

Veritas™ Cluster Server Agent for DB2 インストール/設定 ガイド

Linux

5.0 Maintenance Pack 3



Veritas Cluster Server Agent for DB2 インストール/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。

Product version:5.0 MP3

Document version:5.0MP3.0

法定通知

Copyright © 2008 Symantec Corporation. All rights reserved.

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

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

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

ライセンス対象ソフトウェアおよび資料は、FAR 12.212 の規定によって商業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 Licensed Software - Restricted Rights」、DFARS 227.7202 「Rights in Commercial Computer Licensed Software or Commercial Computer Licensed 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。

Symantec Corporation
20330 Stevens Creek Blvd.
Cupertino, CA 95014

<http://www.symantec.com>

テクニカルサポート

ご購入先にお問い合わせください。

テクニカルサポート	3	
第 1 章	DB2 用の Veritas エージェントの紹介	9
	Veritas High Availability Agent for DB2 について	9
	エージェントが DB2 で高可用性を実現するしくみ	9
	このリリースの新機能	10
	DB2 用のサポート対象のソフトウェア	10
	DB2 エージェント機能	10
	オンラインエージェント機能	11
	オフラインエージェント機能	11
	監視エージェント機能	11
	クリーンエージェント機能	12
	情報エージェント機能	12
	アクションエージェント機能	12
	情報エージェント機能の実行	13
	VCS クラスタの標準的な DB2 の設定	14
	DB2 UDB のクラスタを設定するためのロードマップ	15
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
	DB2 UDB をインストールするための前提条件 (非 MPP 構成)	17
	DB2 UDB をインストールするための前提条件 (MPP 構成)	18
	DB2 インスタンス用のファイルシステムの作成	18
	DB2 のユーザーアカウントとグループアカウントの定義	21
	VCS 環境での DB2 UDB のインストール	22
	共有メモリパラメータの設定	22
	バイナリのインストール	22
	DB2 ライセンスのインストール	23
	インスタンスのインストール	23
	DB2 UDB 構成の設定	23
	/etc/services の確認	23
	\$DB2InstHome/.rhosts の作成	24
	\$DB2InstHome/sql/lib/db2nodes.cfg ファイルの変更	25
	MPP 構成と非 MPP 構成の DB2 UDB のインストールの 設定確認	26

第 3 章	DB2 UDB 用のエージェントのインストール、アップグレード、削除	29
	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	29
	DB2 用エージェントのインストール	30
	DB2 UDB 用のエージェントのアップグレード	30
	DB2 UDB 用のエージェントの削除	31
第 4 章	DB2 の VCS サービスグループの設定	33
	DB2 UDB のサービスグループの設定について	33
	VCS の DB2 UDB の設定について	34
	非 MPP 構成の DB2 UDB サービスグループ	34
	MPP 構成の DB2 UDB サービスグループ	35
	DB2 のサービスグループを設定する前に	36
	dbtypes Types.cf ファイルのインポート	36
	DB2 UDB エージェントの Cluster Manager (Java コンソール) からの設定	37
	Cluster Manager (Java コンソール) による DB2 UDB MPP のサービスグループの追加	37
	Cluster Manager (Java コンソール) による DB2 UDB 非 MPP のサービスグループの追加	39
	main.cf ファイルの編集による DB2 UDB エージェントの設定	41
	MPP 構成 DB2 UDB を使うためのエージェントの設定	42
	非 MPP 構成 DB2 UDB を使うためのエージェントの設定	43
	設定の確認	44
	DB2 UDB インスタンスの詳細監視の設定	45
	DB2 UDB インスタンスの詳細監視の有効化	46
	コマンドラインからの詳細監視の有効化	46
	詳細監視中の DB2 エラーコードの処理	47
	詳細監視の無効化	48
第 5 章	DB2 の VCS サービスグループの管理	49
	DB2 のサービスグループの管理について	49
	DB2 のサービスグループのオンライン化	49
	DB2 のサービスグループのオフライン化	50
	DB2 のサービスグループの切り替え	50
	DB2 エージェントの無効化	50
付録 A	DB2 エージェントのトラブルシューティング	53
	環境変数の db2profile の作成	53
	RestartLimit 属性の設定	53

付録 B	DB2 UDB のリソースタイプ定義	55
	DB2 UDB のリソースタイプ定義について	55
	DB2 UDB タイプの定義ファイル: Db2udbTypes.cf	55
	Db2udb のリソースタイプの属性	56
付録 C	設定例	61
	Linux 設定ファイルの例	61
	Linux 用の非 MPP main.cf の設定	61
	Linux 用の MPP main.cf の設定	64
索引	71

