

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

Linux

6.0.1

Veritas Cluster Server Agent for Sybase 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 Sybase の概要	9
	Veritas Cluster Server Agent for Sybase について	9
	VCS agent for Sybase でサポートされるソフトウェア	10
	エージェントが Sybase の高可用性を確保するしくみ	10
	Sybase エージェントと SybaseBk エージェントがインテリジェントなリソース 監視をサポートするしくみ	11
	Sybase エージェントの関数について	12
	Sybase 用エージェントの機能	12
	SybaseBk エージェントの関数	15
	Sybase 用エージェントの IPC クリーンアップ機能の使用	16
	Sybase エージェントの監視オプション	18
	Sybase 用の VCS エージェントの Action 機能	18
	VCS クラスタにおける典型的な Sybase 設定	19
第 2 章	Sybase のインストールと設定	21
	Sybase をインストールするための VCS の必要条件	21
	Sybase インストールディレクトリ	21
	共有ディスク上の SYBASE ディレクトリ	22
	Sybase ホームディレクトリのシステムグループとユーザー	22
	データベース dbspace	22
	透過的 TCP/IP フェールオーバー	22
	Sybase エージェントの言語設定	23
	VCS 環境での Sybase のインストール	23
	詳細監視のための Sybase の設定	23
第 3 章	Sybase 用のエージェントのインストール、アップグ レード、および削除	27
	Sybase 用の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	27
	agentSybase 用のインストール	28
	Sybase 用エージェントの無効化	29
	Sybase エージェントの削除	30
	Sybase 用エージェントのアップグレード	31

第 4 章	Sybase のための VCS サービスグループの設定	35
	Sybase のサービスグループの設定について	35
	Sybase サービスグループを設定する前に	35
	SybaseTypes.cf ファイルのインポート	36
	Sybase 用サービスグループの設定 Sybase 用	38
	Cluster Manager (Java コンソール) からの Sybase のサービスグループの 設定	38
	コマンドラインを使用した、Sybase のサービスグループの設定	40
	Sybase のパスワードの暗号化	42
	Sybase のエージェントに対する詳細監視の設定について	43
	Sybase のエージェントに対する詳細監視の無効化	43
	Sybase のエージェントに対する詳細監視の有効化	43
	エージェントに対するインテリジェントなリソース監視の有効化と無効化	44
	AMF カーネルドライバの管理	46
第 5 章	Sybase のための VCS サービスグループの管理	49
	Sybase のサービスグループの管理について	49
	Sybase サービスグループのオンライン化	49
	Sybase サービスグループのオフライン化	50
	Sybase サービスグループの切り替え	50
	Sybase サービスグループ設定の修正	51
	Sybase のエージェントログの表示	51
付録 A	Sybase のリソースタイプ定義	53
	Sybase のリソースタイプと属性の定義について	53
	Sybase リソースタイプ	53
	Sybase エージェントのタイプ定義	53
	Sybase エージェントの属性の定義	54
	SybaseBk リソースタイプ	62
	SybaseBk エージェントのタイプ定義	62
	SybaseBk 用エージェントの属性の定義	63
付録 B	Sybase の設定例	67
	Sybase 用エージェントの設定例について	67
	Sybase 用エージェントのリソースの依存関係グラフ	67
	Linux システムの Sybase サービスグループの設定例	68
	Sybase ASE CE の設定のための main.cf ファイルのサンプル	70
	VCS の制御下にある、Sybase バイナリをインストールするための CFS の共有マウントポイントを持つ基本的な Sybase ASE CE クラスタ 設定向けの main.cf のサンプル	71

	Sybase バイナリをインストールするための VxFS のローカルマウントポ イントを持つ基本的な Sybase ASE CE クラスタ設定向けの main.cf のサンプル	75
	CVM VVR のプライマリサイトの main.cf のサンプル	80
	CVM VVR のセカンダリサイトの main.cf のサンプル	86
付録 C	以前のリリースで導入された変更	93
	以前のリリースで導入された変更点	93
索引	95

Veritas Cluster Server Agent for Sybase 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 Sybase について
- VCS agent for Sybase でサポートされるソフトウェア
- エージェントが Sybase の高可用性を確保するしくみ
- Sybase エージェントと SybaseBk 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
- Sybase エージェントの関数について
- Sybase エージェントの監視オプション
- Sybase 用の VCS エージェントの Action 機能
- VCS クラスタにおける典型的な Sybase 設定

Veritas Cluster Server Agent for Sybase について

Veritas Cluster Server Agent for Sybase は、設定した Sybase サーバーのオンライン化、監視およびオフライン化を行います。

Sybase ASE Enterprise Edition では、次のエージェントが連携して VCS クラスタ内の Sybase の高可用性を実現します。

- SQL サーバー用のエージェント - Sybase
- バックアップサーバー用のエージェント - SybaseBk

これらのエージェントには、リソースタイプの宣言文と実行ファイルが含まれており、Sybase は Sybase リソースタイプ、SybaseBk は SybaseBk リソースタイプとなります。

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

Sybase Cluster Edition では、Sybase が提供する `qrmutil` ユーティリティを使って、エージェントがインスタンスの状態(動作中または停止中)を確認します。`qrmutil` がインスタンスの状態をエラーによる停止状態として報告した場合は、エージェントはノードを再起動し、インスタンスは自動的に再開されます。

設定された Sybase サーバーがシステムで動作していないことを Sybase エージェントが検出すると、Sybase サーバークラスタの `OnlineRetryLimit` 属性の値に基づいて、サービスグループはグループが停止したのと同じシステムで再起動します。

ASE Enterprise Edition では、設定した Sybase サーバーがシステム上で動作していないことがエージェントによって検出されると、その Sybase サービスグループは、サービスグループの `SystemList` にある次の使用可能なシステムにフェールオーバーされます。設定した Sybase サーバーが新しいシステム上で起動されることにより、Sybase サーバーおよびデータの高可用性が保証されます。

Sybase エージェントと SybaseBk 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。

Sybase と SybaseBk の VCS エージェントのインテリジェントなリソース監視機能は、`haimfconfig` スクリプトを使うかまたは手動で、有効または無効にできます。

p.44 の「[エージェントに対するインテリジェントなリソース監視の有効化と無効化](#)」を参照してください。

IMF に対応したエージェントが起動すると、エージェントは非同期監視フレームワーク(AMF)カーネルドライバを初期化します。リソースが安定状態になった後に、エージェントは、リソースを監視するために必要な詳しい情報を AMF カーネルドライバに登録します。たとえば、Sybase エージェントはプロセスの PID を AMF カーネルドライバに登録します。エージェントの `imf_getnotification` 関数は、リソースのあらゆる状態変化を待ち機します。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が監視エージェントの関数を実行し、リソースの状態を確認します。エージェントが状態の変化を 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。

Sybase エージェントの関数について

エージェントが実行する機能はエントリポイントと呼ばれます。Veritas agent for Sybase 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、クリーニング、アクションがあります。

Sybase 用の Veritas Cluster Server エージェントスイートの一部である次のエージェントの機能を見直します。

- Sybase (SQL サーバー) 用エージェントの機能
p.12 の「[Sybase 用エージェントの機能](#)」を参照してください。
- SybaseBk (Backup Server) エージェントの関数
p.15 の「[SybaseBk エージェントの関数](#)」を参照してください。

Sybase 用エージェントの機能

Sybase のエージェントは、Sybase ASE データサーバーの起動、サーバープロセスの監視、サーバーのシャットダウンを行います。

Sybase エージェントは IMF 対応です。

[表 1-1](#) に、SQL サーバー用の Sybase 用エージェントの機能の一覧を示します。

表 1-1 SQL サーバー用の Sybase 用エージェントの機能

エージェントの機能	説明
オンライン化	<p>次のコマンドを使って、Sybase ASE データサーバーを起動します。</p> <pre>startserver -f \$SYBASE/\$SYBASE_ASE/install/RUN_\$Server</pre> <p>これは、\$Server が instance_name で、\$SYBASE/\$SYBASE_ASE/install/RUN_\$Server が Run サーバーファイルのデフォルトの場所の場合のコマンドです。Run_ServerFile の属性値を指定した場合は、デフォルトの場所の代わりに指定した値が使われます。</p> <p>WaitForRecovery 属性が有効な場合、エージェントは、リカバリが完了するか、オンラインにできるすべてのデータベースがオンライン化されるまで待機します。エージェントは、isql セッションに接続してリカバリの状態を問い合わせます。OnlineTimeout 属性は、OnlineTimeout に到達する前にリカバリが完了できるよう、十分に余裕のある値に設定する必要があります。</p> <p>デフォルトでは、WaitForRecovery 属性は無効になっています。</p> <p>インターフェースファイルの場所が interfaces_File 属性を使って指定されている場合、エージェントは、isql セッションに接続している間、[-I interfaces file] オプションを使います。</p> <p>DelayAfterOnline 属性が設定されている場合、監視機能は、オンライン機能の完了後、DelayAfterOnline 属性で指定された秒数が経過してから呼び出されます。</p>
監視	<p>基本監視モードでは、プロセステーブルで dataserver プロセスをスキャンします。詳細監視モードでは、エージェントは MonScript でオプションとして指定されているスクリプトを実行します。</p> <p>Sybase が提供する qrmutil ユーティリティを使って、エージェントがインスタンスの状態（動作中または停止中）を確認します。qrmutil がインスタンスの状態をエラーによる停止状態として報告した場合は、エージェントはノードを再起動し、インスタンスは自動的に再開されます。</p> <p>p.18 の「Sybase エージェントの監視オプション」を参照してください。</p>

エージェントの機能	説明
オフライン	<p>次の方法で <code>isql</code> コマンドを使って、Sybase SQL Server を停止します。</p> <p>インターフェースファイルの場所が <code>interfaces_File</code> 属性を使って指定されている場合、エージェントは、<code>isql</code> セッションに接続している間、指定されたファイルを使います。</p> <p>エージェントは最初に <code>shutdown with wait</code> コマンドを実行します。</p> <p><code>timeout</code> オプションがサポートされている場合、Sybase エージェントは Sybase データサーバーのシャットダウンの間にこのオプションを使います。</p> <p>Sybase ASE Enterprise Edition の場合、<code>shutdown</code> コマンドのこのタイムアウトオプションは、12.5.4 と 15.0.2 以降のバージョンのためにサポートされます。このタイムアウトオプションが Sybase ASE Enterprise Edition でサポートされない場合は、データサーバーが完全に停止するまで、オフラインスクリプトがループで待機します。エージェントは OfflineTimeout 期間が経過するまで待機します。処理がまだ動作中の場合、オフラインスクリプトは <code>shutdown with nowait</code> コマンドを発行します。</p> <p>Sybase ASE Cluster Edition では、<code>shutdown</code> コマンドのタイムアウトオプションはバージョン 15.5 ESD #1 以降でサポートされます。タイムアウトオプションが Sybase ASE Cluster Edition でサポートされない場合は、データサーバーが完全に停止するまで、オフラインスクリプトがループで待機します。エージェントは OfflineTimeout 期間が経過するまで待機します。プロセスがまだ実行中の場合は、オフラインスクリプトはプロセスを強制終了します。</p> <p>DelayAfterOffline 属性が設定されている場合、監視機能は、オフライン機能の完了後、DelayAfterOffline 属性で指定された秒数が経過してから呼び出されます。</p>
sybase_imf_init	<p>Sybase エージェント用の IMF 通知モジュールである AMF カーネルドライバと連動するようにエージェントを初期化します。この機能は、エージェントが起動すると実行されます。</p>
sybase_imf_register	<p>リソースエンティティを AMF カーネルモジュールに登録または登録解除します。この機能は、リソースが安定した状態(オンラインまたはオフライン)に入った後に、リソースごとに実行されます。</p>
sybase_imf_getnotification	<p>リソースの状態の変更についての通知を取得します。この機能は、AMF カーネルモジュールと連動するようにエージェントが初期化された後に実行されます。この機能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。</p>

エージェントの機能	説明
クリーニング	<p>次の方法で isql コマンドを使って、Sybase SQL Server を強制終了します。</p> <p>エージェントは最初に shutdown with wait コマンドを実行します。</p> <p>Sybase ASE Enterprise Edition では、isql コマンドが失敗した場合、クリーンアップスクリプトが shutdown with nowait コマンドを実行します。プロセスが shutdown コマンドに応答しない場合、エージェントは Sybase データサーバープロセスを強制終了します。</p> <p>Sybase ASE Cluster Edition では、shutdown with wait コマンドがデータサーバーを停止しない場合は、エージェントが直接データサーバープロセスを強制終了するために続行します。</p>
アクション	<p>リソースで事前定義済みの処理を実行します。</p> <p>p.18 の「Sybase 用の VCS エージェントの Action 機能」を参照してください。</p>

