS 環境に DB2 UDB を設定するためのロードマップ

Veritas™ Cluster Server Agent for DB2 について

Veritas™ Cluster Server Agent for DB2 for DB2 UDB (バージョン 6.0.1) は、リレーショナルデータベース管理システムの高可用性ソリューションです。

このマニュアルでは、DB2 UDB エージェント、エージェントの処理モードとその動作モード、およびその属性について説明します。このエージェントのインストールと設定の方法についても説明します。リソースが動作するサービスグループを管理する方法も説明します。

Veritas™ Cluster Server Agent for DB2 が DB2 の高可用性を実現するしくみ

エージェントはパーティションのレベルで DB2 UDB データベース処理を監視します。システムが失敗すれば、エージェントはエラーを検出し、該当するデータベースのパーティションをオフライン状態にします。VCS はクラスタの別のノードにエラーのインスタンスを移動します。ここでエージェントがデータベースのパーティション (複数可) をオンラインにします。

エージェントは、設定可能な、様々なレベルでの監視と様々な処理を実行します。またエージェントが実行できる多数のアクションを設定できます。

DB2 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 (インテリジェントな監視フレームワーク) 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。DB2 エージェントは IMF 対応であり、リソースの状態変化の通知に非同期監視フレームワーク (AMF) カーネルドライバを使います。

p.11 の「[VCS Agent for DB2 のエージェント関数について](#)」を参照してください。

DB2 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

インテリジェントな監視フレームワーク (IMF) では、VCS はポーリングベースの監視に加えてインテリジェントなリソース監視をサポートします。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。VCS agents for DB2 のインテリジェントなリソース監視機能は有効または無効にできます。

p.50 の「[Db2udb リソースに対するインテリジェントなリソースの監視の有効化と無効化](#)」を参照してください。

IMF 対応の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、DB2 エージェントは自身の `imf_register` 関数を使ってプロセスの 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、リソースのすべての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` 関数に通知すると、エージェントフレームワークは監視エージェント関数を実行し、リソースの状態を確認します。エージェントが状態の変化を VCS に通知すると、VCS は適切な処理を実行します。

詳しくは、『Veritas Cluster Server 管理者ガイド』を参照してください。

DB2 の VCS エージェントの のサポート対象ソフトウェア

Veritas™ Cluster Server Agent for DB2 (バージョン 6.0.1) は DB2 Universal Database Enterprise Server Edition (ESE) をサポートします。

ESE のマルチパーティションインスタンスでは、次の構成の両方をサポートします

- SMP (対称型マルチプロセッシング) ハードウェア構成
- MPP (Massively Parallel Processing) ハードウェア構成

メモリの必要条件は、使っている DB2 のバージョンによって異なります。メモリの必要条件については、該当する IBM DB2 のマニュアルを確認してください。

サポートされるプラットフォームのバージョンについて詳しくは、『Veritas Cluster Server リリースノート』を参照してください。

VCS Agent for DB2 のエージェント関数について

エージェントはデータベースに対してさまざまな操作または機能を実行できます。これらの機能には、オンライン、オフライン、監視、クリーン、情報、アクションがあります。アクションエージェント機能では、エージェントに対するカスタムアクションを追加できます。

カスタムアクションを定義する方法について詳しくは、『Veritas Cluster Server エージェント開発者ガイド』を参照してください。

VCS agent for DB2 のオンラインエージェント関数について

DB2 インスタンスまたはデータベースのパーティションを開始するためにエージェントは `db2gcf` プログラムを使います。コマンドは、次のようになります。

```
# su $DB2InstOwner -c "$InstHome/sqllib/bin/db2gcf -u -I¥  
$DB2InstOwner -p $nodenum"
```

このエージェントは、すべての DB2 パーティションを開始するために、`db2start` コマンドを使用することもできます。

VCS agent for DB2 のオフラインエージェント関数について

エージェントは DB2 データベースのパーティションを停止するために `db2gcf` プログラムを使います。1つのパーティションのみがあるデータベースでは、インスタンスを停止します。コマンドは、次のようになります。

```
# su $DB2InstOwner -c "$InstHome/sqllib/bin/db2gcf -d -I¥  
$DB2InstOwner -p $nodenum"
```

このエージェントは、すべての DB2 パーティションを停止するために、`db2stop` コマンドを使用することもできます。

DB2 の VCS エージェントの監視エージェント機能について

エージェントは `su $DB2InstOwner -c "$InstHome/sqlllib/bin/db2gcf -s -I $DB2InstOwner -p $nodenum"` コマンドを実行し、ノード番号やデータベースパーティションの状態を確認します。`db2gcf` コマンドの `exit` ステータスが 0 である場合、`monitor` は `exit` コード 110 を返します。0 ではない場合は、`exit` コード 100 を返し、リソースはオフライン状態になります。エージェントはリソースを再起動、またはフェールオーバーします。この処理は `RestartLimit` または `ToleranceLimit` などの他のタイプ独立属性によって、決まります。

詳細監視を行う場合は、`IndepthMonitor` 属性に 1 を設定します。エージェントは、`/opt/VRTSagents/ha/bin/Db2udb` ディレクトリで `monitor_custom_$db2instance_$nodenum` ファイルを検索します。ファイルが存在し、実行可能な場合、エージェントはカスタマイズされたこの詳細監視ファイルを実行します。`/etc/VRTSagents/ha/conf/Db2udb/sample_db2udb` ディレクトリに、カスタム監視スクリプトの例があります。

カスタム監視にエラーや問題があると、`Db2udb` エージェントの `WarnOnlyIfDBQueryFailed` 属性の値が確認されます。`db2error.dat` ファイルが `/opt/VRTSagents/ha/bin/Db2udb` ディレクトリ内にある場合、エージェントはこのファイルを調べ、エラー設定に従ってエラーを処理します。

p.51 の「[詳細監視中の DB2 エラーコードの処理](#)」を参照してください。

`WarnOnlyIfDBQueryFailed` の属性を 1 (デフォルト) に設定し、`Notifie` リソースを設定した場合は、エージェントは次の処理を実行します。

- 通知を送信します
- `exit` コード 110 が戻ります

`WarnOnlyIfDBQueryFailed` 属性を 0 に設定した場合、エージェントは `db2error.dat` ファイルのエラー処理を実行します。エラーの処理を実行するためにファイルが必要であることを注意します。ファイルがなければ、デフォルトである `exit` コード 100 が戻ります。

VCS Agent for DB2 クリーンエージェント機能について

このエージェントは最初に `force` オプションを付けて `db2stop` コマンドを使用します。コマンドでエージェント機能をクリーンアップできない場合、次にエージェントは `db2_kill` スクリプトを使用してインスタンスを強制終了します。`db2_kill` スクリプトが失敗した場合、`kill -9` コマンドが呼び出されます。

```
# su $DB2InstOwner -c "$InstHome/sqlllib/bin/db2stop force
nodenum $nodenum"
```

VCS agent for DB2 用情報エージェント機能について

このエージェントは、データベースパーティションや重要なプロセスに関する静的な情報や動的な情報を提供する、情報機能をサポートしています。

情報エージェント機能について詳しくは、次のマニュアルを参照します。

- Veritas Cluster Server 管理者ガイド
- Veritas Cluster Server エージェント開発者ガイド

p.14 の「[VCS Agent for DB2 データベース情報を取得するための情報エージェント機能の実行について](#)」を参照してください。

アクションエージェント機能について VCS Agent for DB2

DB2 エージェントは、リソースに対して定義済みのアクションまたはカスタムアクションを実行できる、アクション機能をサポートしています。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]
```

エージェントは、次の事前定義アクションをサポートします。

- **VRTS_GetInstanceName** トークンは設定された Db2udb リソースの DB2 インスタンス名を取得します。
- **VRTS_GetRunningServices** のトークンはエージェントが Db2udb のリソースのために監視する処理のリストを取り込みます。

次に例を示します。

```
# hares -action db2udb1 VRTS_GetInstanceName -sys systemName
```

```
VCS NOTICE V-16-13323 Resource (db2udb1): action  
(VRTS_GetInstanceName) completed successfully. Output is:
```

```
db2inst1
```

```
# hares -action db2udb0 VRTS_GetRunningServices -sys systemName
```

```
VCS NOTICE V-16-13323 Resource (db2udb0): action  
(VRTS_GetRunningServices) completed successfully. Output is:
```

```
PARTITION: 0
```

```
PID TTY TIME CMD
```

```
9800 - 0:06 db2sysc
```

VCS Agent for DB2 の IMF の統合機能について

次のセクションでは、VCS Agent for DB2 の IMF の統合機能について説明します。

VCS agent for DB2 の imf_init 関数について

DB2 エージェントの IMF 通知モジュールである AMF カーネルドライバと連動するようにエージェントを初期化します。This function エージェントが起動すると実行されます。

VCS agent for DB2 の imf_register 関数について

リソースエンティティの AMF カーネルモジュールへの登録および登録解除を行います。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。

VCS agent for DB2 の imf_getnotification 関数について

リソースの状態の変更についての通知を取得します。この関数は、AMF カーネルモジュールでエージェントが初期化された後に実行されます。この関数は継続的に通知を待機し、通知後にリソースでアクションを実行します。

VCS Agent for DB2 データベース情報を取得するための情報エージェント機能の実行について

データベース情報を返すために情報エージェント機能を実行できます。この例では、情報エージェント機能はデータベース情報を取り込みます。

データベースの情報を取り込むには

- 1 設定を読み書き両用にします。

```
# haconf -makerw
```

- 2 情報エージェント機能を実行する一定の間隔を秒単位で指定します。このデフォルト値は 0 であり、情報エージェント機能を実行しないことを示します。

```
# hatype -modify Db2udb InfoInterval 300
```

このコマンドでは、Db2udb は DB2 リソースタイプの名前です。情報エージェント機能が情報のスクリプトを実行した後の時間が InfoInterval 300 (秒) です。スクリプトはエージェントが監視するプロセスの情報を取得します。

- 3 要求された **ResourceInfo** 値を表示します。次の出力例はエージェントが DB2 リソースに対して監視するプロセスを示します。

ResourceInfo が 300 秒(5分)ごとに更新することに注意してください。以前の手順で **InfoInterval** に 300 を設定したためです。