DB2 用の Veritas エージェントの紹介

この章では以下の項目について説明しています。

- [Veritas High Availability Agent for DB2](#) について
- エージェントが DB2 で高可用性を実現するしくみ
- このリリースの新機能
- DB2 用のサポート対象のソフトウェア
- DB2 エージェント機能
- VCS クラスタの標準的な DB2 の設定
- DB2 UDB のクラスタを設定するためのロードマップ

Veritas High Availability Agent for DB2 について

Veritas High Availability agent for DB2 UDB(バージョン 5.0 MP3)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。

エージェントが DB2 で高可用性を実現するしくみ

DB2 のエージェントはパーティションのレベルで DB2 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

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。エージェントが持つ各種機能の設定について詳しくは、

p.10 の「[DB2 エージェント機能](#)」を参照してください。

このリリースの新機能

このリリースでは DB2 9.5 がサポートされます。

DB2 用のサポート対象のソフトウェア

Veritas High Availability agent for DB2 UDB (バージョン 5.0 MP3) 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

DB2 ESE のエージェントは、次のプラットフォーム、または単一パーティションインスタンスとマルチパーティションインスタンスをサポートする必要があります。

サポート対象のオペレーティングシステムと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。

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。

DB2 エージェント機能

このエージェントはデータベースで各種処理や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。

オンラインエージェント機能

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/sqlllib/bin/db2gcf -u -i  
$DB2InstOwner -p $nodenum
```

オフラインエージェント機能

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/sqlllib/bin/db2gcf -d -i  
$DB2InstOwner -p $nodenum
```

監視エージェント機能

エージェントは `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/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。エラー処理情報は、
p.47 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に注意します。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。

クリーンエージェント機能

エージェントは DB2 データベースのパーティションを強制終了するために 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/sqlllib/bin/db2gcf -k -i  
$DB2InstOwner -p $nodenum"
```

情報エージェント機能

DB2 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 ユーザーズガイド』
- 『Veritas Cluster Server エージェント開発者ガイド』

データベース情報を取り込む情報エージェント機能の例については:

p.13 の「[情報エージェント機能の実行](#)」を参照してください。

アクションエージェント機能

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。リソースでアクションを実行するには、次のコマンドを入力します。

```
# hares -action res token [-actionargs arg1 ...]  
[-sys system] [-rclus 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 (db2udb0): 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
```

情報エージェント機能の実行

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そのプロセス情報を手に入れます。

- 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 リソースを監視した処理を示したものです。

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
```

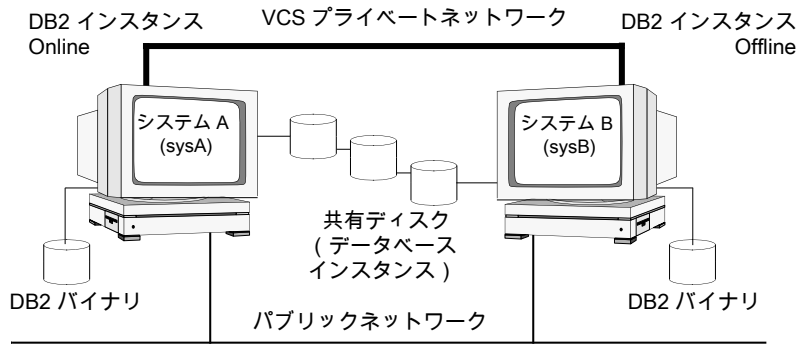
```
TS Fri Jan 14 18:11:52 2005
```

VCS クラスタの標準的な DB2 の設定

次の例は、2 つのシステムクラスタで VCS が設定されています。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。

