

Veritas[™] High Availability Agent for Oracle Installation and Configuration Guide

Linux

5.0

Veritas High Availability Agent for Oracle Installation and Configuration Guide

Copyright © 2006 Symantec Corporation. All rights reserved.

VCS 5.0

Symantec, the Symantec logo, Veritas are trademarks or registered trademarks of Symantec Corporation or its affiliates in the U.S. and other countries. Other names may be trademarks of their respective owners.

The product described in this document is distributed under licenses restricting its use, copying, distribution, and decompilation/reverse engineering. No part of this document may be reproduced in any form by any means without prior written authorization of Symantec Corporation and its licensors, if any.

THIS DOCUMENTATION IS PROVIDED “AS IS” AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID, SYMANTEC CORPORATION SHALL NOT BE LIABLE FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES IN CONNECTION WITH THE FURNISHING PERFORMANCE, OR USE OF THIS DOCUMENTATION. THE INFORMATION CONTAINED IN THIS DOCUMENTATION IS SUBJECT TO CHANGE WITHOUT NOTICE.

The Licensed Software and Documentation are deemed to be “commercial computer software” and “commercial computer software documentation” as defined in FAR Sections 12.212 and DFARS Section 227.7202.

Symantec Corporation
20330 Stevens Creek Blvd.
Cupertino, CA 95014
www.symantec.com

Third-party legal notices

Third-party software may be recommended, distributed, embedded, or bundled with this Symantec product. Such third-party software is licensed separately by its copyright holder. All third-party copyrights associated with this product are listed in the accompanying release notes.

Linux is a registered trademark of Linus Torvalds.
Oracle is a registered trademark of Oracle Corporation.

Licensing and registration

Veritas Cluster Server is a licensed product. See the *Veritas Cluster Server Installation Guide* for license installation instructions.

Technical support

For technical assistance, visit <http://support.veritas.com> and select phone or email support. Use the Knowledge Base search feature to access resources such as TechNotes, product alerts, software downloads, hardware compatibility lists, and our customer email notification service.

Contents

Chapter 1	Introducing the Veritas High Availability Agent for Oracle	
	About the Veritas High Availability Agent for Oracle	9
	What's new in this release	10
	Supported software	10
	Oracle agent functions	11
	Startup and shutdown options	12
	Startup options	12
	Shutdown options	13
	Monitor options for Oracle agent	13
	Basic monitoring options	14
	Detail monitoring	14
	Info entry point	14
	Action entry point	15
	Netlsnr agent functions	16
	How the agent makes Oracle highly available	17
	How the agent handles Oracle error codes	
	during detail monitoring	18
	Typical Oracle configuration in a VCS cluster	19
	About setting up an Oracle cluster	20
Chapter 2	Installing and configuring Oracle	
	Before you install Oracle	21
	VCS requirements for installing Oracle	21
	Linux configuration	21
	Disk partitions	22
	Linux package group	22
	Shared memory	22
	Swap size	22
	Semaphores	23
	File handles	23
	JDK requirements for Linux	23
	Location of the \$ORACLE_HOME	23
	Multiple Oracle instances (SIDs)	24
	Location of database tablespaces	24
	Transparent listener failover	25

Hot backup of Oracle database and VCS	25
Installing Oracle in a VCS environment	25
Overview of Oracle installation tasks	26
Installation tasks for \$ORACLE_HOME on shared disks	26
Installation tasks for \$ORACLE_HOME on local disks	27
Installing Oracle binaries	28
Disabling the Oracle 10g clustering daemon	29
Configuring the Oracle database	30
Copying the \$ORACLE_BASE/admin/SID directory	32
Starting the database	32
Starting the database when your	
\$ORACLE_HOME is on shared disks	32
Starting the database when your	
\$ORACLE_HOME is on local disks	32

Chapter 3 Installing the agent for Oracle

Before you install or upgrade the agent for Oracle	35
Installing the agent software	35
Upgrading the agent	36
Upgrading VCS agent 4.0 and 4.1 for Oracle	36

Chapter 4 Configuring VCS service groups for Oracle

About configuring a service group for Oracle	39
About Oracle configurations in VCS	39
Configuring single Oracle instance in VCS	40
Configuring multiple Oracle instances (single listener) in VCS	41
Configuring multiple Oracle instances (multiple listeners) in VCS	42
Configuring Oracle instance with shared server support in VCS	43
Before you configure the service group	45
Importing the OracleTypes.cf file	45
Configuring the service group	46
Configuring the service group	
using the agent configuration wizard	47
Configuring the service group	
using cluster manager (Java console)	55
Configuring the service group using the command-line	57
Encrypting passwords	59
Additional considerations for basic monitoring	59
Setting up detail monitoring	60
Setting up detail monitoring for Oracle	61
Enabling and disabling detail monitoring for Oracle	62
Setting up detail monitoring for Netlsnr	62

Chapter 5	Administering VCS service groups for Oracle	
	About administering VCS service groups	65
	Bringing the service group online	65
	Taking the service group offline	66
	Switching the service group	66
	Modifying the service group configuration	66
	Disabling the Veritas High Availability Agent for Oracle	67
	Removing the Veritas High Availability Agent for Oracle	68
Chapter 6	Troubleshooting	
	Error messages common to the Oracle and Netlsnr agents	69
	Error messages specific to the Oracle agent	71
	Error messages specific to the Netlsnr agent	73
Appendix A	Resource type definitions	
	About the resource type and attribute definitions	75
	Oracle resource type	75
	Oracle attribute definitions	76
	Required attributes	76
	Optional attributes	77
	Internal attribute	80
	Netlsnr resource type	80
	Netlsnr attribute definitions	80
	Required attributes	81
	Optional attributes	81
	Internal attribute	83
Appendix B	Sample configurations	
	About the sample configurations for Oracle enterprise agent	85
	Sample single Oracle instance configuration	85
	Sample configuration file for single Oracle instance	86
	Sample multiple Oracle instances (single listener) configuration	88
	Sample configuration file for	
	multiple Oracle instances (single listener)	90
	Sample multiple instance (multiple listeners) configuration	93
	Sample configuration file	93
	Sample Oracle configuration with shared server support	97
	Sample configuration file for	
	Oracle instance configured with shared server support	98

Appendix C	Best practices	
	Best practices for multiple Oracle instance configurations	101
Appendix D	Using the SPFILE in a VCS cluster for Oracle	
	About the Oracle initialization parameter files	103
	Starting an Oracle instance	103
	Using the SPFILE in a VCS cluster	104
	Specifying the SPFILE location in the PFILE	104
	Creating the SPFILE from a PFILE	104
Index		107

Introducing the Veritas High Availability Agent for Oracle

This chapter contains the following topics:

- [About the Veritas High Availability Agent for Oracle](#)
- [What's new in this release](#)
- [Supported software](#)
- [Oracle agent functions](#)
- [Netlsnr agent functions](#)
- [How the agent makes Oracle highly available](#)
- [Typical Oracle configuration in a VCS cluster](#)
- [About setting up an Oracle cluster](#)

About the Veritas High Availability Agent for Oracle

The Veritas High Availability Agent for Oracle provides high availability for Oracle. Veritas high availability agents monitor specific resources within an enterprise application, determine the status of these resources, and start or stop them according to external events.

The Veritas High Availability Agent for Oracle monitors the Oracle and listener processes, brings them online, and takes them offline.

The package contains two agents:

- The Oracle agent monitors the Oracle database processes.

- The Netlsnr agent monitors the listener process.
- The agents include resource type declarations and agent executables, and are represented with the Oracle and Netlsnr resource types, respectively. Both agents work together to make Oracle highly available.

What's new in this release

The Veritas High Availability Agent for Oracle includes the following new or enhanced features:

- New monitoring option
The basic monitoring option of the Oracle agent now allows health check monitoring in addition to the process check monitoring. You can choose health check monitoring option for Oracle 10g and later.
See “[Monitor options for Oracle agent](#)” on page 13.
- Allows performing virtual fire drills
VCS requires you to keep the configurations in sync with the underlying infrastructure on a node. Virtual fire drills detect such discrepancies that prevent a service group from going online on a specific node. Refer to the *Veritas Cluster Server User's Guide* for more information.
The Veritas High Availability Agent for Oracle uses the Action entry point to support the virtual fire drill functionality.
See “[Action entry point](#)” on page 15.
- With Veritas High Availability Agent for Oracle 5.0, the location of agent binary directory is /opt/VRTSagents/ha/bin. The AgentDirectory attribute indicates this location.
See “[Resource type definitions](#)” on page 75.

Supported software

The Veritas High Availability Agent for Oracle 5.0 supports the following software versions in a Veritas Cluster Server (VCS) environment:

Oracle	Oracle9i, Oracle 10g R1, and Oracle 10g R2 (including 64-bit versions) Note: Veritas High Availability Agent for Oracle supports the specified Oracle versions on Linux if Oracle supports that version on Linux. Refer to the <i>Veritas Cluster Server Release Notes</i> for more details.
VCS	VCS 5.0 on Linux
Linux	<ul style="list-style-type: none">■ Red Hat Enterprise Linux 4 (Update 3)■ SUSE Linux Enterprise Server 9 with SP3

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

Oracle agent functions

The Oracle agent monitors the database processes. [Table 1-1](#) lists the Oracle agent operations. The functions an agent performs are called entry points.

