VERITAS™

VERITAS Cluster Server Enterprise Agent 4.1 for Sybase

Installation and Configuration Guide

HP-UX

N16822G

June 2005

Disclaimer

The information contained in this publication is subject to change without notice. VERITAS Software Corporation makes no warranty of any kind with regard to this manual, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. VERITAS Software Corporation shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this manual.

VERITAS Legal Notice

VERITAS Software Corporation
350 Ellis Street
Mountain View, CA 94043
USA
Phone 650–527–8000 Fax 650–527–2908
www.veritas.com

Contents

Preface

This book describes how to install and configure the VERITAS Cluster Server (VCS) enterprise agent 4.1 for Sybase.

If this document is dated more than six months prior to the date you are installing your enterprise agent, contact VERITAS Technical Support to confirm you have the latest supported versions of the application and operating systems.

How This Guide is Organized

Chapter 1, "Introduction" on page 1 introduces the VCS enterprise agent for Sybase.

Chapter 2, "Installing and Configuring Sybase" on page 9 provides instructions on installing and configuring Sybase in a VCS cluster.

Chapter 3, "Installing the Sybase Agent" on page 13 provides instructions on install VCS enterprise agent for Sybase in a VCS cluster.

Chapter 4, "Configuring the Sybase Service Group" on page 17 provides instructions on configuring a Sybase service group.

Chapter 5, "Administering the Sybase Service Group" on page 27 provides instructions on administering a Sybase service group.

Appendix A, "Resource Type Definitions" on page 31 lists the resource type definitions and attribute definitions for the VCS enterprise agent for Sybase.

Appendix B, "Sample Configuration" on page 35 provides the sample configurations for the Sybase service group.

VERITAS Cluster Server Documentation

The following documents, along with the online help and the Release Notes, comprise the VCS documentation for this release:

Title	File Name
VERITAS Cluster Server Installation Guide	`vcs_install.pdf`
VERITAS Cluster Server User's Guide	`vcs_users.pdf`
VERITAS Cluster Server Bundled Agents Reference Guide	`vcs_bundled_agents.pdf`
VERITAS Cluster Server Agent Developer's Guide	`vcs_agent_dev.pdf`

See the Release Notes for a complete list of documents, including VCS enterprise agent guides.

Conventions

Convention	Usage	Example
`monospace`	Used for path names, commands, output, directory and file names, functions, and parameters.	Read tunables from the `/etc/vx/tunefstab` file. See the `ls`(1) manual page for more information.
`monospace` (**`bold`**)	Indicates user input.	# **`ls pubs`**
italic	Identifies book titles, new terms, emphasized text, and variables replaced with a name or value.	See the *User's Guide* for details. The variable *system_name* indicates the system on which to enter the command.
bold	Depicts GUI objects, such as fields, list boxes, menu selections, etc. Also depicts GUI commands.	Enter your password in the **Password** field. Press **Return.**
blue text	Indicates hypertext links.	See "Getting Help" on page ix.
#	Unix superuser prompt (all shells).	# **`cp /pubs/4.1/user_book /release_mgnt/4.1/archive`**

Getting Help

For technical assistance, visit http://support.veritas.com and select phone or email support. This site also provides access to resources such as TechNotes, product alerts, software downloads, hardware compatibility lists, and our customer email notification service. Use the Knowledge Base Search feature to access additional product information, including current and past releases of VERITAS documentation.

Diagnostic tools are also available to assist in troubleshooting problems associated with the product. These tools are available on disc or can be downloaded from the VERITAS FTP site. See the `README.VRTSspt` file in the `/support` directory for details.

Additional Resources

For license information, software updates and sales contacts, visit https://my.veritas.com/productcenter/ContactVeritas.jsp. For information on purchasing product documentation, visit http://webstore.veritas.com.

Documentation Feedback

Your feedback on product documentation is important to us. Send suggestions for improvements and reports on errors or omissions to clusteringdocs@veritas.com. Include the title and part number of the document (located in the lower left corner of the title page), and chapter and section titles of the text on which you are reporting. Our goal is to ensure customer satisfaction by providing effective, quality documentation. For assistance with topics other than documentation, visit http://support.veritas.com.

Introduction 1

VERITAS Cluster Server (VCS) enterprise agents monitor specific resources within an enterprise application, determine the status of these resources, and start or stop them according to external events. The VCS enterprise agent for Sybase provides high availability for Sybase in a VCS Cluster.

VCS enterprise agent for Sybase provides “active/passive” support for Sybase. For “active/active” support, contact Sybase for their agent.

Supported Software

- Sybase Adaptive Server Enterprise (ASE) 12.5 and 12.5.2.
- VCS 4.1 on HP-UX
- September 2004 HP-UX 11i version 2.0

Note Within the cluster, all systems must use the same operating system version and patch level.

About the VCS Enterprise Agent for Sybase

The Sybase agent brings the configured Sybase servers online, monitors them, and takes them offline.

The package contains two agents:

- SQL Server Agent: Sybase
- Backup Server Agent: SybaseBk

The agents include type declarations and agent executables, and are represented with Sybase and SybaseBk resource types, respectively. Both agents work together to make Sybase highly available in a VCS cluster.

SQL Server Agent: Sybase

The Sybase agent starts a Sybase SQL Server, monitors the server processes, and shuts down the server. Specific agent operations include:

- Online— Starts the Sybase SQL Server by using the following `sql` command:

  ```
  startserver -f $SYBASE/$SYBASE_ASE/install/RUN_$Server
  ```

- Monitor— In basic monitoring mode, the agent scans process table for the `dataserver` process. In detail monitoring mode, the agent runs the script specified in `MonScript` as an option. For more information on the monitoring options, see "Monitoring Options" on page 4.
- Offline— Stops the Sybase SQL Server by using the `isql` command in the following manner:

  The agent first executes the command `shutdown with wait`. If this fails, the offline script executes `shutdown with nowait`.