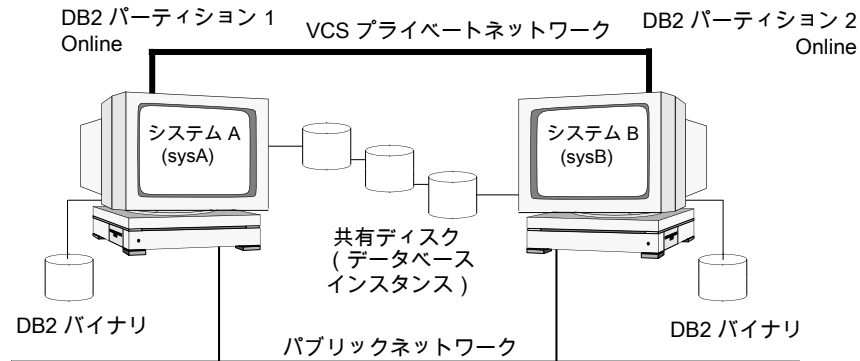
図 1-1 非 MPP 構成での DB2 インストール



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



DB2 UDB のクラスタを設定するためのロードマップ

表 1-1 は、クラスタを設定するロードマップです。

表 1-1 DB2 UDB のクラスタの設定

タスク	参照
次の事項の確認 ■ サポート対象のソフトウェア ■ エージェント機能 ■ エージェントの必要条件	<ul style="list-style-type: none"> ■ p.10 の「DB2 用のサポート対象のソフトウェア」を参照してください。 ■ p.10 の「DB2 エージェント機能」を参照してください。 ■ 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>p.22 の「VCS 環境での DB2 UDB のインストール」を参照してください。</p> <p>p.23 の「DB2 UDB 構成の設定」を参照してください。</p>
DB2 の VCS エージェントのインストール	p.30 の「DB2 用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 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>p.37 の「DB2 UDB エージェントの Cluster Manager (Java コンソール)からの設定」を参照してください。</p> <p>p.45 の「DB2 UDB インスタンスの詳細監視の設定」を参照してください。</p>
サービスグループのオンライン化	p.49 の「DB2 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。

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 UDB のインストール](#)
- [DB2 UDB 構成の設定](#)

DB2 をインストールするための VCS の必要条件

DB2 をインストールするための次の必要条件と前提条件を確認します。

DB2 UDB をインストールするための前提条件 (非 MPP 構成)

DB2 の非 MPP バージョンをインストール前に次の前提条件を実行します。

- 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。DB2 の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 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 5.0 MP3 をインストールし、設定します。インストールの手順について詳しくは、『Veritas Cluster Server インストールガイド』を参照してください。
- DB2 UDB をインストールする前に、DB2 UDB のユーザーアカウントとグループアカウントを次のとおり定義します。
 - DB2 のユーザーアカウントとグループアカウントの定義
p.21 の「[DB2 の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 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 をインストールするための前提条件 (MPP 構成)

DB2 の MPP パージョンをインストールする前に次の前提条件を実行します。

- 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。DB2 の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。
- 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。
- 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)
- クラスタ機能が有効である Veritas File System (CFS)

これらの製品のインストール手順や情報について詳しくは、次のマニュアルを参照してください。

- 『Veritas Storage Foundation Cluster File System インストールガイド』
- 『Veritas Storage Foundation Cluster File System 管理者ガイド』
- DB2 UDB をインストールする前に、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、次の事項を確認します。
 - DB2 のユーザーアカウントとグループアカウントの定義
p.21 の「DB2 の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。
 - DB2 UDB の関連マニュアル

DB2 インスタンス用のファイルシステムの作成

次の項では、DB2 データベースインスタンス用のディスクグループを作成する例について説明します。

DB2 非 MPP インスタンス用のファイルシステムの作成

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 システム上に作成するには

- 1 共有ディスク上に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。この場合は、グループを sdc の 1 つのディスクで構成します。次に例を示します。

```
# vxdg init db2db_dg /dev/sdc
```

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 import db2db_dg
```

- 2 vxassist コマンドを使って、3 GB の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 MPP インスタンス用の共有クラスタファイルシステムの作成

共有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。

Storage Foundation Cluster File System ソフトウェアがインストールされている必要があります。

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 共有ディスクグループを作成するためのコマンドは、CVM マスターノードから発行する必要があります。ノードがマスターまたはスレーブのどちらであるかを判別するには、次のコマンドを入力します。

```
# vxctl -c mode
```

出力の中の、次の行を探します。

```
cluster active - MASTER
```

または

```
cluster active - SLAVE
```

- 2 マスターノードから、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。
- 3 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 sdc です。

```
# vxvg -s init db2db_dg hdisk5
```

- 4 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 -s import db2db_dg
```

- 5 vxassist コマンドを使って、7 GB のボリュームを作成します。

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

- 6 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
```

- 7 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/dbq2db_vol ¥  
/db2_mnt/db2inst1
```

DB2 の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システム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にローカルに存在する必要があります。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るため、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。

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 システム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。たとえば、次のように入力します。

```
# groupadd -g 999 db2iadm1
```

- 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。たとえば、次のように入力します。

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

- 3 DB2 管理サーバー (DAS) のグループを作成します。たとえば、次のように入力します。

```
# groupadd -g 997 db2asgrp
```

Linux システムでの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。共有ストレージの DB2 UDB のインスタンスのホームディレクトリをホストするファイルシステムはこのマウントポイントを使います。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 **db2as** 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 db2fadml -u 1003 -d /home/db2fenc1 -m -s /bin/ksh  
db2fenc1  
# useradd -g db2asgrp -u 1002 -d /home/db2as -m -s /bin/ksh db2as
```

VCS 環境での DB2 UDB のインストール

VCS 環境で DB2 UDB 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。

共有メモリパラメータの設定

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。

バイナリのインストール

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。

DB2 ライセンスのインストール

各ノードに DB2 ライセンスをインストールします。たとえば、次のように入力します。