Table 1-1 Oracle agent operations

Agent operation	Description
Online	<p>Starts the Oracle database by using the following <code>svrmgrl</code> or <code>sqlplus</code> command:</p> <pre>startup force pfile=\$PFile</pre> <p>The default Startup option is <code>STARTUP_FORCE</code>. You can also configure the agent to start the database using different Startup options for Oracle.</p> <p>See “Startup options” on page 12.</p>
Offline	<p>Stops the Oracle database with the specified options by using the following <code>svrmgrl</code> or <code>sqlplus</code> command:</p> <pre>shutdown immediate</pre> <p>The default Shutdown option is <code>IMMEDIATE</code>. You can also configure the agent to stop the database using different Shutdown options for Oracle.</p> <p>See “Shutdown options” on page 13.</p>
Monitor	<p>Verifies the status of the Oracle processes. The Oracle agent provides two levels of monitoring: basic and detail.</p> <p>See “Monitor options for Oracle agent” on page 13.</p>
Clean	<p>Forcibly stops the Oracle database by using the following <code>svrmgrl</code> or <code>sqlplus</code> command:</p> <pre>shutdown abort</pre> <p>If the process does not respond to the <code>shutdown</code> command, the agent scans the process table for processes associated with the configured instance and kills them.</p>
Info	<p>Provides static and dynamic information about the state of the database.</p> <p>See “Info entry point” on page 14.</p>

Table 1-1 Oracle agent operations

Agent operation	Description
Action	Performs predefined actions on a resource. See “ Action entry point ” on page 15.

Startup and shutdown options

You can specify Startup and Shutdown options for Oracle instances that are configured.

Startup options

[Table 1-2](#) lists the startup options that the agent supports.

Table 1-2 Startup options

Option	Description
STARTUP_FORCE (Default)	Runs <code>startup force pfile='location_of_pfile'</code> if the pfile is configured. If the pfile is not configured, the agent runs <code>startup force</code> . It picks up the default parameter files from their default locations.
STARTUP	Runs <code>startup pfile='location_of_pfile'</code> if the pfile is configured. If the pfile is not configured, the agent picks up the default parameter files from their default locations and runs <code>startup</code> .
RESTRICTED	Starts the database in the <code>RESTRICTED</code> mode.
RECOVERDB	Performs a database recovery on instance startup.
CUSTOM	Uses a predefined SQL script (<code>start_custom_\${SID}.sql</code>) and runs custom startup options. The script must be in the <code>/opt/VRTSagents/ha/bin/Oracle</code> directory and must have access to the Oracle Owner OS user. If the file is not present, the agent logs an error message. With a custom script, the agent takes the following action: <pre>sqlplus /nolog <<! connect / as sysdba; @start_custom_\${SID}.sql exit; !</pre>

Table 1-2 Startup options

Option	Description
SRVCTLSTART	Uses the <code>srvctl</code> utility to start an instance of the database. For RAC clusters with Oracle9i R2 and later, you must manually set the default startup option as SRVCTLSTART.

Shutdown options

Table 1-3 lists the shutdown options that the agent supports.

Table 1-3 Shutdown options

Option	Description
IMMEDIATE (Default)	Shuts down the Oracle instance by running <code>shutdown immediate</code> .
TRANSACTIONAL	Runs the <code>shutdown transactional</code> command. This option is valid only for database versions that support this option.
CUSTOM	Uses a predefined SQL script (<code>shut_custom_<SID>.sql</code>) and runs custom shutdown options. The script must be in the <code>/opt/VRTSagents/ha/bin/Oracle</code> directory and must have access to the Oracle Owner OS user. If the file is not present, the agent shuts the agent down with the default option.
SRVCTLSTOP	Uses the <code>srvctl</code> utility to stop an instance of the database. For RAC clusters for Oracle9i R2 and later, you must manually set the default option as SRVCTLSTOP.

Monitor options for Oracle agent

The Oracle agent provides two levels of monitoring: basic and detail. By default, the agent does a basic monitoring.

Basic monitoring options

The basic monitoring mode has two options: Process check and Health check. [Table 1-4](#) describes the basic monitoring options.

Table 1-4 Basic monitoring options

Option	Description
0 (Default)	Process check The agent scans the process table for the ora_dbw, ora_smon, ora_pmon, and ora_lgwr processes to verify that Oracle is running.
1	Health check (supported on Oracle 10g and later) The agent uses the Health Check APIs from Oracle to monitor the SGA and retrieve the information about the instance.

Detail monitoring

In the detail monitoring mode, the agent performs a transaction on a test table in the database to ensure that Oracle is functioning properly.

Info entry point

The Veritas High Availability Agent for Oracle supports the Info entry point, which provides static and dynamic information about the state of the database.

To invoke the Info entry point, type the following command:

```
# hares -refreshinfo resource [-sys system] \
[-clus cluster | -localclus]
```

The entry point retrieves the following static information:

- Version

■ InstanceNo

■ InstanceName
- DatabaseName

■ HostName

■ StartupTime
- Parallel

■ Thread

■ InstanceRole

The entry point retrieves the following dynamic information:

- InstanceStatus

■ Logins

■ OpenMode
- LogMode

■ ShutdownPending

■ DatabaseStatus
- Shared Pool Percent free

■ Buffer Hits Percent

■

You can add additional attributes by adding sql statements to the file `/opt/VRTSagents/ha/bin/Oracle/resinfo.sql`. For example:

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

The format of the selected record must be as follows:

attribute_type:*userkey_name*:*userkey_value*

The variable *attribute_type* can take the value static and/or dynamic.

Action entry point

The Veritas High Availability Agent for Oracle supports the Action entry point, which enables you to perform predefined actions on a resource. [Table 1-5](#) describes the agent's predefined actions.

See [Table 1-6, "Predefined virtual fire drill actions."](#)

To perform an action on a resource, type the following command:

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

You can also add custom actions for the agent. For further information, refer to the *Veritas Cluster Server Agent Developer's Guide*.

Table 1-5 Predefined agent actions

Action	Description
VRTS_GetInstanceName	Retrieves the name of the configured instance. You can use this option for the Oracle and the Netlsnr resources.
VRTS_GetRunningServices	Retrieves the list of processes monitored by the agent. You can use this option for the Oracle and the Netlsnr resources.
DBRestrict	Changes the database session to enable the RESTRICTED mode.
DBUndoRestrict	Changes the database session to disable the RESTRICTED mode.
DBSuspend	Suspends a database.
DBResume	Resumes a suspended database.
DBTbspBackup	Backs up a tablespace; <i>actionargs</i> contains name of the tablespace to be backed up.

Table 1-6 lists the virtual fire drill actions of the Veritas High Availability Agent for Oracle allows you to run infrastructure checks and fix specific errors.

Table 1-6 Predefined virtual fire drill actions

Virtual fire drill action	Description
getid (Oracle agent)	Verifies that the Oracle Owner exists on the node.
home.vfd (Oracle agent)	Verifies the following: <ul style="list-style-type: none">■ ORACLE_HOME is mounted on the node and corresponding entry is in the fstab. If the ORACLE_HOME is not mounted, the action entry point checks if any other resource has already mounted ORACLE_HOME.■ Pfile is provided and it exists on the node.■ Password file from \$ORACLE_HOME/dbs/orapw[SID] is present.
owner.vfd (Oracle agent)	Verifies the uid and gid of the Oracle Owner attribute. Checks if uid and gid of Owner attribute is the same on the node where the Oracle resource is currently ONLINE.
tnsadmin.vfd (Netlsnr agent)	Checks if listener.ora file is present. If listener.ora file, it checks if ORACLE_HOME is mounted and displays appropriate messages.

Netlsnr agent functions

The listener is a server process that listens to incoming client connection requests and manages traffic to the database. The Netlsnr agent brings the listener services online, monitors their status, and takes them offline. Table 1-7 lists the Netlsnr agent operations.

Table 1-7 Netlsnr agent operations

Agent operation	Description
Online	Starts the listener process by using the following command: <code>lsnrctl start \$LISTENER</code>

Table 1-7 Netlsnr agent operations

Agent operation	Description
Offline	<p>Stops the listener process by using the following command:</p> <pre>lsnrctl stop \$LISTENER</pre> <p>If the listener is configured with a password, the agent uses the password to stop the listener.</p>
Monitor	<p>Verifies the status of the listener process.</p> <p>The Netlsnr agent provides two levels of monitoring: basic and detail.</p> <ul style="list-style-type: none"> ■ In the basic monitoring mode, the agent scans the process table for the <code>tnslsnr</code> process to verify the listener process is running. (Default) ■ In the detail monitoring mode, the agent uses the <code>lsnrctl status \$LISTENER</code> command to verify the status of the Listener process.
Clean	Scans the process table for <code>tnslsnr \$Listener</code> and kills it.
Action	<p>Performs predefined actions on a resource.</p> <p>See “Action entry point” on page 15.</p>

How the agent makes Oracle highly available

The Veritas High Availability Agent for Oracle continuously monitors the Oracle database and listener processes to verify they are functioning properly. The agent provides two levels of application monitoring:

- **Primary or Basic monitoring**
In the Basic monitoring mode, with the default Process check option, the agent verifies that the Oracle and listener processes are present in the process table. Process check cannot detect whether processes are in a hung or stopped states.
See [“Monitor options for Oracle agent”](#) on page 13.
- **Secondary or Detail monitoring**
In the Detail monitoring mode, the agent runs a perl script that executes commands against the database and listener to verify their status.

The agent detects application failure if the monitoring routine reports that the Oracle or listener processes are not functioning properly. When this occurs, the Oracle service group fails over to another node in the cluster, thus ensuring high availability for the Oracle services and the database.

How the agent handles Oracle error codes during detail monitoring

The Veritas High Availability Agent for Oracle handles Oracle errors during detail monitoring. The agent classifies Oracle errors according to their severity and associates predefined actions with each error code.

The agent includes a reference file called oraerror.dat, which lists Oracle errors and the action to be taken when the error is encountered.