- Clean— Forcefully stops the Sybase SQL Server by using the `isql` command in the following manner:

  The agent first executes the command `shutdown with wait`. If this fails, the clean script executes `shutdown with nowait`.

  If the process does not respond to the `shutdown` command, the agent scans the process table for processes associated with the configured database and kills them.

Backup Server Agent: SybaseBk

The Backup server agent starts a Sybase Backup Server, monitors the server process, and shuts down the server. Specific agent operations include:

- Online— Starts the Sybase Backup Server by using the following `sql` command:

  ```
  startserver -f $SYBASE/$SYBASE_ASE/install/RUN_$BackupServer
  ```

- Monitor— Scans process table for the `backupserver` process.
- Offline— Stops the Sybase Backup Server by using the following `isql` command:

  The agent first executes the command `shutdown SYB_BACKUP with wait`. If this fails, the offline script executes `shutdown SYB_BACKUP with nowait`.

- Clean— Forcefully stops the Sybase Backup Server by using the `isql` command in the following manner:

  The agent first executes the command `shutdown SYB_BACKUP with wait`. If this fails, the clean script executes `shutdown SYB_BACKUP with nowait`.

  If the process does not respond to the `shutdown` command, the agent scans the process table for processes associated with the configured Sybase Backup Server and kills them.

Monitoring Options

The VCS enterprise agent for Sybase provides two levels of application monitoring: basic and detail.

In the basic monitoring mode, the Sybase agent monitors the Sybase daemon processes to verify whether they are running. In the detail monitoring mode, the agent performs a transaction on a test table in the database to ensure that Sybase is functioning properly. See "Setting Up Detail Monitoring for Sybase Agent" on page 24 for more information.

How the Agent Makes Sybase Highly Available

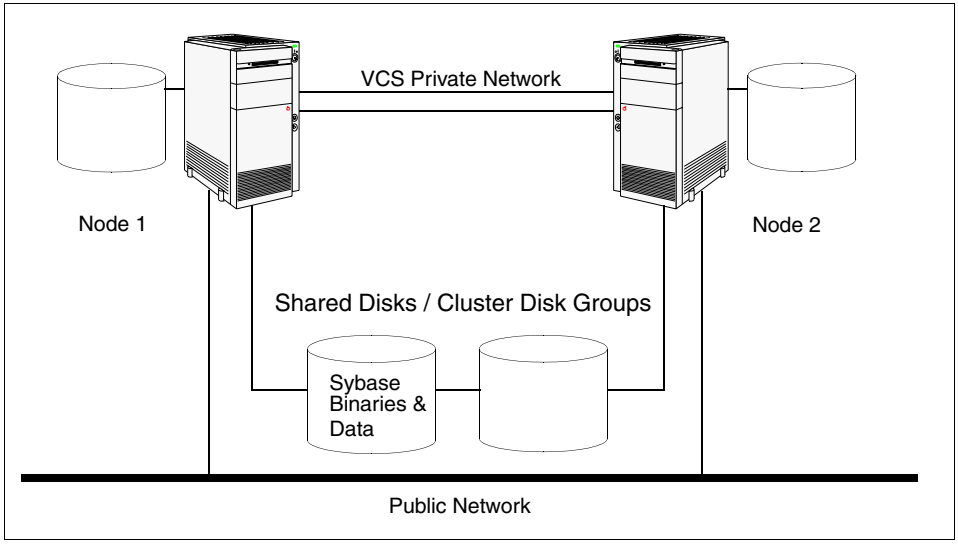
In the basic monitoring mode, the agent detects an application failure if a configured Sybase server process is not running. In the detail monitoring mode, the agent detects application failure if it cannot perform a transaction in the test table in the Sybase database server.

When the agent detects that the configured Sybase server is not running on a system, the Sybase service group is failed over to the next available system in the service group's SystemList. The configured Sybase servers are started on the new system, thus ensuring high availability for the Sybase server and data.

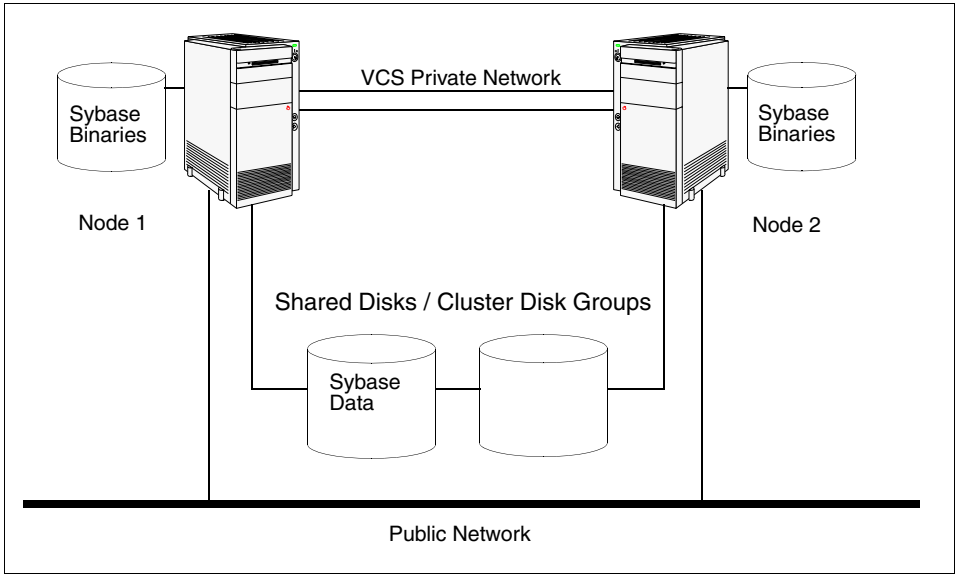
Typical Sybase Configuration in a VCS Cluster

In a typical configuration, VCS is configured in a two node cluster. The Sybase data is installed on shared disks. The Sybase server binaries can be installed locally on both nodes or on shared disks. The Sybase agent is installed on both nodes. The shared disks can be managed using VERITAS Volume Manager (VxVM).