```
# /opt/IBM/db2/V8.1/adm/db2licm -a db2ese.lic
```

インスタンスのインストール

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。

DB2 UDB 構成の設定

VCS 環境で DB2 UDB を設定するために、次の手順を使用します。

/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 UDB 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。

\$DB2InstHome/.rhosts の作成

各システムで、`$DB2InstHome/.rhosts` というファイルを作成し、このファイルの中に「+」文字を記述します。このファイルにより、パスワードを入力することなくシステムがデータベースにアクセスできます。

セキュリティが問題である場合は、次の例のように、ホスト名とユーザー ID を `.rhosts` ファイルの中に設定します。

```
dbmach01  db2inst1
dbmach02  db2inst1
dbmach03  db2inst1
dbmach04  db2inst1
```

または

```
+ db2inst1
```

`rsh system_name` コマンドを使うと、パスワード入力がないリモートログインをテストできます。クラスタの 1 つのシステムから別のシステムに、このコマンドでは `DB2` インスタンスの (たとえば、`db2inst1`) アカウントでリモートログインできることをテストします。パスワードのプロンプトが表示されません。クラスタの各システムから他のすべてのシステムに対してこのコマンドをテストします。

SUSE での ssh の設定

SUSE で `ssh` を設定するために次の手順を実行します。

SUSE で `ssh` を設定するには

- 1 VCS のインストールを実行するシステムにログインします。
- 2 このシステムで、次のコマンドを実行して DSA キーペアを生成します。

```
# ssh-keygen -t dsa
```


- 3 デフォルトの場所 `~/.ssh/id_dsa` を使います
- 4 プロンプトで、パスフレーズを入力し、確定します。
- 5 次のように入力して `.ssh` ディレクトリの権限を変更します。

```
# chmod 755 ~/.ssh
```

- 6 ファイル `~/.ssh/id_dsa.pub` には、`ssh_dss` で始まり、ファイルが作成されたシステムの名前で終わる行が含まれています。VCS をインストールする予定のすべてのシステムで、この行を `/root/.ssh/authorized_keys2` ファイルにコピーします。

ローカルシステムがクラスタの一部である場合は、必ずそのシステムの `authorized_keys2` ファイルを編集してください。

- 7 インストールを実行するシステムで次のコマンドを実行します。

```
# exec /usr/bin/ssh-agent $SHELL  
# ssh-add
```

このステップはシェルに固有なので、そのシェルが存続している間は有効です。

- 8 インストーラがプロンプトを表示するとき、DSA のパスフレーズを入力します。

VCS をインストールする準備ができています。これで、インストールを行う予定の任意のシステム、またはクラスタ外部の独立したコンピュータで `installvcs` スクリプトを実行することによって、複数のシステムにインストールできます。

各シェルでの `ssh` エージェントの実行を避けるためには X ウィンドウシステムを実行します。パスフレーズを要求するプロンプトを表示しないように設定します。詳しくは Red Hat のマニュアルを参照してください。

\$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 構成の場合

各 DB2 UDB のインスタンス(非 MPP)のデータベースのパーティションでは、ファイル `$DB2InstHome/sqllib/db2nodes.cfg` を修正します。各データベースのパーティションのエントリを作成し、ホスト名として仮想 IP アドレスを割り当てる必要があります。次に例を示します。

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

`virtualhostname` は、`/etc/hosts` ファイルで仮想 IP アドレスと関連付けを行います。仮想 IP アドレスが起動し、動作することを確認します。

MPP 構成の場合

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/sqlllib/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」が、デフォルトのホスト名です。MPP 構成の場合、db2nodes.cfg ファイルでは仮想 IP は使われません。

次に例を示します。

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

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。次に例を示します。

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

MPP 構成と非 MPP 構成の DB2 UDB のインストールの設定確認

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 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。

- 3 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
```

- 4 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。
- 5 データベースを作成します。

```
$ db2 create database dbname
```

- 6 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。

DB2 UDB 用のエージェント のインストール、アップグ レード、削除

この章では以下の項目について説明しています。

- 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
- DB2 用エージェントのインストール
- DB2 UDB 用のエージェントのアップグレード
- DB2 UDB 用のエージェントの削除

エージェントをインストールまたはアップグレード する前に

Veritas High Availability Agent for 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
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 環境での DB2 UDB のインストール
p.22 の「[VCS 環境での DB2 UDB 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。
- Symantec では VCS GUI、Cluster Manager (Java コンソール) をインストールすることをお勧めします。