The file stores information in the following format:

```
Oracle_error_string:action_to_be_taken
```

For example:

```
01035:WARN
01034:FAILOVER
```

[Table 1-8](#) lists the predefined actions that the agent takes when an Oracle error is encountered.

Table 1-8 Predefined agent actions for Oracle errors

Action	Description
IGNORE	Ignores the error.
UNKNOWN	Marks the resource state as UNKNOWN and sends a notification if the Notifier resource is configured. See the <i>Veritas Cluster Server User's Guide</i> for more information about VCS notification. This action is typically associated with configuration errors or program interface errors.
WARN	Marks the resource state as ONLINE and sends a notification if the Notifier resource is configured. This action is typically associated with errors due to exceeded quota limits, session limits/restricted sessions so on.
FAILOVER (Default)	Marks the resource state as OFFLINE. This faults the service group, which fails over to the next available system. This is agent's default behavior. If the file oraerror.dat is not available, the agent assumes this default behavior for every Oracle error encountered.
NOFAILOVER	Freezes the service group temporarily and marks the resource state as OFFLINE. The agent also sends a notification if the Notifier resource is configured. This action is typically associated with errors that are not system-specific. For example, if a database does not open from a node due to corrupt Oracle files, failing it over to another node will not help.

Typical Oracle configuration in a VCS cluster

A typical Oracle configuration in a VCS cluster has the following characteristics:

- VCS is configured in a two-node cluster.
- The Oracle data is installed on shared storage.
- The Oracle binaries is installed locally on both nodes or on shared disks.
- The Veritas High Availability Agent for Oracle is installed on both nodes.

Figure 1-1 depicts a configuration where Oracle binaries and data are installed completely on shared disks.

Figure 1-1 Oracle binaries and data on shared disks

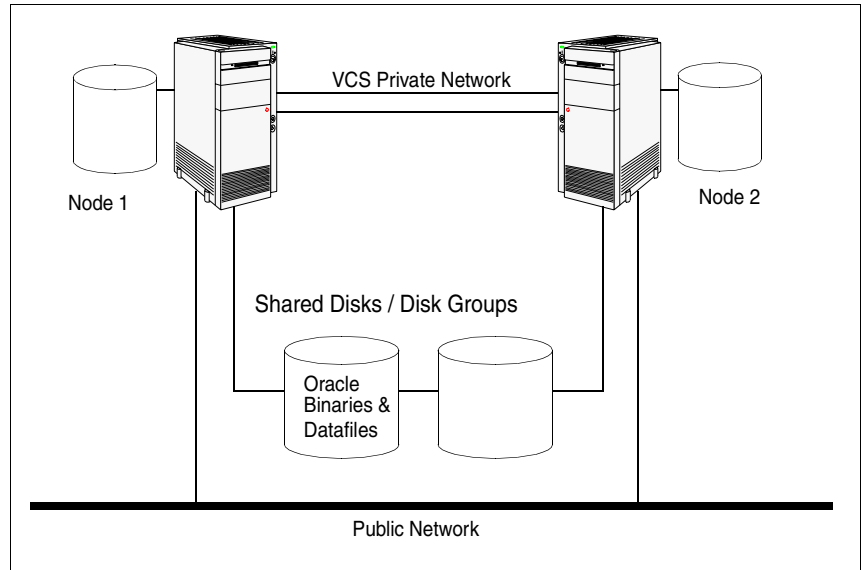
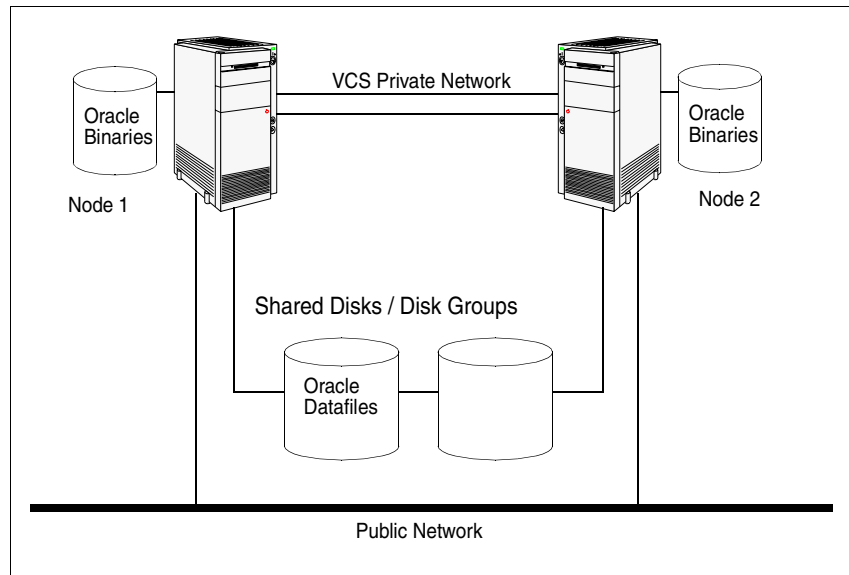


Figure 1-2 depicts a configuration where Oracle binaries are installed locally on each node and Oracle data is installed on shared disks.

Figure 1-2 Oracle binaries on local disk and Oracle data on shared disk



About setting up an Oracle cluster

Tasks involved in setting up Oracle in a VCS environment include:

- Setting up a VCS cluster
Refer to *Veritas Cluster Server Installation Guide* for more information on installing and configuring VCS.
- [Installing and configuring Oracle](#)
- [Installing the agent for Oracle](#)
- [Configuring VCS service groups for Oracle](#)

Installing and configuring Oracle

This chapter contains the following topics:

- [Before you install Oracle](#)
- [VCS requirements for installing Oracle](#)
- [Installing Oracle in a VCS environment](#)

Before you install Oracle

- Verify that VCS is installed on all nodes in the cluster.
- Verify that all nodes in the cluster have adequate resources to run Oracle and VCS.
- Verify that the network supports the TCP/IP protocol.
- Make sure that you meet the VCS requirements for installing Oracle. See “[VCS requirements for installing Oracle](#)” on page 21.

VCS requirements for installing Oracle

Review the requirements before you install Oracle in a VCS cluster on Linux and make sure that you meet the requirements.

Linux configuration

Make sure the operating system configuration requirements are met for nodes on which you want to install Oracle9*i* or later.

Disk partitions

Create partitions on the OS disk on which the Oracle database will be deployed. [Table 2-9](#) specifies the recommended size for each partition.

Table 2-9 Recommended disk partition size

Partition	Minimum recommended size
/tmp	1 GB
/usr	3 GB
swap	at least twice the physical memory
/home	2 GB

Linux package group

Symantec recommends you to select the **Software Development package** option when installing Linux. This is required for relinking Oracle at install time and to take advantage of some Advanced Server features.

Shared memory

Set the shared memory at the time of database creation. [Table 2-10](#) presents the information that you must use to set the shared memory.

Table 2-10 Recommended semaphore value

Semaphore	Recommended value
SHMMAX	Set the value such that Oracle will allocate the SGA in one segment. For example, for a system with 4 GB RAM, the suggested value of SHMMAX is 2 GB. <code>echo "2147483648" > /proc/sys/kernel/shmmax</code>
SHMMNI	Set this parameter to at least 4096. <code>echo "4096" > /proc/sys/kernel/shmmni</code>
SHMALL	Set this parameter to at least 2097152. <code>echo "2097152" > /proc/sys/kernel/shmall</code>

Swap size

Set the swap size to at least twice the physical memory size.

Semaphores

Set values for various semaphore variables in the `/proc/sys/kernel/sem` file. [Table 2-11](#) specifies the semaphore and the recommended values for Oracle9i and Oracle 10g.

Table 2-11 Recommended semaphore value

Semaphore	Recommended value	Oracle9i	Oracle 10g
SEMMSL	The largest PROCESSES parameter of any Oracle database plus 10.	100	250
SEMMNS	The sum of the PROCESSES parameter for each Oracle database, adding the largest one twice, and then adding an additional 10 for each database.	256	32000
SEMOPM	The maximum number of operations per semop call.	100	100
SEMMNI	The maximum number of semaphore sets in the entire system.	100	128

File handles

Set the value of File Handles to at least 65536.

```
echo "65536" > /proc/sys/fs/file-max
ulimit -n 65536
```

JDK requirements for Linux

Make sure that Sun's JDK 1.3.1_02 or Blackdown's JDK 1.1.8 is installed on the system.

Location of the \$ORACLE_HOME

You can place the Oracle home directory (`$ORACLE_HOME`), which contains the Oracle binaries and configuration files, locally on each server's disk. Alternatively, you can place the Oracle home directory on the shared storage. The correct location for Oracle binaries depends on your environment. The following points discuss the advantages of each approach.

`$ORACLE_HOME` directory on shared disks

When the Oracle Database Server (`$ORACLE_HOME`) is installed on shared disks, each node in the cluster must have the same mount point directory for the shared file system. Placing Oracle binaries on shared storage simplifies setting up a given node in a cluster to run an instance. Each database service group is

self-contained. An instance can be moved to a new node in the cluster that shares the storage.

For example, in a cluster with four nodes, it is possible to have three database instances, or service groups, each at a different version of Oracle. If the Oracle binaries are placed on shared storage, three copies of Oracle, that is, one per version are required on shared storage. By contrast, placing the Oracle binaries on local storage, would require as many as 12 copies of the binaries (three versions on four nodes).

The disadvantage of this approach is that a rolling upgrade of Oracle binaries on shared storage is not possible.