SybaseBk エージェントの関数

SybaseBk のエージェントは、Sybase Backup Server の起動、サーバープロセスの監視、サーバーのシャットダウンを行います。

SybaseBk エージェントは IMF 対応です。

表 1-2 に、バックアップサーバー用の Sybase 用エージェントの機能の一覧を示します。

表 1-2 バックアップサーバー用の Sybase 用エージェントの機能

エージェントの操作	説明
オンライン化	<p>次のコマンドを使って、Sybase Backup Server を起動します。</p> <pre>startserver -f \$SYBASE/\$SYBASE_ASE/ install/RUN_\$BackupServer</pre> <p>これは、Run_ServerFile のデフォルトの場所が \$SYBASE/\$SYBASE_ASE/install/RUN_\$BackupServer の場合のコマンドです。Run_ServerFile の属性値を指定した場合は、デフォルトの場所の代わりに指定した値が使われます。</p>
Monitor	<p>プロセステーブルで backupserver プロセスをスキャンします。</p>

エージェントの操作	説明
Offline	<p>次の方法で <code>isql</code> コマンドを使って、Sybase Backup Server を停止します。</p> <p>インターフェースファイルの場所が <code>interfaes_File</code> 属性を使って指定されている場合、エージェントは、<code>isql</code> セッションに接続している間、指定されたファイルを使います。</p> <p>エージェントが最初にコマンド <code>shutdown SYB_BACKUP with wait</code> を実行します。このコマンドが失敗すると、オフラインスクリプトは <code>shutdown SYB_BACKUP with nowait</code> を実行します。</p>
クリーニング	<p>次の方法で <code>isql</code> コマンドを使って、Sybase Backup Server を強制終了します。</p> <p>エージェントが最初にコマンド <code>shutdown SYB_BACKUP with wait</code> を実行します。このコマンドが失敗すると、クリーンスクリプトは <code>shutdown SYB_BACKUP with nowait</code> を実行します。</p> <p>プロセスが <code>shutdown</code> コマンドに応答しない場合、エージェントは、プロセステーブルをスキャンして設定済み Sybase Backup Server に関連付けられたプロセスを特定し、それらのプロセスに対して <code>kill</code> コマンドを実行します。</p>
SybaseBk_imf_init	<p>SybaseBk エージェント用の IMF 通知モジュールである AMF カーネルドライバとインターフェースをとるためにエージェントを初期化します。この機能は、エージェントが起動すると実行されます。</p>
SybaseBk_imf_register	<p>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。</p>
SybaseBk_imf_getnotification	<p>リソースの状態の変更についての通知を取得します。この機能は、AMF カーネルモジュールと連動するようにエージェントが初期化された後に実行されます。この機能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。</p>

Sybase 用エージェントの IPC クリーンアップ機能の使用

Adaptive Server は開始するときに、それが使う共有メモリセグメントに関する情報を格納する共有メモリファイルを `$SYBASE` 内に作成します。**Adaptive Server** 起動パラメータの `-M` を使って、共有メモリファイルを格納するディレクトリの場所を変更できます。起動パラメータ `-M` は、`RUN_$$Server` ファイル内で更新する必要があります。

Sybase ホームディレクトリがマウントされていない場合、Sybase クリーンアップスクリプトは共有メモリファイルにアクセスできず、Sybase プロセスにより割り当てられている IPC リソースをクリーンアップしません。したがって、エージェントでは、共有メモリファイルがローカルシステム上の `/var/tmp/sybase_shm/$Server` ディレクトリにあることが必要です。

`/$SYBASE/$SYBASE_ASE/install` ディレクトリで、`RUN_$Server` ファイルを編集します。共有メモリファイルを格納するディレクトリの場所を、`-M` オプションを使って `/var/tmp/sybase_shm/$Server` に変更します。

たとえば、変更前、`RUN_Sybase_Server` は次のようなパス名です。

```
/home/sybase/ASE-15_0/bin/dataserver ¥  
  
-sSybase_Server ¥  
  
-d/home/sybase/data/master.dat ¥  
  
-e/home/sybase/ASE-15_0/install/Sybase_Server.log ¥  
  
-c/home/sybase/ASE-15_0/Sybase_Server.cfg ¥  
  
-M/home/sybase/ASE-15_0 ¥
```

置換後、このファイルは次のようになります。

```
/home/sybase/ASE-15_0/bin/dataserver ¥  
  
-sSybase_Server ¥  
  
-d/home/sybase/data/master.dat ¥  
  
-e/home/sybase/ASE-15_0/install/Sybase_Server.log ¥  
  
-c/home/sybase/ASE-15_0/Sybase_Server.cfg ¥  
  
-M/var/tmp/sybase_shm/Sybase_Server ¥
```

ここで `Sybase_Server` は、`Adaptive Server` の名前です。

メモ: `/var/tmp/sybase_shm/Sybase_Server` ディレクトリが正しい権限で作成されていることを確認します。

Sybase エージェントの監視オプション

Veritas agent for Sybase で提供されるアプリケーションの監視には、基本と詳細の 2 種類のレベルがあります。

基本監視モードでは、Sybase 用のエージェントは Sybase デーモンプロセスを監視し、プロセスが実行中かどうかを検証します。

Sybase Cluster Edition では、Sybase インスタンスの状態を取得するために、エージェントは Sybase が提供する qrmutil ユーティリティを使います。qrmutil ユーティリティが返す状態が「エラーによる停止状態」の場合は、エージェントによるノードパニックが発生します。設定された Sybase サーバーがシステムで動作していないことを Sybase エージェントが検出すると、Sybase サービスグループの OnlineRetryLimit 属性の値に基づいて、サービスグループはグループが停止したのと同じシステムで再起動します。

次に例を示します。

```
# qrmutil --quorum_dev=/quorum/quorum.dat --monitor=ase1
Executing 'monitor' command for instance 'ase1'
Instance 'ase1' has a failure pending.
# echo $?
99
```

この例では、インスタンス「ase1」がエラーによる停止状態です。インスタンス「ase1」を実行しているノードで、エージェントによるパニックが発生します。ノードは再起動の後、自動的にクラスタに再参加します。

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

p.43 の「[Sybase のエージェントに対する詳細監視の設定について](#)」を参照してください。

Sybase ASE Enterprise Edition では、設定した Sybase サーバーがシステム上で動作していないことがエージェントによって検出されると、その Sybase サービスグループは、サービスグループの SystemList にある次の使用可能なシステムにフェールオーバーされます。設定した Sybase サーバーが新しいシステム上で起動されることにより、Sybase サーバーおよびデータの高可用性が保証されます。

Sybase 用の VCS エージェントの Action 機能

Sybase 用のエージェントは、リソースに対して定義済みのアクションを実行できる、Action 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]
```

エージェントのカスタム処理を追加することもできます。

詳しくは、『Veritas Cluster Server エージェント開発者ガイド』を参照してください。

表 1-3 に、エージェントの事前定義済み処理の説明を示します。

表 1-3 事前定義済みのエージェント処理

Action	説明
checkpoint_all	isql セッションに接続することによって、Sybase データサーバーに対して「checkpoint all」を実行します。 メモ: Action 機能 checkpoint_all は Sybase Cluster Edition には適用できません。

VCS クラスタにおける典型的な Sybase 設定

典型的な設定では、VCS を 2 ノードのクラスタ設定で使います。Sybase データは、共有ディスクにインストールされます。Sybase サーバーバイナリは、両方のノードにローカルにインストールすることも、共有ディスクにインストールすることもできます。Sybase 用のエージェントは、両方のノードにインストールされます。共有ディスクは、VxVM (Veritas Volume Manager) を使って管理できます。

図 1-1 では設定例が示されており、バイナリとデータが含まれる Sybase サーバーは、VxVM を使って管理される共有ディスクまたは共有クラスタディスクグループに完全にインストールされています。

図 1-1 共有ディスク上の Sybase バイナリおよびデータ

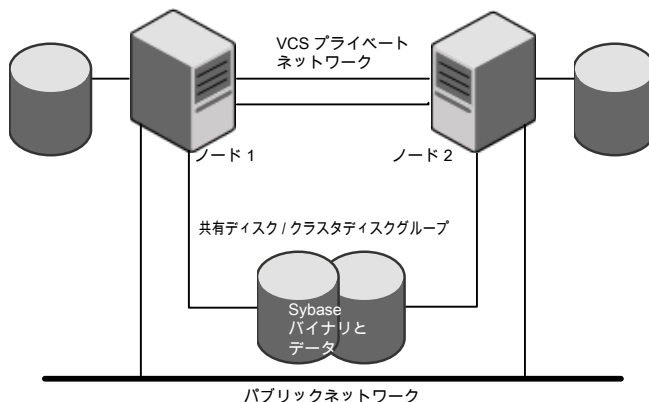
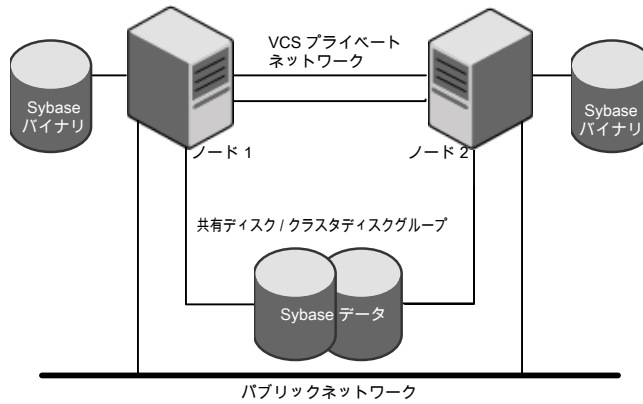


図 1-2 に設定例を示します。この例では、Sybase バイナリはクラスタ内の各ノードにローカルにインストールされており、Sybase データは、VxVM を使って管理される共有ディスクまたは共有クラスタディスクグループにあります。

図 1-2 ローカルディスク上のバイナリおよび共有ディスク上の Sybase データ



Sybase 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。

- Sybase をインストールするための VCS の必要条件
- VCS 環境での Sybase のインストール
- 詳細監視のための Sybase の設定

Sybase をインストールするための VCS の必要条件

VCS クラスタに Sybase をインストールする前に、以下の必要条件を確認してください。
Sybase をインストールする前に、Sybase と VCS を実行するための十分なリソースがクラスタ内のシステムにあることを確認してください。

Sybase インストールディレクトリ

Sybase インストールディレクトリは、ローカルディスクまたは共有ストレージ上に配置できます。

次の前提条件を確認してください。

- Sybase バイナリをローカルディスクにインストールする場合は、クラスタ内のすべてのノードでインストールパスが同じであることを確認します。クラスタ内のすべてのノードで Sybase の設定ファイルが同一であることを確認します。
- Sybase バイナリを共有ディスクにインストールする場合は、共有ディスクのマウントポイントがすべてのノードで同じであることを確認します。Sybase のインストールディレクトリは、環境変数 \$SYBASE によって示されます。また、各システムで、共有ディスク上の \$SYBASE 用に同一のマウントポイントを作成します。

共有ディスク上の SYBASE ディレクトリ

マスターデバイス、`sybssystemprocs` および Sybase ユーザーに関する情報を含むすべてのデータベースデバイスは、共有ディスク上にある必要があります。データベースデバイスをファイルシステム上で作成する場合は、ファイルシステムも共有ディスク上にある必要があります。また、各システムで、共有ディスクにアクセスする同一のファイルシステムマウントポイントを作成します。

Sybase ASE Cluster Edition では、Sybase ASE のクォーラムデバイス用に Storage Foundation Cluster File System (SFCFS) を使ってください。

Sybase ホームディレクトリのシステムグループとユーザー

同じ名前と ID を持つシステムグループとシステムユーザーがすべてのクラスタノードに存在することを確認します。また、このシステムユーザーは、共有ディスク上の Sybase ホームディレクトリの所有権を持っている必要があります。次のコマンドを入力します。

```
# groupadd -g group_id group_name
# useradd -u user_id -g group_name user_name
# chown -R user_name:group_name $SYBASE
```

データベース dbspace

`dbspace` 用の共有ディスクを使っている場合、Sybase データが格納されているディスクグループ上で権限またはアクセスモードを変更します。`sybase` の権限を `660` に変更します。

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

```
# vxedit -g diskgroup_name set group=dba¥
user=sybase mode=660 volume_name
```

透過的 TCP/IP フェールオーバー

Sybase クライアントに対して Sybase サーバーのフェールオーバーが透過的であるようにするには、Sybase サービスグループの一部として IP アドレスを作成します。この IP アドレスは、`$SYBASE/interfaces` ファイル内のデータサーバーおよびバックアップサーバーエントリと一致する必要があります。

エントリを `$SYBASE/interfaces` ファイルに追加するための書式について詳しくは、Sybase のマニュアルを参照してください。

Sybase エージェントの言語設定

Veritas agent for Sybase が特定のロケールで機能するために、Sybase のインストールに正しいローカライズファイルが存在することを確認してください。たとえば、Sybase サーバーで言語変数 `LANG=en_US.UTF-8` が求められる場合は、言語 `en_US.UTF-8` に対応したローカライズファイルが Sybase と一緒にインストールされていることを確認してください。

また、ファイル `$VCS_HOME/bin/vcsenv` を編集して、次の行を追加します。