```
# hares -value db2udb1 ResourceInfo
State Valid
Msg
PARTITION: 0
      PID  TTY  TIME CMD
      413924  -   0:00 db2sysc
TS Fri Jan 14 18:11:52 2011
```

- 4 設定を永続的にします。

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

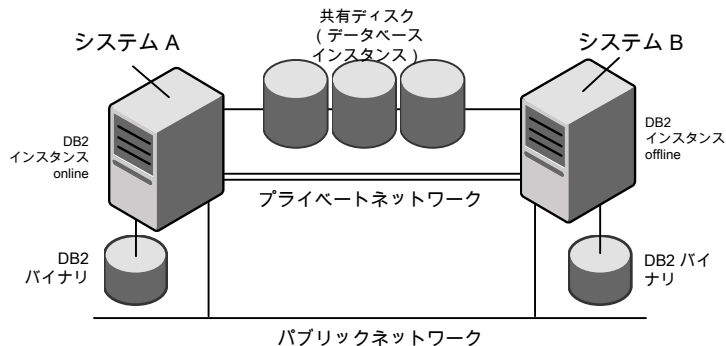
VCS クラスタの標準的な DB2 設定

DB2 UDB は 2 つのシステムクラスタで設定されます。DB2 UDB システムバイナリは、**sysA**と**sysB**上のローカルファイルシステムでまったく同じになるように、インストールします。インスタンスホームディレクトリ、インスタンスバイナリおよびデータベースは、両方のノードから利用できるように共有ストレージにインストールします。

非 MPP 構成の場合には、インスタンスがオンラインになるのは一度に 1 つのシステムのみです。他のシステムはフェールオーバーシステムです。

図 1-1 は非 MPP 構成での DB2 インストールを示します。

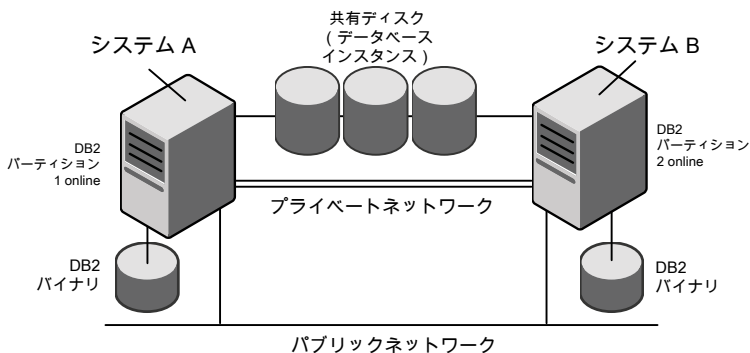
図 1-1 非 MPP 構成での DB2 インストール



MPP 構成の場合、データベースパーティションは各システム上で実行でき、各システムはフェールオーバーシステムになることができます。

図 1-2 は MPP 構成での DB2 インストールを示します。

図 1-2 MPP 構成での DB2 インストール



VCS 環境に DB2 UDB を設定するためのロードマップ

次のタスクと情報を確認します。

- サポート対象のソフトウェアの確認
- エージェント機能の確認
p.11 の「[VCS Agent for DB2 のエージェント関数について](#)」を参照してください。
- エージェントの必要条件の確認
p.17 の「[DB2 をインストールするための VCS の必要条件](#)」を参照してください。
- DB2 のインストールと設定
p.22 の「[VCS 環境での DB2 のインストール](#)」を参照してください。
p.23 の「[DB2 の設定](#)」を参照してください。
- DB2 の VCS エージェントのインストール
p.30 の「[DB2 ソフトウェアの VCS エージェントのインストール](#)」を参照してください。
- DB2 の VCS エージェントのサービスグループの設定
p.40 の「[Cluster Manager \(Java コンソール\)からの VCS Agent for DB2 の設定](#)」を参照してください。
- 省略可能なオプションとして、詳細監視と自動化された処理の設定
p.48 の「[DB2 インスタンスの詳細監視の設定](#)」を参照してください。
- サービスグループをオンラインにします。
p.53 の「[DB2 サービスグループのオンライン化](#)」を参照してください。

DB2 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。

- [DB2 をインストールするための VCS の必要条件](#)
- [VCS 環境での DB2 のインストール](#)
- [DB2 の設定](#)

DB2 をインストールするための VCS の必要条件

インストールする前に、次のタスクを実行します。

- 非 MPP 構成 DB2 UDB をインストールするための準備を確認します。
[p.17 の「非 MPP 構成 DB2 UDB をインストールするための準備」](#)を参照してください。
- MPP 構成 DB2 UDB をインストールするための準備を確認します。
[p.18 の「MPP 構成 DB2 UDB をインストールするための準備」](#)を参照してください。
- 非 MPP インスタンス用のファイルシステムを作成します。
[p.19 の「DB2 の非 MPP インスタンス用のファイルシステムの作成」](#)を参照してください。
- MPP インスタンス用の共有ファイルシステムを作成します。
[p.19 の「DB2 の MPP インスタンス用の共有ファイルシステムの作成」](#)を参照してください。
- ユーザーアカウントとグループアカウントを定義します。
[p.20 の「DB2 用ユーザーアカウントとグループアカウントの定義」](#)を参照してください。

非 MPP 構成 DB2 UDB をインストールするための準備

次の事項を確認します。

- クラスタ内のすべてのノードに DB2 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。
- DB2 をインストールするための VCS の必要条件を満たしていることを確認してください。
- DB2 UDB のユーザーアカウントとグループアカウントを定義します。
p.20 の「[DB2 用ユーザーアカウントとグループアカウントの定義](#)」を参照してください。
使用している DB2 UDB の固有バージョンについては、該当する DB2 UDB マニュアルを参照してください。
- DB2 UDB システムバイナリをローカルにインストールします。
- 共有ストレージに DB2 UDB データベースのインスタンスをインストールします。
- クラスタのすべてのノードで VCS バージョン 6.0.1 をインストールし、設定します。インストール手順については、『Veritas Cluster Server インストールガイド』を参照してください。

MPP 構成 DB2 UDB をインストールするための準備

次の事項を確認します。

- DB2 UDB を実行するために十分なリソースがすべてのシステムにあることを確認します。このようなリソースの例として共有メモリがあります。メモリの必要条件を確認します。これは、DB2 のバージョンやハードウェア構成により異なります。
- DB2 UDB のユーザーアカウントとグループアカウントを定義します。
p.20 の「[DB2 用ユーザーアカウントとグループアカウントの定義](#)」を参照してください。
使用している DB2 UDB の固有バージョンについては、該当する DB2 UDB マニュアルを参照してください。
- 各システムのローカルファイルシステムに DB2 UDB システムバイナリをインストールします。
- 共有ストレージに DB2 UDB データベースのインスタンスをインストールします。
- クラスタのすべてのノードで VCS バージョン 6.0.1 をインストールし、設定します。インストール手順については、『Veritas Cluster Server インストールガイド』を参照してください。
- MPP 構成には、Storage Foundation Cluster File System ソフトウェアが必要です。このソフトウェアに次のコンポーネントが含まれます。
 - Veritas Cluster Server (VCS)
 - クラスタ機能が有効である Veritas Volume Manager (CVM)

『Veritas Storage Foundation Cluster File System High Availability インストールガイド』

- クラスタ機能が有効である Veritas File System (CFS)

『Veritas Storage Foundation Cluster File System 管理者ガイド』

DB2 の非 MPP インスタンス用のファイルシステムの作成

非 MPP インスタンス用のファイルシステムを作成するには、最初に物理共有ディスクにディスクグループを作成します。それからディスクグループ内に十分なサイズのボリュームを作成します。

非 MPP インスタンス用のファイルシステムを Linux 上に作成するには

1 # vxvg init db2db_dg sdc

ディスクグループをデポートし、インポートします。

```
# vxvg deport db2db_dg
```

```
# vxvg import db2db_dg
```

2 vxassist コマンドを使ってボリュームを作成します。

```
# vxassist -g db2db_dg make db2db_vol 3g
```

3 ファイルシステムを作成します。

```
# mkfs -t vxfs /dev/vx/dsk/db2db_dg/db2db_vol
```

4 マウントポイントディレクトリを作成し、ファイルシステムをマウントします。共有ストレージではなく、ローカルファイルシステムのクラスタにあるすべてのノードにマウントポイントがあることを確認します。

```
# mkdir /db2_mnt/db2inst1
```

```
# mount -t vxfs /dev/vx/dsk/db2db_dg/db2db_vol ¥  
/db2_mnt/db2inst1
```

DB2 の MPP インスタンス用の共有ファイルシステムの作成

Storage Foundation Cluster File System ソフトウェアをインストールします。MPP インスタンス用の共有ファイルシステムを作成するには、物理共有ディスクに共有ディスクグループを作成します。それからディスクグループ内に十分なサイズのボリュームを作成します。

MPP インスタンス用の共有ファイルシステムを Linux 上に作成するには

- 1 どれかのノードから、ディスクグループを作成します。vxdisk list コマンドを使って、ディスクを一覧表示します。
- 2 共有ディスクグループを作成します。この場合は、グループを1つのディスクで構成します。この例では、ディスクは **sd**c です。

```
# vxdg -s init db2db_dg sdc
```

- 3 ディスクグループをデポートし、インポートします。

```
# vxdg deport db2db_dg
# vxdg -s import db2db_dg
```

- 4 vxassist コマンドを使って、ボリュームを作成します。

```
# vxassist -g db2db_dg make db2db_vol 7g
```

- 5 ファイルシステムを作成します。

```
# mkfs -t vxfs -o largefiles /dev/vx/rdisk/db2db_dg/db2db_vol
```

- 6 マウントポイントディレクトリを作成し、ファイルシステムをマウントします。

```
# mkdir /db2_mnt/db2inst1
# mount -t vxfs -o cluster /dev/vx/dsk/db2db_dg/db2db_vol ¥
/db2_mnt/db2inst1
```

DB2 用ユーザーアカウントとグループアカウントの定義

DB2 UDB のバイナリをインストールしてインスタンスを作成する前に、各システム上の各インスタンスに対して DB2 UDB のユーザーアカウントとグループアカウントを定義する必要があります。

次の必要条件に注意してください。

- DB2 のユーザーとグループの ID は、クラスタ内のすべてのノードで同一になるようにしてください。
- DB2 インスタンス所有者のホームディレクトリは各ノードからアクセス可能である必要があります。このディレクトリは DB2 インスタンスが使用するマウントポイントです。マウントするデータベースは共有ストレージになければなりません。すでに作成されていなければ、各ノードのマウントポイントディレクトリをローカルに作成します。
- すべての DB2 ユーザーアカウントがローカルシステム上に存在している必要があります。NIS または NIS+ の使用は、これらのサービスは高可用性に欠ける場合があります。

め、お勧めしません。またこれらのサービスが中断されると、VCS が正しく動作できない場合があります。

DB2 のユーザーグループのアカウントの作成

クラスタ内のノードにはそれぞれ 3 つのユーザーグループアカウントが必要です。

Linux のクラスタ内の各ノードでグループアカウントを作成するには

- 1 DB2 インスタンス所有者のグループを作成します。次のように実行します。

```
# groupadd -g 999 db2iadml
```

- 2 ブロックされたユーザー定義関数 (UDF: User-Defined Function) またはストアードプロシージャを実行するユーザーのためのグループを作成します。次のように実行します。

```
# groupadd -g 998 db2fadm1
```

- 3 DB2 管理サーバー (DAS) のグループを作成します。次のように実行します。

```
# groupadd -g 997 dasadm1
```

Linux での DB2 用ユーザーアカウントの追加

ユーザーアカウントを作成する次の例では、以下のオプションを使うことができます。

- **-g** オプションはグループを指定します
- **-u** オプションはユーザー ID を指定します
- **-d** オプションはホームディレクトリを指定します
- **-m** オプションはホームディレクトリを作成します (存在しない場合)
- **-s** オプションはユーザーのログオンのシェルです
- 最終的な式はユーザーのログオンです

クラスタ内の各ノードでユーザーアカウントを作成します。

この例では、DB2 UDB のインスタンス所有者である **db2inst1** を作成する方法を示します。インスタンスのホームディレクトリはマウントポイント **/db2_mnt/db2inst1** でもあります。DB2 UDB のインスタンスのホームディレクトリはすべてのノードに存在する必要があります。次に例を示します。

```
# useradd -g db2iadml -u 1004 -d /db2_mnt/db2inst1 -m -s ¥  
/bin/ksh db2inst1
```

次の例は、**db2fenc1** と **dasusr1** のユーザーアカウントの作成方法を示します。これらのユーザーのホームディレクトリは、各ノードのローカルファイルシステムの **/home** の下に置かれます。

```
# useradd -g db2fadm1 -u 1003 -d /home/db2fenc1 -m -s ¥  
/bin/ksh db2fenc1  
# useradd -g dasadm1 -u 1002 -d /home/dasusr1 -m -s /bin/ksh dasusr1
```

MPP モードの DB2 設定では、マウントポイント **/db2_mnt/db2inst1** を使うファイルシステムは、DB2 UDB インスタンスのホームディレクトリをホストするために **Cluster File System** を使います。

VCS 環境での DB2 のインストール

VCS 環境で DB2 をインストールするために Symantec は IBM DB2 UDB の関連マニュアルのインストール手順に従うことを推奨します。

各ノードのローカルディスク上にバイナリをインストールし、各クラスターノードからアクセス可能な共有ストレージ上にデータベースインスタンスをインストールします。

VCS 環境に DB2 をインストールするには

- 1 共有メモリパラメータを設定します。メモリ必要条件が満たされることを確かめるために関連した **IBM DB2 UDB** マニュアルを参照します。
- 2 バイナリをインストールします。DB2 UDB システムのバイナリは、共有ストレージではなく、各ノードのローカルディスクにインストールします(ミラー化したディスクへのインストールをお勧めします)。IBM の **db2setup** ツールを使用できます。
- 3 DB2 ライセンスをインストールします。各ノードに DB2 ライセンスをインストールします。次のように実行します。