In the configuration depicted in the illustration below, the Sybase servers, including binaries and data, are installed completely on shared disks or shared cluster disk groups managed using VxVM.

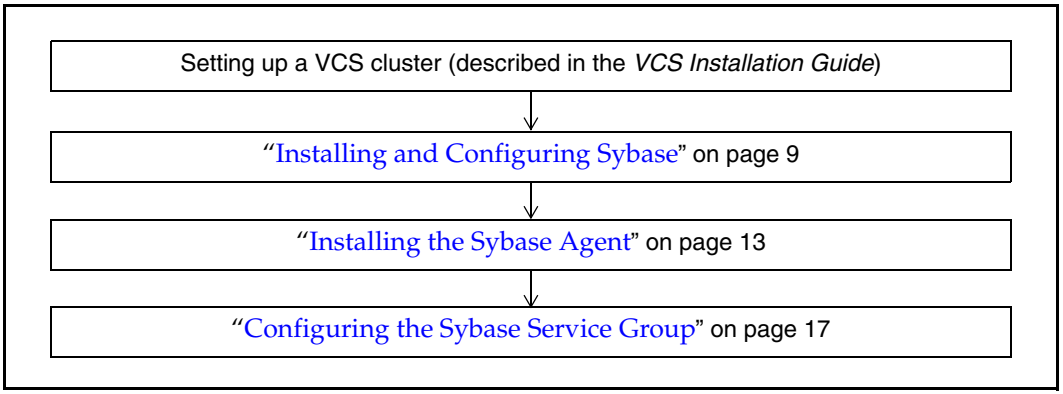


In the configuration depicted in the illustration below, the Sybase binaries are installed locally on each node in the cluster and the Sybase data is on shared disks or shared cluster disk groups managed using VxVM.



Setting Up a Sybase Cluster

Setting up a Sybase cluster in a VCS environment involves the following tasks. Each task is described in detail in subsequent chapters.



Installing and Configuring Sybase 2

This chapter lists the VCS specific requirements for installing Sybase in a VCS cluster. For information on how to install Sybase, refer to Sybase documentation. Before installing Sybase, make sure the systems in the cluster have adequate resources to run Sybase and VCS.

VCS Requirements for Installing Sybase

Make sure you meet the requirements mentioned here before installing Sybase in a VCS cluster.

Sybase Installation Directory

You can have the Sybase installation directory on a local disk or a shared storage.

- If the Sybase binaries are installed on local disk, verify the installation path is same on all the nodes in the cluster. Make sure the Sybase configuration files are identical on all the nodes in the cluster.
- If the Sybase binaries are installed on shared disks, make sure the mount points for the shared disks are same on all the nodes. The Sybase installation directory is specified by the environment variable `$SYBASE`. Create the same `$SYBASE` mount points on each system.

$SYBASE Directory on Shared Disks

All database devices, including master devices, `sybsystemprocs`, and information about Sybase user must be located on shared disks. If the database devices are created on file systems, the file systems must also be located on shared disks. Create the same file system mount points on each system to access the shared disks.

Database Dbspaces

If you are using shared disks for dbspaces, change the permissions, or access mode on the disk groups that store the Sybase data. Change the permissions for `sybase` to `660`.

For example, if you are using VERITAS Volume Manager, type:

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

Transparent TCP/IP Failover

For Sybase server failover to be transparent to Sybase clients, create an IP address as part of the Sybase service group. This IP address must match the dataserver and backup server entries in the `$SYBASE/interfaces file`. For information on the format for adding entries to the `$SYBASE/interfaces` file, refer to the Sybase documentation.

Language Settings

For the VCS enterprise agent for Sybase to function with the desired locale, make sure that the Sybase installation has the correct localization files. For example, if the Sybase server requires 'LANG=en_US.UTF-8' environment variable, verify that the localization files corresponding to language 'en_US.UTF-8' are installed with Sybase. Also, edit the file `$VCS_HOME/bin/vcsenv` to contain the following:

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

The above change would affect all VCS agents configured on the nodes.

Configuring Sybase for Detail Monitoring

This section describes the tasks to be performed for configuring a Sybase server to be monitored in detail. For more information about detail monitoring, see "Setting Up Detail Monitoring for Sybase Agent" on page 24.

Note The steps described here are specific to the sample script, `SqlTest.pl`, provided with the agent. If you will use a custom script for detail monitoring, you must configure the Sybase database accordingly.

▼ To set up Sybase for detail monitoring

Perform these steps only once in a Sybase cluster.

1. Start the Sybase server.

   ```
   # $SYBASE/ASE-12_5/install/RUN_$DSQUERY
   ```

2. Start the Sybase client on any cluster node.

   ```
   # isql -Usa
   ```

   Enter the administrator password when prompted to do so.

3. Connect to the master database.

   ```
   1> use master
   2> go
   ```

4. Create a Sybase user account.

   ```
   1> sp_addlogin <user_name>, <password>
   2> go
   ```

   The detail monitor script should use this account to make transactions on the database.

5. Create a database.

   ```
   1> create database <database_name>
   2> go
   ```

   The detail monitor script should make transactions on this database.

6. If required, restrict the size of the log file for the database.

   ```
   1> sp_dboption <database_name>, "trunc log on chkpt", true
   2> go
   ```


7. Connect to the database created in step 5.

```
1> use <database_name>
2> go
```

8. Associate the user created in step 4 with the database created in step 5.

```
1> sp_adduser vcsuser
2> go
```

9. Change the user to the one created in step 4.

```
1> setuser "user_name"
2> go
```

10. Create a table in the database.

```
1> create table <table_name> (lastupd datetime)
2> go
```

The detail monitor script should make transactions on this table.

Note If you will use the `SqlTest.pl` for detail monitoring, make sure you create a table with a `lastupd` field of type `datetime`.