『Veritas Cluster Server インストールガイド』を参照してください。

DB2 用エージェントのインストール

ディスクをマウントし、エージェントをインストールします。

Linux システム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 DB2 UDB エージェントソフトウェアが収録されているソフトウェアディスクを、システムのドライブに挿入します。ディスクは自動的にマウントされます。
ディスクが自動的にマウントされない場合は、次のように入力します。

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

- 3 rpm がある次のディレクトリに移動します。

■ RHEL 4 または RHEL 5 の場合

```
# cd /mnt/cdrom/dist_arch/cluster_server_agents/db2_agent/rpms
```

ここで、*dist* は **rhel4** または **rhel5**、*arch* は **i686** または **x86_64** です。

■ SLES 9 または SLES 10 の場合

```
# cd /mnt/cdrom/disk_arch/cluster_server_agents/db2_agent/rpms
```

ここで、*dist* は **sles9** または **sles10**、*arch* は **i586** または **x86_64** です。

- 4 DB2 UDB エージェントソフトウェアをインストールします。

```
# rpm -i VRTSvcsdb-5.0.30.00-MP3_GENERIC.noarch.rpm
```

- 5 パッケージがインストールされていることを確認します。次のとおりに入力します。

```
# rpm -q VRTSvcsdb
```

出力は次のようになります。

```
VRTSvcsdb-5.0.30.00-MP3_GENERIC
```

DB2 UDB 用のエージェントのアップグレード

手動でアップグレードできるのは、DB2 用の HA エージェントのみです。installvcs プログラムでは、VRTSvcsdb パッケージは自動的にアップグレードされません。

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 4.1 または 5.0 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 エージェントの無効化
p.50 の「[DB2 エージェントの無効化](#)」を参照してください。
 - DB2 UDB 用のエージェントの削除
p.31 の「[DB2 UDB 用のエージェントの削除](#)」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.30 の「[DB2 用エージェントのインストール](#)」を参照してください。
- 3 新しい Db2udbTypes.cf を /etc/VRTSagents/ha/conf ディレクトリから /etc/VRTSagents/ha/conf/config ディレクトリにコピーします。
- 4 main.cf ファイルの include 文の Db2udbTypes.cf ファイルの場所を更新します。
たとえば、次のようにこの文を変更します。

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

読み込むには、次のように指定します。

```
include "/etc/VRTSagents/ha/conf/config/Db2udbTypes.cf"
```
- 5 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.47 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。

DB2 UDB 用のエージェントの削除

エージェントを削除する前に、無効にする必要があります。

p.50 の「[DB2 エージェントの無効化](#)」を参照してください。

エージェントを削除するには、次の手順を実行します。

Linux システム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 VRTSvcsdb
```


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

DB2 UDB のサービスグループの設定について

この章では、DB2 UDB エージェントの設定方法について説明します。

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 DB2 UDB エージェントのサービスグループテンプレートを編集できます。
[p.37 の「DB2 UDB エージェントの Cluster Manager \(Java コンソール\)からの設定」](#)を参照してください。
- VCS 管理コンソールを使います。
VCS 管理コンソールでサービスグループを設定できます。VCS 管理コンソールについて詳しくは、『Veritas Cluster Server ユーザーズガイド』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 ユーザーズガイド』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。
 - `haconf -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 ユーザーズガイド』を参照してください。
 - **main.cf** ファイルの編集による DB2 UDB エージェントの設定
p.41 の「**main.cf** ファイルの編集による DB2 UDB エージェントの設定」を参照してください。

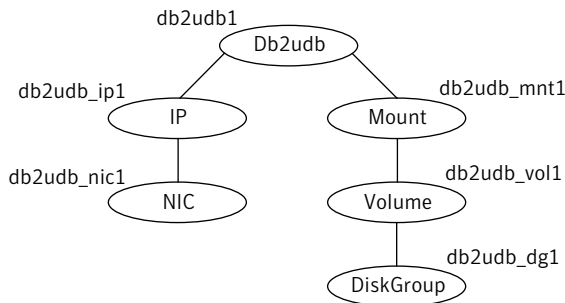
VCS の DB2 UDB の設定について

MPP と非 MPP 構成の DB2 UDB のサービスグループを設定できます。

非 MPP 構成の DB2 UDB サービスグループ

図 4-1 は、非 MPP 構成 DB2 UDB 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ツリー



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 UDB 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。

MPP 構成の DB2 UDB サービスグループ

DB2 UDB 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。

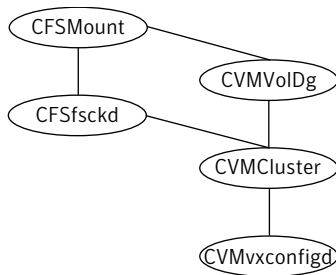
- パラレルの CVM
- DB2 フェールオーバー

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

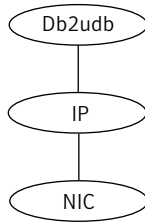


DB2 フェールオーバーサービスグループ

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 のサービスグループを設定する前に

DB2 の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計画のクラスタにあるすべてのノードで VCS 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 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で DB2 がインストールされ、同一に設定されていることを確認します。
p.22 の「VCS 環境での DB2 UDB 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で DB2 用 Veritas エージェントがインストールされていることを確認します。
p.30 の「DB2 用エージェントのインストール」を参照してください。

dbtypes Types.cf ファイルのインポート

DB2 UDB のテンプレートを使う前に、VCS エンジンに Db2udbTypes.cf ファイルをインポートするために Cluster Manager (Java コンソール) を使います。

dbtypesTypes.cf ファイルを Cluster Manager を使って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。次のとおりに入力します。