\$ORACLE_HOME directory on the local disk

Installing Oracle Database Server (\$ORACLE_HOME) on the local disk has the advantage of allowing an upgrade of the Oracle database binaries on an offline node while the database server runs on another node. The database server can later be switched to the upgraded node (provided the database is compatible), permitting a minimum amount of downtime.

The disadvantage of this approach is that with a large number of nodes, it becomes difficult to maintain the various Oracle installations.

Multiple Oracle instances (SIDs)

If multiple Oracle instances are defined in a single cluster configuration, the parameter file for each instance must be accessible on all nodes listed in the service group's SystemList attribute.

Location of database tablespaces

If the tablespaces are created using regular (UFS or VxFS) files, the file systems containing these files must be located on shared disks. Create the same file system mount points on each node.

If you are using raw devices on shared disks for Oracle tablespaces, change the ownership to Oracle dba user and the permissions, or access mode, to 660 on the raw devices storing the Oracle data.

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

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

Note: The user **oracle** and the group **dba** must be local and not Network Information Service (NIS and NIS+) users

Transparent listener failover

To enable Oracle Server clients to reconnect after a node switch without reconfiguring, include at least one IP resource in the service group for the Oracle resource. The hostname mapping the IP address of this resource must be used for the Host field in the file `$TNS_ADMIN/listener.ora`.

If you are using the TCP/IP protocol for Oracle client/server communication, verify that the file `/etc/services` on each node defined in the service group's SystemList attribute contains entries for the service name of the Oracle Net Service.

Hot backup of Oracle database and VCS

If a node in the cluster fails while running a hot backup of an Oracle database, the failover to another node can succeed only if the Oracle resource attribute, `AutoEndBkup`, is set to a non-zero value.

Note: You must set the Startup option of Oracle agent as `STARTUP` or `STARTUP_FORCE` to use the hot backup feature in a VCS environment. See “[Startup and shutdown options](#)” on page 12.

See “[Resource type definitions](#)” on page 75.

Otherwise, the database in the backup mode on the failover node cannot be opened and VCS cannot online the Oracle resource group.

The following errors are displayed to indicate this condition:

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

or

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

Before VCS can online the Oracle resource group on the failover node, you must take the datafiles in the database out of the backup mode and shut down the database so that it can be reopened. Refer to the Oracle documentation for instructions on how to change the state of the database files.

Installing Oracle in a VCS environment

The strategy for installing Oracle into a VCS cluster is aimed at ensuring that installations on all nodes in the cluster are uniform. For more information on installing Oracle, refer to the Oracle documentation on Linux.

You can install Oracle in the following two ways in a VCS environment:

<code>\$ORACLE_HOME</code> on the shared disk	Oracle binaries and Oracle data are installed on shared disks.
---	--

\$ORACLE_HOME on the local disk

Oracle binaries are installed locally on each node and Oracle data is installed on shared disks.

Note that Oracle data includes the datafiles, control files, redo log files, and archive log files.

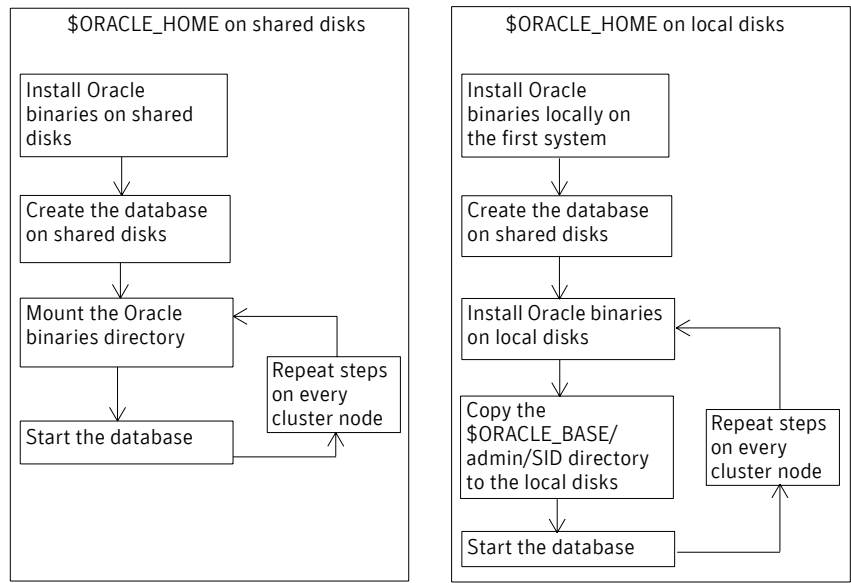
When installing Oracle, ensure that the login_id, id_name, group_id, and group_name for the Oracle owner is the same on all the nodes. The user oracle and the group dba must be local and not Network Information Service (NIS and NIS+) users.

Overview of Oracle installation tasks

Tasks to complete Oracle installation in a VCS cluster depend on whether you want the \$ORACLE_HOME on shared disk or local disk.

Figure 2-3 illustrates the flow of the installation when the \$ORACLE_HOME is on shared disk and on local disk.

Figure 2-3 Comparison of the installation flow



Installation tasks for \$ORACLE_HOME on shared disks

Table 2-12 lists the tasks to install Oracle such that \$ORACLE_HOME is on shared disks. The methodology is to install the Oracle binaries and to create the

Oracle database on shared disks during the first installation. Mount the Oracle binaries directory and verify that the database can be started from all nodes.

Table 2-12 Installation tasks for \$ORACLE_HOME on shared disks

Task	Reference
From any node in the cluster, install Oracle binaries on shared disks.	See “Installing Oracle binaries” on page 28.
From the node where you installed Oracle, create the database on shared disks.	See “Configuring the Oracle database” on page 30.
<ul style="list-style-type: none"> ■ From each node that would be a part of the Oracle cluster, verify access to the database on shared disks. ■ On any node, start the Oracle database. 	See “Starting the database” on page 32.

Installation tasks for \$ORACLE_HOME on local disks

[Table 2-13](#) lists the tasks to install Oracle such that \$ORACLE_HOME is on local disks. The methodology is to install the Oracle binaries on the local disk and to create the Oracle database on shared disks during the first installation. Then, install the Oracle binaries on local disks of other nodes. This ensures that all Oracle installations are identical and access the database from the same location on the shared disk.

Table 2-13 Installation tasks for \$ORACLE_HOME on local disks

Task	Reference
On the first node in the cluster, install Oracle binaries on the local disk.	See “Installing Oracle binaries” on page 28.
From the first node where you installed Oracle, create the database on shared disks.	See “Configuring the Oracle database” on page 30.
On each node in the cluster, perform the following tasks: <ul style="list-style-type: none"> ■ Install Oracle binaries on the local disk. ■ Copy the \$ORACLE_BASE/admin/SID directory to the local disk. 	<ul style="list-style-type: none"> ■ See “Installing Oracle binaries” on page 28. ■ See “Copying the \$ORACLE_BASE/admin/SID directory” on page 32.
On any nodes, start the Oracle database.	See “Starting the database” on page 32.

Installing Oracle binaries

Install a version of Oracle that the Veritas High Availability Agent for Oracle supports. Review the procedure to install Oracle9i or Oracle 10g for VCS. Note that the screens and options presented by installers for other Oracle versions may differ.

To install Oracle9i for VCS

- 1 Insert the Oracle CD.
- 2 Set the *DISPLAY* variable, mount the CD drive, and run the Oracle installer as Oracle User.
`/mnt/cdrom/runInstaller`
- 3 Read the Welcome screen and click **Next**.
- 4 If this is the first Oracle installation, the Inventory Location dialog box is displayed. Specify the location of your base directory and click **OK**.
 - If in your installation, the \$ORACLE_HOME will be on local disks, specify a location on the local disk.
 - If the \$ORACLE_HOME will be on shared disks, specify a location on the shared disk.
- 5 On the UNIX Group Name dialog box, enter the name of a UNIX group that will have permissions to update the Oracle software and click **Next**.
- 6 The Oracle installer displays a message asking you to run the script `/tmp/orainstRoot.sh`. Run the script and click **Continue** to return to the Oracle installer.
- 7 On the File Locations dialog box, enter or select a name and a path for the Oracle Home and click **Next**.
- 8 On the Available Products dialog box, select the **Oracle9i Database** option and click **Next**.
- 9 On the Installation Types dialog box, select your installation type and click **Next**.
- 10 On the Database Configuration dialog box, select the **Software Only** option and click **Next**.
- 11 On the Summary dialog box, review your selections. Click **Back** to make changes. Otherwise click **Install**.
- 12 Click **Next** after the installation is complete.
- 13 On the End of Installation dialog box, click **Exit**.
- 14 Proceed to configure the database on shared disks.
See “[Configuring the Oracle database](#)” on page 30.

To install Oracle 10g for VCS

- 1 Insert the Oracle CD.
- 2 Set the *DISPLAY* variable, mount the CD drive, and run the Oracle installer as Oracle User.

```
# /mnt/cdrom/runInstaller -ignoreSysPrereqs
```
- 3 Read the Welcome screen and click **Next**.
- 4 If this is the first Oracle installation, the Inventory Location dialog box is displayed. Specify the location of your base directory and click **OK**. Do one of the following:
 - If in your installation, the \$ORACLE_HOME will be on local disks, specify a location on the local disk.
 - If the \$ORACLE_HOME will be on shared disks, specify a location on the shared disk.
- 5 The Oracle installer displays a message asking you to run the script /tmp/orainstRoot.sh. Run the script and click **Continue** to return to the Oracle installer.
- 6 In the File Locations dialog box, enter or select a name and a path for the Oracle Home and click **Next**.
- 7 In the Installation Types dialog box, select your installation type and click **Next**.
- 8 In the Select Database Configuration dialog box, select the **Do not create a starter database** option and click **Next**.
- 9 In the Summary dialog box, review your selections. Click **Back** to make changes. Otherwise click **Install**.
- 10 Click **Next** after the installation is complete.
- 11 In the End of Installation dialog box, click **Exit**.
- 12 Disable the Oracle 10g clustering daemon.
See [“Disabling the Oracle 10g clustering daemon”](#) on page 29.
- 13 Proceed to configure the database on shared disks.
See [“Configuring the Oracle database”](#) on page 30.