11. Verify the configuration by adding an initial value to the table.

```
1> insert into <table_name> (lastupd) values (getdate())
2> go
```

12. Exit the database.

```
1> exit
```

Installing the Sybase Agent 3

This chapter describes how to install the Sybase agent in a VCS cluster. If you have an earlier version of the agent installed, refer to "Upgrading the Agent" on page 15.

The VCS enterprise agent for Sybase is installed from the software disc. You must install the Sybase SQL Server and Sybase Backup Server agents on all the systems that will host the Sybase service group.

Prerequisites

- ✔ Verify VCS is install and configured. VERITAS recommends installing the VCS graphical user interface. If required, review the *VERITAS Cluster Server Installation Guide*.
- ✔ Verify Sybase is installed and configured, with the considerations described under "VCS Requirements for Installing Sybase" on page 9, on all cluster nodes where you will be installing the agent. Review the Sybase documentation for more information.
- ✔ Verify that the `sybase` account is valid and identical on all cluster systems that will run Sybase. Also, verify the `sybase` user account has execute permissions to Sybase binaries.

Installing the Agent

Install the Sybase enterprise agent on each node in the cluster.

▼ To install the agent

1. Insert the software disk in a disk drive connected to the system.
2. Log in as root.
3. Set the display permission on your workstation:

   # **xhost +myws**

4. Configure the shell environment variable DISPLAY on your workstation. For example, if your workstation has the name "myws." type:

   For Bourne or Korn shell (`sh` or `ksh`):

   ```
   # DISPLAY=myws:0.0
   # export DISPLAY
   ```

   For C shell (`csh` or `tcsh`):

   ```
   # setenv DISPLAY=myws:0.0
   ```

   Setting the DISPLAY environment variable this way enables you to use the SD-UX graphical interface.

5. Create a mount point directory, `/cdrom`, if it does not exist. The directory must have read-write permissions.
6. Determine the block device file for the disk drive. The device file should have the form `/dev/dsk/C#t#d#`. Type:

   # **ioscan -fnC disk**

7. Run the following commands to start PFS (Portable File System):

   ```
   # nohup pfs_mountd &
   # nohup pfsd &
   ```

8. Mount the CD. For example, to mount the CD to the mount point /cdrom, type:

   # **/usr/sbin/pfs_mount -t rrip /dev/dsk/c1t2d0 /cdrom**

   where `/dev/dsk/c1t2d0` is the drive's block device file.

9. Type the following command to install the agent:

   ```
   # swinstall -s /cdrom/sybase_agent/pkgs VRTSvcssy
   ```

Note Repeat step 1 through step 9 on all nodes that will become part of the Sybase service group.

Upgrading the Agent

This section describes steps to upgrade from the VCS enterprise agent 3.5 for Sybase in a VCS cluster.

1. Save the VCS configuration and stop the VCS engine.

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

2. Back up the configuration file, main.cf to a location on the cluster node.

3. Follow step a through step c on all systems that have the Sybase agent installed.

   a. Remove the Sybase agent

      ```
      # swremove VRTSvcssy
      ```

   b. Delete the file `/etc/VRTSvcs/conf/config/SybaseTypes.cf`.

   c. Install the VCS enterprise agent 4.1 for Sybase. See "Installing the Agent" on page 14.

4. Copy the file SybaseTypes.cf from the `/etc/VRTSvcs/conf/` directory to `/etc/VRTSvcs/conf/config` directory.

5. Copy the main.cf from the backed up location to `/etc/VRTSvcs/conf/config`.

6. Edit the main.cf located under `/etc/VRTSvcs/conf/config`.

   a. If you configured the `Monscript` attribute for the earlier version of the agent, replace `Monscript` with `MonScript`.

   a. Remove the `Monscript` attribute for the SybaseBk resource type.

   b. Replace the password specified for the `SApswd` attribute (of Sybase and SybaseBk resources) and `UPword` attribute (of Sybase resource) with encrypted passwords. See "Encrypting Passwords" on page 23 for instructions on encrypting passwords.

   c. Save the main.cf

7. Verify the configuration.

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

8. Start VCS on the local node.

9. Start VCS on other nodes.

Configuring the Sybase Service Group 4

Configuring the Sybase service group involves configuring service group resources and defining attribute values for the configured resources. You must have administrator privileges to create and configure a service group.

You can configure the Sybase service using the following methods:

- By using VCS Cluster Manager (Java Console) to edit a resource group template for the Sybase agent. See "Configuring the Service Group From Cluster Manager (Java Console)" on page 19).
- By using the types configuration files and directly editing the sample `main.cf` file supplied with the Sybase agent. This method requires you to restart VCS before the configuration takes effect. See "Configuring the Service Group From the Command Line" on page 21.

Before configuring the agent, see Appendix A, "Resource Type Definitions" on page 31 to review the Sybase resource types and their attributes.

Prerequisites

- ✔ Verify Sybase is installed and configured, with the considerations described under "VCS Requirements for Installing Sybase" on page 9, on all cluster nodes where you will be configuring the service group.
- ✔ Verify the Sybase agent is installed on all cluster systems. For more information, see "Installing the Sybase Agent" on page 13.
- ✔ Verify the Sybase agent type definition is imported into the VCS engine. See "Importing the SybaseTypes.cf File" on page 18 for instructions.

Importing the SybaseTypes.cf File

Before configuring the Sybase service group, you must import the `SybaseTypes.cf` file to the VCS engine.

▼ To import using the Cluster Manager

1. Start Cluster Manager and log on to the cluster.
2. From the **File** menu select **Import Types...**
3. In the Import Types dialog box, select the file:

   `/etc/VRTSvcs/conf/sample_sybase/SybaseTypes.cf`
4. Click **Import**.
5. Save the configuration.