```
# haguie
```

- 2 クラスタにログインし、Cluster Explorer が起動するのを待ちます。
- 3 ファイルメニューから、[タイプ of インポート (Import Types)] を選択します。要求があった場合は、クラスタを読み書き両用モードに切り替えます。
- 4 [タイプ of インポート (Import Types)] ダイアログボックスで、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、ファイルがインポートされるのを待ちます。
- 6 設定を保存します。

Db2udb タイプが VCS エンジンにインポートされるとき、Db2udb エージェントは設定できます。

以下を参照してください。

- MPP 構成 DB2 UDB を使う場合
p.37 の「[Cluster Manager \(Java コンソール\) による DB2 UDB MPP のサービスグループの追加](#)」を参照してください。
- 非 MPP 構成 DB2 UDB を使う場合
p.39 の「[Cluster Manager \(Java コンソール\) による DB2 UDB 非 MPP のサービスグループの追加](#)」を参照してください。

DB2 UDB エージェントの Cluster Manager (Java コンソール) からの設定

DB2 UDB 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 UDB 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、『VeritasCluster Server ユーザーズガイド』を参照してください。

Cluster Manager (Java コンソール) による DB2 UDB 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.36 の「[dbtypes Types.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 UDB 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 UDB 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 UDB 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 UDB 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 UDB 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.36 の「[dbtypes Types.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。

- 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。

main.cf ファイルの編集による DB2 UDB エージェントの設定

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。この方法を使用する場合は、設定を実装するために VCS を停止して、再起動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 DB2 用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 を編集時に 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 UDB 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 UDB を使うためにエージェントを設定するには

- 1 Db2udbTypes.cf ファイルを指す完全修飾パスを追加します。

```
include "/etc/VRTSagents/ha/conf/Db2udb/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、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.64 の「Linux 用の 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.56 の「Db2udb のリソースタイプの属性」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 ユーザーズガイド』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 UDB 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 UDB を使うためにエージェントを設定するには

- 1 Db2udbTypes.cf ファイルを指す完全修飾パスを追加します。

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

- 2 DB2 UDB 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.56 の「[Db2udb のリソースタイプの属性](#)」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。db2udb_grp1 グループでは、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 ユーザーズガイド』を参照してください。

- 5 ファイルを保存して閉じます。

設定の確認

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 エンジンを開始します。

```
# 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あるとき、システム B の VCS を開始します。

```
# hstart
```

- 5 すべての DB2 UDB 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。

```
# hagrps -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。

```
# hagrps -offline db2udb_grp1 -sys sysa
# hagrps -offline db2udb_grp3 -sys sysa
# hagrps -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。

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

- 8 DB2 UDB サービスグループをシステム B に切り替えます。

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

- 9 すべての DB2 UDB 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。

```
# hagrps -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 UDB インスタンスの詳細監視の設定

DB2 用の Veritas エージェント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 ユーザーズガイド』を参照してください。

DB2 UDB インスタンスの詳細監視の有効化

DB2 UDB 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。

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 UDB 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ついては次のテーブルを参照します。

p.57 の [表 B-2](#) 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_grp1
```

- 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_grp1
```

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のあるアクション) 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 通知について詳しくは、『Veritas Cluster Server ユーザーズガイド』を参照してください。 このアクションは、通常、設定エラーに関連付けられます。
WARN	リソースの状態をオンラインと判断し、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 <code>db2error.dat</code> 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。

アクション	説明
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。

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にするには

- 1 VCS 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_grp1
```

- 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 -modify  
resource InDepthMonitor 0
```

次に例を示します。

```
# hares -modify db2udb InDepthMonitor 0  
# haconf -dump -makero  
# hagrps -unfreeze db2udb_grp1
```


DB2 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 のサービスグループのオンライン化
- DB2 のサービスグループのオフライン化
- DB2 のサービスグループの切り替え
- DB2 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