```
LANG=en_US.UTF-8;export LANG
```

この変更は、ノードで設定されているすべてのエージェントに影響します。

VCS 環境での Sybase のインストール

Sybase のインストール方法については、Sybase のマニュアルを参照してください。

詳細監視のための Sybase の設定

このセクションでは、詳細監視を行う Sybase サーバーを設定するために実行するタスクについて説明します。

p.43 の「[Sybase のエージェントに対する詳細監視の設定について](#)」を参照してください。

メモ:ここで説明する手順は、エージェントに付属するサンプルスクリプト `SqlTest.pl` に固有のものです。詳細監視のためにカスタムスクリプトを使う場合は、それに応じて Sybase データベースを設定する必要があります。

次の手順は、1 つの Sybase クラスタ内で 1 回のみ実行します。

詳細監視のために Sybase を設定するには

- 1 ユーザーのシェルに応じて `SYBASE.sh` ファイルまたは `SYBASE.csh` ファイルをソースとして指定し、`$SYBASE` 環境変数や `$SYBASE_ASE` 環境変数を設定します。
- 2 Sybase サーバーを起動します。

```
# startserver -f ./SYBASE/$SYBASE_ASE/install/RUN_server_name
```

- 3 任意のクラスタノードで Sybase クライアントを起動します。

```
# isql -Usa -SSYBASE_SERVER_NAME
```

メッセージが表示されたら管理者のパスワードを入力します。

- 4 マスターデータベースに接続します。

```
# use master  
# go
```

- 5 Sybase ユーザーアカウントを作成します。

```
# sp_addlogin user_name, password  
# go
```

詳細監視スクリプトはこのアカウントを使って、データベースでトランザクションを実行する必要があります。

- 6 データベースを作成します。

```
# create database database_name  
# go
```

詳細監視スクリプトは、このデータベースでトランザクションを実行する必要があります。

- 7 必要に応じて、データベースのログファイルのサイズを制限してください。

```
# sp_dboption database_name, "trunc log on chkpt", true  
# go
```

- 8 ステップ 6 で作成したデータベースに接続します。

```
# use database_name  
# go
```

- 9 ステップ 5 で作成したユーザーをステップ 6 で作成したデータベースに関連付けます。

```
# sp_adduser user_name  
# go
```

- 10 ユーザーをステップ 5 で作成したユーザーに変更します。

```
# setuser user_name  
# go
```


- 11 データベース内にテーブルを作成します。

```
# create table table_name (lastupd datetime)
# go
```

詳細監視スクリプトは、このテーブルでトランザクションを実行する必要があります。

詳細監視のために **SqlTest.pl** を使う場合は、**datetime** タイプの **lastupd** フィールドがあるテーブルを作成するようにします。

- 12 テーブルに初期値を追加することによって、設定を確認します。

```
# insert into table_name (lastupd) values (getdate())
# go
```

- 13 データベースを終了します。

```
# exit
```


Sybase 用のエージェントのインストール、アップグレード、および削除

この章では以下の項目について説明しています。

- Sybase 用の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
- agentSybase 用のインストール
- Sybase 用エージェントの無効化
- Sybase エージェントの削除
- Sybase 用エージェントのアップグレード

Sybase 用の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 agent for Sybase をインストールするには、次の前提条件を満たしていることを確認します。

- 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。必要に応じて『Veritas Cluster Server インストールガイド』を参照してください。
 - エージェントをインストールする予定のすべてのクラスターノードに、Sybase がインストールされ、設定されていることを確認します。詳しくは Sybase のマニュアルを参照してください。
- p.21 の「[Sybase 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。

- Sybase を実行するすべてのクラスタシステム上で sybase アカウントが有効かつ同一であることを確認します。sybase ユーザーアカウントに Sybase バイナリを実行する権限があることを確認します。
p.22 の「[Sybase ホームディレクトリのシステムグループとユーザー](#)」を参照してください。

agentSybase 用のインストール

Veritas Cluster Server agent for Sybase のエージェントバイナリは VRTSvcsea RPM の一部です。VCS のインストール時にすべてのパッケージまたは推奨パッケージのインストールを選択した場合は、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 RPM を手動でインストールする必要があります。Sybase ASE Cluster Edition では、VCS 製品のインストーラは他のコンポーネントと一緒に Sybase エージェントの VRTSvcsea RPM をインストールします。

Veritas Cluster Server agent for Sybase は、製品ディスクからもインストールできます。クラスタ内の各ノードに Sybase 用のエージェントをインストールします。

エージェントを Linux システムに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。コマンドが Sybase 用の VCS エージェントを返す場合は、この手順を飛ばすことができます。
- 3 Sybase 用エージェントソフトウェアが含ま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、次のとおり入力してください。

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

- 4 エージェントソフトウェアが含まれているフォルダにナビゲートします。

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

各オプションの説明

`dist` は `rhel5`、`rhel6`、`sles10`、`sles11` のいずれかです。

`RHEL` と `SLES` の場合、`arch` は `x86_64` です。

- 5 エージェントソフトウェアをインストールします。

RHEL5 の場合:

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

RHEL6 の場合:

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

SLES 10 の場合:

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

SLES 11 の場合:

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

- 6 VRTSvcsea RPM には、Sybase エージェントのエージェントバイナリが含まれます。この RPM は、DB2 用と Oracle 用の VCS エージェントもインストールします。
- 7 SybaseTypes.cf ファイルを /etc/VRTSagents/ha/conf/Sybase/ ディレクトリから /etc/VRTSvcscs/conf/config ディレクトリにコピーします。

Sybase 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 Sybase 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。

```
# hagrpr -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。

```
# hagrpr -switch service_group -to system_name
```

または

```
# hagrpr -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 Sybase -sys system_name  
# haagent -stop SybaseBk -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。

Sybase エージェントの削除

このセクションでは、クラスタからエージェントを削除する手順について説明します。

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

Sybase 用のエージェントを削除するには

- 1 Sybase リソースと SybaseBk リソースをオフライン状態にします。
- 2 Sybase 用のエージェントを停止します。

```
# haagent -stop Sybase -sys system_name
```

Sybase 用のエージェントが実行されているすべてのノードでこの手順を実行します。

- 3 SybaseBk 用のエージェントを停止します。

```
# haagent -stop SybaseBk -sys system_name
```

SybaseBk 用のエージェントが実行されているすべてのノードでこの手順を実行します。

- 4 次のコマンドを入力して、クラスタ内のすべてのノードからエージェントを削除します。プロンプトに対して適宜応答してください。

```
# rpm -e VRTSvcsea
```

Sybase 用エージェントのアップグレード

この項では、Veritas Cluster Server (VCS) クラスタで、Veritas agent Sybase 用 をアップグレードする手順について説明します。VCS クラスタ内の各ノードで、次の手順を実行します。

メモ: VCS agent Sybase 用 をバージョン 5.0、5.1、6.0 から 6.0.1 にアップグレードできます。p.28 の「[agentSybase 用のインストール](#)」を参照してください。

VCS agentSybase 用 をアップグレードするには

- 1 VCS の設定を保存し、VCS エンジンを停止します。

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

- 2 クラスタノード上の場所に設定ファイル (main.cf) のバックアップを作成します。

- 3 現在の VCS Agent for Sybase のバージョンに基づいて、エージェントがインストールされているすべてのシステムで適切な手順を実行します。

目的

次の手順の実行

バージョンが 5.1 より前の VCS エージェントのアップグレード

1. 次のコマンドを使って VCS agent for Sybase を削除します。

```
# rpm -e VRTSvcssy
```

2. 次のファイルを削除します。

```
/etc/VRTSvcscs/conf/config/SybaseTypes.cf
```

3. Veritas Cluster Server Agent for Sybase をインストールします。

p.28 の「agentSybase 用のインストール」を参照してください。

VCS エージェントの 5.1 以降からのアップグレード

1. 次のコマンドを実行して、エージェントソフトウェアがあるフォルダに移動します。

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

各オプションの説明

dist は rhel5、rhel6、sles10、sles11 のいずれかです。

RHEL と SLES の場合、arch は x86_64 です。

2. 次のコマンドを実行してエージェントをアップグレードします。

注意: 次のコマンドを実行すると、Oracle 用と DB2 用の VCS エージェントもアップグレードされます。

RHEL5 の場合:

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

RHEL6 の場合

```
rpm -U VRTSvcsea-6.0.100.000-GA_RHEL6.i686.rpm
```

SLES 10 の場合:

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

SLES 11 の場合:

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


- 4 SybaseTypes.cf ファイルが /etc/VRTSagents/ha/conf/Sybase/ ディレクトリから /etc/VRTSvcs/conf/config ディレクトリにコピーされたことを確認します。

必ず、古い SybaseTypes.cf ファイルに対して加えたタイプレベルの変更のすべてが、新しくコピーされた SybaseTypes.cf ファイルに反映されるようにします。

たとえば、MonitorInterval 属性の値をデフォルトの 60 秒から 120 秒に変更した場合は、SybaseTypes.cf ファイルが更新されます。これらのタイプレベルの変更は、新しくコピーされた SybaseTypes.cf ファイルに対して手動で適用する必要があります。

- 5 バックアップした場所から /etc/VRTSvcs/conf/config ディレクトリに main.cf をコピーします。
- 6 設定を確認します。

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

- 7 ローカルノードの VCS を起動します。
- 8 次に、他のノードの VCS を起動します。

Sybase 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。

- Sybase のサービスグループの設定について
- Sybase サービスグループを設定する前に
- Sybase 用サービスグループの設定 Sybase 用
- Cluster Manager (Java コンソール) からの Sybase のサービスグループの設定
- コマンドラインを使用した、Sybase のサービスグループの設定
- Sybase のパスワードの暗号化
- Sybase のエージェントに対する詳細監視の設定について
- エージェントに対するインテリジェントなリソース監視の有効化と無効化
- AMF カーネルドライバの管理

Sybase のサービスグループの設定について

Sybase サービスグループを設定するときは、サービスグループのリソースを設定し、設定したリソースの属性値を定義する必要があります。サービスグループの作成および設定には、Administrator 権限が必要です。エージェントを設定する前に、Sybase リソースタイプとその属性を見直してください。

Sybase サービスグループを設定する前に

Sybase 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。

注意: Sybase Cluster Edition では、Sybase エージェントは `membership-mode` が `vcs` に設定されていると想定します。Sybase インスタンスが VCS の制御下で設定されている場合、インストーラは自動的に `membership-mode` を `vcs` に設定します。Sybase サーバーインスタンスが VCS の制御下で手動で設定される場合、またはクォラムデバイスが「ネイティブ」モードで再構築された場合は、`membership-mode` が `vcs` に設定されていることを確認します。

- クラスタのすべてのノードに、Sybase 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.21 の「[Sybase 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。
- クラスタのすべてのノードに、Veritas agent for Sybase がインストールされていることを確認します。
p.28 の「[agentSybase 用のインストール](#)」を参照してください。
- Veritas agent for Sybase のタイプ定義が VCS エンジンにインポートされていることを確認します。
p.36 の「[SybaseTypes.cf ファイルのインポート](#)」を参照してください。

メモ: VCS 6.0 以降では、VCS Cluster Manager (Java コンソール) を使って属性を設定している場合は、パスワードを暗号化する必要はありません。

注意: Sybase Cluster Edition の `Quorum_dev` 属性を設定しない場合、Sybase エージェントは `qrmutil` ベースの検査を実行しません。この設定のエラーは望ましくない結果を引き起こす可能性があります。たとえば、`qrmutil` がエラーによる停止状態を返した場合、エージェントはシステムをパニック状態にしません。このとき、`Quorum_dev` 属性が設定されていないため、Sybase のエージェントは `qrmutil` ベースの検査を実行しません。

SybaseTypes.cf ファイルのインポート

Sybase サービスグループを設定する前に、`SybaseTypes.cf` ファイルを VCS エンジンにインポートする必要があります。

メモ: Java GUI について詳しくは、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 SybaseTypes.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。
- 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 Type)]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/Sybase/SybaseTypes.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。

コマンドラインを使って SybaseTypes.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。

```
# hstop -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、SybaseTypes.cfファイルをインクルードします。

```
# include "/etc/VRTSagents/ha/conf/Sybase/SybaseTypes.cf"
```

Sybase タイプ定義は VCS エンジンにインポートされます。Sybase 用のエージェントを中断したり、停止したりすることなく VCS を設定できます。

Sybase 用サービスグループの設定 Sybase 用

Sybase サービスグループは、次の方法を使って設定できます。

- VCS Cluster Manager (Java コンソール)を使って、エージェントのリソースグループテンプレートを編集します。
p.38 の「[Cluster Manager \(Java コンソール\)からの Sybase のサービスグループの設定](#)」を参照してください。
- タイプ設定ファイルを使って、エージェントに付属の **main.cf** サンプルファイルを直接編集します。この方法では、VCS を再起動して設定を有効にする必要があります。
p.40 の「[コマンドラインを使用した、Sybase のサービスグループの設定](#)」を参照してください。
- Veritas Operations Manager (VOM)を使って設定します。
詳しくは、VOM のマニュアルを参照してください。