▼ To import using the command line

1. Log in to a cluster system as `root`.
2. Ensure that all changes to the existing configuration have been saved and that further changes are prevented while you modify `main.cf`:

   ```
   # haconf -dump -makero
   ```
3. To ensure that VCS is not running while you edit `main.cf`, issue the following command to stop the VCS engine on all systems and leave the resources available:

   ```
   # hastop -all -force
   ```
4. Make a backup copy of the `main.cf` file:

   ```
   # cd /etc/VRTSvcs/conf/config
   # cp main.cf main.cf.orig
   ```
5. Edit the `main.cf` file to include the `SybaseTypes.cf` file:

   ```
   include "../SybaseTypes.cf"
   ```

At this point, the Sybase types definition has been imported to the VCS engine. The Sybase agent can be configured without interrupting or stopping VCS.

Configuring the Service Group From Cluster Manager (Java Console)

A template for the Sybase resource groups is automatically installed with the VCS enterprise agent for Sybase. Using the VCS Cluster Manager (Java Console), you can view the template, which displays the Sybase service group, its resources and their attributes. You can dynamically modify the attributes' values as necessary for your configuration.

▼ To configure a service group from the Java Console

1. Make sure the Sybase type definition file `SybaseTypes.cf` is imported in your configuration. See "Importing the SybaseTypes.cf File" on page 18 for instructions.

2. Launch the Cluster Configuration wizard using any of the following two ways:
   - From the Cluster Explorer menu, select **Tools** > **Configuration Wizard**.
   - If no service groups exist on the system, **Cluster Explorer** prompts you to launch the **Cluster Configuration wizard.** Click **Yes** when prompted.

   The Loading Templates Information window appears, and launches the wizard.

3. Review the information in the Welcome dialog box and click **Next**.

4. Specify the name of the service group and the target systems on which the service group will be configured.

   a. Enter the name of the service group.

   b. From the **Available Systems** box, select the systems on which to configure the service group.

   c. Click the right arrow to move the selected systems to the **Systems for Service Group** box. To remove a system from the box, select the system and click the left arrow.

   d. Indicate the order in which the systems will fail over (priority). System priority is numbered sequentially, with 1 denoting the system that starts first following a failover.

   e. Click **Next**.

5. On the Would you like to use a template to configure the service group? dialog box, click **Next**.

6. On the Select a template to configure the service group dialog box, select the SybaseGroup template on which to base the service group.

   If applicable, a window opens notifying that names of some resources within the new service group are already in use. Resolve the name clashes, if any and click **Next**.

7. Click **Next** to create the service group based on the selected template. A window opens indicating that commands are being sent to add the group, its resources, and the attributes and dependencies specified in the template. A progress indicator displays the percentage of the commands fired. The actual commands are displayed at the top of the indicator.

8. Click **Next** when prompted that the service group has been successfully created.

9. A window opens listing the service group's resources and their associated attributes. Click **Finish** to accept the default values and complete the configuration. You can modify the default values of the resources according to your specific configuration requirements, as instructed in the following section.

Editing Resource attributes

1. Select the resource from the list on the left pane. The resource attributes appear in the right pane.

2. Select the attribute to be modified and click the edit icon in the **Edit** column.

3. In the Edit Attribute dialog box, enter the attribute values. To modify the scope of the attribute, choose the **Global** or **Local** option.

4. Click **OK**.

5. Repeat the procedure for each resource and its attributes and click **Finish**. Edit the attributes for all the resources according to your configuration.

   Caution For added security, you must always provide a secure value for passwords See "Encrypting Passwords" on page 23 for instructions on setting secure passwords.

6. Follow the wizard instructions to complete the configuration. Click **Finish** to quit the wizard.

Configuring the Service Group From the Command Line

The VCS enterprise agent for Sybase comes with a sample configuration file that can be used as reference to directly modify your present configuration file. This method requires you to restart VCS before the configuration takes effect.

▼ To configure a service group from the command line

1. Log in to a cluster system as `root`.
2. Make sure the Sybase type definition is imported into VCS engine. See "Importing the SybaseTypes.cf File" on page 18 for instructions.
3. Edit the `main.cf` file. Use the file `/etc/VRTSvcs/conf/sample_sybase/main.cf` for reference:
   a. Create a Sybase service group.
   b. Create the Sybase and SybaseBk resources. Refer to the type definitions under "Sybase Resource Type" on page 31 and "SybaseBk Resource Type" on page 34.
   c. Edit the default attributes to match the parameters in your configuration.

   Caution For added security, you must always provide a secure value for passwords See "Encrypting Passwords" on page 23 for instructions on setting secure passwords.

   d. Assign dependencies to the newly created resources. Refer to the sample file `/etc/VRTSvcs/conf/sample_sybase/main.cf`. (See the *VERITAS Cluster Server User's Guide* for more information on assigning dependencies.)
4. Save and close the file.
5. Verify the syntax of the file `/etc/VRTSvcs/conf/config/main.cf`:

   ```
   # hacf -verify config
   ```
6. Start the VCS engine:

   ```
   # hastart
   ```
7. Verify all Sybase service group resources are brought online:

   ```
   # hagrp -state
   ```


8. Take the service group offline and verify that all resources are stopped:

```
# hagrp -offline <service_group> -sys <system_name>
# hagrp -state
```

9. Bring the service group online again and verify that all resources are available:

```
# hagrp -online <service_group> -sys <system_name>
# hagrp -state
```

10. Start the VCS engine on sysb:

```
# hastart
```

11. Switch the Sybase service group to sysb:

```
# hagrp -switch <service_group> -to <system_name>
```

12. Verify that all Sybase service group resources are brought online on sysb:

```
# hagrp -state
```

13. On all systems, look at the following log files for any errors or status:

```
/var/VRTSvcs/log/engine_A.log
/var/VRTSvcs/log/Sybase_A.log
/var/VRTSvcs/log/SybaseBk_A.log
```

Encrypting Passwords

VCS provides a `vcsencrypt` utility to encrypt user passwords. Encrypt passwords before specifying them for Sybase and SybaseBk resource type definition.