```
# /opt/IBM/db2/V9.7/adm/db2licm -a db2ese.lic
```

- 4 インスタンスをインストールします。インスタンスのホームディレクトリがマウントされているいずれか 1 つのノードで、共有ストレージにデータベース インスタンスをインストールします。シングルパーティションインスタンスをインストールすることも、マルチパーティションインスタンスをインストールすることもできます。IBM の **db2setup** ツールを使用できます。

インストール時、次の点に留意してください

- **db2setup** を使う場合は、[DB2 インスタンスのプロパティ(DB2 Instance Properties)] ウィンドウで、[システム・ブート時に DB2 インスタンスを自動開始してください。(Auto start DB2 instance at system boot)] オプションを選択しないでください。このオプションはすべての DB2 バージョンに必ずしも存在しません。インスタンスをオンラインにする前に、VCS は特定の順序で DB2 インスタンスのリソースを起動する必要があります。

- インスタンスのホームディレクトリは、ローカルシステム上のマウントポイントです。

メモ: DB2 のインストールについて詳しくは、IBM のマニュアルを参照してください。

DB2 の設定

VCS 環境で DB2 UDB を設定するために、次の手順を使用します。

次のタスクを実行する必要があります。

- `/etc/services` を確認します。
p.23 の「[DB2 用 /etc/services の確認](#)」を参照してください。
- すべてのクラスタノードの DB2 ユーザーに対してパスワードなし通信を有効にします。
p.24 の「[すべてのクラスタノードの DB2 ユーザーに対するパスワードなし通信の有効化](#)」を参照してください。
- DB2 の `$DB2InstHome/sqllib/db2nodes.cfg` ファイルを変更します。
p.24 の「[DB2 の \\$DB2InstHome/sqllib/db2nodes.cfg ファイルの変更](#)」を参照してください。
- DB2 インストールの設定を確認します。
p.27 の「[DB2 インストールの確認](#)」を参照してください。

DB2 用 /etc/services の確認

クラスタ内の各システムで、`more` コマンドを使って `/etc/services` ファイルを確認します。

以下の点を忘れずに実行してください。

- 各パーティションにポート番号が設定されていることを確認します。予約されているポートの数は、パーティションの数によって異なります。
- 他のサービスがポートを使用しないことを確認します。必要に応じて手動で新たに番号を設定します。
- クラスタ内のすべてのシステムの `/etc/services` ファイルに同じエントリが用意されていることを確認してください。

次は、2 つの DB2 インスタンス (`db2inst1` と `db2inst2`) の例です。どちらのインスタンスにも 2 つのパーティションがあります。各インスタンスには、パーティションごとに 2 つのポートと 1 つの予備ポートが必要であり、したがってインスタンスごとに 4 つの回線が必要です。

```
# more /etc/services
DB2_db2inst1      60000/tcp
DB2_db2inst1_1   60001/tcp
```

```
DB2_db2inst1_2    60002/tcp
DB2_db2inst1_END  60003/tcp
DB2_db2inst2     60004/tcp
DB2_db2inst2_1   60005/tcp
DB2_db2inst2_2   60006/tcp
DB2_db2inst2_END 60007/tcp
```

このファイルで、重複して設定されているポート番号がないことを確認してください。

すべてのクラスタノードの DB2 ユーザーに対するパスワードなし通信の有効化

デフォルトでは、リモートのデータベースパーティションを起動するときは、DB2 データベースシステムは通信プロトコルとして **rsh** を使います。リモートパーティションの起動時にパスワードを要求する画面が表示されないように、パスワードを要求しない **rsh** を設定します。システム通信で **ssh** を使う場合は、パスワードやパスフレーズを要求せずに動作するよう設定する必要があります。

DB2 の \$DB2InstHome/sqllib/db2nodes.cfg ファイルの変更

DB2 は、ノード間のフェールオーバー中に \$DB2InstHome/sqllib/db2nodes.cfg ファイルを使います。

非 MPP 構成の \$DB2InstHome/sqllib/db2nodes.cfg ファイルを変更するには

- 1 \$DB2InstHome/sqllib/db2nodes.cfg ファイルを開きます。
- 2 データベースパーティションのエントリを追加します。
- 3 IPv4 の場合、ホスト名として仮想 IP アドレスを割り当てます。たとえば、この手順と前の手順で、次のように指定します。

```
0 virtualhostname 0
1 virtualhostname 1
```

virtualhostname は、`/etc/hosts` ファイルの仮想 IP アドレスに対応します。仮想 IP アドレスが起動し、動作することを確認します。

- 4 IPv6 の場合、DB2 には `db2nodes.cfg` のホスト名エントリがノードのホスト名に対応している必要があるという制限があります。たとえば、ホスト名が `sysA-v6` の場合、`db2nodes.cfg` に `sysA-v6` が含まれている必要があります。

VCS では、仮想ホスト名を使用します。サービスグループが別のノードにフェールオーバーする場合、フェールオーバー先のノードに、仮想 IP アドレスが提供されます。`db2nodes.cfg` のホスト名エントリがノードのホスト名に対応する必要がある制限の回避策として、次のような設定を使用できます。

各ノードで、`$DB2InstHome` と同じアクセス権限で `/local/$DB2InstHome/db2nodes.cfg` ファイルを作成し、`$DB2InstHome/sqlllib/db2nodes.cfg` からこのファイルへのシンボリックリンクを作成します。

```
# mkdir -p /local/$DB2InstHome/  
# ln -s /local/$DB2InstHome/db2nodes.cfg ¥  
$DB2InstHome/sqlllib/db2nodes.cfg
```

パス `/local/$DB2InstHome/db2nodes.cfg` は各システムにローカルです。各ノードの `/local/$DB2InstHome/db2nodes.cfg` に IPv6 ホスト名を入力します。

たとえば、ノード `sysA-v6` と `sysB-v6` で、それぞれの `db2nodes.cfg` ファイルに次のように入力します。

■ `sysA-v6` の場合

```
0 sysA-v6 0
```

■ `sysB-v6` の場合

```
0 sysB-v6 0
```

MPP 構成の \$DB2InstHome/sql/lib/db2nodes.cfg ファイルを変更するには

- 1 MPP 構成の場合、各データベースパーティションを起動するホスト名を使って \$DB2InstHome/sql/lib/db2nodes.cfg ファイルを変更します。DB2 は、データベースパーティションをノード間でフェールオーバーできるように、db2nodes.cfg ファイルを自動的に変更し、更新します。DB2 は「netname」のための第 4 列を追加します。netname は、Fast Communication Manager (FCM) の通信のための高速相互接続の IP アドレスです。netname は、データベースパーティションの通信に高速相互接続を使う場合にのみ必要です。netname が指定されていない場合は、デフォルトでホスト名が使われます。複数のデータベースパーティションを設定する場合、仮想 IP は、MPP 構成の db2nodes.cfg ファイルでは使われません。

高速相互接続 (FCM スイッチ) が使われない場合は、次の例のようになります。

```
0 sysA 0
1 sysB 0
2 sysC 0
3 sysD 0
```

パーティション間の通信に高速相互接続 (FCM スイッチ) が使われる場合は、次の例のようになります。

```
0 sysA 0 switchA
1 sysB 0 switchB
2 sysC 0 switchC
3 sysD 0 switchD
```

3 列目で指定する相対ポート番号が、ホスト上の各パーティションで一意であるようにします。高速相互接続 (FCM スイッチ) が使われない場合は、次の例のようになります。

```
0 sysA 0
1 sysA 1
2 sysB 0
3 sysC 0
4 sysD 0
```

パーティション間の通信に高速相互接続 (FCM スイッチ) が使われる場合は、次の例のようになります。

```
0 sysA 0 switchA
1 sysA 1 switchA
2 sysB 0 switchB
3 sysC 0 switchC
4 sysD 0 switchD
```

- 2 高速通信スイッチや **netname** がパーティション間通信のために設定されている場合、**db2start** または **db2gcf** コマンドを使ってパーティションをオンラインにする前に、**db2greg** コマンドを使って **db2** グローバルレジストリに **switchname** と **netname** のエントリを追加します。次のように、**DB2** インスタンスのホームディレクトリから **db2profile** をソースとして指定した後、**db2greg** コマンドを実行します。

```
# source /db2_mnt/db2inst1/sqlllib/db2profile
```

```
# db2greg -addvarrec variable=switchname, value=$switch
```

switchname のエントリが **db2** グローバルレジストリであることを検証するには、次のコマンドを使います。

```
# db2greg -dump
```

DB2 インストールの確認

共有ファイルシステムがマウントされるホストで各インスタンスを開始し、停止できるかどうかを調べます。DB2 インストールを確認するためにこの手順をします。

DB2 インスタンスの起動と停止が可能かどうかを確認するには

- 1 インスタンス所有者としてログインします。

```
# su - db2inst1
```

- 2 IPv6 環境で DB2 を設定する場合、変数 **DB2FCMCOMM** の設定を適用します。

```
# db2set DB2FCMCOMM=TCPIP6
```

- 3 インスタンスの起動を試みます。

```
$ db2start
```

DB2 は **db2nodes.cfg** のファイルのパーティションで開始します。DB2 が開始しない場合、エラーコードを調べます。

- 4 前のコマンドが成功した場合、インスタンスを停止します。

```
$ db2stop
```

- 5 各ノードでアプリケーションが正しく停止しない場合は、設定エラーを調べます。DB2 UDB のマニュアルでエラーコードを確認します。

- 6 データベースを作成します。

```
$ db2 create database dbname
```

- 7 データベースディレクトリを表示します。

```
$ db2 list database directory
```

クラスタ内のその他の DB2 設定を確認するには

- 1 VCS クラスタの各ノードごとにディスクグループをインポートし、ディスクグループ内のすべてのボリュームを起動します。
- 2 DB2 インスタンスとデータベースを含むボリュームのファイルシステムをマウントします。
- 3 ディスクグループをマウント解除し、デポートします。
- 4 クラスタ内の各ノードでこの手順を繰り返します。

Veritas™ Cluster Server Agent for DB2 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。

- [VCS Agent for DB2](#) をインストールまたはアップグレードする前に
- DB2 ソフトウェアの [VCS エージェントのインストール](#)
- [VCS Agent for DB2](#) のアップグレード
- [VCS Agent for DB2](#) の無効化
- [VCS Agent for DB2](#) の削除

VCS Agent for DB2 をインストールまたはアップグレード する前に

Veritas High Availability Agent for DB2 をインストールまたはアップグレードする前に次の前提条件を満たす必要があります。

- DB2 をインストールするための VCS の必要条件を満たします。
p.17 の「[DB2 をインストールするための VCS の必要条件](#)」を参照してください。
- VCS 環境に DB2 をインストールします。
p.22 の「[VCS 環境での DB2 のインストール](#)」を参照してください。
- クラスタに VCS がインストールされていることを確認します。
- DB2 がインストールおよび設定されていることを確認します。

- VCS Single Cluster Manager (Java コンソール) は VCS のパッケージに含まれなくなりました。マルチクラスタ環境での管理、監視、レポートには、Veritas Operations Manager (VOM) を使うことを推奨します。VOM は、<http://go.symantec.com/vom> からダウンロードできます。引き続き VCS Single Cluster Manager を使う場合は、http://go.symantec.com/vcsm_download の Web サイトから無料で取得できます。

DB2 ソフトウェアの VCS エージェントのインストール

Veritas™ Cluster Server Agent for DB2 のエージェントバイナリは VRTSvcsea RPM に含まれています。VCS のインストール時に、すべてのパッケージまたは推奨されるパッケージをインストールするように選択した場合、VRTSvcsea RPM はすでにインストールされています。VCS のインストール時に最小限のパッケージをインストールした場合は、VRTSvcsea を手動でインストールする必要があります。

Veritas™ Cluster Server Agent for DB2 は製品ディスクからインストールできます。DB2 サービスグループをホストする予定のすべてのノードに VCS Agent for DB2 エンタープライズエージェントをインストールする必要があります。

Linux 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。
コマンドで、DB2 の VCS エージェントが返された場合、この手順を飛ばすことができます。
- 3 ノードに接続されているドライブに、適切なソフトウェアディスクを挿入します。
このソフトウェアは、ディスクを `/mnt/cdrom` として自動的にマウントします。
- 4 ディスクが自動的にマウントされない場合は、ディスクを手動でマウントする必要があります。次に例を示します。

```
# mount -o ro /dev/cdrom/mnt/cdrom
```

- 5 エージェントソフトウェアをインストールします

- RHEL6(x86_64) の場合

```
# cd /mnt/cdrom/rhel6_x86_64/rpms
```

```
# rpm -i VRTSvcsea-6.0.100.000-GA_RHEL6.i686.rpm
```

- RHEL5 または OEL5(x86_64) の場合:

```
# cd /mnt/cdrom/rhel5_x86_64/rpms
```

```
# rpm -i VRTSvcsea-6.0.100.000-GA_RHEL5.i686.rpm
```

■ SLES10(x86_64)の場合:

```
# cd /mnt/cdrom/sles10_x86_64/rpms
```

```
# rpm -i VRTSvcsea-6.0.100.000-GA_SLES10.i586.rpm
```

■ x86_64 上の SLES11:

```
# cd /mnt/cdrom/sles11_x86_64/rpms
```

```
# rpm -i VRTSvcsea-6.0.100.000-GA_SLES11.i686.rpm
```

VRTSvcsea RPM には、DB2 エージェントのエージェントバイナリが含まれていません。RPM は、Oracle と Sybase の VCS エージェントもインストールします。

- 6 DB2 サービスグループの一部にする各ノードに対して、ステップ 1 からステップ 4 までを繰り返します。

VCS Agent for DB2 のアップグレード

以前の設定情報を保存しないでアップグレードするには、エージェントを無効にし、削除してから再インストールする必要があります。

アップグレードして以前の設定情報を再利用する手順を次に示します。

DB2 エージェントからアップグレードするには

- 1 エージェントを無効にして削除します。
詳しくは以下を参照してください。
 - DB2 用エージェントの無効化
p.32 の「[VCS Agent for DB2 の無効化](#)」を参照してください。
 - DB2 UDB 用エージェントの削除
p.33 の「[VCS Agent for DB2 の削除](#)」を参照してください。
- 2 DB2 用の HA エージェントが収録されているディスクから、新しいパッケージを追加します。
p.22 の「[VCS 環境での DB2 のインストール](#)」を参照してください。

- 3 Db2udbTypes.cf ファイルを /etc/VRTSagents/ha/conf/Db2udb/ ディレクトリから /etc/VRTSvcs/conf/config ディレクトリにコピーしていることを確認します。

新しくコピーされた Db2udbTypes.cf ファイルを、古い Db2udbTypes.cf ファイルに行ったすべてのタイプレベルの変更で更新します。たとえば、MonitorInterval 属性の値をデフォルトの 60 秒から 120 秒に変更している場合、Db2udbTypes.cf ファイルを更新します。新しくコピーした Db2udbTypes.cf ファイルに、これらのタイプレベルの変更を手動で適用する必要があります。

- 4 詳細監視の使用を続行するには、カスタム監視スクリプトの例または任意のユーザー定義スクリプトを使います。

p.51 の「[詳細監視中の DB2 エラーコードの処理](#)」を参照してください。

VCS Agent for DB2 の無効化

Veritas™ Cluster Server Agent for DB2 を無効にするには、VCS Agent for DB2 サービスグループを OFFLINE 状態に変更する必要があります。アプリケーションを完全に停止したり、エージェントを他のシステムに切り替えることが可能です。

エージェントを無効にするには

- 1 サービスグループの SystemList からシステムを削除するには、そのサービスグループがオンラインになっているかどうかを確認します。

```
# 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。

- あるノードでサービスグループをオフラインにし、別のノードでオンラインにするには、**-switch** オプションを使えます。

```
# hagrps -switch service_group -to system_name
```

- サービスグループをオフラインにして、クラスタ内の他のノードでオンラインにしない場合は、次のように入力します。

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

- 3 ノード上のエージェントを停止します。

```
# haagent -stop Db2udb -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 すべてのノード上のエージェントを無効にしたら、サービスグループまたはリソースタイプ、またはその両方を VCS 設定から削除できます。
詳しくは、『Veritas Cluster Server 管理者ガイド』を参照してください。

VCS Agent for DB2 の削除

エージェントを削除する前に、すべてのノードで無効にする必要があります。エージェントパッケージ `VRTSvcsea` には DB2、Oracle、Sybase 用の VCS エージェントが含まれています。次の手順で、すべての VCS エージェントを削除します。

p.32 の「[VCS Agent for DB2 の無効化](#)」を参照してください。

ノードからエージェントを削除するには

- ◆ エージェントがある各システムで、次のように入力します。

```
# rpm -e VRTSvcsea
```


DB2 のための 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。

- [DB2 用サービスグループの設定について](#)
- [VCS の DB2 の設定について](#)
- [DB2 サービスグループを設定する前に](#)
- [Cluster Manager \(Java コンソール\)からの VCS Agent for DB2 の設定](#)
- [main.cf ファイルの編集による VCS Agent for DB2 の設定](#)
- [DB2 インスタンスの詳細監視の設定](#)

DB2 用サービスグループの設定について

DB2 サービスグループを設定するときには、DB2 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。

エージェントは、次の方法で設定できます。

- VCS Cluster Manager (Java コンソール)を使います。
VCS Cluster Manager (Java コンソール)を使って Veritas™ Cluster Server Agent for DB2 のサービスグループテンプレートを編集できます。
[p.40 の「Cluster Manager \(Java コンソール\)からの VCS Agent for DB2 の設定」](#)を参照してください。
- Veritas Operations Manager を使います。
詳しくは、Veritas Operations Manager のマニュアルを参照してください。
- VCS コマンドを使います。

コマンドラインから、エージェントを設定できます。コマンドラインからサービスグループを設定する方法については、『Veritas Cluster Server 管理者ガイド』を参照してください。

- **main.cf** ファイルを編集します。
main.cf ファイルを編集できます。この方法では、編集前にVCSを停止し、編集後に再起動して新しい設定を有効にする必要があります。
main.cf ファイルについて詳しくは、次のマニュアルを参照してください。
 - `hacf -verify` コマンドで変更された **main.cf** ファイルが動作するかどうかを確認できます。このコマンドについて詳しくは、『Veritas Cluster Server 管理者ガイド』を参照してください。
 - **main.cf** ファイルの編集による Veritas™ Cluster Server Agent for DB2 の設定 p.44 の「**main.cf** ファイルの編集による VCS Agent for DB2 の設定」を参照してください。

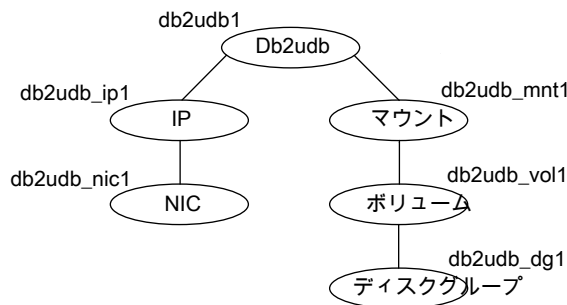
VCS の DB2 の設定について

MPP と非 MPP 構成の DB2 のサービスグループを設定できます。

DB2 用非 MPP 構成サービスグループ

図 4-1 は、非 MPP 構成 DB2 インスタンスリソースグループ用に設定されたリソース間の依存関係を示しています。

図 4-1 DB2udb リソースの依存関係図



この設定は Db2udb リソースのサービスグループを示します。db2udb1 リソース(データベース)には、IP リソースと Mount リソースが必要です。DB2 サーバーのサービスグループの IP アドレスは、IP リソース (db2udb_ip1) と NIC リソース (db2udb_nic1) を使って設定されます。Mount リソース (db2udb_mnt1) は Volume リソース (db2udb_vol1) に依存し、Volume リソースは DiskGroup リソース (db2udb_dg1) に依存しています。これらのリソースのそれぞれが利用可能になると、サービスグループを開始できます。

DB2 用 MPP 構成サービスグループ

DB2 のエージェントは MPP 構成をサポートするために 2 個のサービスグループを使用します。

これらのサービスグループは次のとおりです。

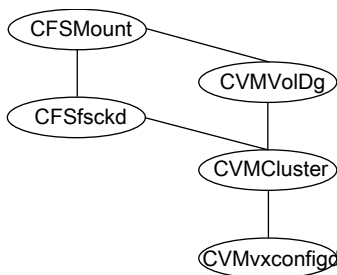
- パラレルの CVM
- DB2 フェールオーバー

DB2 用 MPP 構成の CVM パラレルサービスグループ

クラスタノードごとに 1 つの CVM/Infrastructure グループが存在します。このサービスグループには、CVM リソースと、CFS のサポートに必要なリソースがあります。また、このグループには、すべてのクラスタノードで共有されるインスタンスのホームディレクトリなど、DB2 に必要なすべての共通コンポーネントも含まれます。

図 4-2 は MPP 構成をサポートする 2 個のサービスグループの 1 番目である CVM パラレルサービスグループを説明します。

図 4-2 CVM パラレルサービスグループ

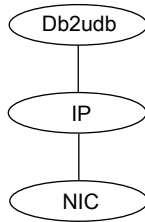


DB2 用 MPP 構成のフェールオーバーサービスグループ

このサービスグループは MPP 構成で 1 つのデータベースのパーティションを監視します。DB2 サービスグループ (フェールオーバー) と CVM サービスグループ (パラレル) は、online local firm の依存関係を持ちます。

図 4-3 は MPP 構成をサポートする 2 個のサービスグループの 2 番目である DB2 フェールオーバーサービスグループを説明します。

図 4-3 DB2 フェールオーバーサービスグループ



DB2 サービスグループを設定する前に

DB2 サービスグループを設定する前に、次の前提条件を満たす必要があります。

- サービスグループを設定する予定のクラスタのすべてのノードに、VCS がインストールされ、設定されていることを確認します。VCS のインストールについては、『Veritas Cluster Server インストールガイド』を参照してください。
- クラスタのすべてのノードに、DB2 が同じよう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.22 の「VCS 環境での DB2 のインストール」を参照してください。
- クラスタのすべてのノードに、Veritas agent for DB2 がインストールされていることを確認します。
p.22 の「VCS 環境での DB2 のインストール」を参照してください。

Db2udbTypes.cf ファイルのインポート

DB2 UDB テンプレートを使う前に、Cluster Manager (Java コンソール) を使って Db2udbTypes.cf ファイルを VCS エンジンにインポートします。

メモ: Java GUI については、以下の手順を開始する前に『Veritas Cluster Server 6.0.1 インストールガイド』を参照してください。

Cluster Manager (Java コンソール) を使って Db2udbTypes.cf ファイルをインポートするには

- 1 クラスタのノードの 1 つで、Cluster Manager (Java コンソール) を開始します。次のように入力します。

```
# haguic
```

- 2 クラスタにログインし、Cluster Explorer の起動を待機します。

- 3 [ファイル (File)]メニューの[タイプのインポート (Import Types)]をクリックします。メッセージが表示されたら、読み取り/書き込みモードに切り替えます。
- 4 [タイプのインポート (Import Types)]ダイアログボックスで、次のファイルを選択します。

```
/etc/VRTSagents/ha/conf/Db2udb/Db2udbTypes.cf
```

- 5 [インポート (Import)]をクリックして、ファイルのインポートを待機します。
- 6 設定を保存します。

Db2udb のタイプが VCS エンジンにインポートされたら、Db2udb のエージェントを設定できます。

次の項を参照してください。

- DB2 UDB MPP の設定を使う場合:
p.40 の「[Cluster Manager \(Java コンソール\)による DB2 MPP のサービスグループの追加](#)」を参照してください。
- DB2 UDB 非 MPP の設定を使う場合:
p.42 の「[Cluster Manager \(Java コンソール\)による DB2 非 MPP のサービスグループの追加](#)」を参照してください。

コマンドラインを使って Db2udbTypes.cf ファイルをインポートするには

- 1 スーパーユーザーとしてクラスタシステムにログインします。
- 2 クラスタ設定を読み取り専用にします。このアクションにより、既存の設定に加えられた変更をすべて保存し、main.cf ファイルの修正中に、さらに変更が加えられないようにします。

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

- 3 main.cf の編集集中に VCS が動作しないことを保証するために、次のコマンドを発行してすべてのシステムで VCS エンジンを停止します。リソースをオンラインで使用可能な状態にします。

```
# hastop -all -force
```

- 4 main.cf ファイルのバックアップを作成します。

```
# cd /etc/VRTSvcs/conf/config  
# cp main.cf main.cf.orig
```

- 5 main.cf ファイルを編集し、Db2udbTypes.cf ファイルをインクルードします。