Cluster Manager (Java コンソール)、クラスタ管理コンソール、またはコマンドラインからサービスグループを管理できます。次の手順では Cluster Manager (Java コンソール) を使用します。

DB2 の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ために次の手順を実行します。サービスグループをオンラインにする初期の処理では、エージェントのメモリ使用状況が瞬間的に急上昇すること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にするには

- 1 Cluster Explorer から、設定ツリーの [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、このグループのすべてのリソースを有効にするために [Enable Resources] をクリックします。

- 3 サービスグループを右クリックし、[Enable]をマウスで指し、サービスグループを有効にする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)]、[Close Configuration]の順に選択します。
- 5 サービスグループを右クリックし、[Online]をマウスで指し、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 の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ために **Cluster Manager (Java コンソール)** からの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初期の処理では、エージェントのメモリ使用状況が瞬間的に急上昇すること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状態にするには

- 1 [Service Groups] タブを選択した **Cluster Explorer** の設定ツリーで、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 [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 のサービスグループの切り替え

サービスグループを切り替える処理は現在のシステムをオフライン状態にして、別のシステムでオンラインにすることです。サービスグループを切り替えるために **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 [Service Groups] タブを選択した **Cluster Explorer** の設定ツリーでサービスグループを右クリックします。
- 2 [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 エージェントの無効化

システムのエージェントを無効にするには、最初に **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、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判断します。プロンプトで、次のとおりに入力します。

```
# hagrpl -state service_group -sys system_name
```

- 2 サービスグループがオンラインの場合は、オフライン状態にします。プロンプトで、次のとおりに入力します。

```
# hagrpl -switch service_group -to system_name
```

または

```
# hagrpl -offline service_group -sys system_name
```

- 3 システムのエージェントを停止します。プロンプトで、次のとおりに入力します。

```
# haagent -stop DB2 -sys system_name
```

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

また、`ps` コマンドを使って、エージェントが停止していることを確認することもできます。

- 5 エージェントが停止したら、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、以下のマニュアル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ます。

Veritas Cluster Server ユーザーズガイド

DB2 エージェント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。

- [環境変数の db2profile の作成](#)
- [RestartLimit 属性の設定](#)

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の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 インスタンスにインスタンスのログオン ID (`$InstHome/sql1lib/db2profile` など) と関連付けされるホームディレクトリがあります。

ここで `db2profile` 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、環境変数は供給されます。

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。DB2 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。

DB2 UDB のリソースタイプ 定義

この付録では以下の項目について説明しています。

- [DB2 UDB のリソースタイプ定義について](#)
- [Db2udb のリソースタイプの属性](#)

DB2 UDB のリソースタイプ定義について

このスタブは共有コンテンツのためのコンテンツです。

p.61 の「[Linux 設定ファイルの例](#)」を参照してください。

DB2 UDB タイプの定義ファイル: Db2udbTypes.cf

以下に示すのは DB2 エージェントのタイプ定義です。

```
type Db2udb (  
    static str ContainerType = Zone  
    static str AgentDirectory = "/opt/VRTSagents/ha/bin/Db2udb"  
    static str AgentFile = "/opt/VRTSagents/ha/bin/Db2udb/  
    Db2udbAgent"  
    static keylist SupportedActions = { VRTS_GetInstanceName,  
    VRTS_GetRunningServices }  
    static int CleanTimeout = 240  
    static int MonitorTimeout = 240  
    static int OfflineTimeout = 240  
    static int OnlineRetryLimit = 2  
    static int OnlineTimeout = 180  
    static int OnlineWaitLimit = 1
```

```

static int RestartLimit = 3
static int ToleranceLimit = 1
static str ArgList[] = { DB2InstOwner, DB2InstHome,
IndepthMonitor, DatabaseName, NodeNumber, StartUpOpt,
ShutDownOpt, AgentDebug, Encoding, WarnOnlyIfDBQueryFailed,
LastWarningDay, ContainerName }
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
str ContainerName
)

```

Db2udb のリソースタイプの属性

DB2 リソースにはいくつかの必須属性と省略可能な属性があります。

表 B-1 は DB2 用エージェントの必須属性を示します。

表 B-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 B-2 は DB2 用エージェントの省略可能な属性を示します。

表 B-2 DB2 用エージェントの省略可能な属性

省略可能な属性	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>メモ: 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>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>Default: 0</p> <p>データ形式と値の種類: 整数 - スカラー</p>
StartupOpt	<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。/opt/VRTSagents/ha/bin/Db2udb ディレクトリの start_custom_\$db2instance_\$nodenum 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: 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 start_custom_db2inst1_1 を実行します。/opt/VRTSagents/ha/bin/Db2udb 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>