Disabling the Oracle 10g clustering daemon

If you installed Oracle 10g on shared disks, you must disable the Oracle clustering daemon.

Oracle 10g provides a clustering daemon called Oracle Cluster Synchronization Service Daemon (CSSD). If the Oracle binary files are on a shared storage, the

`init` command to start the daemon may result in an error. Because a VCS cluster for Oracle does not require this daemon, Symantec recommends you to disable the daemon.

To disable the daemon on Linux

- ◆ Remove the following line from the `/etc/inittab` file on the node from where you ran the Oracle installation wizard:

```
h1:23:respawn:/etc/init.d/init.cssd run >/dev/null 2>&1 >
</dev/null
```

Configuring the Oracle database

Configure an Oracle database on shared disks using the Database Configuration Assistant. You need to configure the database only once, from the node on which you installed Oracle first. Review the procedure to configure Oracle9i or Oracle 10g database.

To configure the Oracle9i database

- 1 Set the `DISPLAY` variable and start the Oracle Database Configuration Assistant as Oracle User.
`dbca`
- 2 Read the Welcome screen, click **Next**.
- 3 On the Operations dialog box, select the **Create a database** option and click **Next**.
- 4 On the Database Templates dialog box, select a template to create the database and click **Next**.
- 5 On the Database Identification dialog box, enter or select a unique name and SID for the global database and click **Next**.
- 6 On the Database Connection Options dialog box, select the **Dedicated Server Mode** or the **Shared Server Mode** and click **Next**.
- 7 On the Initialization Parameters dialog box, specify the locations of the archive logs and the initialization parameter files.
 - If you want to enable archive logs for the database, click the **Archive** tab and select the **Archive Log Mode** check box. In the Archive Log Destination(s) list, enter a path on the shared disk. This ensures that the Archive Log will be created on the shared disk.
 - Click the **File Locations** tab.
 - If installing Oracle locally on all systems, make sure that the initialization parameter file and the trace file directories are on the local disk.

- Modify other parameters, if desired.
 - Click **Next**.
- 8 On the Database Storage dialog box, specify the locations of the Control, Data, and Redo log files.
 - In the left pane, expand the **Storage** folder by clicking the + next to it.
 - Click **Controlfile** and click the **General** tab in the right pane.
 - In the **File Directory** field, enter a path on the shared disk for each control file.
 - Expand the **Datafiles** folder by clicking the + next to it.
 - Select the datafile in the left pane and click the **General** tab in the right pane. In the **Name** field, enter a path on the shared disk for each datafile.
 - Expand the **Redo Log Groups** folder by clicking the + next to it.
 - Select the Redo Log file in the left pane. In the **File Directory** field in the right pane, enter the path of a directory on the shared disk for each Redo Log file. Specify file sizes for the Redo Log files, if desired.
 - Click **Next**.
 - 9 On the Creation Options dialog box, select the **Create Database** check box and click **Finish**.
 - 10 On the Summary dialog box, review your selections and click **OK**.

To configure the Oracle 10g database

- 1 Set the *DISPLAY* variable and start the Oracle Database Configuration Assistant as Oracle User.
dbca
- 2 Read the Welcome screen, click **Next**.
- 3 In the Operations dialog box, select the **Create a database** option and click **Next**.
- 4 In the Database Templates dialog box, select a template to create the database and click **Next**.
- 5 In the Database Identification dialog box, enter or select a unique name and SID for the global database and click **Next**.
- 6 Follow the wizard instructions and select other options.
- 7 In the Database File Locations dialog box, specify a location on the shared disk for the database files to be created and click **Next**.
- 8 In the Recovery Configuration dialog box, specify the **Flash Recovery Area** on the shared disk and click **Next**.

- 9 Follow the wizard instructions and complete creating the database.

Copying the \$ORACLE_BASE/admin/SID directory

Follow the instruction on each node in the cluster, only if the \$ORACLE_HOME is on local disks.

To copy the \$ORACLE_BASE/admin/SID directory

- ◆ Copy the directory \$ORACLE_BASE/admin/*SID* from the shared disk to the local disk.

The variable *SID* represents the database instance.

Refer to Oracle's OFA Structure for more information.

Starting the database

Depending on the location of your \$ORACLE_HOME, perform one of the procedures to start the database.

You can verify access to the database by running an SQL query.

Starting the database when your \$ORACLE_HOME is on shared disks

Verify that you can access the shared database from each node. Perform the steps from each node that would be a part of the Oracle cluster.

To start the database when \$ORACLE_HOME is on shared disks

- 1 If the Oracle binaries are mounted on any other node, unmount it.
- 2 If the data files on shared disk are mounted on any node, unmount it.
- 3 Mount the Oracle binaries and data files.
- 4 Start the database.

Starting the database when your \$ORACLE_HOME is on local disks

Start the database individually on each node that would be a part of the Oracle cluster.

To start the database when \$ORACLE_HOME is on local disks

- 1 Make sure you have installed Oracle binaries on each node in the cluster.
- 2 If the data files on shared disk are mounted on any node, unmount it.
- 3 Mount the data files.

- 4 Start the database.

Installing the agent for Oracle

This chapter contains the following topics:

- [Before you install or upgrade the agent for Oracle](#)
- [Installing the agent software](#)
- [Upgrading the agent](#)

Before you install or upgrade the agent for Oracle

Meet the prerequisites to install or upgrade the Veritas High Availability Agent for Oracle.

- Make sure that VCS is installed in the cluster.
Symantec recommends installing the VCS graphical user interface. Refer to the *Veritas Cluster Server Installation Guide*.
- Verify that Oracle Server for Linux and the add-on Listener are installed and configured. For more information, see the Oracle documentation.
See [“VCS requirements for installing Oracle”](#) on page 21.
See [“Installing Oracle in a VCS environment”](#) on page 25.

Installing the agent software

You can install the Veritas High Availability Agent for Oracle from the product disc. You must install the Oracle enterprise agent on all nodes that will host the Oracle service group.

To install the agent on a Linux node

- 1 Log in as superuser.

- 2 Insert the software disc into a drive connected to the node.
 The software automatically mounts the disc as /mnt/cdrom.
- 3 If the disc does not automatically mount, then you must mount the disc manually. For example:

```
# mount -o ro /dev/cdrom /mnt/cdrom
```
- 4 Install the agent software.

```
RHEL4 (i686, x86_64) # cd /mnt/cdrom/rhel4_i686\
                    /cluster_server_agents/oracle_agent/rpms
                    # rpm -i VRTSvcsor-5.0.00.0-GA_RHEL4.i686.rpm
                    # rpm -i \
                    VRTScsocw-5.0.00.0-GA_GENERIC.noarch.rpm

SLES9 (i586, x86_64) # cd /mnt/cdrom/sles9_i586\
                    /cluster_server_agents/oracle_agent/rpms
                    # rpm -i VRTSvcsor-5.0.00.0-GA_SLES9.i586.rpm
                    # rpm -i \
                    VRTScsocw-5.0.00.0-GA_GENERIC.noarch.rpm
```

This installs the following RPMs:

- VRTSvcsor, the agent binaries
 - VRTScsocw, the agent configuration wizard
- 5 Repeat [step 1](#) through [step 4](#) on each node that will become part of the Oracle service group.

Upgrading the agent

You can upgrade Veritas High Availability Agent for Oracle from versions 4.0 and 4.1 in a VCS cluster.

Upgrading VCS agent 4.0 and 4.1 for Oracle

Perform the following steps on each node of the VCS cluster.

To upgrade VCS agent for Oracle to version 5.0

- 1 Stop VCS locally.

```
# hastop -local -force
```
- 2 Remove the VCS enterprise agent 4.0 for Oracle.

```
# pkgrm VRTSvcsor
# pkgrm VRTScsocw
```
- 3 Install the VCS enterprise agent 5.0 for Oracle.
 See “[Installing the agent software](#)” on page 35.

- 4 Copy the file OracleTypes.cf from the directory /etc/VRTSagents/ha/conf/Oracle to the /etc/VRTSvcs/conf/config directory.
- 5 Restart the VCS engine.

```
# hastart
```


Configuring VCS service groups for Oracle

This chapter contains the following topics:

- [About configuring a service group for Oracle](#)
- [About Oracle configurations in VCS](#)
- [Before you configure the service group](#)
- [Configuring the service group](#)
- [Additional considerations for basic monitoring](#)
- [Setting up detail monitoring](#)

About configuring a service group for Oracle

Configuring the Oracle service group involves creating the Oracle service group, its resources, and defining attribute values for the configured resources. You must have administrator privileges to create and configure a service group.

VCS provides several ways to configure an enterprise agent:

- Using the agent configuration wizard
 - Using the cluster manager (Java console)
 - Using the command-line
- See [“Configuring the service group”](#) on page 46.

About Oracle configurations in VCS

You can set up Oracle in different ways in a VCS environment. [Table 4-14](#) lists the different Oracle configurations you can set up in a VCS environment for

high availability. Configuring Oracle for VCS involves configuring the Oracle files listener.ora and tnsnames.ora as per VCS requirements.