```
# include "Db2udbTypes.cf"
```

DB2タイプ定義がVCSエンジンにインポートされます。VCSを中断または停止せずに、DB2のエージェントを設定できます。

Cluster Manager (Java コンソール)からのVCS Agent for DB2の設定

DB2 リソースグループのテンプレートは、DB2用のエージェントをインストールしたときに自動的にインストールされます。Cluster Manager (Java コンソール)でテンプレート (/etc/VRTSagents/ha/Templates/Db2udbGroup.tf)を使って、DB2 サービスグループ、サービスグループのリソース、その属性を設定できます。また、Cluster Manager (Java コンソール)を使って、設定の属性値を動的に変更することもできます。

Cluster Manager (Java コンソール)について詳しくは、『Veritas Cluster Server 管理者ガイド』を参照してください。

Cluster Manager (Java コンソール)によるDB2 MPPのサービスグループの追加

Db2udbTypes.cf ファイルがインポートされている場合、テンプレート (/etc/VRTSagents/ha/Templates/Db2udbGroup.tf)を使ってサービスグループを設定できます。

p.38の「[Db2udbTypes.cf ファイルのインポート](#)」を参照してください。

Cluster Manager にログインすると、[状態 (Status)] タブには、クラスタの各システムでCVM サービスグループがオンラインであることが示されます。CVM サービスグループは、Veritas Storage Foundation Cluster File System (SFCFS) ソフトウェアのインストールが完了すると自動的に設定されます。

MPP 構成 DB2 データベースのサービスグループの追加を開始するには

- 1 [Cluster Explorer] ウィンドウで、ツールバーの[サービスグループの追加 (Add Service Group)] アイコンをクリックします。
- 2 [サービスグループの追加 (Add Service Group)] ウィンドウで、追加作成するサービスグループの名前を入力します。たとえば、db2mpp_grp1 と入力します。Return キーまたは Enter キーは押さないでください。

- 3 [使用可能なシステム (Available Systems)] ボックスで、設定に含めるシステムをダブルクリックします。
- 4 自動的にサービスグループを開始するシステムを選択します。設定内のシステムを示すウィンドウで、システムの隣にあるチェックボックスにチェックマークを付けます。
- 5 [フェールオーバー (Failover)] ラジオボタンを選択し、サービスグループタイプを指定します。

DB2 MPP のデータベースのテンプレートを選択するには

- 1 [テンプレート (Templates)] ボタンをクリックします。
- 2 [テンプレートの選択 (Select Templates)] ウィンドウで、[テンプレート (Templates)] ボックスのリストから `db2udb_mpp_grp` を選択します。依存関係のグラフ情報とタイプ情報が、選択したテンプレートに応じて変更されます。[OK] をクリックします。[サービスグループの追加 (Add Service Group)] ウィンドウに、選択に応じたテンプレートの名前が表示されます。
- 3 [サービスグループの追加 (Add Service Group)] ウィンドウの下部にある [OK] をクリックします。グループが追加されます。[Cluster Manager] (Java コンソール) ウィンドウの左のペインに、追加したサービスグループが CVM サービスグループの下に表示されます。[状態 (Status)] タブには、各システムのグループが **Offline** で表示されます。

DB2 MPP データベースのリソースを設定するには

- 1 左のペインで、`db2mpp_grp1` サービスグループをダブルクリックします。グループに設定できるリソースのタイプ (Db2udb、IP、NIC) が表示されます。
- 2 Db2udb リソースタイプをダブルクリックします。リソース `db2udb` (Db2udb タイプの下にある) を選択します。[プロパティ (Properties)] タブをクリックします。
- 3 `db2udb` リソースの [プロパティ (Properties)] タブに [タイプに固有の属性 (Type Specific Attributes)] のリストが表示されます。設定する各属性の [編集 (Edit)] アイコンをクリックします。[属性の編集 (Edit Attribute)] ウィンドウで、必要な属性値情報を入力します。たとえば、`DB2InstOwner` の値として `db2inst1` を入力します。
- 4 `db2udb` リソースに値を設定したときと同じ方法で IP および NIC リソースの値を設定します。この場合、タイプをダブルクリックしてリソースを表示し、選択します。[プロパティ (Properties)] タブが表示された状態で、各リソースの [タイプに固有の属性 (Type Specific Attributes)] を編集できます。
- 5 左のペインで `db2mpp_grp1` サービスグループを右クリックします。ドロップダウンメニューの [リンク (Link)] をクリックします。

[サービスグループのリンク設定 (Link Service Groups)] ウィンドウには次の事項が表示されます。

- 親グループ `db2mpp_grp1`
- 子グループ CVM

- 関係「online local」
 - 依存関係タイプ「firm」
- 6 [OK]をクリックし、依存関係のリンクを作成します。
- MPP 構成 DB2 データベースのサービスグループを最終化するには**
- 1 [設定の保存 (Save Configuration)] アイコンをクリックします。
 - 2 **db2udb** および IP リソースを有効にします。リソースを右クリックし、ドロップダウンメニューの[有効 (Enabled)]をクリックします。必要に応じて、設定を読み書き両用モードにします。
 - 3 [サービスグループをオンラインにします (Online Service Group)] アイコンをクリックします。
 - 4 ウィンドウで、オンライン状態にするサービスグループとシステムを選択します。[OK]をクリックします。

Cluster Manager (Java コンソール)によるDB2 非 MPP のサービスグループの追加

Db2udbTypes.cf ファイルをインポートしている場合、Db2udb_Group テンプレートを使ってサービスグループを設定できます。

p.38 の「[Db2udbTypes.cf ファイルのインポート](#)」を参照してください。

Db2udbTypes.cf ファイルをインポートしている場合にサービスグループを設定するには

- 1 [Cluster Explorer] ウィンドウで設定ウィザードを使うかを確認するプロンプトが表示されたら、[いいえ (No)] を選択します。ウィザードの使用を選択した場合も、後続の手順は同じであることに注意してください。
- 2 [Cluster Explorer] ウィンドウで、ツールバーの[サービスグループの追加 (Add Service Group)] アイコンをクリックします。
- 3 [サービスグループの追加 (Add Service Group)] ウィンドウで、追加作成するサービスグループの名前を入力します。たとえば、**db2_group1** と入力します。**Return** キーまたは **Enter** キーは押さないでください。
- 4 [使用可能なシステム (Available Systems)] ボックスのシステムのうち、設定に含めるシステムをダブルクリックします。
- 5 自動的にサービスグループを開始するシステムを選択します。設定内のシステムを示すウィンドウで、システムの隣にあるチェックボックスにチェックマークを付けます。
- 6 [フェールオーバー (Failover)] ラジオボタンを選択し、サービスグループタイプを指定します。

Db2udbTypes.cf ファイルをインポートしている場合にテンプレートを設定するには

- 1 [テンプレート(Templates)] ボタンをクリックします。
- 2 [テンプレートの選択 (Select Templates)] ウィンドウで、[テンプレート(Templates)] ボックスのリストから **db2udb_grp** を選択します。依存関係のグラフ情報とタイプ情報が、選択したテンプレートに応じて変更されます。[OK] をクリックします。[サービスグループの追加 (Add Service Group)] ウィンドウに、選択に応じたテンプレートの名前が表示されます。

Db2udbTypes.cf ファイルをインポートしている場合にサービスグループを最終化するには

- 1 [サービスグループの追加 (Add Service Group)] ウィンドウの下部にある [OK] をクリックします。グループが追加されます。[Cluster Manager] (Java コンソール) ウィンドウの左のペインに、追加したサービスグループがクラスタ名の下に表示されます。[状態 (Status)] タブには、各システムのグループが **Offline** で表示されます。
- 2 左のペインで、**db2_group1** サービスグループをダブルクリックします。グループに対して設定できるリソースのタイプ (**Db2udb**、**DiskGroup**、**IP**、**Mount**、**NIC**、**Volume**) が表示されます。
- 3 **Db2udb** リソースタイプをダブルクリックします。リソース **db2udb** (**Db2udb** タイプの下にある) を選択します。[プロパティ (Properties)] タブをクリックします。
- 4 **db2udb** リソースの [プロパティ (Properties)] タブに [タイプに固有の属性 (Type Specific Attributes)] のリストが表示されます。設定する各属性の [編集 (Edit)] アイコンをクリックします。[属性の編集 (Edit Attribute)] ウィンドウで、必要な属性値情報を入力します。たとえば、**DB2InstOwner** の値として **db2inst1** を入力します。
DB2InstOwner と **DB2InstHome** は必須の属性です。これらの属性を編集する必要があります。
- 5 **db2udb** リソースに値を設定したときと同じ方法で **DiskGroup**、**IP**、**Mount**、**NIC** および **Volume** リソースの値を設定します。この場合、タイプをダブルクリックしてリソースを表示し、選択します。[プロパティ (Properties)] タブが表示された状態で、[タイプに固有の属性 (Type Specific Attributes)] を編集できます。
必須属性の一覧とそれらの属性の説明については『Veritas Cluster Server 付属エージェントリファレンスガイド』を参照してください。
- 6 **db2_group1** のリソースを有効にします。各リソースを右クリックし、ドロップダウンメニューの [有効 (Enabled)] をクリックします。必要に応じて、設定を読み書き両用モードにします。
- 7 [設定を保存して閉じます (Save and Close Configuration)] アイコンをクリックします。

- 8 [サービスグループをオンラインにします (Online Service Group)] アイコンをクリックします。
- 9 ウィンドウで、オンライン状態にするサービスグループとシステムを選択します。オンライン状態にするシステムをクリックします。[OK]をクリックします。確認メッセージで[はい(Yes)]をクリックします。

メモ: VCS Single Cluster Manager (Java コンソール) は VCS のパッケージに含まれなくなりました。マルチクラスタ環境での管理、監視、レポートには、Veritas Operations Manager (VOM) を使うことを推奨します。VOM は、<http://go.symantec.com/vom> からダウンロードできます。引き続き VCS Single Cluster Manager を使う場合は、http://go.symantec.com/vcsm_download の Web サイトから無料で取得できます。

main.cf ファイルの編集による VCS Agent for DB2 の設定

DB2 用の VCS エージェントでは、3 つの VCS 設定ファイルの例が /etc/VRTSagents/ha/conf/Db2udb/sample_db2udb ディレクトリにあります。

サンプルは次のとおりです。

- 単一パーティションインスタンスの設定
- マルチパーティションインスタンス SMP の設定
- マルチパーティションインスタンス MPP の設定

適切なファイルを参考にして、現在の main.cf 設定ファイルを直接変更できます。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。

これらの手順を完了した後に、次に記載したエージェントの設定方法について、適切な項目に進んでください。

main.cf ファイルの編集準備を行うには

- 1 root ユーザーとしてシステム A にログインします。
- 2 main.cf ファイルを編集する前に、現行の各システムの状態、アプリケーションやシステムの設定を main.cf ファイルに保存し、クラスタを読み取り専用モードに切り替えます。

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

- 3 main.cf を編集中に VCS を必ず停止します。hastop コマンドを使って、すべてのシステムで VCS エンジンを停止し、リソースを利用可能な状態にします。

```
# 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
```

MPP 構成 DB2 を使うためのエージェントの設定

main.cf ファイルを編集しま

す。/etc/VRTSagents/ha/conf/Db2udb/sample_db2udb/main.cf.MPP を参考に使
います。CVM サービスグループは設定ファイルにあります。

MPP 構成 DB2 を使うためにエージェントを設定するには

- 1 Db2udbTypes.cf ファイルを含めます。

```
include "Db2udbTypes.cf"
```

- 2 Db2udb リソースのサービスグループを作成します。設定ファイルの例を参照してく
ださい。この例は、次のような 4 つの DB2 MPP サービスグループと CVM サービス
グループを示します。

- Linux 用の MPP main.cf の設定

p.70 の「[Linux での VCS agent for DB2 の MPP main.cf の設定](#)」を参照して
ください。

- 3 DB2 MPP サービスグループでは、Db2udb、IP、NIC リソースに対する定義を追加
します。ユーザーの設定のパラメータに一致するようにリソースの属性に値を割り当
てます。

詳しくは、次の情報を確認してください。

- Db2udb のリソースタイプの属性

p.61 の「[Db2udb リソースタイプと属性の定義について](#)」を参照してください。

- 設定ファイルの例

- IP と NIC リソースについて詳しくは、『Veritas Cluster Server 付属エージェン
トリファレンスガイド』を参照してください。

- 4 DB2 udb サービスグループと cvm サービスグループを、online local firm の依存
関係に設定します。次に例を示します。

```
requires group cvm online local firm
```

- 5 サービスグループの依存関係を設定した後、新しく作成したリソースの依存関係を設定します。該当する設定ファイルの例を参照してください。db2mpp_grp0 グループでは、次のように入力します。

```
db2udb0 requires Db2_IP0
Db2_IP0 requires Db2_NIC0
```

依存関係の設定について詳しくは、『Veritas Cluster Server 管理者ガイド』を参照してください。

- 6 ファイルを保存して閉じます。

非 MPP 構成 DB2 を使うためのエージェントの設定

main.cf ファイルを編集しま

す。/etc/VRTSagents/ha/conf/Db2udb/sample_db2udb/main.cf.EE または /etc/VRTSagents/ha/conf/Db2udb/sample_db2/main.cf.EEE を参考に使います。

非 MPP 構成 DB2 を使うためにエージェントを設定するには

- 1 Db2udbTypes.cf ファイルを含めます。