省略可能な属性	説明
ShutDownOpt	<p>この属性で許可される値は STOP と 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"> ■ 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。 <p>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。</p> <p>例: パーティションのオフライン機能が <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。<code>DB2</code> 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> <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。このリリースより前の <code>IndepthMonitor</code> では、データベースに対してデフォルトの SQL クエリーが実行されていました。5.0 では、このデフォルトのクエリーは存在しなくなりました。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。</p> <p>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>

省略可能な属性	説明
AgentDebug	<p>この属性の値が1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>
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>

表 B-3 は DB2 のエージェントの内部属性を示します。

表 B-3 DB2 用エージェントの内部属性

必須属性	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>使用禁止。内部使用限定。</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 設定ファイルの例](#)

Linux 設定ファイルの例

このセクションでは Linux 用の DB2 UDB の設定例を示します。

Linux 用の非 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。2 つのデータベースパーティションが設定されています。

```
include "types.cf"
include "/etc/VRTSagents/ha/conf/Db2udb/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 用の 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 "/etc/VRTSagents/ha/conf/Db2udb/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 vxfscckd
//          {
//          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

.rhosts
データベースアクセスのための編集 24
/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 37
DB2 サービスグループの設定 37
CVM サービスグループの平行モード 35
CVM (インフラストラクチャ) サービスグループ 35

D

DatabaseName 属性 57
DB2
エラー処理 47
db2gcf -d コマンド 11
db2gcf -u コマンド 11
DB2InstHome 属性 56
DB2InstOwner 属性 56
db2nodes.cfg
パーティションの仮想ホスト 25
db2setup のツール
バイナリのため
インスタンス 22
DB2 UDB ESE (Enterprise Server Edition)
設定ファイルの例 61、64
Db2udbTypes.cf
main.cf ファイルに追加 42~43
タイプ定義ファイル 55
DB2 UDB インスタンス
共有ストレージにインストール 23
DB2 UDB エージェントの設定
Cluster Manager (Java コンソール) の使用 37
設定ファイルの編集 41
DB2 UDB タイプの定義ファイル 55
DB2 UDB のインストール
共有ストレージのインスタンス 23
システムバイナリ 22

設定の確認 26
必要条件 22

Db2udb リソース
IP への依存
Mount リソース 34
Db2udb リソースタイプの属性 56
DB2 インスタンスの停止 11
dbtypesTypes.cf ファイル
インポート 36
dbtypes Types.cf ファイルのインポート 36
DiskGroup リソース
Volume リソースに必要 34

E

Encoding 属性 58

H

hagrp コマンド 50
hagui コマンド 36

I

IP アドレス
仮想 25
IP リソース
NIC リソースが必要 34

J

Java コンソール
サービスグループの設定 37、39

M

main.cf ファイル
DB2 UDB EEE の例 61
DB2 UDB ESE マルチパーティションの例 64
設定の確認 44
編集 41
例 41
Mount リソース
Volume リソースが必要 34

MPP (Massively Parallel Processing)

サポート 10

N

NIC リソース

IP リソースに必要 34

NodeNumber 属性 57

R

ResourceInfo 属性 14

rsh コマンド 24

S

SMP (対称型マルチプロセッシング)

サポート 10

V

VCS サポートバージョン 10

Volume リソース

DiskGroup リソースが必要 34

W

WarnOnlyIfDBQueryFailed 属性 11

あ

依存関係

DB2 と CVM のサービスグループ間 39

main.cf での定義 44

インスタンス

起動と停止 26

共有ディスクへのインストール 23

エージェント

設定

概要 33

か

仮想 IP アドレス 25

監視

詳細監視の設定 46

通常と詳細 46

グループ (ユーザー)

追加 21

コマンド

hagrp 50

hagui 36

rsh 24

さ

サービスグループ

Cluster Manager (Java コンソール) の追加 37

CVM 35

DB2 と CVM の依存関係リンク 39

main.cf ファイルに作成 42~43

MPP 構成

依存関係 35

ウィザードを使った追加 39

非 MPP 構成

依存関係 34

詳細監視

無効化 48

有効化 46

設定ファイル

db2nodes.cfg

パーティションのためのホスト 25

Db2udbTypes.cf 55

main.cf

例 61, 64

た

通常監視

記述 46

テスト

サービスグループの設定 45

リモートログオン 24

テンプレート

DB2 MPP サービスグループ 39

非 MPP サービスグループに使用 41

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) 57

ファイル

.rhosts 24

db2nodes.cfg 25

Db2udbTypes.cf 55

main.cf (VCS 設定) 41

ファイルシステム

MPP 構成の作成 19

非 MPP インスタンス用の作成 18