Note You need not encrypt passwords when the VCS Cluster Manager (Java Console) to configure attributes.

1. From the path `$VCS_HOME/bin/`, run the `vcsencrypt` utility.

   a. Type the following command:

   ```
   # vcsencrypt -agent
   ```

   b. The utility prompts you to enter the password twice. Enter the password and press Return.

   ```
   # Enter New Password:
   # Enter Again:
   ```

2. The utility encrypts the password and displays the encrypted password.

3. Enter this encrypted password as the value for the attribute.

   Copy the encrypted password for future reference.

Setting Up Detail Monitoring for Sybase Agent

The VCS enterprise agent for Sybase provides two levels of application monitoring: basic and detail. In basic monitoring, Sybase resource monitors the Sybase daemon processes to verify that they are continuously active.

In detail monitoring, the Sybase resource performs transactions on a test table in the database to ensure that the Sybase server is functioning properly. The agent uses the script defined in the attribute `MonScript` of the Sybase resource. During detail monitoring, the agent tries to execute the specified script. If the script successfully executes, the agent considers the database available. You can customize the default script according to your configurations.

To activate detail monitoring, the `DetailMonitor` attribute must be set to "1" and `User`, `UPword`, `Db`, and `Table` must not be empty (""). The attribute `MonScript`, which contains the path of the detail monitor script, must also exist and have execute permissions for the `root`.

Enabling Detail Monitoring

1. Make sure Sybase server is configured for detail monitoring. See "Configuring Sybase for Detail Monitoring" on page 11 for instructions.

1. Make the VCS configuration writable:

   ```
   # haconf -makerw
   ```

2. Freeze the service group to avoid automated actions by VCS caused by incomplete reconfiguration:

   ```
   # hagrp -freeze <service_group>
   ```

3. Enable detail monitoring for Sybase:

   ```
   # hares -modify <Sybase_resource>  DetailMonitor 1
   # hares -modify <Sybase_resource>  User <user>
   # hares -modify <Sybase_resource>  UPword <password>
   # hares -modify <Sybase_resource>  Db <database_name>
   # hares -modify <Sybase_resource>  Table <table_name>
   # hares -modify <Sybase_resource>  MonScript
     "/opt/VRTSvcs/bin/Sybase/SqlTest.pl"
   ```

4. Unfreeze and save the configuration:

   ```
   # hagrp -unfreeze <service_group>
   # haconf -dump [-makero]
   ```

Disabling Detail Monitoring

1. Make the VCS configuration writable:

   ```
   # haconf -makerw
   ```

2. Freeze the service group to avoid automated actions by VCS caused by incomplete reconfiguration:

   ```
   # hagrp -freeze <service_group>
   ```

3. Disable detail monitoring for Sybase:

   ```
   # hares -modify <Sybase_resource>  DetailMonitor 0
   ```

4. Unfreeze and save the configuration:

   ```
   # hagrp -unfreeze <service_group>
   # haconf -dump [-makero]
   ```


Administering the Sybase Service Group 5

This chapter describes the administrative tasks you can perform on the Sybase service group, including bringing the service group online, taking it offline, switching it, and modifying the agent configuration. The chapter also describes how to disable the agent and to remove it.

Bringing the Service Group Online

1. In the Cluster Explorer configuration tree, select the newly created service group.
2. Right-click the service group name, and select **Enable Resources**. This enables all the resources in the group.
3. Right-click the service group name, and select the systems on which to enable the service group (Right-click>Enable>*system_name* or Right-click>Enable>All).
4. Save your configuration (File>Close Configuration).
5. Right-click the service group and select to online the service group on the system (Right-click>Online>*system_name*).

Taking the Service Group Offline

1. On the **Service Groups** tab of the Cluster Explorer configuration tree, right-click the service group.

   or

   Select the cluster in the Cluster Explorer configuration tree, select the **Service Groups** tab, and right-click the service group icon in the view panel.
2. Choose **Offline**, and choose the appropriate system from the pop-up menu (Right-click>Offline>*system_name*).

Switching the Service Group

The process of switching a service group involves taking it offline on its current sytem and bringing it online on the another system

1. On the **Service Groups** tab of Cluster Explorer configuration tree, right-click the service group.

   or

   Select the cluster in the Cluster Explorer configuration tree, select the **Service Groups** tab, and right-click the service group icon in the view panel.

2. Choose **Switch To**, and choose the appropriate system from the pop-up menu (Right-click>Switch To>*system_name*).

Viewing the Agent Log

The VCS enterprise agent for Sybase logs messages to the following files:

```
/var/VRTSvcs/log/engine_A.log
/var/VRTSvcs/log/Sybase_A.log
/var/VRTSvcs/log/SybaseBk_A.log
```

Modifying the Service Group Configuration

You can dynamically modify the VCS enterprise agent for Sybase using several methods, including the Cluster Manager (Java Console), Cluster Manager (Web Console), and the command line. See the *VERITAS Cluster Server User's Guide* for more information.

Disabling the Agent

To disable the agent on a system, you must first take the Sybase service group offline. You can stop the application completely, or switch the service group to another system.

1. Determine if the service group is online by entering:

   # **hagrp -state *<service_group>* -sys *<system_name>***

2. If the service group is online, take it offline by entering:

   # **hagrp -offline *<service_group>* -sys *<system_name>***

   or

   # **hagrp -switch *<service_group>* -to *<system_name>***

3. Stop the agents on the system by entering:

   # **haagent -stop Sybase -sys *<system_name>***
   # **haagent -stop SybaseBk -sys *<system_name>***

   When you get the message "`Please look for messages in the log file`," check the file `/var/VRTSvcs/log/engine_A.log` for a message confirming the agent has stopped.

   You can also use the `ps` command to verify whether the agent is stopped.

You can remove the system, service group, or resource type from the VCS configuration after disabling the agent on all systems. See the chapter on reconfiguring VCS from the command line in the *VERITAS Cluster Server User's Guide* for more information.