Table 4-14 Oracle configurations in VCS

Configuration type	Reference
Single Oracle instance configuration	See “Configuring single Oracle instance in VCS” on page 40.
Multiple Oracle instances (single listener) configuration	See “Configuring multiple Oracle instances (single listener) in VCS” on page 41.
Multiple Oracle instances (multiple listeners) configuration	See “Configuring multiple Oracle instances (multiple listeners) in VCS” on page 42.
Configuration with shared server support	See “Configuring Oracle instance with shared server support in VCS” on page 43.

See [“Best practices for multiple Oracle instance configurations”](#) on page 101.

Configuring single Oracle instance in VCS

Review the resource dependency graph and sample configuration of service groups involving a single Oracle instance.

See [“Sample single Oracle instance configuration”](#) on page 85.

To configure an Oracle instance in VCS

- 1 Review the Oracle and Netlsnr resource types and their attribute definitions. See [“Resource type definitions”](#) on page 75.
- 2 Configure the Oracle file tnsnames.ora as per VCS requirements. The changes required in the file depends on your Oracle configuration. For clients to connect to the failover instance, in the file tnsnames.ora located at \$TNS_ADMIN, change the host name for all TCP protocol address databases to the virtual IP address for the service group. The following example assumes that the host name for the database is set to oraprod, which represents the virtual IP address for the service group.

```
prod =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP) (HOST = oraprod) (PORT = 1521))
    )
    (CONNECT_DATA =
      (SERVICE_NAME = prod)
    )
  )
```


- 3 Configure the Oracle file listener.ora as per VCS requirements. The changes required in the file depends on your Oracle configuration.

In the file listener.ora located at \$TNS_ADMIN, edit the “Host=” line in the ADDRESS_LIST section and add the name of the high availability address for the service group, in this case, oraprod.

```
LISTENER_PROD =  
(DESCRIPTION_LIST =  
(DESCRIPTION =  
(ADDRESS_LIST =  
(ADDRESS = (PROTOCOL = TCP) (HOST = oraprod) (PORT = 1521))  
)  
)  
)
```

- 4 Create the Oracle service groups using the agent configuration wizard, Cluster Manager (Java Console), or the command-line.

See “[Configuring the service group](#)” on page 46.

- 5 Bring the Oracle service group online.

See “[Bringing the service group online](#)” on page 65.

Configuring multiple Oracle instances (single listener) in VCS

Review the resource dependency graph and sample configuration of a service group involving multiple Oracle instances.

See “[Sample multiple Oracle instances \(single listener\) configuration](#)” on page 88.

To configure multiple Oracle instances (single listener)

- 1 Review the Oracle and Netlsnr resource types and their attribute definitions. See “[Resource type definitions](#)” on page 75.

- 2 Configure the Oracle file tnsnames.ora as per VCS requirements. The changes required in the file depends on your Oracle configuration. For clients to connect to the failover instance, in the file tnsnames.ora located at \$TNS_ADMIN, change the host name for all TCP protocol address databases to the virtual IP address for the service group.

The following example assumes that the host name for the database is set to oraprod, which represents the virtual IP address for the service group.

```
prod =  
(DESCRIPTION =  
(ADDRESS_LIST =  
(ADDRESS = (PROTOCOL = TCP) (HOST = oraprod) (PORT = 1521))  
)  
(CONNECT_DATA =  
(SERVICE_NAME = prod)  
)  
)
```

```
prod =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP) (HOST = oramktg) (PORT = 1522))
    )
    (CONNECT_DATA =
      (SERVICE_NAME = mktg)
    )
  )
```

- 3 Configure the Oracle file listener.ora as per VCS requirements. The changes required in the file depends on your Oracle configuration.

In the file listener.ora located at \$TNS_ADMIN, edit the “Host=” line in the ADDRESS_LIST section and add the name of the high availability address for the service group, in this case, oraprod.

```
LISTENER_ORACLE =
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (ADDRESS_LIST =
        (ADDRESS = (PROTOCOL = TCP) (HOST = oraprod) (PORT = 1521))
        (ADDRESS = (PROTOCOL = TCP) (HOST = oramktg) (PORT = 1522))
      )
    )
  )
```

- 4 Create the Oracle and Listener service groups using Cluster Manager (Java Console.) You can also use the command-line to create the service group. See [“Configuring the service group”](#) on page 46.
- 5 Bring the Oracle service group online. See [“Bringing the service group online”](#) on page 65.

Configuring multiple Oracle instances (multiple listeners) in VCS

Review the resource dependency graph and sample configuration of a service group involving multiple Oracle instance.

See [“Sample multiple instance \(multiple listeners\) configuration”](#) on page 93.

To configure multiple Oracle instances (multiple listeners)

- 1 Review the Oracle and Netlsnr resource types and their attribute definitions. See [“Resource type definitions”](#) on page 75.
- 2 Configure the Oracle file tnsnames.ora as per VCS requirements. The changes required in the file depends on your Oracle configuration. For clients to connect to the failover instance, in the file tnsnames.ora located at \$TNS_ADMIN, change the host name for all TCP protocol address databases to the virtual IP address for the service group.

```
prod =
  (DESCRIPTION =
    (ADDRESS_LIST =
```

```

        (ADDRESS = (PROTOCOL = TCP) (HOST = oraprod) (PORT = 1521))
    )
    (CONNECT_DATA =
        (SERVICE_NAME = prod)
    )
)
mktg =
    (DESCRIPTION =
        (ADDRESS_LIST =
            (ADDRESS = (PROTOCOL = TCP) (HOST = oramktg) (PORT = 1522))
        )
        (CONNECT_DATA =
            (SERVICE_NAME = mktg)
        )
    )
)

```

- 3 Configure the Oracle file listener.ora as per VCS requirements. The changes required in the file depends on your Oracle configuration.

In the file listener.ora, create independent configurations for each listener.

```

LISTENER_PROD =
    (DESCRIPTION_LIST =
        (DESCRIPTION =
            (ADDRESS_LIST =
                (ADDRESS = (PROTOCOL = TCP) (HOST = oraprod) (PORT = 1521))
            )
        )
    )

LISTENER_MKTG =
    (DESCRIPTION_LIST =
        (DESCRIPTION =
            (ADDRESS_LIST =
                (ADDRESS = (PROTOCOL = TCP) (HOST = oramktg) (PORT = 1522))
            )
        )
    )

```

- 4 Create the Oracle service groups using the agent configuration wizard, Cluster Manager (Java Console,) or the command-line.
See [“Configuring the service group”](#) on page 46.
- 5 Bring the Oracle service group online.
See [“Bringing the service group online”](#) on page 65.

Configuring Oracle instance with shared server support in VCS

Review the resource dependency graph and sample configuration.

See [“Sample Oracle configuration with shared server support”](#) on page 97.

To configure Oracle with shared server support

- 1 Review the Oracle and Netlsnr resource types and their attribute definitions.

See “[Resource type definitions](#)” on page 75.

- 2 Configure the Oracle file tnsnames.ora as per VCS requirements. The changes required in the file depends on your Oracle configuration.
 For clients to connect to the failover instance, in the file tnsnames.ora located at \$TNS_ADMIN, change the host name for all TCP protocol address databases to the virtual IP address for the service group.
 The following example assumes that the host name for the database is set to oraprod, which represents the virtual IP address for the service group.

```
prod =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP) (HOST = oraprod) (PORT = 1521))
    )
    (CONNECT_DATA =
      (SERVICE_NAME = prod)
    )
  )
```

- 3 Configure the Oracle file listener.ora as per VCS requirements. The changes required in the file depends on your Oracle configuration.
 In the file listener.ora located at \$TNS_ADMIN, edit the “Host=” line in the ADDRESS_LIST section and add the name of the high availability address for the service group, in this case, oraprod.

```
LISTENER_PROD =
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (ADDRESS_LIST =
        (ADDRESS = (PROTOCOL = TCP) (HOST = oraprod) (PORT = 1522))
      )
    )
  )
```

- 4 In the initialization parameter file, for the dispatchers parameter, set the host name for all TCP protocol address dispatchers to the virtual IP address for the service group.

In the following example, the host name for the dispatcher is set to oraprod.

```
dispatchers =
  "(ADDRESS = (PROTOCOL = TCP) (HOST = oraprod))
  (MUL=ON) (TICK=15) (POOL=true) "
```

- 5 In the initialization parameter file, set the LOCAL_LISTENER attribute to a listener alias, so that the Oracle PMON process registers information with the listener.

```
local_listener=listener_alias
```

This listener alias gets appended by the default domain set in the file sqlnet.ora.

- 6 In the file tnsnames.ora, create an entry for the listener alias to resolve the address information. In the address parameter, set the host name to the

virtual IP address for the service group, without the CONNECT_DATA portion of the connect descriptor.

In the following example, the listener *listener_alias* uses TCP/IP port 1521 and the virtual IP address for the service group is oraprod.

```
listener_alias=  
  (address = (PROTOCOL = TCP) (HOST = oraprod)  
    (PORT= 1521))
```

- 7 Create the Oracle service groups using the agent configuration wizard, Cluster Manager (Java Console,) or the command-line.
See “[Before you configure the service group](#)” on page 45.
- 8 Bring the Oracle service group online.
See “[Bringing the service group online](#)” on page 65.

Before you configure the service group

Before you configure the Oracle service group, you must:

- Verify that VCS is installed and configured on all nodes in the cluster where you will configure the service group.
Refer to the *Veritas Cluster Server Installation Guide* for more information.
- Verify that Oracle is installed and configured identically on all nodes in the cluster.
See “[Installing and configuring Oracle](#)” on page 21.
- Verify that the Veritas High Availability Agent for Oracle is installed on all nodes in the cluster.
See “[Installing the agent for Oracle](#)” on page 35.
- Verify that the type definition for Veritas High Availability Agent for Oracle is imported into the VCS engine.
See “[Importing the OracleTypes.cf file](#)” on page 45.
- If you have installed Oracle 10g, make sure that the Oracle services are not started on reboot by the init script.