```
include "Db2udbTypes.cf"
```

- 2 DB2 リソースのサービスグループを作成します。

この例では、2 つのパーティションが定義されている「db2_grp1」グループを示します。

- 3 サービスグループのすべてのリソースが含まれます。これらのリソースには、Db2udb、DiskGroup、IP、Mount、NIC、Volume の各リソースが含まれます。ユーザーの設定のパラメータに一致するようにリソースの属性に値を割り当てます。

詳しくは、

- Db2udb のリソースタイプの属性
p.61 の「Db2udb リソースタイプと属性の定義について」を参照してください。
- 設定ファイルの例

DiskGroup、IP、Mount、NIC、Volume の各リソースについて詳しくは、『Veritas Cluster Server 付属エージェントリファレンスガイド』を参照してください。

- 4 新しく作成されたリソースの依存関係を設定します。グループ `db2udb_grp1` については、次のように入力します。

```
db2udb1 requires db2udb_ip1
db2udb1 requires db2udb_mnt1
db2udb_ip1 requires db2udb_nic1
db2udb_mnt1 requires db2udb_vol1
db2udb_vol1 requires db2udb_dg1
```

また、グループ `db2udb_grp3` については、次のように入力します。

```
db2udb3 requires db2udb_ip3
db2udb3 requires db2udb_mnt3
db2udb_ip3 requires db2udb_nic3
db2udb_mnt3 requires db2udb_vol3
db2udb_vol3 requires db2udb_dg3
```

詳しくは、該当する設定ファイルの例を参照してください。

依存関係の設定について詳しくは、『Veritas Cluster Server 管理者ガイド』を参照してください。

- 5 ファイルを保存して閉じます。

VCS agent for DB2 の設定の検証

`main.cf` を編集した後、設定を調べます。

設定を確認するには

- 1 `/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 .
```

- 2 システム A で VCS エンジンを起動します。

```
# hstart
```

- 3 `hastatus` コマンドを入力します。

```
# hastatus
```

- 4 「LOCAL_BUILD」がメッセージの列に表示されたら、システム B で VCS を開始します。

```
# hstart
```

- 5 すべての DB2 サービスグループリソースが、システム A 上でオンライン状態であることを確認します。

```
# hagr -display
```

- 6 システム A でサービスグループをオフライン状態にして、すべてのリソースが停止していることを確認します。

```
# hagr -offline db2udb_grp1 -sys sysa
# hagr -offline db2udb_grp3 -sys sysa
# hagr -display
```

- 7 システム A でサービスグループを再度オンラインにして、すべてのリソースが利用可能であることを確認します。

```
# hagr -online db2udb_grp1 -sys sysa
# hagr -online db2udb_grp3 -sys sysa
# hagr -display
```

- 8 DB2 サービスグループをシステム B に切り替えます。

```
# hagr -switch db2udb_grp1 -to sysb
# hagr -switch db2udb_grp3 -to sysb
```

- 9 すべての DB2 サービスグループリソースが、システム B 上でオンライン状態であることを確認します。

```
# hagr -display
```

- 10 すべてのシステムについて、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/Db2udb_A.log
```

DB2 インスタンスの詳細監視の設定

Veritas™ Cluster Server Agent for DB2 の設定を動的に変更するには、Cluster Manager (Java コンソール) を使うか、VCS コマンドラインを使います。以降の説明では、詳細監視に関する設定変更について、コマンドラインから VCS コマンドを使う方法を示します。コマンドラインからの VCS の再設定について詳しくは、『Veritas Cluster Server 管理者ガイド』を参照してください。

DB2 インスタンスの詳細監視の有効化

DB2 のインスタンスの通常監視は `db2gcf` のコマンドの `exit` ステータスを調べます。

それに対して、詳細監視ではインスタンスやパーティション、およびそのデータベースの可用性においてより高いレベルの信頼性が得られます。データベースに対する追加のクエリーを実行してデータベースが利用可能かどうかを検証します。

DB2 のコマンドラインからの詳細監視の有効化

詳細監視は、動的に設定できます。詳細監視を開始する前に、エージェントのデフォルトの監視(通常監視)を使って DB2 を正常に実行することを推奨します。MPP 構成の場合、データベースパーティションごとにデータベースにローカルにアクセスできることを確認してください。

カスタム監視スクリプトが必要です。詳細監視の属性の情報に関しては次のテーブルを参照します。

p.61 の「[Db2udb リソースタイプと属性の定義について](#)」を参照してください。

特定のインスタンスの詳細監視を開始するには

- 1 VCS 設定を書き込み可能にします。

```
# haconf -makerw
```

- 2 設定が不完全なために、VCS が自動的にフェールオーバーを行わないよう、サービスグループをフリーズさせます。

```
# hagrps -freeze db2udb_group
```

- 3 次のコマンドを使って詳細監視を有効にします。

```
# hares -modify resource DatabaseName name
```

```
# hares -modify resource IndepthMonitor 1
```

次に例を示します。

```
# hares -modify db2udb DatabaseName SAMPLE
```

```
# hares -modify db2udb IndepthMonitor 1
```

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

```
# hagrps -unfreeze db2udb_group
```

Db2udb リソースに対するインテリジェントなリソースの監視の有効化と無効化

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の監視機能は、デフォルトで有効に設定されるようになりました。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の監視を実行する必要があるかどうかが決まります。

Db2udb リソースに対してインテリジェントなリソースの監視を有効にするには

- 1 次のコマンドを実行して VCS 設定を書き込み可能にします。

```
# haconf -makerw
```

- 2 種類が Db2udb のリソースすべてに対してインテリジェントなリソースの監視を有効にするには、次のコマンドを実行します。

```
# hatype -modify Db2udb IMF -update Mode 2
```

- 3 IMF 属性の MonitorFreq キーと RegisterRetryLimit キーの値を変更します。

p.61 の「Db2udb リソースタイプと属性の定義について」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。

```
# /opt/VRTS/bin/amfconfig
```

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コマンドから返された場合は、AMFドライバを設定します。

- 6 Db2udb 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 Db2udb -force -sys sys_name
```

```
# haagent -start Db2udb -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 (Db2udb リソース)

- 1 VCS 設定を書き込み可能にします。

```
# haconf -makerw
```

- 2 種類が Db2udb のリソースすべてに対して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 Db2udb IMF -update Mode 0
```

- 3 特定の Db2udb 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: また、IMF の有効化または無効化には、haimfconfig ユーティリティも使用できます。haimfconfig について詳しくは、『Veritas Cluster Server 管理者ガイド』を参照してください。

詳細監視中の DB2 エラーコードの処理

DB2 用のエージェントでは、詳細監視中に発生する DB2 エラーの処理機能が強化されています。エージェントは、DB2 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。

カスタム エラー処理ファイル **db2error.dat** を作成できます。このファイルには、DB2 のエラーと、そのエラーに関連付けられたアクション (エージェントがそのエラーを検出したときに実行する必要があるアクション) が記載されます。

このファイルには、次の形式で情報が格納されています。

```
SQL_error_string:action_to_be_taken
```

次に例を示します。

```
SQL1034N: IGNORE
SQL1039N: WARN
SQL1234N: FAILOVER
```

表 4-1 では詳細監視に利用できるアクションを示します。

表 4-1 詳細監視に利用できるアクション

アクション	説明
IGNORE	エラーを無視します。
UNKNOWN	リソースの状態を UNKNOWN と判断し、Notifier リソースが設定されている場合は通知を送信します。VCS の通知について詳しくは、『Veritas Cluster Server 管理者ガイド』を参照してください。 このアクションは、通常、設定エラーに関連付けられます。

アクション	説明
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、重大度が低いエラーに関連付けられます。
FAILOVER (デフォルト)	リソースの状態を OFFLINE と判断します。これはサービスグループのエラーで、次の使用可能なシステムにフェールオーバーされます。 このアクションはエージェントのデフォルト動作です。検出された DB2 エラーコードが db2error.dat ファイル内に存在しない場合、エージェントは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、別のノードへのフェールオーバーは破損したデータベースで役に立ちません。

DB2 用詳細監視の無効化

詳細監視は、動的に無効にできます。

詳細監視を動的に無効化するには

- 1 VCS の設定を読み書き両用にします。

```
# haconf -makerw
```

- 2 設定が不完全なために、VCS が自動的にフェールオーバーを行わないよう、サービスグループをフリーズさせます。

```
# hagrps -freeze db2udb_group
```

- 3 詳細監視を無効化するには **InDepthMonitor** 属性に **NULL** 値を割り当ててください。次のコマンドを使います。

```
# hares -modifyresource InDepthMonitor 0
```

次に例を示します。

```
# hares -modify db2udb InDepthMonitor 0  
# haconf -dump -makero  
# hagrps -unfreeze db2udb_group
```

DB2 のための 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。

- [DB2 のサービスグループの管理について](#)
- [DB2 サービスグループのオンライン化](#)
- [DB2 サービスグループのオフライン化](#)
- [DB2 サービスグループの切り替え](#)
- [DB2 用エージェントの無効化](#)

DB2 のサービスグループの管理について

サービスグループは、Cluster Manager (Java コンソール)、Veritas Operations Manager、またはコマンドラインを使って管理できます。次の手順では、Cluster Manager (Java コンソール)を使っています。

DB2 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状態にするには、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最初の数サイクルの間、エージェントによるメモリの使用状況は急上昇する可能性があることに注意してください。

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、[サービスグループ (Service Groups)] タブをクリックします。
- 2 サービスグループを右クリックし、[リソースの有効化 (Enable Resources)] をクリックして、このグループ内のすべてのリソースを有効にします。

- 3 サービスグループを右クリックして、[有効化 (Enable)]の上にカーソルを移動し、サービスグループを有効にする特定のノードまたはすべてのノードを選択します。
- 4 設定を保存して閉じます。[ファイル (File)]>[設定の保存 (Save Configuration)]をクリックし、さらに[設定を閉じる (Close Configuration)]をクリックします。
- 5 サービスグループを右クリックし、[オンライン (Online)]にカーソルを置いて、サービスグループをオンラインにするシステムを選択します。

DB2 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状態にするには、Cluster Manager (Java コンソール) から次の手順を実行します。サービスグループをオフライン状態にする最初の数サイクルの間、エージェントによるメモリの使用状況は急上昇する可能性があることに注意してください。

サービスグループをオフライン化するには

- 1 Cluster Explorer 設定ツリーで[サービスグループ (Service Groups)]タブをクリックし、オフライン状態にするサービスグループ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。

DB2 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、Cluster Manager (Java コンソール) から次の手順を実行します。

サービスグループを切り替えるには

- 1 Cluster Explorer の設定ツリーで[サービスグループ (Service Groups)]タブをクリックし、サービスグループ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。

DB2 用エージェントの無効化

システム上でエージェントを無効にするには、まず DB2 サービスグループをオフライン状態に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、サービスグループを他のシステムに切り替えることが可能です。

エージェントを無効にするには

- 1 サービスグループがオンラインかどうかを確認します。プロンプトで、次のように入力します。

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

- 2 サービスグループがオンラインの場合、サービスグループを別のシステムに切り替えるか、またはオフラインにします。プロンプトで、次のように入力します。

```
# hagr -switch service_group -to system_name
```

または

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

- 3 システム上でエージェントを停止します。プロンプトで、次のように入力します。

```
# haagent -stop Db2udb -sys system_name
```

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

`ps` コマンドを使うと、エージェントが停止しているかどうかも確認できます。

- 5 エージェントが停止している場合は、システム、サービスグループ、またはリソースタイプを VCS 設定から削除できます。

詳しくは、『Veritas Cluster Server 管理者ガイド』でコマンドラインからの VCS の再設定に関する章を参照してください。

Veritas™ Cluster Server Agent for DB2 のトラブル シューティング

この付録では以下の項目について説明しています。

- DB2 用環境変数の **db2profile** の作成
- VCS agent for DB2 用 **RestartLimit** 属性の設定
- MPP モードの **DB2 V9.5** と **DB2 V9.7** のパーティションまたはノードの切り替えシナリオ

DB2 用環境変数の **db2profile** の作成

DB2 の各インスタンスに対するプロファイルのファイルを作成し、プロファイルのファイルに環境変数を設定できます。各データベースのユーザーに一意的な環境変数を作成するためにこのプロファイルを使うことができます。各 DB2 インスタンスには、インスタンスのログオン ID と関連付けられる次のようなホームディレクトリがあります。

```
$InstHome/sql1lib/db2profile
```