Cluster Manager (Java コンソール)からの Sybase のサービスグループの設定

Sybase リソースグループのテンプレートは、Veritas agent for Sybase とともに自動的にインストールされます。VCS Cluster Manager (Java コンソール)を使って、テンプレートを表示し、Sybase サービスグループ、およびそのリソースと属性を見ることができます。属性の値は、設定に合わせて動的に変更できます。

メモ: Java GUI について詳しくは、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。

Java コンソールからサービスグループを設定するには

- 1 リソースタイプ Sybase タイプの定義ファイル SybaseTypes.cf が、設定にインポートされていることを確認します。
p.38 の「[Sybase 用サービスグループの設定 Sybase 用](#)」を参照してください。
- 2 次のいずれかの方法でクラスタ設定ウィザードを起動します
 - Cluster Explorer メニューで、[ツール (Tools)]、[設定ウィザード (Configuration Wizard)] の順に選択します。
 - システム上にサービスグループが存在しない場合、Cluster Explorer からクラスタ設定ウィザードを起動するよう求めるメッセージが表示されます。このプロンプトで [はい (Yes)] をクリックします。

[テンプレート情報を読み込んでいます.... (Loading Templates Information....)]
ウィンドウが表示され、ウィザードが起動します。

- 3 [ようこそ (Welcome)] ダイアログボックスの情報を見直し、[次へ (Next)] をクリックします。
- 4 サービスグループの名前と、そのサービスグループを設定する対象のシステムを指定します
 - サービスグループの名前を入力します。
 - [使用可能なシステム (Available Systems)] ボックスで、サービスグループを設定するシステムを選択します。
 - 右矢印ボタンをクリックして、選択したシステムを [サービスグループのシステム (Systems for Service Group)] ボックスに移動します。ボックスからシステムを削除するには、システムを選択して左矢印をクリックします。
 - フェールオーバーするサービスグループの、システムの優先度を指定します。システムの優先度は数字で順序を指定します。最小の数字が割り当てられたシステムが最高の優先度を持ちます。
 - [サービスグループタイプ (Service Group Type)] で [フェールオーバー] を選択し、[次へ (Next)] をクリックします。
- 5 [テンプレートを使ってサービスグループを設定しますか? (Would you like to use a template to configure the service group?)] ダイアログボックスで [次へ (Next)] をクリックし、テンプレートを使ってサービスグループを設定します。
- 6 Sybase サービスグループを設定するための [SybaseGroup] テンプレートを選択します。

場合によっては、新しいサービスグループ内のリソースの名前の中にすでに使われている名前があることを通知するウィンドウが表示されることがあります。名前の競合があった場合はそれを解決し、[次へ (Next)] をクリックします。
- 7 [次へ (Next)] をクリックし、選択したテンプレートに基づいてサービスグループを作成します。

進行状況インジケータに、サービスグループを作成するために実行されたコマンドの進捗率が表示されます。現在のコマンドがインジケータの上部に表示されます。
- 8 サービスグループが作成されたら、[次へ (Next)] をクリックしてリソースの属性を編集します。

ダイアログボックスの左ペインには、Sybase サービスグループのすべてのリソースの一覧が表示されます。右ペインに属性を表示するリソースを左ペインから選択します。太字の属性は必須の属性を示します。属性値は、以下の手順に従って修正できます。
- 9 デフォルト値を受け入れて設定を完了する場合は、[完了 (Finish)] をクリックします。

Sybase サービスグループのリソース属性を編集するには

- 1 左ペインの一覧からリソースを選択します。リソース属性が右ペインに表示されます。
- 2 修正する属性を選択し、[編集 (Edit)]列の編集アイコンをクリックします。
- 3 [属性の編集 (Edit Attribute)]ダイアログボックスで、属性値を入力します。属性の範囲を修正するには、[グローバル (Global)]または[ローカル (Local)]オプションを選択します。
- 4 [OK]をクリックします。
- 5 リソースごとにこの手順を繰り返し、[完了 (Finish)]をクリックします。設定に応じてすべてのリソースの属性を編集します。
- 6 ウィザードの指示に従って設定を完了します。[完了 (Finish)]をクリックしてウィザードを終了します。

注意: セキュリティを確保するため、パスワードには必ず安全な値を指定する必要があります。

p.42 の「[Sybase のパスワードの暗号化](#)」を参照してください。

コマンドラインを使用した、Sybase のサービスグループの設定

Veritas agent for Sybase には、現在の設定ファイルを直接変更するときの参考として使える、設定ファイルの例が付属しています。この方法では、VCS を再起動して設定を有効にする必要があります。

コマンドラインから Sybase 用のサービスグループを設定するには

- 1 スーパーユーザーとしてクラスタシステムにログインします。
- 2 Sybase タイプの定義が VCS エンジンにインポートされていることを確認してください。
p.38 の「[Sybase 用サービスグループの設定 Sybase 用](#)」を参照してください。
- 3 `/etc/VRTSvcs/conf/config/main.cf` の `main.cf` を編集します。参照用として、`/etc/VRTSagents/ha/conf/Sybase` のサンプルファイルを使います。
 - Sybase サービスグループを作成します。
 - Sybase リソースと SybaseBk リソースを作成します。
p.53 の「[Sybase リソースタイプ](#)」を参照してください。
p.62 の「[SybaseBk リソースタイプ](#)」を参照してください。
 - 設定内のパラメータに一致するようにデフォルトの属性を編集。

セキュリティを確保するため、パスワードには必ず安全な値を指定する必要があります。

p.42 の「[Sybase のパスワードの暗号化](#)」を参照してください。

- 新しく作成されたリソースへの依存関係の割り当て。
/etc/VRTSagents/ha/conf/Sybase/ のサンプルファイルを参照してください。
依存関係の割り当てについて詳しくは、『Veritas Cluster Server 管理者ガイド』を参照してください。

4 ファイルを保存して閉じます。

5 /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 .
```

6 ローカルノードの VCS を起動します。

```
# hastart
```

7 他のノードで VCS を起動します。

8 システムが Sybase サービスグループの `AutoStartList` 属性に含まれている場合は、すべての Sybase サービスグループリソースがオンライン状態になっていることを確認します。

```
# hagr -state
```

9 サービスグループをオフライン状態にして、すべてのリソースが停止したことを確認します。

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

```
# hagr -state
```

10 サービスグループを再びオンライン状態にして、すべてのリソースが利用可能であることを確認します。

```
# hagr -online service_group -sys system_name
```

```
# hagr -state
```

11 Sybase サービスグループを別のノードに切り替えます。

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

- 12 すべての Sybase サービスグループリソースが、別のノード上でオンライン状態であることを確認します。

```
# hagrps -state
```

- 13 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/Sybase_A.log  
/var/VRTSvcs/log/SybaseBk_A.log
```

Sybase のパスワードの暗号化

VCS には、ユーザーパスワードを暗号化する `vcscrypt` ユーティリティが用意されています。Sybase や SybaseBk のリソースタイプ定義のためにパスワードを指定する前に、パスワードを暗号化します。

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

パスワードを暗号化するには

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

```
# vcscrypt -agent
```

ユーティリティを実行すると、パスワードを2回入力するように要求されます。パスワードを入力し、Return キーを押します。

```
Enter Password:  
Enter Again:
```

- 3 ユーティリティがパスワードを暗号化し、暗号化パスワードを表示します。
- 4 この暗号化パスワードを属性値として入力します。

今後の参照用に、暗号化パスワードのコピーを保存します。

Sybase のエージェントに対する詳細監視の設定について

Veritas agent for Sybase で提供されるアプリケーションの監視には、基本と詳細の 2 種類のレベルがあります。基本監視では、Sybase リソースは Sybase デーモンプロセスを監視し、それらのプロセスが継続的にアクティブであるかどうかを確認します。

詳細監視では、Sybase リソースはデータベースのテストテーブルでトランザクションを実行し、Sybase サーバーが正しく機能しているかどうかを確認します。エージェントはこのテストテーブルを内部の目的で使用します。テストテーブルではその他のトランザクションを実行しないことをお勧めします。エージェントは、Sybase リソースの **Monscript** 属性に定義されているスクリプトを使います。詳細監視中、エージェントは指定されたスクリプトを実行します。スクリプトが正常に実行された場合は、データベースが使用可能であると判断されます。デフォルトのスクリプトは設定に応じてカスタマイズできます。

詳細監視を有効にするには、**LevelTwoMonitorFreq** 属性を正の整数に設定する必要があります。また、**User**、**UPword**、**Db**、**Table** の各属性は空("")にしないでください。詳細監視スクリプトのパスを格納する **Monscript** 属性が存在し、**root** の実行権限を持っている必要もあります。

Sybase のエージェントに対する詳細監視の無効化

- 1 次のように実行して、VCS 設定を書き込み可能にします。

```
# haconf -makerw
```

- 2 Sybase の詳細監視を無効にするには、次のコマンドを実行します。

```
# hatype -modify Sybase LevelTwoMonitorFreq 0
```

- 3 次のように実行して、設定を保存します。

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

Sybase のエージェントに対する詳細監視の有効化

次の手順を実行し、データベースに対する詳細監視を有効にします。

詳細監視を有効にするには

- 1 Sybase サーバーが、詳細監視のために設定されていることを確認します。
p.23 の「[詳細監視のための Sybase の設定](#)」を参照してください。
- 2 VCS 設定を書き込み可能にします。

```
# haconf -makerw
```

- 3 Sybase の詳細監視を有効にします。

```
# hatype -modify Sybase LevelTwoMonitorFreq <value>
# hares -modify Sybase_resource User user_name
# hares -modify Sybase_resource UPword encrypted-password
# hares -modify Sybase_resource Db database_name
# hares -modify Sybase_resource Table table_name
# hares -modify Sybase_resource Monscript
"/opt/VRTSagents/ha/bin/Sybase/SqlTest.pl"
```

メモ: 詳細監視を有効にするには、`LevelTwoMonitorFreq` 属性は正の値に設定する必要があります。リソースレベルでこの属性の値を上書きすることもできます。

- 4 設定を保存します。

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

メモ: 詳細監視が設定されている場合は、データベースがいっぱいであると、SQLクエリーで結果をコミットするのに非常に長い時間がかかります。そのような場合、エージェントの監視ルーチンは失敗し、サービスグループをフェールオーバーしようとします。詳細監視が設定されていない場合は、この問題は発生しません。

エージェントに対するインテリジェントなリソース監視の有効化と無効化

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

メモ: VCS は、エージェントの IMF を有効および無効にするために、`haimeconfig` スクリプトを提供します。このスクリプトは、VCS が実行中か、VCS が停止しているときに使うことができます。スクリプトを使って、IMF 対応の付属エージェント、エンタープライズエージェント、カスタムエージェントの IMF を有効または無効にします。

詳しくは、『Veritas Cluster Server 管理者ガイド』の「Enabling and disabling IMF for agents by using script」セクションを参照してください。

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

インテリジェントなリソースの監視を有効にするには

- 1 VCS 設定を書き込み可能にします。

```
# haconf -makerw
```

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

- オフラインのリソースのインテリジェントな監視を有効にするには:

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

- オンラインのリソースのインテリジェントな監視を有効にするには:

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

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

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

また、IMF 属性の `MonitorFreq` キーと `RegisterRetryLimit` キーの値を変更できます。

p.54 の「[Sybase エージェントの属性の定義](#)」を参照してください。

p.63 の「[SybaseBk 用エージェントの属性の定義](#)」を参照してください。

- 3 VCS 設定を保存します。

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

- 4 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。

```
■ /etc/init.d/amf status
```

- 5 エージェントを再起動します。各ノードで次のコマンドを実行します。

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

インテリジェントなリソースの監視を無効にするには

- 1 VCS 設定を書き込み可能にします。

```
# haconf -makerw
```

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

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

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

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

- 4 VCS 設定を保存します。

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

メモ: Sybase Cluster Edition では、**monitor** エントリポイントが実行される時は必ず、**qrmutil** に基づいた監視チェックがエージェントによって実行されます。この監視チェックは、**IMF MonitorFreq** の値に到達するか **LevelTwoMonitorFreq** の値に到達すると実行されます。たとえば、**IMF MonitorFreq** が 10 などの高い値に設定されている場合は、**qrmutil** に基づいた監視チェックは、**MonitorInterval** の 10 倍の秒数が経過してから実行されます。

AMF カーネルドライバの管理

AMF カーネルドライバを起動または停止するための次の手順を実行します。

AMF カーネルドライバを起動するには

- 1 次のファイルで、AMF_START 変数の値を 1 に設定します。

```
# /etc/sysconfig/amf
```

- 2 AMF カーネルドライバを起動します。次のコマンドを実行します。

```
# /etc/init.d/amf start
```

AMF カーネルドライバを停止するには

- 1 次のコマンドを実行します。

```
# /etc/init.d/amf stop
```

- 2 次のファイルで、AMF_START 変数の値を 0 (ゼロ) に設定します。

```
# /etc/sysconfig/amf
```


Sybase 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。

- Sybase のサービスグループの管理について
- Sybase サービスグループのオンライン化
- Sybase サービスグループのオフライン化
- Sybase サービスグループの切り替え
- Sybase サービスグループ設定の修正
- Sybase のエージェントログの表示