Importing the OracleTypes.cf file

Before configuring the Veritas High Availability Agent for Oracle, you must import the OracleTypes.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 Cluster Explorer’s **File** menu, choose **Import Types**.

- 3 In the Import Types dialog box, select the `/etc/VRTSagents/ha/conf/Oracle/OracleTypes.cf` file.
- 4 Click **Import**.
- 5 Save the configuration.

To import using the command line

- 1 Log in to sysa as superuser.
- 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`, stop the VCS engine on all nodes 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 Copy the Oracle types configuration file into place:

```
# cp /etc/VRTSagents/ha/conf/Oracle/OracleTypes.cf \
/etc/VRTSvcs/conf/config/OracleTypes.cf
```
- 6 Edit the `main.cf` file to include the `OracleTypes.cf` file:

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

Configuring the service group

You can configure Oracle in a VCS environment in one of the ways that VCS supports.

See “[About Oracle configurations in VCS](#)” on page 39.

Review the following to configure the service group:

- Sample configuration files and resource dependency graphs of the Oracle service group.
See “[Sample configurations](#)” on page 85.
- Oracle resource type and the attribute definitions of the Oracle and Netlsnr agents.
See “[Resource type definitions](#)” on page 75.

VCS provides several ways to configure an enterprise agent:

- [Configuring the service group using the agent configuration wizard](#)
- [Configuring the service group using cluster manager \(Java console\)](#)

- [Configuring the service group using the command-line](#)

Configuring the service group using the agent configuration wizard

VCS provides an agent configuration wizard that guides you through the process of configuring the agent. The wizard creates and modifies Oracle service groups. You can also modify an existing service group.

See “[Modifying the service group configuration](#)” on page 66.

Make sure that you meet the following requirements:

- The Oracle instances and listeners to be configured must be running. All listeners to be configured must listen to the same virtual IP address.

Note: For the default listener, the configuration wizard requires the listener parameter file, `listener.ora`, to reside in `$ORACLE_HOME/network/admin`. No such condition applies for non-default listeners.

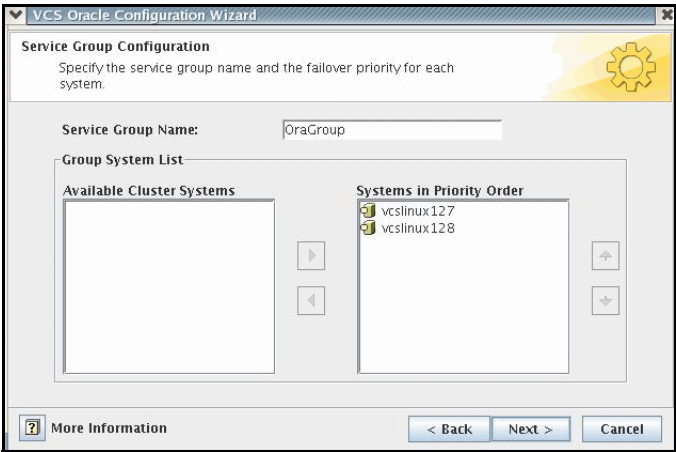
- The Oracle files (control, data, redo-log, and archive files) must be on shared disks.
- The IP addresses and host names specified in the files `listener.ora` and `tnsnames.ora` must be the same.
- If detail monitoring is to be used for a database instance, the table used for detail monitoring must be set up, with user and password assigned.
See “[Setting up detail monitoring](#)” on page 60.

To configure the service group using the agent configuration wizard

- 1 Set the `DISPLAY` variable, start the VCS Oracle configuration wizard as `root`.

```
# hawizard oracle
```
- 2 Read the information on the Welcome screen and click **Next**.
- 3 In the Wizard Options dialog box, select the **Create Oracle Service Group** option and click **Next**.

- 4
- In the Service Group Configuration dialog box, specify information about the service group.



Specify the following information:

- Service Group Name

Enter a name for the Oracle service group.
- Available Cluster Systems

Select the systems on which to configure the service group and click the right-arrow icon to move the systems to the service group's system list.
- Systems in Priority Order

■

To remove a system from the service group's system list, select the system in the **Systems in Priority Order** box and click the left arrow icon.

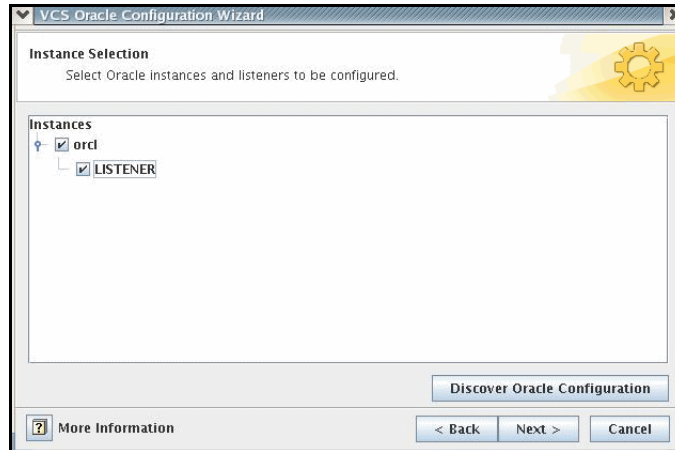
■

To change a system's priority in the service group's system list, select the system in the **Systems in Priority Order** box and click the buttons with the up and down arrow icons.

The system at the top of the list has the highest priority while the system at the bottom of the list has the lowest priority.

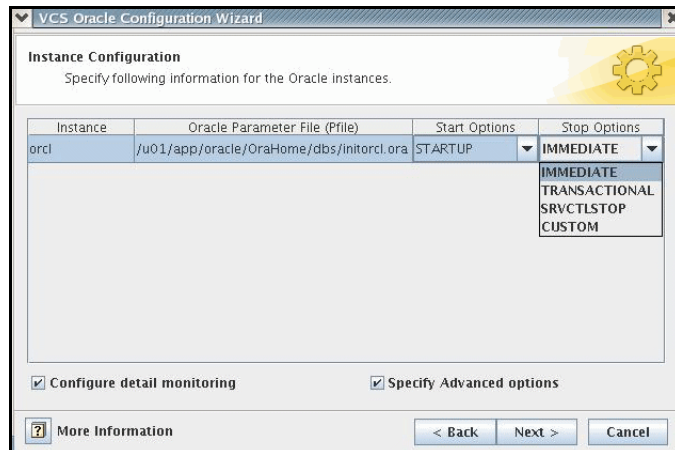
- 5
- Click **Next**.

- 6 In the Instance Selection dialog box, select the Oracle instances to be configured along with their listeners. Click **Discover Oracle Configuration**, if required.



Click **Next**.

- 7 In the Instance Configuration dialog box, specify information for all selected instances.



Specify the following information for the Oracle instances that you selected:

Oracle Parameter File (Pfile) Verify the location of the Oracle Parameter File.

- Start Options

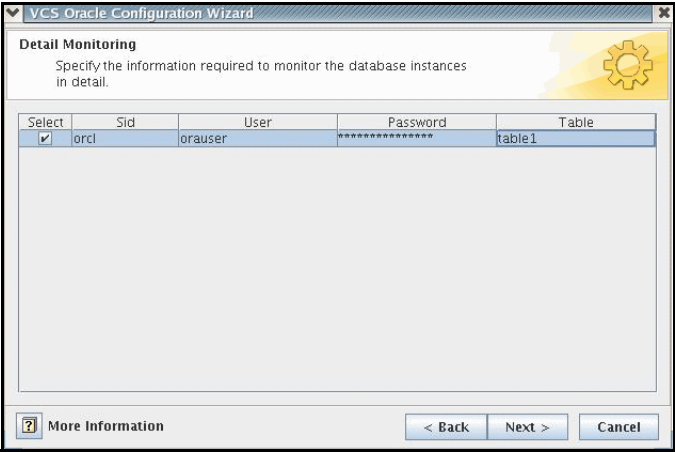
Choose the Start options, if desired. Default is STARTUP_FORCE.
See “Startup options” on page 12.
- Stop Options

Choose the Stop options, if desired. Default is IMMEDIATE.
See “Shutdown options” on page 13.
- Configure detail monitoring

Select the check box if you want to monitor the database in detail.
See “Detail monitoring” on page 14.
- Specify Advanced Options

Select the check box to enter advanced configuration information for the database instances.

- 8 Click **Next**.
- 9 If you chose to monitor the database in detail, the Detail Monitoring dialog box is displayed.



Specify the following information for the database instances that you want the agent to monitor in detail and click **Next**:

- Select

Select the check box corresponding to the database to be monitored in detail.
- User

Enter a valid user name for the database.

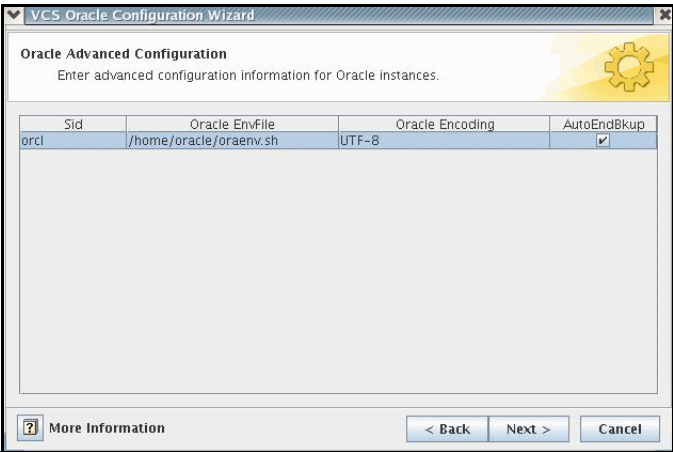
- Password

Enter a valid password for the database user.
- Note:

Do not encrypt passwords when entering them through the Agent Configuration Wizard; the wizard takes care of encrypting passwords.
- Table

Enter the name of a table that will be queried to validate the status of the database.