ここで **\$InstHome** はログオン ID の名前です。

プロファイルのファイルに使用しようとする変数 (たとえば、**TimeZone** 変数) を設定します。su - db2instX コマンド (X はインスタンスの名前) を発行するときに、環境変数が参照されます。

VCS agent for DB2 用 RestartLimit 属性の設定

VCSは複数のパーティションを同時に開始します。これは競合状態の原因になる場合があります。エージェントの **RestartLimit** 属性の値を 3 に設定して、この状態を避けるようにします。各パーティションに対してリソースの依存関係を構築することによってこの状態の可能性を軽減できます。たとえば、サービスグループの中では **Db2udb** リソース 3 (**nodenum=2**) に **Db2udb** リソース 4 (**nodenum=1**) が依存するように設定できます。依存関係ツリーに構築されたパーティションがある場合は、**RestartLimit** の値を 0 に設定できます。

MPP モードの DB2 V9.5 と DB2 V9.7 のパーティションまたはノードの切り替えシナリオ

パーティションがシステムでオンラインの場合に切り替えるには

- 1 ターゲット上のポート 0 に関連付けられないソースマシンから、任意のパーティションを連続して移動します。
- 2 最後のソースのポート 0 に関連付けられているパーティションを使って、残りのパーティションを 1 つずつ移動します。

詳しくは、IBM のケース番号 IC66748 を参照してください。

Veritas™ Cluster Server Agent for DB2 のリソースタイプ情報

この付録では以下の項目について説明しています。

- [DB2 のリソースタイプ定義について](#)

DB2 のリソースタイプ定義について

リソースタイプは、エージェントの VCS 設定の定義を表します。さらに、設定ファイル `main.cf` でのエージェントの定義方法も指定しています。エージェントを設定する場合、リソースタイプ属性に値を割り当てる必要があります。

p.67 の「[Linux 上の VCS agent for DB2 用設定ファイルの例](#)」を参照してください。

Linux での DB2 UDB タイプの定義ファイル: Db2udbTypes.cf

以下に示すのはエージェントのタイプ定義です。

```
Type Db2udb (
    static keylist SupportedActions = { VRTS_GetInstanceName,
    VRTS_GetRunningServices }
    static int CleanTimeout = 240
    static int OfflineTimeout = 240
    static int MonitorTimeout = 240
    static int OnlineRetryLimit = 2
    static int OnlineTimeout = 180
    static int OnlineWaitLimit = 1
    static int RestartLimit = 3
```

```
static int ToleranceLimit = 1
static str AgentFile = "/opt/VRTSagents/ha/bin/Db2udb/Db2udbAgent"
static str AgentDirectory = "/opt/VRTSagents/ha/bin/Db2udb"
static str ArgList[] = { DB2InstOwner, DB2InstHome, IndepthMonitor,
DatabaseName, NodeNumber, StartUpOpt, ShutDownOpt, AgentDebug,
Encoding, WarnOnlyIfDBQueryFailed, LastWarningDay, UseDB2start }
static str IMFRegList[] = { DB2InstOwner, DB2InstHome }
static int IMF{} = { Mode=2, MonitorFreq=5, RegisterRetryLimit=3 }
str DB2InstOwner
str DB2InstHome
int IndepthMonitor
str DatabaseName
int NodeNumber
str StartUpOpt = START
str ShutDownOpt = STOP
boolean AgentDebug = 0
str Encoding
boolean WarnOnlyIfDBQueryFailed = 1
temp str LastWarningDay
boolean UseDB2start = 0
)
```

DB2 のリソースタイプ属性

この付録では以下の項目について説明しています。

- [Db2udb リソースタイプと属性の定義について](#)

Db2udb リソースタイプと属性の定義について

DB2 用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 C-1](#) に、DB2 用エージェントの必須属性を示します。必須属性には必ず値を割り当てる必要があります。

表 C-1 DB2 用エージェントの必須属性

必須属性	説明
DB2InstHome	DB2 インスタンスの設定ファイルと重要データが格納されている DB2 UDB インスタンスホームディレクトリへのパス。 データ形式と値の種類: 文字列 - スカラー
DB2InstOwner	DB2 UDB インスタンスを起動するインスタンス所有者のユーザー ID。各インスタンスには、一意のユーザー ID が必要です。 データ形式と値の種類: 文字列 - スカラー 注意: この属性に不正な変更を加えると、DB2 が一貫性のない状態になる可能性があります。

[表 C-2](#) は DB2 用エージェントのオプション属性を示します。

表 C-2 DB2 用エージェントのオプション属性

オプション属性	説明
DatabaseName	<p>詳細監視用のデータベースの名前。詳細監視が有効な場合 (IndepthMonitor = 1) に必要です。</p> <p>データベースのすべてのパーティションで障害が発生するのでデータベース名の属性を変更するときに注意します。無効か不正な値に DataBaseName の属性を変更しないでください。</p> <p>メモ: DatabaseName で名前が指定されたデータベースが、Db2udb エージェントリソースが設定されたパーティションと同じパーティションに作成されていることを確認してください。</p> <p>データ形式と値の種類: 文字列 - スカラー</p>
NodeNumber	<p>データベースのノード番号またはパーティション番号。特定のデータベースのパーティションを監視した場合使われます。</p> <p>デフォルト: 0</p> <p>データ形式と値の種類: 整数 - スカラー</p>

オプション属性	説明
<p>StartUpOpt</p>	<p>起動オプションを提供します。許可される値は START、ACTIVATEDB、CUSTOM です。</p> <p>次のオプションを確認します。</p> <ul style="list-style-type: none"> ■ START (デフォルト) DB2 インスタンスまたはパーティションを起動します。 ■ ACTIVATEDB db2 プロセスの起動後にデータベースをアクティブにするコマンドを実行します。 ■ CUSTOM エージェントは、StartUpOpt 属性が CUSTOM に設定されている場合、すべてのオンライン機能をユーザーに完全に任せます。<code>/opt/VRTSagents/ha/bin/Db2udb</code> ディレクトリの <code>start_custom_\${db2instance}_\${nodenum}</code> というファイルを見つけます。このファイルが存在し、実行可能である場合、エージェントは代わりに、カスタマイズされたこのオンラインファイルを実行します。 <p>例:</p> <p>パーティションのオンライン機能が db2 インスタンス db2inst1 の nodenum 1 をカスタマイズするには、DB2 エージェントでこのカスタマイズされたファイル <code>start_custom_db2inst1_1</code> を実行します。<code>/opt/VRTSagents/ha/bin/Db2udb</code> のディレクトリの下でこのファイルを実行します。</p> <p>データ形式と値の種類: 文字列 - スカラー</p>

オプション属性	説明
ShutDownOpt	<p>この属性で許可される値は STOP と CUSTOM です。 次のオプションを確認します。</p> <ul style="list-style-type: none"> ■ STOP DB2 インスタンスまたはパーティションを通常の方法でシャットダウンします。 ■ CUSTOM ShutDownOpt 属性が CUSTOM に設定されている場合、すべてのオフライン処理をユーザーに完全に任せます。<code>/opt/VRTSagents/ha/bin/Db2udb</code> ディレクトリの <code>stop_custom_\$db2instance_\$nodenum</code> というファイルを見つけます。 このファイルが存在し、実行可能である場合、エージェントは代わりに、カスタマイズされたこのオフラインファイルを実行します。 例: パーティションのオフライン機能か <code>db2inst1</code> という <code>db2</code> インスタンスの <code>nodenum 0</code> をカスタマイズするとします。 DB2 エージェントでカスタマイズされたファイル: <code>stop_custom_db2inst1_0</code> を実行します。このファイルは、<code>/opt/VRTSagents/ha/bin/Db2udb</code> ディレクトリにあります。 <p>データ形式と値の種類: 文字列 - スカラー</p>
IndepthMonitor	<p>詳細監視を有効にする場合は、<code>IndepthMonitor</code> 属性の値を <code>1</code> に設定します。現在、エージェントは、<code>/opt/VRTSagents/ha/bin/Db2udb</code> ディレクトリで <code>monitor_custom_\$db2instance_\$nodenum</code> ファイルを検索します。 ファイルが存在し、実行可能な場合、エージェントはカスタマイズされたこの詳細監視ファイルを実行します。 <code>sample_db2udb</code> ディレクトリに、カスタム監視スクリプトの例があります。</p> <p>データ形式と値の種類: 文字列 - 整数</p>
Encoding	<p>DB2 UDB の出力表示で使う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>データ形式と値の種類: 文字列 - スカラー</p>
AgentDebug	<p>この属性の値が <code>1</code> のとき、エージェントで追加のデバッグメッセージがログに記録されます。</p> <p>データ形式と値の種類: ブール - スカラー</p>

オプション属性	説明
WarnOnlyIfDBQueryFailed	<p>この属性では、SQL エラーをログに記録するか、特別な処理が必要なエラーを確認するかを指定します。</p> <p>WarnOnlyIfDBQueryFailed 属性の値を 1 に設定すると有効になります。この属性が有効であるとき、すべての SQL エラーは無視され、1 日に 1 回警告メッセージがエージェントログに記録されます。</p> <p>WarnOnlyIfDBQueryFailed 属性の値を 0 に設定すると無効になります。無効にすると、db2error.dat ファイルでエラーコードに対して特別な処理が必要かどうかを確認されます。db2error.dat ファイル内にそのエラーコードが存在しない場合は、監視のために OFFLINE が返されます。そのエラーコードが存在する場合は、db2error.dat ファイルに記載されたそのエラーコードのアクションが実行されます。</p> <p>データ形式と値の種類: ブール - スカラー</p>
UseDB2start	<p>DB マネージャを起動するための代替方法を選択できます。エージェントは、UseDBStart が 1 に設定されている場合は db2start を実行しますが、UseDBStart が 0 に設定されている場合は db2gcf を使います。設定が非 MPP の場合、UseDB2start 属性の値を 1 に設定する必要があります。</p> <p>設定が MPP の場合、UseDB2start 属性の値を 0 に設定する必要があります。MPP の場合は、VCS は実際のホスト名を使い、各パーティションを開始、停止できる必要があります。このため、VCS は db2gcf コマンドを使い、UseDB2start 属性の値を 0 にする必要があります。</p> <p>データ形式と値の種類: ブール - スカラー デフォルト: 0</p>

表 C-3 は DB2 のエージェントの内部属性を示します。

表 C-3 DB2 用エージェントの内部属性

内部属性	説明
AgentDirectory	<p>エージェントに関連するその他のファイルとスクリプトの場所を示します。</p> <p>使用しないでください。内部使用の属性です。</p>

内部属性	説明
IMF	<p>このリソースタイプレベルの属性は、DB2 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。</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"> ■ Mode: 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。有効な値は次の通りです： <ul style="list-style-type: none"> ■ 0 - インテリジェントなリソースの監視を実行しない ■ 1 - 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 ■ 2 - 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 ■ 3 - 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 デフォルト: 2 ■ MonitorFreq: 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し頻度を指定します。このキーの値は整数です。デフォルト: 5 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 0 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。リソースが AMF 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。 <ul style="list-style-type: none"> ■ オンラインリソースの場合は (MonitorFreq x MonitorInterval) 秒おき ■ オフラインリソースの場合は (MonitorFreq x OfflineMonitorInterval) 秒おき ■ RegisterRetryLimit: 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 <code>imf_register</code> エージェント関数を呼び出して、リソースを AMF カーネルドライバに登録します。 RegisterRetryLimit キーの値は、エージェントがリソースの登録を再試行する最小の回数を決定します。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、Mode 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。 デフォルト: 3 <p>p.50 の「Db2udb リソースに対するインテリジェントなリソースの監視の有効化と無効化」を参照してください。</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。

- [Linux 上の VCS agent for DB2 用設定ファイルの例](#)