Sybase 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。

Sybase 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。

Sybase 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。

Sybase 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。

Sybase サービスグループ設定の修正

Sybase エージェントは、いくつかの方法を使って動的に修正できます。これらの方法には、Cluster Manager (Java コンソール)、Veritas Operations Manager、コマンドラインがあります。

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

Sybase のエージェントログの表示

VCS agent for Sybase は、次のファイルにメッセージを記録します。

`/var/VRTSvcs/log/engine_A.log`

`/var/VRTSvcs/log/Sybase_A.log`

`/var/VRTSvcs/log/SybaseBk_A.log`

Sybase 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。

- [Sybase のリソースタイプと属性の定義について](#)
- [Sybase リソースタイプ](#)
- [SybaseBk リソースタイプ](#)

Sybase のリソースタイプと属性の定義について

リソースタイプとは、エージェントの VCS 設定の定義を表すものであり、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。エージェントの設定には、値をリソースタイプの属性に割り当てる関係があります。

リソースタイプの属性についての参照先

p.54 の「[Sybase エージェントの属性の定義](#)」を参照してください。

p.63 の「[SybaseBk 用エージェントの属性の定義](#)」を参照してください。

`main.cf` サンプル設定ファイルについての参照先

p.68 の「[Linux システムの Sybase サービスグループの設定例](#)」を参照してください。

Sybase リソースタイプ

Sybase リソースタイプのタイプ定義と属性定義について、次に説明します。

Sybase エージェントのタイプ定義

Sybase 用のエージェントのリソースタイプの定義は次のとおりです。

```
type Sybase (  
    static boolean AEPTimeout = 1  
    static keylist SupportedActions = { "checkpoint_all" }  
    str Server  
    str Owner  
    str Home  
    str Version  
    str SA  
    str SApwd  
    str Run_ServerFile  
    int DetailMonitor = 0  
    str User  
    str UPword  
    str Db  
    str Table  
    str Monscript = "/opt/VRTSagents/ha/bin/Sybase/SqlTest.pl"  
    boolean WaitForRecovery = 0  
    str Quorum_dev  
    str interfaces_File  
    int ShutdownWaitLimit = 60  
    int DelayAfterOnline = 10  
    int DelayAfterOffline = 2  
    static int ToleranceLimit = 1  
    static str ArgList[] = { Server, Owner, Home, Version, SA,  
        SApwd, User, UPword, Db, Table, Monscript, DetailMonitor,  
        WaitForRecovery, Run_ServerFile, Quorum_dev, State,  
        interfaces_File, ShutdownWaitLimit, DelayAfterOnline,  
        DelayAfterOffline }  
    static int IMF{} = { Mode=3, MonitorFreq=5, RegisterRetryLimit=3 }  
    static str IMFRegList[] = { Server, Owner, Quorum_dev }  
    static str AgentDirectory = "/opt/VRTSagents/ha/bin/Sybase"  
)
```

Sybase エージェントの属性の定義

Sybase 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 A-1 に、必須属性の一覧を示します。

表 A-1 必須属性

必須属性	定義
Server	<p>\$DSQUERY ASE 名。1 つの Sybase サービスグループには、1 つのサーバーのみを設定するようにしてください。個別のサービスグループで Sybase リソースを設定する利点は、各 Sybase データサーバーを独立してフェールオーバーできることにあります。</p> <p>1 つのサービスグループに複数の Sybase リソースが設定されていると、「n」個のデータサーバーのうちの 1 つでエラーが発生した場合でも、サービスグループ全体がフェールオーバーされます。</p> <p>データ形式と値の種類: 文字列 - スカラー</p> <p>デフォルト値: デフォルト値なし</p>
Owner	<p>パスワード入力用の /etc/nsswitch.conf ファイルで指定された、いずれかのソース (NIS+, /etc/hosts など) の実行ファイルおよびデータベースファイルの定義済み所有者としての Sybase ユーザー。</p> <p>Sybase の実行ファイルとデータベースファイルは、このユーザーのコンテキストでアクセスされます。</p> <p>データ形式と値の種類: 文字列 - スカラー</p> <p>p.22 の「Sybase ホームディレクトリのシステムグループとユーザー」を参照してください。</p>
Home	<p>Sybase バイナリおよび設定ファイルへのパス \$SYBASE。</p> <p>データ形式と値の種類: 文字列 - スカラー</p> <p>デフォルト値: デフォルト値なし</p>
Version	<p>Sybase ASE のバージョン。</p> <p>データ形式と値の種類: 文字列 - スカラー</p> <p>デフォルト値: デフォルト値なし</p> <p>注意: Sybase リソースが VCS でオンラインになったら、Home と Version の属性は変更しないでください。Sybase Cluster Edition では、リソースがオンラインの状態にあるときに Home と Version の属性に無効な値を設定すると、ノードでパニックが発生します。</p>
SA	<p>Sybase データベース管理者。この属性はシャットダウンのために ASE に接続するために必要です。</p> <p>データ形式と値の種類: 文字列 - スカラー</p> <p>デフォルト値: デフォルト値なし</p>

必須属性	定義
SAPswd	<p>Sybase データベース管理者の暗号化されたパスワード。このパスワードはシャットダウンのために ASE に接続するために必要です。</p> <p>データ形式と値の種類: 文字列 - スカラー</p> <p>デフォルト値: デフォルト値なし</p> <p>p.42 の「Sybase のパスワードの暗号化」を参照してください。</p> <p>メモ: SA ユーザーがパスワードを必要としていない場合は、この属性の値を指定する必要はありません。</p>
Quorum_dev	<p>クォラムデバイスは、クラスタメンバーシップを管理し、クラスタ設定データを保存し、サーバーインスタンスとノード間で共有される情報を格納します。クォラムデバイスはクラスタ内のすべてのノードからアクセスできるディスクです。完全修飾のクォラムデバイス名を指定します。</p> <p>データ形式と値の種類: 文字列 - スカラー</p> <p>デフォルト値: デフォルト値なし</p> <p>メモ: この属性は、クラスタエディションに対してのみ指定する必要があります。</p> <p>注意: Sybase Cluster Edition の Quorum_dev 属性を設定しない場合、Sybase エージェントは qrmutil ベースの検査を実行しません。この設定のエラーは望ましくない結果を引き起こす可能性があります。たとえば、qrmutil がエラーによる停止状態を返した場合、エージェントはシステムをパニック状態にしません。このとき、Quorum_dev 属性が設定されていないため、Sybase のエージェントは qrmutil ベースの検査を実行しません。</p>

表 A-2 に、オプション属性を一覧で示します。

表 A-2 オプション属性

オプション属性	定義
DetailMonitor	<p>Sybase サーバーの詳細監視を行うかどうかを指定します。正の整数値は、リソースが Sybase サーバーの詳細監視を行うことを示し、値 0 は詳細監視を行わないことを示します。デフォルトは 0 です。</p> <p>データ形式と値の種類: 整数 - スカラー</p> <p>メモ: DetailMonitor 属性は VCS 6.0.1 では推奨されなくなりました。代わりに、Sybase エージェントの LevelTwoMonitorFreq 属性を使用できます。LevelTwoMonitorFreq 属性のデフォルト値は 0 (ゼロ) です。</p>

オプション属性	定義
User	<p>データベースユーザー。トランザクションはデータベースで、そのユーザーのコンテキストで実行されます。LevelTwoMonitorFreq が正の整数値に設定されている場合は、この属性の値を指定する必要があります。</p> <p>データ形式と値の種類: 文字列 - スカラー</p> <p>デフォルト値: デフォルト値なし</p>
UPword	<p>データベースユーザーの暗号化されたパスワード。LevelTwoMonitorFreq が正の整数値に設定されている場合は、この属性の値を指定する必要があります。ただし、データベースユーザーがパスワードを必要としない場合は、この属性の値を指定する必要はありません。</p> <p>p.42 の「Sybase のパスワードの暗号化」を参照してください。</p> <p>データ形式と値の種類: 文字列 - スカラー</p> <p>デフォルト値: デフォルト値なし</p>
Db	<p>詳細監視に使用されるデータベースの名前。このデータベースにある詳細監視スクリプトにより使用されるテーブル。LevelTwoMonitorFreq が正の整数値に設定されている場合は、この属性の値を指定する必要があります。</p> <p>データ形式と値の種類: 文字列 - スカラー</p> <p>デフォルト値: デフォルト値なし</p>
Table	<p>詳細監視スクリプトがトランザクションを実行する対象のテーブルの名前。LevelTwoMonitorFreq が正の整数値に設定されている場合は、この属性の値を指定する必要があります。</p> <p>データ形式と値の種類: 文字列 - スカラー</p> <p>デフォルト値: デフォルト値なし</p>
Monscript	<p>詳細監視スクリプトのパス。この属性のデフォルト値は、エージェントに付属するスクリプト SqlTest.pl のパスです。LevelTwoMonitorFreq が正の整数値に設定されている場合は、この属性の値を指定する必要があります。</p> <p>データ形式と値の種類: 文字列 - スカラー</p> <p>デフォルト値: デフォルト値なし</p> <p>メモ: デフォルトでは、SqlTest.pl スクリプトに実行の権限セットがあります。カスタムの詳細監視スクリプトを指定する場合は、カスタムの詳細監視スクリプトにも実行の権限セットがあることを確認します。</p>

オプション属性	定義
WaitForRecovery	<p>回復手順は、サーバーのデータベースをトランザクションログから再構築します。このブール属性は、エージェントがオンラインエージェント機能中に、リカバリが完了し、オンラインにできるすべてのデータベースがオンライン化されるまで待機するかどうかを指定します。この機能は、12.5.1 以前の Sybase ASE サーバーではサポートされていません。</p> <p>データ形式と値の種類: ブール - スカラー</p> <p>デフォルト値: 0</p>
Run_ServerFile	<p>Sybase インスタンス用の RUN_SERVER ファイルの場所を指定します。この属性の値を指定しない場合、このファイルのデフォルトの場所が使われます。</p> <p>データ形式と値の種類: 文字列 - スカラー</p> <p>デフォルト値: デフォルト値なし</p>

オプション属性	定義
IMF	

オプション属性	定義
	<p>このリソースタイプレベルの属性は、Sybase エージェントがインテリジェントなリソース監視を実行する必要があるかどうかを決定します。リソースレベルでこの属性の値を上書きすることもできます。</p> <p>この属性は次のキーを含んでいます</p> <ul style="list-style-type: none"> ■ 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: 3 ■ 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す頻度を指定します。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>sybase_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 キーの値によって、エージェントがリソースの登録を再試行する必要がある回数が決まります。エージェントが、指定された制限内にリソースを登録できない場合、リソースの状態が変化するか、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.44 の「エージェントに対するインテリジェントなリソース監視の有効化と無効化」を参照してください。</p>

オプション属性	定義
	データ形式と値の種類: 整数 - 関連
interfaces_File	<p>Sybase インスタンスのディレクトリ名とファイル名を含む、インターフェースファイルの場所を指定します。この属性が設定されると、isql セッションに接続するときに [-I interfaces file] オプションが使われます。この属性が設定されていない場合は、エージェントは -I オプションをしません。</p> <p>データ形式と値の種類: 文字列 - スカラー</p> <p>デフォルト値: デフォルト値なし</p> <p>例: <code>/sybase/my_interfaces_file</code></p> <p>メモ: interfaces_File 属性が設定されていると、RUN_ServerFile がデフォルト以外のインターフェースファイルの場所に変更されていると想定されます。</p>
DelayAfterOnline	<p>Online エントリポイントが完了してから次の監視サイクルが呼び出されるまでに経過する秒数を指定します。</p> <p>データ形式と値の種類: 整数 - スカラー</p> <p>デフォルト値: 10</p>
DelayAfterOffline	<p>Offline エントリポイントが完了してから次の監視サイクルが呼び出されるまでに経過する秒数を指定します。</p> <p>データ形式と値の種類: 整数 - スカラー</p> <p>デフォルト値: 2</p>
ShutdownWaitLimit	<p><code>shutdown with wait</code> コマンドの発行後に、必要な場合に <code>kill -15 <data server-pid></code> コマンドの発行を試みるまでに Sybase インスタンスが停止するのをエージェントが待機する最大秒数。</p> <p>データ形式と値の種類: 整数 - スカラー</p> <p>デフォルト値: 60</p>