Removing the Agent

Type the following command on each sytem to remove the agent. Answer prompts accordingly:

swremove VRTSvcssy

Resource Type Definitions A

This appendix lists resource type definitions and attribute definitions of the Sybase and SybaseBk agents.

The resource type represents the VCS configuration definitions of the agent and specifies how the agent is defined in the configuration file main.cf. The Attribute Definitions table explains the attributes associated with the agent. The Required Attributes table lists the attributes that must be configured for the agent to function properly.

Sybase Resource Type

```
type Sybase (
  static str ArgList[] = { Server, Owner, Home, Version, SA,
                           SApswd, User, UPword, Db, Table, MonScript,
                           DetailMonitor }
  str Server
  str Owner
  str Home
  str Version
  str SA
  str SApswd
  int DetailMonitor = 0
  str User
  str UPword
  str Db
  str Table
  str MonScript="./bin/Sybase/SqlTest.pl"
)
```


Attribute Definitions

Required Attributes	Type and Dimension	Definition
Server	string-scalar	The `$DSQUERY` ASE name. Only one server must be configured in a Sybase service group.
Owner	string-scalar	Sybase user as defined `/etc/passwd`. The Sybase executables and database files are accessed in the context of this user.
Home	string-scalar	The `$SYBASE` path to Sybase binaries and configuration files.
Version	string-scalar	Version of Sybase ASE.
SA	string-scalar	Sybase database administrator. This is required to connect to the ASE for shutdown.
SApswd	string-scalar	Encrypted password for Sybase database administrator. This password is required to connect to the ASE for shutdown. See "Encrypting Passwords" on page 23 for instructions on encrypting passwords. **Note** You need not specify a value for this attribute if the SA user does not require a password.

Optional Attributes	Type and Dimension	Definition
DetailMonitor	int-scalar	Specifies whether the Sybase server is monitored in detail. Value 1 indicates that the resource monitors the Sybase server in detail. Value 0 denotes it does not. Default is 0.
User	string-scalar	The database user, in the context of which, the transactions are performed on the database. Make sure you specify the user name created in step 4 on page 11. **Note** You must specify a value for this attribute if DetailMonitor is set to a non-zero value.

Optional Attributes	Type and Dimension	Definition
UPword	string-scalar	Encrypted password for the database user. Make sure you provide the password specified in step 4 on page 11. See "Encrypting Passwords" on page 23 for instructions on encrypting passwords. **Note** You must specify a value for this attribute if DetailMonitor is set to a non-zero value. However, you need not specify a value for this attribute if the database user does not require a password.
Db	string-scalar	Name of the database, in which the table resides, on which the detail monitor script will perform the transactions. Make sure you specify the name of the database created in step 5 on page 11. **Note** You must specify a value for this attribute if DetailMonitor is set to a non-zero value.
Table	string-scalar	Name of the table on which the detail monitoring script will perform the transactions. Make sure you specify the name of the table created in step 10 on page 12. **Note** You must specify a value for this attribute if DetailMonitor is set to a non-zero value.
MonScript	string-scalar	The Path to the detail monitor script; the default value for this attribute is the path for the script, SqlTest.pl, provided with the agent. **Note** You must specify a value for this attribute if DetailMonitor is set to a non-zero value.

SybaseBk Resource Type

```
type SybaseBk (
  static str ArgList[] = { Backupserver, Owner, Home, Version,
                           Server, SA, SApswd }
  str Server
  str Owner
  str Home
  str Version
  str Backupserver
  str SA
  str SApswd
)
```

Attribute Definitions

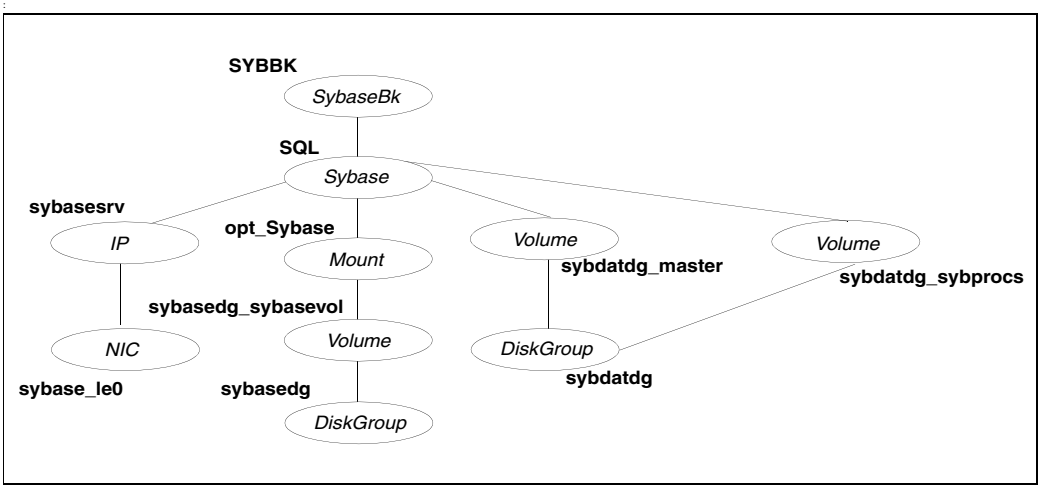
Attributes	Type and Dimension	Definition
Server	string-scalar	The `$DSQUERY` Backup server name.
Owner	string-scalar	Sybase user as defined `/etc/passwd`. The Sybase executables and database files are accessed in the context of this user.
Home	string-scalar	The `$SYBASE` path to Sybase binaries and configuration files.
Version	string-scalar	Version of Sybase Backup Server.
Backupserver	string-scalar	The `$BACKUP` SYBASE Backup Server name.
SA	string-scalar	Sybase database administrator. This is required to connect to the ASE for shutdown.
SApswd	string-scalar	Encrypted password of Sybase database administrator. This password is required to connect to the ASE for shutdown. See "Encrypting Passwords" on page 23 for instructions on encrypting passwords. **Note** You need not specify a value for this attribute if the SA user does not require a password.