Linux 上の VCS agent for DB2 用設定ファイルの例

このセクションでは DB2 UDB の設定例を示します。

Linux での Veritas Cluster Server Agent for DB2 用非 MPP main.cf の設定

次の main.cf 設定ファイルは、ESE マルチパーティションインスタンス SMP 環境での DB2 UDB を示します。

```
include "types.cf"
include "Db2udbTypes.cf"

cluster vcs (
    CounterInterval = 5
)

system vcstc1 (
    CPUUsageMonitoring = { Enabled = 0, ActionThreshold = 0,
    ActionTimeLimit = 0, Action = NONE, NotifyThreshold = 0,
    NotifyTimeLimit = 0 }
)

system vcstc2 (
    CPUUsageMonitoring = { Enabled = 0, ActionThreshold = 0,
    ActionTimeLimit = 0, Action = NONE, NotifyThreshold = 0,
    NotifyTimeLimit = 0 }
```

```
)

group db2udb_grp1 (
    SystemList = { vcstc1= 0, vcstc2 = 1 }
    AutoStartList = { vcstc1 }
)

Db2udb db2udb1 (
    DB2InstOwner = db2inst1
    DB2InstHome = "/db2inst1"
    IndepthMonitor = 1
    DatabaseName = SAMPLE
    NodeNumber = 0
)

Db2udb db2udb2 (
    DB2InstOwner = db2inst1
    DB2InstHome = "/db2inst1"
    IndepthMonitor = 0
    NodeNumber = 1
)

DiskGroup db2udb_dg1 (
    DiskGroup = db2_dg1
)

IP db2udb_ip1 (
    Device = eth0
    Address = "166.98.9.163"
    NetMask = "255.255.252.0"
)

Mount db2udb_mnt1 (
    MountPoint = "/db2inst1"
    BlockDevice = "/dev/vx/dsk/db2_dg1/inst1_vol"
    FSType = vxfs
    MountOpt = rw
    FsckOpt = "-n"
)

NIC db2udb_nic1 (
    Device = eth0
)
```

```

Volume db2udb_vol1 (
    Volume = inst1_vol
    DiskGroup = db2_dg1
)

db2udb1 requires db2udb_ip1
db2udb1 requires db2udb_mnt1
db2udb2 requires db2udb_ip1
db2udb2 requires db2udb_mnt1
db2udb_ip1 requires db2udb_nic1
db2udb_mnt1 requires db2udb_vol1
db2udb_vol1 requires db2udb_dg1

// resource dependency tree
//
//     group db2udb_grp1
//     {
//     Db2udb db2udb1
//         {
//             IP db2udb_ip1
//                 {
//                     NIC db2udb_nic1
//                 }
//             Mount db2udb_mnt1
//                 {
//                     Volume db2udb_vol1
//                         {
//                             DiskGroup db2udb_dg1
//                         }
//                 }
//         }
//     Db2udb db2udb2
//         {
//             IP db2udb_ip1
//                 {
//                     NIC db2udb_nic1
//                 }
//             Mount db2udb_mnt1
//                 {
//                     Volume db2udb_vol1
//                         {
//                             DiskGroup db2udb_dg1

```

```
//  
//  
//  
// }
```

Linux での VCS agent for DB2 の MPP main.cf の設定

次の設定ファイルは、ESE マルチパーティションインスタンス MPP 環境での DB2 UDB を示します。4 つのデータベースパーティションが設定されています。各クラスタノードで 1 つのパーティションが設定されています。各データベースサービスグループは、クラスタの共有ストレージを管理する同じ CVM サービスグループに依存しています。

```
include "types.cf"  
include "CFSTypes.cf"  
include "CVMTypes.cf"  
include "Db2udbTypes.cf"  
cluster db2_mpp (  
    CounterInterval = 5  
)  
system vcstc1 (  
)  
system vcstc2 (  
)  
system vcstc3 (  
)  
system vcstc4 (  
)  
group cvm (  
    SystemList = { vcstc1 = 0, vcstc2 = 1, vcstc3 = 2, vcstc4 = 3 }  
    AutoFailOver = 0  
    Parallel = 1  
    AutoStartList = { vcstc1, vcstc2, vcstc3, vcstc4 }  
)  
CFSMount db2cfsmnt (  
    MountPoint = "/db2_mnt/db2inst1"  
    BlockDevice = "/dev/vx/dsk/cdb2dg1/cdb2dg1home"  
)  
CFSfsckd vxfsckd (  
)  
CVMCluster cvm_clus (  
    CVMClustName = db2_mpp  
    CVMNodeId = { vcstc1 = 0, vcstc2 = 1, vcstc3 = 2, vcstc4 = 3  
    }
```

```

        CVMTransport = gab
        CVMTimeout = 200
    )
    CVMVolDg db2dg (
        CVMDiskGroup = cdb2dg1
        CVMVolume = { cdb2dg1home }
        CVMActivation = sw
    )
    CVMVxconfigd cvm_vxconfigd (
        Critical = 0
        CVMVxconfigdArgs = { syslog }
    )
    cvm_clus requires cvm_vxconfigd
    db2cfsmnt requires db2dg
    db2cfsmnt requires vxfsckd
    db2dg requires cvm_clus
    vxfsckd requires cvm_clus

// resource dependency tree
//
//     group cvm
//     {
//         CFSSMount db2cfsmnt
//         {
//             CVMVolDg db2dg
//             {
//                 CVMCluster cvm_clus
//                 {
//                     CVMVxconfigd cvm_vxconfigd
//                 }
//             }
//             CFSfsckd vxfsckd
//             {
//                 CVMCluster cvm_clus
//                 {
//                     CVMVxconfigd cvm_vxconfigd
//                 }
//             }
//         }
//     }
group db2mpp_grp0 (
    SystemList = { vcstc1 = 0, vcstc2 = 1, vcstc3 = 2, vcstc4 = 3 }
    AutoStartList = { vcstc1 }

```

```
AutoStart = 1
)
Db2udb db2udb0 (
    DB2InstOwner = db2inst1
    DB2InstHome = "/db2_mnt/db2inst1"
    IndepthMonitor = 1
    DatabaseName = SAMPLE
    NodeNumber = 0
)
IP Db2_IP0 (
    Device = eth0
    Address = "10.118.2.144"
    NetMask = "255.255.248.0"
)
NIC Db2_NIC0 (
    Device = eth0
)
requires group cvm online local firm
Db2_IP0 requires Db2_NIC0
db2udb0 requires Db2_IP0
// resource dependency tree
//
//     group db2mpp_grp0
//     {
//     Db2udb db2udb0
//         {
//             IP Db2_IP0
//                 {
//                     NIC Db2_NIC0
//                 }
//             }
//         }
//     }
group db2mpp_grp1 (
    SystemList = { vcstc1 = 0, vcstc2 = 1, vcstc3 = 2, vcstc4 = 3 }
    AutoStartList = { vcstc2 }
    AutoStart = 1
)
Db2udb db2udb1 (
    DB2InstOwner = db2inst1
    DB2InstHome = "/db2_mnt/db2inst1"
    IndepthMonitor = 1
    DatabaseName = TEST1
    NodeNumber = 1
```



```
)
IP Db2_IP1 (
    Device = eth0
    Address = "10.118.2.145"
    NetMask = "255.255.248.0"
)
NIC Db2_NIC1 (
    Device = eth0
)
requires group cvm online local firm
Db2_IP1 requires Db2_NIC1
db2udb1 requires Db2_IP1
// resource dependency tree
//
//     group db2mpp_grp1
//     {
//     Db2udb db2udb1
//         {
//             IP Db2_IP1
//                 {
//                     NIC Db2_NIC1
//                 }
//             }
//         }
//     }
group db2mpp_grp2 (
    SystemList = { vcstc1 = 0, vcstc2 = 1, vcstc3 = 2, vcstc4 = 3 }
    AutoStartList = { vcstc3 }
    AutoStart = 1
)
Db2udb db2udb2 (
    DB2InstOwner = db2inst1
    DB2InstHome = "/db2_mnt/db2inst1"
    IndepthMonitor = 1
    DatabaseName = TEST2
    NodeNumber = 2
)
IP Db2_IP2 (
    Device = eth0
    Address = "10.118.2.146"
    NetMask = "255.255.248.0"
)
NIC Db2_NIC2 (
    Device = eth0
```

```
    )
    requires group cvm online local firm
    Db2_IP2 requires Db2_NIC2
    db2udb2 requires Db2_IP2
    // resource dependency tree
    //
    //     group db2mpp_grp2
    //     {
    //     Db2udb db2udb2
    //     {
    //     IP Db2_IP2
    //     {
    //     NIC Db2_NIC2
    //     }
    //     }
    //     }
    //     }
group db2mpp_grp3 (
    SystemList = { vcstc1 = 0, vcstc2 = 1, vcstc3 = 2, vcstc4 = 3 }
    AutoStartList = { vcstc4 }
    AutoStart = 1
)
Db2udb db2udb3 (
    DB2InstOwner = db2inst1
    DB2InstHome = "/db2_mnt/db2inst1"
    NodeNumber = 3
)
IP Db2_IP3 (
    Device = eth0
    Address = "10.118.2.147"
    NetMask = "255.255.248.0"
)
NIC Db2_NIC3 (
    Device = eth0
)
requires group cvm online local firm
Db2_IP3 requires Db2_NIC3
db2udb3 requires Db2_IP3

// resource dependency tree
//
//     group db2mpp_grp3
//     {
//     Db2udb db2udb3
```

```
//      {  
//      IP Db2_IP3  
//      {  
//      NIC Db2_NIC3  
//      }  
//      }  
//      }
```


記号

/etc/services
割り当て済みのポートの確認 23

C

Cluster Manager (Java コンソール)
Db2udbTypes.cf のインポートに使用 40
DB2 サービスグループの設定 40
CVM サービスグループの平行モード 37
CVM (インフラストラクチャ) サービスグループ 37

D

DatabaseName 属性 62
DB2
エラー操作 51
db2gcf -d コマンド 11
db2gcf -u コマンド 11
DB2InstHome 属性 61
DB2InstOwner 属性 61
db2nodes.cfg
パーティションのための仮想ホスト 24
db2setup ツール
バイナリ用
インスタンス 22
DB2 UDB ESE (Enterprise Server Edition)
サンプル設定ファイル 67、70
Db2udbTypes.cf
main.cf ファイルに追加 45~46
タイプ定義ファイル 59
Db2udbTypes.cf ファイル
インポート 38
Db2udbTypes.cf ファイルのインポート 38
DB2 UDB インスタンス
共有ストレージにインストール 22
DB2 UDB タイプの定義ファイル 59
DB2 UDB のインストール
共有ストレージのインスタンス 22
システムバイナリ 22
設定の確認 27
必要条件 22

Db2udb リソース
IP への依存
Mount リソース 36
Db2udb リソースタイプの属性 61
DB2 インスタンスの停止 11
DB2 エージェントの設定
Cluster Manager (Java コンソール) の使用 40
設定ファイルの編集 44
DiskGroup リソース
Volume リソースに必要 36

E

Encoding 属性 64
Enterprise Agent
無効化 32

H

hagrp コマンド 54
hagui コマンド 38

I

IMF の統合機能 14
IP アドレス
仮想 24
IP リソース
NIC リソースが必要 36

J

Java コンソール
サービスグループの設定 40、42

M

main.cf ファイル
DB2 UDB ESE の例 67
DB2 UDB ESE マルチパーティションの例 70
設定の検証 47
編集 44
例 44

Mount リソース

Volume リソースが必要 36

MPP (Massively Parallel Processing)

サポート 11

N**NIC** リソース

IP リソースに必要 36

NodeNumber 属性 62**R****ResourceInfo** リソース属性 15**S****SMP** (対称型マルチプロセッシング)

サポート 11

V**VCS** サポートバージョン 11**Volume** リソース

DiskGroup リソースが必要 36

W**WarnOnlyIfDBQueryFailed** 属性 12**あ**

アンインストール 33

依存関係

DB2 と CVM のサービスグループ間 42

main.cf での定義 47

インスタンス

起動と停止 27

共有ディスクへのインストール 22

インテリジェントなリソース監視のサポート

IMF 10

エージェント

設定

概要 35

エンタープライズエージェント

インストール 30

か

仮想 IP アドレス 24

監視

詳細監視の設定 49

通常と詳細 49

グループ (ユーザー)

追加 21

コマンド

hagrp 54

hagui 38

さ

削除 33

サービスグループ

Cluster Manager (Java コンソール) による追加 40

CVM 37

DB2 と CVM の依存関係リンク 42

main.cf ファイルに作成 45~46

MPP 構成

依存関係 37

ウィザードを使った追加 42

非 MPP 構成

依存関係 36

詳細監視

無効化 52

有効化 49

設定ファイル

db2nodes.cfg

パーティションのためのホスト 24

Db2udbTypes.cf 59

main.cf

例 67、70

た

通常監視

説明 49

テスト

サービスグループの設定状態 48

テンプレート

DB2 MPP サービスグループ 42

非 MPP サービスグループに使用 44

は

バイナリ

DB2 UDB

ローカルにインストール 22

パーティション番号 (NodeNumber) 62

ファイル

db2nodes.cfg 24

Db2udbTypes.cf 59

main.cf (VCS 設定) 44

ファイルシステム

MPP 構成のための作成 19

非 MPP インスタンス用の作成 19