オプション属性	定義
Quorum_dev	<p>クォーラムデバイスは、クラスタメンバーシップを管理し、クラスタ設定データを保存し、サーバーインスタンスとノード間で共有される情報を格納します。クォーラムデバイスは、クラスタのすべてのノードがアクセス可能なディスクである必要があります。完全修飾のクォーラムデバイス名を指定してください。</p> <p>注意: Sybase Cluster Edition の Quorum_dev 属性を設定しない場合、Sybase エージェントは qrmutil ベースの検査を実行しません。この設定のエラーは望ましくない結果を引き起こす可能性があります。たとえば、qrmutil がエラーによる停止状態を返した場合、エージェントはシステムをパニック状態にしません。このとき、Quorum_dev 属性が設定されていないため、Sybase のエージェントは qrmutil ベースの検査を実行しません。</p> <p>データ形式と値の種類: 文字列 - スカラー</p> <p>デフォルト値: デフォルト値なし</p>
Run_ServerFile	<p>Sybase インスタンスの RUN_SERVER ファイルの場所を指定します。この属性の値を指定しない場合、ファイルのデフォルトの場所が使われます。</p> <p>データ形式と値の種類: 文字列 - スカラー</p> <p>デフォルト値: デフォルト値なし</p>

表 A-3 は、Sybase 用エージェントの内部属性を一覧します。

この属性は内部使用のみを目的としています。この属性の値は修正しないことを推奨します。

表 A-3 内部属性

内部属性	定義
AgentDirectory	<p>Sybase 用のエージェントに関連するバイナリ、スクリプト、その他のファイルの場所を指定します。</p> <p>データ形式と値の種類: 静的 - 文字列</p>

SybaseBk リソースタイプ

SybaseBk エージェントのタイプ定義

Sybase 用のエージェントのリソースタイプの定義は次のとおりです。

```

type SybaseBk (
    str Backupserver
    str Owner
    str Home
    str Version
    str Server
    str SA
    str SApwd
    str Run_ServerFile
    str interfaces_File
    static str ArgList[] = { Backupserver, Owner, Home, Version,
    Server, SA, SApwd, Run_ServerFile, interfaces_File }
    static int IMF{} = { Mode=3, MonitorFreq=5, RegisterRetryLimit=3 }
    static str IMFRegList[] = { Backupserver, Owner }
    static str AgentDirectory = "/opt/VRTSagents/ha/bin/SybaseBk"
)

```

SybaseBk 用エージェントの属性の定義

SybaseBk 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と内部に分類されます。

表 A-4 に、SybaseBk リソースの必須属性を一覧で示します。

表 A-4 必須属性

必須属性	定義
Server	<p>\$DSQUERY バックアップ名。1つの Sybase サービスグループには、1つのサーバーのみを設定するようにしてください。個別のサービスグループで Sybase リソースを設定する利点は、各 Sybase データサーバーを独立してフェールオーバーできることにあります。1つのサービスグループに複数の Sybase リソースが設定されていると、「n」個のデータサーバーのうちの1つでエラーが発生した場合でも、サービスグループ全体がフェールオーバーされます。</p> <p>データ形式と値の種類: 文字列 - スカラー</p>
Owner	<p>パスワード入力用の /etc/nsswitch.conf ファイルで指定された、いずれかのソース (NIS+、/etc/hosts など) の実行ファイルおよびデータベースファイルの定義済み所有者としての Sybase ユーザー。Sybase の実行ファイルとデータベースファイルは、このユーザーのコンテキストでアクセスされます。</p> <p>データ形式と値の種類: 文字列 - スカラー</p>
Home	<p>Sybase バイナリおよび設定ファイルへのパス \$SYBASE。</p> <p>データ形式と値の種類: 文字列 - スカラー</p>

必須属性	定義
Version	Sybase バックアップサーバーのバージョン。 データ形式と値の種類: 文字列 - スカラー
Backupserver	\$BACKUP SYBASE バックアップサーバー名。 データ形式と値の種類: 文字列 - スカラー
SA	Sybase データベース管理者。この属性はシャットダウンのために ASE に接続するために必要です。 データ形式と値の種類: 文字列 - スカラー
SAPswd	Sybase データベース管理者の暗号化されたパスワード。このパスワードはシャットダウンのために ASE に接続するために必要です。 データ形式と値の種類: 文字列 - スカラー p.42 の「 Sybase のパスワードの暗号化 」を参照してください。 メモ: SA ユーザーがパスワードを必要としない場合は、この属性の値を指定する必要はありません。

表 A-5 に、SybaseBk エージェントのオプション属性の一覧を示します。

表 A-5 オプション属性

オプション属性	定義
Run_ServerFile	Sybase インスタンス用の RUN_SERVER ファイルの場所を指定します。この属性の値を指定しない場合、このファイルのデフォルトの場所が使われます。 データ形式と値の種類: 文字列 - スカラー デフォルト値: デフォルト値なし
interfaces_File	Sybase インスタンス用のインターフェースファイルの場所を指定します。この属性が設定されると、isql セッションに接続するときに [-I interfaces file] オプションが使われます。この属性が設定されていない場合は、エージェントは -I オプションをしません。 データ形式と値の種類: 文字列 - スカラー デフォルト値: デフォルト値なし 例: /sybase/my_interfaces_file メモ: interfaces_File 属性が設定されていると、Run_ServerFile がデフォルト以外のインターフェースファイルの場所に変更されていると想定されます。

オプション属性	定義
IMF	<p>このリソースタイプレベルの属性は、Sybase エージェントがインテリジェントなリソース監視を実行する必要があるかどうかを決定します。リソースレベルでこの属性の値を上書きすることもできます。</p> <p>この属性は次のキーを含んでいます。</p> <ul style="list-style-type: none"> ■ 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。 <p>有効な値は次の通りです:</p> <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 <p>デフォルト: 3</p> <ul style="list-style-type: none"> ■ 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す頻度を指定します。このキーの値は整数です。 <p>デフォルト: 5</p> <p>エージェントがポーリングベースの監視とインテリジェントなリソースの監視の両方を実行する必要がある場合は、このキーをゼロ以外の値に設定できます。値が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。</p> <p>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。</p> <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 <ul style="list-style-type: none"> ■ 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 oracle_imf_register 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。 <p>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。</p> <p>デフォルト: 3</p> <p>データ形式と値の種類: 整数 - 関連</p>

表 A-6 は、SysbaseBk 用エージェントの内部属性を一覧します。

この属性は内部使用のみを目的としています。この属性の値は修正しないことを推奨します。

表 A-6 内部属性

内部属性	定義
AgentDirectory	SybaseBk のエージェントに関連するバイナリ、スクリプト、その他のファイルの場所を指定します。 データ形式と値の種類: 静的 - 文字列

Sybase の設定例

この付録では以下の項目について説明しています。

- Sybase 用エージェントの設定例について
- Sybase 用エージェントのリソースの依存関係グラフ
- Linux システムの Sybase サービスグループの設定例
- Sybase ASE CE の設定のための main.cf ファイルのサンプル

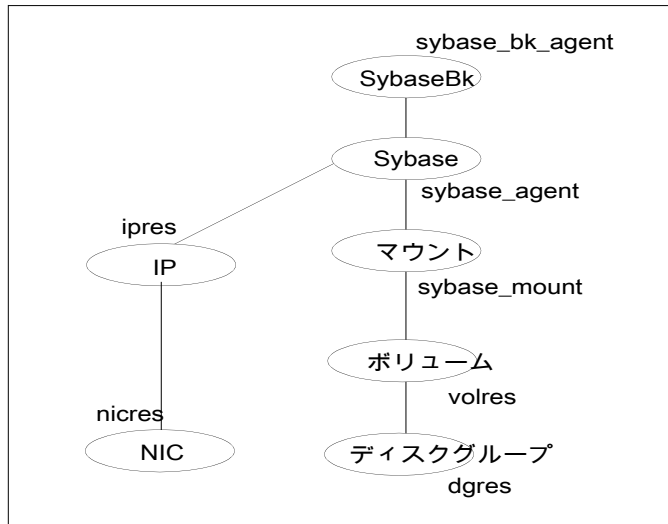
Sybase 用エージェントの設定例について

Sybase エージェントのインストール後、サンプルの main.cf ファイルのいくつかの種類を /etc/VRTSagents/ha/conf/Sybase で参照できます。この付録には、VCS クラスタにおける Sybase サーバーの状態を監視するよう設定された、典型的なサービスグループが収録されています。設定例では、リソースタイプ、リソースおよびサービスグループ間のリソース依存関係を図示します。エージェントを設定する前に、これらの依存関係を十分確認してください。これらのリソースタイプについて詳しくは『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。

Sybase 用エージェントのリソースの依存関係グラフ

図 B-1 は、Linux の設定の依存関係グラフを示しています。

図 B-1 依存関係グラフ



依存関係グラフは、単一の Sybase インスタンスの設定を示しています。設定には、単一のボリュームを持つディスクグループが含まれます。ボリュームは Volume リソースを使って監視され、Mount リソースを使ってマウントされます。Mount リソースには Volume リソースが必要で、Volume リソースには DiskGroup リソースが必要です。Sybase サーバーのサービスグループ IP アドレスは、IP および NIC リソースタイプを使って監視されています。これらの各リソースがオンライン状態にされた後で、Sybase サーバーを起動できます。バックアップサーバーは Sybase SQL Server がオンラインになった後に起動されます。

メモ: 設定に Veritas Volume Manager を使っていない場合、共有ストレージの設定には DiskGroup および Volume リソースタイプではなく、LVMVolumeGroup リソースタイプ、LVMLogicalVolume リソースタイプ、DiskReservation リソースタイプを使ってください。

Linux システムの Sybase サービスグループの設定例

次の設定例は、1 枚の NIC を使って設定された Sybase サービスグループ、詳細監視を使って設定された 1 つのデータベースインスタンス、1 つのバックアップインスタンスを備えた基本的な設定を示しています。

```
include "types.cf"  
include "/etc/VRTSagents/ha/conf/Sybase/SybaseTypes.cf"
```

```

cluster vcs_cluster (
    UserNames = { admin = anoGniNkoJooMwoInl }
    CounterInterval = 5
)

system system1 (
)

system system2 (
)

group Sybase_group (
    SystemList = { system1 = 0, system2 = 1 }
)

IP ipres (
    Device = eth0
    Address = "16.9.1.9"
    NetMask = "255.255.0.0"
)

Mount sybase_mount (
    MountPoint = "/sybase"
    BlockDevice = "/dev/vx/dsk/vxdgG01/vxvolG01"
    FSType = vxfs
    MountOpt = rw
    FsckOpt = "-y"
)

Volume sybase_volres (
    Volume="vxvolG01"
    DiskGroup="vxdgG01"
)

DiskGroup sybase_dres (
    DiskGroup = "vxdgG01"
    StartVolumes=0
    StopVolumes=0
)

NIC nicres (
    Device = eth0
    NetworkHosts = { "16.9.10.1", "16.9.10.2" }
)

```

```
)

Sybase sybase_agent (
    Server = SYB15LNX32
    Owner = sybase
    Home = "/home/sybase"
    Version = "15.0.0"
    SA = sa
    LevelTwoMonitorFreq = 1
    User = vcsuser
    UPword = GSNsLSoSLsPSt
    Db = vcsdb
    Table = vcstable
    Monscript = "/opt/VRTSagents/ha/bin/Sybase/SqlTest.pl"
    WaitForRecovery = 1
    Run_ServerFile = "/home/sybase/RUN_SYB15LNX32"
    interfaces_File = "/newfs/interfaces"
)

SybaseBk sybase_bk_agent (
    Server = SYB15LNX32
    Owner = sybase
    Home = "/home/sybase"
    Version = "15.0.0"
    Backupserver = SYB15LNX32_BS
    SA = sa
    Run_ServerFile = "/home/sybase/RUN_SYB15LNX32_BS"
    interfaces_File = "/newfs/interfaces"
)

ipres requires nicres
sybase_agent requires ipres
sybase_agent requires sybase_mount
sybase_mount requires sybase_volres
sybase_volres requires sybase_dres
sybase_bk_agent requires sybase_agent
```

Sybase ASE CE の設定のための main.cf ファイルのサンプル

次の Sybase ASE CE の設定のための main.cf ファイルのサンプルが提供されます。

- 基本的なクラスタ設定
 - Sybase バイナリをインストールするための CFS の共有マウントポイントがある場合
 - Sybase バイナリをインストールするための VxFS のローカルマウントポイントがある場合
- 2 つのクラスタ間のデータのレプリケーション用の設定
 - CVM VVR 設定のプライマリサイト
 - CVM VVR 設定のセカンダリサイト

VCS の制御下にあり、Sybase バイナリをインストールするための CFS の共有マウントポイントを持つ基本的な Sybase ASE CE クラスタ設定向けの main.cf のサンプル

この main.cf のサンプルは、Sybase バイナリをインストールするための CFS の共有マウントポイントを持った、基本的なクラスタ設定の単一のサイト向けです。