Sample Configuration B

This appendix describes a typical service group configured to monitor the state of Sybase servers in a VCS cluster. The sample configuration graphically depicts the resource types, resources, and resource dependencies within the service group. Review these dependencies carefully before configuring the agent. For more information about these resource types, see the *VCS Bundled Agents Reference Guide*.

Resource Dependency Graph

The figure below illustrates the configuration's dependency graph



This configuration has two disk groups on which three volumes are created. One disk group is for the Sybase server, and one is for the database. The mount resource requires the Volume resource, which in turn requires the DiskGroup resource. The service group IP address for the Sybase server is configured using the IP and NIC resource types. The Sybase server can be started after each of these resources is brought online. The Backup Server is started after the Sybase SQL Server is online.

Note If your configuration does not use VERITAS Volume Manager, use the LVMVolumeGroup and LVMLogicalVolume resource types to configure shared storage instead of DiskGroup and Volume resource types.

Sample Configuration for a Service Group Using LVM

```
include "types.cf"
include "../SybaseTypes.cf"

cluster hpslia3031 (
        UserNames = { admin = IhiAhcHeiDiiGqiChf }
        Administrators = { admin }
        CredRenewFrequency = 0
        CounterInterval = 5
        )

system hpslia30 (
        )

system hpslia31 (
        )


group  Sybase_group(
SystemList = { hpslia30 = 0, hpslia31 = 1 }
AutoStartList = { hpslia30,hpslia31 }
)

IP ipres (
  Device = lan0
  Address = "192.168.175.28"
  NetMask = "255.255.0.0"
  )

Mount sybase_mount (
  MountPoint = "/sybase_mnt"
  BlockDevice = "/dev/vg03/sybase"
  FSType = vxfs
  MountOpt = rw
  FsckOpt = "-n"
  )

NIC nicres (
  Device = lan0
  NetworkHosts = { "192.168.170.107", "192.168.170.108" }
  )

Sybase sybase_agent (
  Server = SYBASE_SERVER
  Owner = sybase
```


```
    Home = "/sybase_mnt/IA"
    Version = "12.5.2"
    SA = sa
    SApswd= ""
    User = vcsuser
    UPword = "jxpVmxMvrXjxKvk"
    Db = vcsdb
    Table = vcstable
    Monscript = "/opt/VRTSvcs/bin/Sybase/SqlTest.pl"
    DetailMonitor = 1
    )

SybaseBk sybase_bk_agent (
    Server = SYBASE_SERVER
    Owner = sybase
    Home= "/sybase_mnt/IA"
    Version= "12.5.2"
    Backupserver = SYBASE_SERVER_BS
    SA = sa
    SApswd =  ""
    )

LVMLogicalVolume lvolres (
    LogicalVolume = sybase
    VolumeGroup = vg03
    )

LVMVolumeGroup lvgres (
    VolumeGroup = vg03
    )

lvolres requires lvgres
sybase_mount requires lvolres
ipres requires nicres
sybase_agent requires ipres
sybase_agent requires sybase_mount
sybase_bk_agent requires sybase_agent
```

Sample Configuration for a Service Group Using VxVM

```
include "types.cf"
include "../SybaseTypes.cf"


cluster vcs_test (
        UserNames = { admin = anoGniNkoJooMwoInl,
                      "root@hpslia30.veritas.com" = en,
                      "root@hpslia31.veritas.com" = en }
        Administrators = { admin, "root@hpslia30.veritas.com",
                           "root@hpslia31.veritas.com" }
        SecureClus = 1
        CredRenewFrequency = 0
        CounterInterval = 5
        )

system hpslia30 (
        )

system hpslia31 (
        )

group  Sybase_group(
 SystemList = { hpslia30 = 0, hpslia31 = 1 }
 AutoStartList = { hpslia30,hpslia31 }
 )

 IP ipres (
  Device = lan0
  Address = "192.168.175.28"
  NetMask = "255.255.0.0"
  )

Mount sybase_mount (
  MountPoint = "/sybase_mnt"
  BlockDevice = "/dev/vx/dsk/sybasedg/sybase_vol"
  FSType = vxfs
  MountOpt = rw
  FsckOpt = "-n"
  )

 NIC nicres (
  Device = lan0
  NetworkHosts = { "192.168.170.107", "192.168.170.108" }
  )
```


```
Sybase sybase_agent (
  Server = SYBASE_SERVER
  Owner = sybase
  Home= "/sybase_mnt/IA"
  Version = "12.5.2"
  SA = sa
  SApswd= ""
  User = vcsuser
  UPword = "jxpVmxMvrXjxKvk"
  Db =  "vcsdb"
  Table ="vcstable"
  Monscript = "/opt/VRTSvcs/bin/Sybase/SqlTest.pl"
  DetailMonitor = 1
  )

 SybaseBk sybase_bk_agent (
  Server = SYBASE_SERVER
  Owner = sybase
  Home = "/sybase_mnt/IA"
  Version = "12.5.2"
  Backupserver = SYBASE_SERVER_BS
  SA = sa
  SApswd =  ""
  )

 DiskGroup dgres (
  DiskGroup = sybasedg
 )

 Volume volres (
  Volume= sybase_vol
  DiskGroup=sybasedg
  )

 volres requires dgres
 sybase_mount requires volres
 ipres requires nicres
 sybase_agent requires ipres
 sybase_agent requires sybase_mount
 sybase_bk_agent requires sybase_agent
```

```
group VxSS (
        SystemList = { hpslia30 = 0, hpslia31 = 1 }
        Parallel = 1
        OnlineRetryLimit = 3
        OnlineRetryInterval = 120
        )

        Phantom phantom_vxss (
        )

        ProcessOnOnly vxatd (
                IgnoreArgs = 1
                PathName = "/opt/VRTSat/bin/pa20_64/vxatd"
                )
```


Index

V