この Sybase ASE CE の設定のサンプル main.cf の設定の詳細を次に示します。

- ファイル名: sybasece_main.cf
- ファイルの場所: /etc/VRTSagents/ha/conf/Sybase/

```
include "types.cf"
include "CFSTypes.cf"
include "CVMTypes.cf"
include "SybaseTypes.cf"

cluster cluster101 (
    UserNames = { admin = HopHojOlpKppNxpJom }
    Administrators = { admin }
    HacliUserLevel = COMMANDROOT
    UseFence=SCSI3
)

system system1 (
)

system system2 (
)

// binmounts group for configuring CFS mounts for Sybase binaries.

group binmnt (
```

```
SystemList = { system1 = 0, system2 = 1 }
Parallel = 1
AutoStartList = { system1, system2 }
OnlineRetryLimit = 3
OnlineRetryInterval = 120
)

CFSMount sybbindg_101_sybbinvol_mnt (
    MountPoint = "/sybase"
    BlockDevice = "/dev/vx/dsk/sybbindg_101/sybbin_vol"
)

CVMVolDg sybbindg_101_voldg (
    CVMDiskGroup = sybbindg_101
    CVMVolume = { sybbin_vol }
    CVMActivation = sw
)

requires group cvm online local firm
sybbindg_101_sybbinvol_mnt requires sybbindg_101_voldg

// resource dependency tree
//
// group binmnt
// {
//   CFSMount sybbindg_101_sybbinvol_mnt
//     {
//       CVMVolDg sybbindg_101_voldg
//     }
// }

// cvm group for CVM and CFS specific agents.

group cvm (
    SystemList = { system1 = 0, system2 = 1 }
    AutoFailOver = 0
    Parallel = 1
    AutoStartList = { system1, system2 }
)

CFSfsckd vxfsckd (
)
```



```

CVMCluster cvm_clus (
    CVMClustName = sfsyb_90
    CVMNodeId = { system1 = 0, system2 = 1 }
    CVMTransport = gab
    CVMTimeout = 200
)

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

cvm_clus requires cvm_vxconfigd
vxfsckd requires cvm_clus

// resource dependency tree
//
// group cvm
// {
//     CFSfsckd vxfsckd
//     {
//         CVMCluster cvm_clus
//         {
//             CVMVxconfigd cvm_vxconfigd
//         }
//     }
// }

// sybasece group for:
// 1. CVM volumes for Sybase database and quorum device
// 2. CFS mount for Sybase database and quorum device
// 3. Process agent for vxfsck process.
// 4. Sybase database instance.

group sybasece (
    SystemList = { system1 = 0, system2 = 1 }
    Parallel = 1
    AutoStartList = { system1, system2 }
    OnlineRetryLimit = 3
    OnlineRetryInterval = 120
)

```

```
CFSMount quorum_101_quorumvol_mnt (  
    MountPoint = "/quorum"  
    BlockDevice = "/dev/vx/dsk/quorum_101/quorumvol"  
)  
  
CFSMount sybdata_101_sybvol_mnt (  
    MountPoint = "/sybdata"  
    BlockDevice = "/dev/vx/dsk/sybdata_101/sybvol"  
)  
  
CVMVolDg quorum_101_voldg (  
    CVMDiskGroup = quorum_101  
    CVMVolume = { quorumvol }  
    CVMActivation = sw  
)  
  
CVMVolDg sybdata_101_voldg (  
    CVMDiskGroup = sybdata_101  
    CVMVolume = { sybvol }  
    CVMActivation = sw  
)  
  
Process vxfsend (  
    PathName = "/sbin/vxfsend"  
    Arguments = "-m sybase -k /tmp/vcmp_socket"  
)  
  
Sybase ase (  
    Server @system1 = ase1  
    Server @system2 = ase2  
    Owner = sybase  
    Home = "/sybase"  
    Version = 15  
    SA = sa  
    Quorum_dev = "/quorum/q.dat"  
)  
  
requires group binmnt online local firm  
ase requires quorum_101_quorumvol_mnt  
ase requires sybdata_101_sybvol_mnt  
ase requires vxfsend  
quorum_101_quorumvol_mnt requires quorum_101_voldg
```

```
sybdata_101_sybvol_mnt requires sybdata_101_voldg

// resource dependency tree
//
// group sybasece
// {
//   Sybase ase
//   {
//     CFSMount quorum_101_quorumvol_mnt
//     {
//       CVMVolDg quorum_101_voldg
//     }
//     CFSMount sybdata_101_sybvol_mnt
//     {
//       CVMVolDg sybdata_101_voldg
//     }
//     Process vxfsend
//   }
// }
```

Sybase バイナリをインストールするための VxFS のローカルマウントポイントを持つ基本的な Sybase ASE CE クラスタ設定向けの main.cf のサンプル

この **main.cf** のサンプルは、Sybase バイナリをインストールするための VxFS のローカルマウントポイントを持った、基本的なクラスタ設定の単一のサイト向けです。

この Sybase ASE CE の設定のサンプル **main.cf** の設定の詳細を次に示します。

- ファイル名: **sybasece_main.cf**
- ファイルの場所: **/etc/VRTSagents/ha/conf/Sybase/**

```
include "types.cf"
include "CFSTypes.cf"
include "CVMTypes.cf"
include "SybaseTypes.cf"

cluster cluster101 (
    UserNames = { admin = HopHojOlpKppNxpJom }
    Administrators = { admin }
    HacliUserLevel = COMMANDROOT
```

```
        UseFence=SCSI3
    )

system system1 (
)

system system2 (
)

// binmounts group for configuring VxFS mounts for Sybase binaries.

group binlocalmnt (
    SystemList = { system1 = 0, system2 = 1 }
    AutoFailOver = 0
    Parallel = 1
    AutoStartList = { system1, system2 }
    OnlineRetryLimit = 3
    OnlineRetryInterval = 120
)

DiskGroup sybbindg_101_voldg (
    DiskGroup = sybbindg
)

Mount sybbindg_101_sybbinvol_mnt (
    MountPoint = "/sybase"
    BlockDevice = "/dev/vx/dsk/sybbindg_101/sybbin_vol"
    FSType = vxfs
    FsckOpt = "-y"
)

Volume sybbindg_101_vol (
    DiskGroup = sybbindg
    Volume = sybbinvol
)

requires group cvm online local firm
sybbindg_101_sybbinvol_mnt requires sybbindg_101_vol
sybbindg_101_vol requires sybbindg_101_voldgdg

// resource dependency tree
//
```

```
// group binlocalmnt
// {
// Mount sybbindg_101_sybbinvol_mnt
// {
//     Volume sybbindg_vol
//     {
//         DiskGroup sybbindg_101_voldg
//     }
// }
// }

// cvm group for CVM and CFS specific agents.

group cvm (
    SystemList = { system1 = 0, system2 = 1 }
    AutoFailOver = 0
    Parallel = 1
    AutoStartList = { system1, system2 }
)

CFSfsckd vxfsckd (
)

CVMCluster cvm_clus (
    CVMClustName = cluster101
    CVMNodeId = { system1 = 0, system2 = 1 }
    CVMTransport = gab
    CVMTimeout = 200
)

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

cvm_clus requires cvm_vxconfigd
vxfsckd requires cvm_clus

// resource dependency tree
//
// group cvm
// {
```

```
// CFSfsckd vxfsckd
//   {
//     CVMCluster cvm_clus
//       {
//         CVMVxconfigd cvm_vxconfigd
//       }
//     }
//   }

// sybasece group for:
// 1. CVM volumes for Sybase database and quorum device
// 2. CFS mount for Sybase database and quorum device
// 3. Process agent for vxfsend process.
// 4. Sybase database instance.

group sybasece (
  SystemList = { system1 = 0, system2 = 1 }
  Parallel = 1
  AutoStartList = { system1, system2 }
  OnlineRetryLimit = 3
  OnlineRetryInterval = 120
)

CFSMount quorum_101_quorumvol_mnt (
  MountPoint = "/quorum"
  BlockDevice = "/dev/vx/dsk/quorum_101/quorumvol"
)

CFSMount sybdata_101_sybvol_mnt (
  MountPoint = "/sybdata"
  BlockDevice = "/dev/vx/dsk/sybdata_101/sybvol"
)

CVMVolDg quorum_101_voldg (
  CVMDiskGroup = quorum_101
  CVMVolume = { quorumvol }
  CVMActivation = sw
)

CVMVolDg sybdata_101_voldg (
  CVMDiskGroup = sybdata_101
  CVMVolume = { sybvol }
  CVMActivation = sw
)
```

```

    )

Process vxfsend (
    PathName = "/sbin/vxfsend"
    Arguments = "-m sybase -k /tmp/vcmp_socket"
)

Sybase ase (
    Server @system1 = ase1
    Server @system2 = ase2
    Owner = sybase
    Home = "/sybase"
    Version = 15
    SA = sa
    Quorum_dev = "/quorum/q.dat"
)

requires group binlocalmnt online local firm
ase requires quorum_101_quorumvol_mnt
ase requires sybdata_101_sybvol_mnt
ase requires vxfsend
quorum_101_quorumvol_mnt requires quorum_101_voldg
sybdata_101_sybvol_mnt requires sybdata_101_voldg

// resource dependency tree
//
// group sybasece
// {
// Sybase ase
// {
//     CFMount quorum_101_quorumvol_mnt
//     {
//         CVMVolDg quorum_101_voldg
//     }
//     CFMount sybdata_101_sybvol_mnt
//     {
//         CVMVolDg sybdata_101_voldg
//     }
//     Process vxfsend
// }
// }

```

CVM VVR のプライマリサイトの main.cf のサンプル

このサンプル `main.cf` は CVM VVR 設定のプライマリサイト向けです。これは、2 つのクラスタ間でのデータのレプリケート向けの 2 つの `main.cf` サンプルのうちの 1 つです。

この Sybase ASE CE の設定のサンプル `main.cf` の設定の詳細を次に示します。

- ファイル名: `sybasece_cvmvvr_primary_main.cf`
- ファイルの場所: `/etc/VRTSagents/ha/conf/Sybase`

```
include "types.cf"
include "CFSTypes.cf"
include "CVMTypes.cf"
include "VVRTypes.cf"
include "SybaseTypes.cf"

cluster cluster101 (
    UserNames = { admin = HopHoj0lpKppNxpJom }
    ClusterAddress = "10.180.88.188"
    Administrators = { admin }
    HacliUserLevel = COMMANDROOT
    UseFence=SCSI3
)

remoteclass syb_cluster102 (
    ClusterAddress = "10.190.99.199"
)

heartbeat Icmp (
    ClusterList = { syb_cluster102 }
    Arguments @syb_cluster102 = { "10.190.99.199" }

)

system system1 (
)

system system2 (
)

group ClusterService (
```



```

SystemList = { system1 = 0, system2 = 1 }
AutoStartList = { system1, system2 }
OnlineRetryLimit = 3
OnlineRetryInterval = 120
)

Application wac (
    StartProgram = "/opt/VRTSvcs/bin/wacstart"
    StopProgram = "/opt/VRTSvcs/bin/wacstop"
    MonitorProcesses = { "/opt/VRTSvcs/bin/wac" }
    RestartLimit = 3
)

IP gcoip (
    Device = eth0
    Address = "10.180.88.188"
    NetMask = "255.255.255.0"
)

NIC csgnic (
    Device = eth0
)

gcoip requires csgnic
wac requires gcoip

// resource dependency tree
//
//     group ClusterService
//     {
//     Application wac
//     {
//     IP gcoip
//     {
//     NIC csgnic
//     }
//     }
//     }
//     }

group RVGgroup (
    SystemList = { system1 = 0, system2 = 1 }

```

```
Parallel = 1
AutoStartList = { system1, system2 }
)

CVMVolDg sybdata_voldg (
    CVMDiskGroup = sybdata_101
    CVMActivation = sw
)

RVGShared sybdata_rvg (
    RVG = syb_rvg
    DiskGroup = sybdata_101
)

requires group binmnt online local firm
sybdata_rvg requires sybdata_voldg

group binmnt (
    SystemList = { system1 = 0, system2 = 1 }
    Parallel = 1
    AutoStartList = { system1, system2 }
    OnlineRetryLimit = 3
    OnlineRetryInterval = 120
)

CFSMount sybbindg_101_sybbinvol_mnt (
    MountPoint = "/sybase"
    BlockDevice = "/dev/vx/dsk/sybbindg_101/sybbin_vol"
)

CVMVolDg sybbindg_101_voldg (
    CVMDiskGroup = sybbindg_101
    CVMVolume = { sybbin_vol }
    CVMActivation = sw
)

requires group cvm online local firm
sybbindg_101_sybbinvol_mnt requires sybbindg_101_voldg

group cvm (
    SystemList = { system1 = 0, system2 = 1 }
    AutoFailOver = 0
    Parallel = 1
```

```
AutoStartList = { system1, system2 }
)

CFSfsckd vxfsckd (
)

CVMCluster cvm_clus (
    CVMClustName = syb_cluster101
    CVMNodeId = { system1 = 0, system2 = 1 }
    CVMTransport = gab
    CVMTimeout = 200
)

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

cvm_clus requires cvm_vxconfigd
vxfsckd requires cvm_clus

// resource dependency tree
//
//     group cvm
//     {
//     CFSfsckd vxfsckd
//         {
//             CVMCluster cvm_clus
//                 {
//                     CVMVxconfigd cvm_vxconfigd
//                 }
//             }
//         }
//     }

group logowner (
    SystemList = { system1 = 0, system2 = 1 }
    AutoStartList = { system1, system2 }
)

IP logowner_ip (
    Device = eth0
```

```
        Address = "10.10.9.101"
        NetMask = "255.255.255.0"
    )

NIC nic (
    Device = eth0
)

RVGLogowner rvg_logowner (
    RVG = syb_rvg
    DiskGroup = sybdata_101
)

requires group RVGgroup online local firm
logowner requires logowner_ip
logowner_ip requires nic

// resource dependency tree
//
//     group logowner
//     {
//         RVGLogowner rvg_logowner
//         {
//             IP logowner_ip
//             {
//                 NIC nic
//             }
//         }
//     }

group sybasece (
    SystemList = { system1 = 0, system2 = 1 }
    Parallel = 1
    ClusterList = { syb_cluster101 = 0, syb_cluster102 = 1 }
    AutoStartList = { system1, system2 }
    ClusterFailOverPolicy = Manual
    Authority = 1
    OnlineRetryLimit = 3
    TriggerResStateChange = 1
    OnlineRetryInterval = 120
)
```

```
CFSMount quorum_101_quorumvol_mnt (
    MountPoint = "/quorum"
    BlockDevice = "/dev/vx/dsk/quorum_101/quorumvol"
)
```

```
CFSMount sybdata_101_sybvol_mnt (
    MountPoint = "/sybdata"
    BlockDevice = "/dev/vx/dsk/sybdata_101/sybvol"
)
```

```
CVMVolDg quorum_101_voldg (
    CVMDiskGroup = quorum_101
    CVMVolume = { quorumvol }
    CVMActivation = sw
)
```

```
Process vxfend (
    PathName = "/sbin/vxfend"
    Arguments = "-m sybase -k /tmp/vcmp_socket"
)
```

```
RVGSharedPri syb_vvr_shpri (
    RvgResourceName = sybdata_rvg
    OnlineRetryLimit = 0
)
```

```
Sybase ase (
    Server @system1 = ase1
    Server @system2 = ase2
    Owner = sybase
    Home = "/sybase"
    Version = 15
    SA = sa
    Quorum_dev = "/quorum/q.dat"
)
```

```
requires group RVGgroup online local firm
sybdata_101_sybvol_mnt requires syb_vvr_shpri
ase requires vxfend
ase requires sybdata_101_sybvol_mnt
ase requires quorum_101_quorumvol_mnt
quorum_101_quorumvol_mnt requires quorum_101_voldg
```

```
// resource dependency tree
//
//     group sybasece
//     {
//     Sybase ase
//     {
//         CFSMount sybdata_101_sybvol_mnt
//         {
//             RVGSharedPri syb_vvr_shpri
//         }
//         Process vxfend
//         CFSMount quorum_101_quorumvol_mnt
//         {
//             CVMVolDg quorum_101_voldg
//         }
//     }
//     }
```

CVM VVR のセカンダリサイトの main.cf のサンプル

このサンプル **main.cf** は CVM VVR 設定のセカンダリサイト向けです。これは、2 つのクラスタ間でのデータのレプリケート向けの 2 つの **main.cf** サンプルのうち 2 つ目です。

この Sybase ASE CE の設定のサンプル **main.cf** の設定の詳細を次に示します。

- ファイル名: **sybasece_cvmvvr_secondary_main.cf**
- ファイルの場所: **/etc/VRTSagents/ha/conf/Sybase**

This is main.cf for CVM VVR configuration on Secondary site.

```
-----
include "types.cf"
include "CFSTypes.cf"
include "CVMTypes.cf"
include "VVRTypes.cf"
include "SybaseTypes.cf"

cluster cluster102 (
    UserNames = { admin = HopHoj0lpKppNxpJom }
    ClusterAddress = "10.190.99.199"
    Administrators = { admin }
    HacliUserLevel = COMMANDROOT
    UseFence=SCSI3
)
```

```

remotecluster syb_cluster101 (
    ClusterAddress = "10.180.88.188"
)

heartbeat Icmp (
    ClusterList = { syb_cluster101 }
    Arguments @syb_cluster101 = { "10.180.88.188" }
)

system system3 (
)

system system4 (
)

group ClusterService (
    SystemList = { system3 = 0, system4 = 1 }
    AutoStartList = { system3, system4 }
    OnlineRetryLimit = 3
    OnlineRetryInterval = 120
)

Application wac (
    StartProgram = "/opt/VRTSvcs/bin/wacstart"
    StopProgram = "/opt/VRTSvcs/bin/wacstop"
    MonitorProcesses = { "/opt/VRTSvcs/bin/wac" }
    RestartLimit = 3
)

IP gcoip (
    Device = eth0
    Address = "10.190.99.199"
    NetMask = "255.255.255.0"
)

NIC csgnic (
    Device = eth0
)

gcoip requires csgnic
wac requires gcoip

```

```
// resource dependency tree
//
// group ClusterService
// {
// Application wac
//   {
//     IP gcoip
//     {
//       NIC csgnic
//     }
//   }
// }

group RVGgroup (
    SystemList = { system3 = 0, system4 = 1 }
    Parallel = 1
    AutoStartList = { system3, system4 }
)

CVMVolDg sybdata_voldg (
    CVMDiskGroup = sybdata_101
    CVMActivation = sw
)

RVGShared sybdata_rvg (
    RVG = syb_rvg
    DiskGroup = sybdata_101
)

requires group binmnt online local firm
sybdata_rvg requires sybdata_voldg

group binmnt (
    SystemList = { system3 = 0, system4 = 1 }
    Parallel = 1
    AutoStartList = { system3, system4 }
    OnlineRetryLimit = 3
    OnlineRetryInterval = 120
)

CFSMount sybbindg_101_sybbinvol_mnt (
```



```

        MountPoint = "/sybase"
        BlockDevice = "/dev/vx/dsk/sybbindg_101/sybbin_vol"
    )
    CVMVolDg sybbindg_101_voldg (
        CVMDiskGroup = sybbindg_101
        CVMVolume = { sybbin_vol }
        CVMActivation = sw
    )
    requires group cvm online local firm
    sybbindg_101_sybbinvol_mnt requires sybbindg_101_voldg

group cvm (
    SystemList = { system3 = 0, system4 = 1 }
    AutoFailOver = 0
    Parallel = 1
    AutoStartList = { system3, system4 }
)

CFSfsckd vxfsckd (
)

CVMCluster cvm_clus (
    CVMClustName = syb_cluster102
    CVMNodeId = { system3 = 0, system4 = 1 }
    CVMTransport = gab
    CVMTimeout = 200
)

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

cvm_clus requires cvm_vxconfigd
vxfsckd requires cvm_clus

// resource dependency tree
//
//     group cvm
//     {
//     CFSfsckd vxfsckd
//         {

```

```
//          CVMcluster cvm_clus
//          {
//          CVMVxconfigd cvm_vxconfigd
//          }
//          }

group logowner (
    SystemList = { system3 = 0, system4 = 1 }
    AutoStartList = { system3, system4 }
)

IP logowner_ip (
    Device = eth0
    Address = "10.11.9.102"
    NetMask = "255.255.255.0"
)

NIC nic (
    Device = eth0
)

RVGLogowner rvg_logowner (
    RVG = syb_rvg
    DiskGroup = sybdata_101
)

requires group RVGgroup online local firm
logowner requires logowner_ip
logowner_ip requires nic

// resource dependency tree
//
// group logowner
// {
//   RVGLogowner rvg_logowner
//   {
//     IP logowner_ip
//     {
//       NIC nic
//     }
//   }
// }
```

```
// }

group sybasece (
    SystemList = { system3 = 0, system4 = 1 }
    Parallel = 1
    ClusterList = { syb_cluster102 = 0, syb_cluster101 = 1 }
    AutoStartList = { system3, system4 }
    OnlineRetryLimit = 3
    OnlineRetryInterval = 120
)

CFSMount quorum_101_quorumvol_mnt (
    MountPoint = "/quorum"
    BlockDevice = "/dev/vx/dsk/quorum_101/quorumvol"
)

CVMVolDg quorum_101_voldg (
    CVMDiskGroup = quorum_101
    CVMVolume = { quorumvol }
    CVMActivation = sw
)

CFSMount sybdata_101_sybvol_mnt (
    MountPoint = "/sybdata"
    BlockDevice = "/dev/vx/dsk/sybdata_101/sybvol"
)

Process vxfsend (
    PathName = "/sbin/vxfsend"
    Arguments = "-m sybase -k /tmp/vcmp_socket"
)

RVGSharedPri syb_vvr_shpri (
    RvgResourceName = sybdata_rvg
    OnlineRetryLimit = 0
)

Sybase ase (
    Server @system3 = ase1
    Server @system4 = ase2
    Owner = sybase
    Home = "/sybase"
    Version = 15
)
```

```
SA = sa
Quorum_dev = "/quorum/q.dat"
)

requires group RVGgroup online local firm
sybdata_101_sybvol_mnt requires syb_vvr_shpri
ase requires vxfsend
ase requires sybdata_101_sybvol_mnt
ase requires quorum_101_quorumvol_mnt
quorum_101_quorumvol_mnt requires quorum_101_voldg
```

以前のリリースで導入された変更

この付録では以下の項目について説明しています。

- [以前のリリースで導入された変更点](#)

以前のリリースで導入された変更点

前のリリースの Veritas Cluster Server Agent for Sybase で拡張された機能は以下のとおりです。

- Sybase の VCS エージェントバイナリが VRTSvcsea RPM に含まれるようになりました。この RPM には、DB2 用と Oracle 用の VCS エージェントバイナリも含まれていません。
- インストールプログラムを使って VCS エージェントバイナリをインストールした場合は、プログラムによって、適切なエージェントの `types.cf` ファイルが含まれるように `main.cf` ファイルが更新されます。
- Sybase 用エージェントは、IPC クリーンアップ機能をサポートします。
- エージェントは、新しい属性である `WaitForRecovery` をサポートします。この属性を有効にすると、オンライン機能中、エージェントは、リカバリが完了し、オンラインにできるすべてのデータベースがオンライン化されるまで待機します。
- エージェントは、`checkpoint_all` という新規アクションをサポートします。`isql` セッションに接続することによって、Sybase データサーバーに対して「`checkpoint_all`」を実行します。
- エージェントは、新しいオプション属性である `Run_ServerFile` をサポートします。この属性には、Sybase インスタンス用の `RUN_SERVER` ファイルの場所を指定します。この属性を指定しない場合、Sybase サーバーインスタンスの起動中、このファイルのデフォルトの場所がアクセスされます。

記号

\$SYBASE 22

B

Backup Server エージェントの属性

- AgentDirectory 64、66
- Backupserver 64
- home 63
- owner 63
- SA 64
- SApswd 64
- server 63
- version 64

Backup Server のエージェント

関数 15

C

Cluster Manager 38

H

hagrp コマンド 29

hagui コマンド 36

I

IMF 11

S

SQL サーバエージェントの属性

- AgentDirectory 62
- Db 57
- DelayAfterOffline 61
- DelayAfterOnline 61
- DetailMonitor 56
- home 55
- monscript 57
- owner 55
- Quorum_dev 56
- Run_ServerFile 58
- SA 55

- SApswd 56
- server 55
- ShutdownWaitLimit 61
- table 57
- UPword 57
- user 57
- version 55
- WaitForRecovery 58

SQL サーバのエージェント

- 機能 12
- 属性の定義 54

SQL サーバ用のエージェント

- タイプの定義 53
- リソースタイプ 53

Sybase

- インストール 23
- インストールディレクトリ 21
- インストールの前提条件 21
- 共有ディスク上のディレクトリ 22
- 言語設定 23
- 詳細監視のための設定 23
- データベース dbspace 22
- 透過的 TCP/IP フェールオーバー 22

SybaseTypes.cf ファイル

- インポート 36

Sybase エージェント

- Cluster Manager を使った設定 38
- アップグレード 31
- アンインストール 30
- インストール 28
- インストールの前提条件 27
- エージェント関数 12
- 概要 9
- 監視オプション 18
- コマンドラインを使用した設定 40
- 削除 30
- サポートされるソフトウェア 10
- 詳細監視 43
- 典型的なセットアップ 19
- 無効化 30

Sybase エージェントのアンインストール 30

Sybase エージェントのインストール

Linux 28

Sybase エージェントの削除 30

Sybase のインストール 23

Sybase の高可用性 10

あ

インテリジェントなリソース監視のサポート 11

か

監視

基本 18

詳細 18

コマンド

hagrp 29

hagui 36

さ

サービスグループ

ログの表示 51

サービスグループの設定 38

Cluster Manager (Java コンソール) 38

概要 38

コマンドライン 40

前提条件 38

サポートされるソフトウェア 10

詳細監視

無効化 43

有効化 43

設定

典型的なセットアップ 19

設定ウィザード 38

設定例

Linux 68

た

タイプ定義 53

データベース dbspace 22

透過的 TCP/IP フェールオーバー 22

は

バックアップサーバー用のエージェント

タイプの定義 62

リソースタイプ 62