10 If you chose to specify advanced options, the Oracle Advanced Configuration dialog box is displayed.



Specify the following information for the Oracle instances and click **Next**:

- Oracle EnvFile

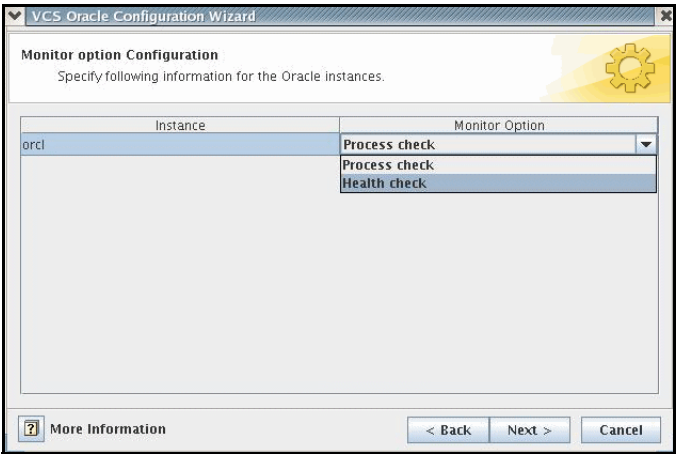
Enter the location of the Oracle Envfile.
- Oracle Encoding

Enter the encoding.
- AutoEndBkup

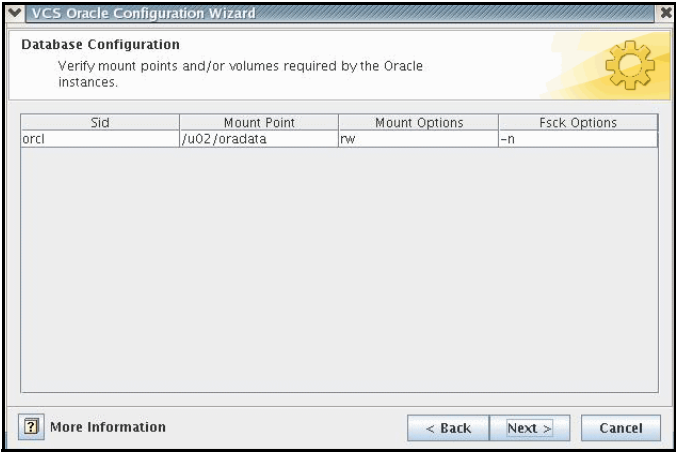
Select the check box, if desired.
See [“Hot backup of Oracle database and VCS”](#) on page 25.

11 In the Monitor option Configuration dialog box, specify the monitor option for the Oracle instances, and click **Next**.
The default monitor option is **Process check**.

See “Basic monitoring options” on page 14.



- 12 In the Database Configuration dialog box, verify the mount points required by each database, specify the Mount and Fsck options, and click **Next**.



- 13 In the Listener Configuration dialog box, configure the listeners for the databases.

Listener Configuration
 Configure the network details for the listeners.

Address: 10 . 212 . 102 . 69
 Netmask: 255 . 255 . 254 . 0

System	Device
vcslinux127	eth0
vcslinux128	eth0

☒ Enable detail monitoring ☒ Specify Advanced options

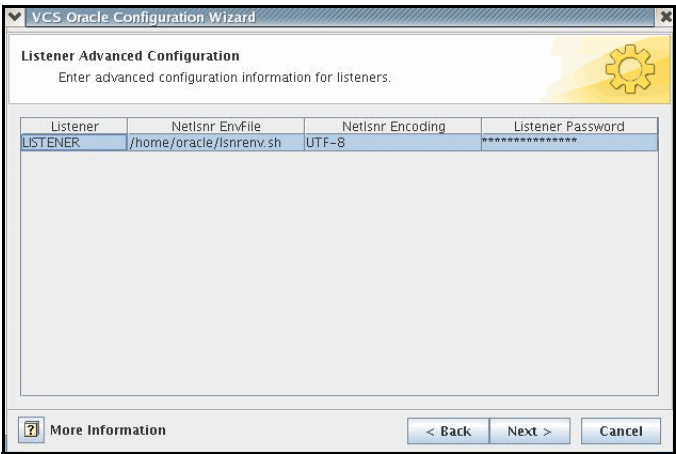
[More Information](#) < Back Next > Cancel

Specify the following information for the listeners:

- | | |
|---------------------------------|---|
| Address | Verify the virtual IP address. |
| Netmask | Verify the Netmask. |
| Device | For each system, select a device. |
| Enable detail monitoring | Select the check box to enable detail monitoring. |
| Specify Advanced Options | Select the check box to enter advanced configuration information for the listeners. |

- 14 Click **Next**.

- 15
- If you chose to specify advanced options for the listeners, the Listener Advanced Configuration dialog box is displayed.



Specify the following information for each listener and click **Next**:

- Netlsnr EnvFile

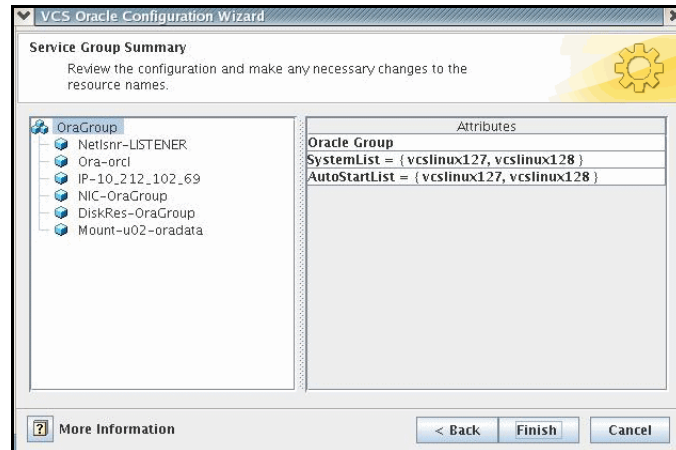
Enter the path of the listener Envfile.
- Netlsnr Encoding

Enter the encoding.
- Listener Password

Enter a valid password for the listener.
- Note:

Do not encrypt passwords when entering them through the agent configuration wizard; the wizard takes care of encrypting passwords.

- 16 In the Service Group Summary dialog, review your configuration.



Click on a resource to view its attributes and their configured values in the **Attributes** box.

- Change names of resources, if desired; the wizard assigns unique names to resources based on their respective name rules.
To edit a resource name, select the resource name and click on it, press Enter after editing each attribute.
- Review your configuration and click **Finish**.

The wizard starts running commands to create the Oracle service group. Various messages indicate the status of these commands.

- 17 In the Completing the Oracle Configuration wizard dialog box, select the **Bring the service group online** check box to bring the service group online on the local system.
- 18 Click **Close**.
The Oracle service group is created in your cluster.

Configuring the service group using cluster manager (Java console)

A template for the Oracle resource groups is automatically installed when you install the Oracle enterprise agent. Using the VCS Cluster Manager, you can view the template, which displays the Oracle service group, its resources and their attributes. You can dynamically modify the attribute values as necessary for your configuration.

Configuration tasks involve:

- Creating a new service group

- Editing the resource attributes

To create a new service group

- 1 Make sure that the Oracle type definition file `OracleTypes.cf` is imported in your configuration.
- 2 Launch the Cluster Configuration wizard. Do one of the following:
 - 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 Read the information on the Welcome screen and click **Next**.
- 4 Specify the name of the service group and the target systems on which it is to be configured.
 - Enter the name of the service group.
 - On the **Available Systems** box, select the systems on which to configure the service group.
 - 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.
 - 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.
 - Click **Next**.
- 5 In the Would you like to use a template to configure the service group? dialog box, click **Next**.
- 6 In the Select a template to configure the service group dialog box, select the OracleGroup template on which to base the service group.

If applicable, a window opens notifying you that names of some resources within the new service group are already in use. This window also includes a list of alternative names. You can select an alternative name, or modify the name. Click **Next** after resolving the name clashes.
- 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.
A window opens listing the service group's resources and their associated attributes.
- 9 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.

To edit the resource attributes

- 1 Select the resource from the list on the left pane.
The resource's attributes appear in the right pane.
- 2 Select the attribute to be modified.
See "[Resource type definitions](#)" on page 75.
- 3 Click the edit icon in the **Edit** column.
- 4 In the Edit Attribute dialog box, enter the attribute values. To modify the scope of the attribute, click the option buttons for **Global** or **Local**.
- 5 Click **OK**.
- 6 Repeat the procedure for each resource and its attributes and click **Finish**.
Edit the attributes for all the resources according to your configuration.
- 7 Follow the wizard instructions to complete the configuration.
- 8 Click **Finish** to quit the wizard.

Configuring the service group using the command-line

The Veritas High Availability Agent for Oracle comes with a sample configuration file that can be used as reference to directly modify your present configuration file. When you use this method, you must stop and restart VCS to implement the configuration.

To configure the service group using the sample configuration file

- 1 Log in to sysa as superuser.
- 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 Copy the Oracle types configuration file into place:

```
# cp /etc/VRTSagents/ha/conf/Oracle/OracleTypes.cf \
/etc/VRTSvcs/conf/config/OracleTypes.cf
```
- 6 Edit the main.cf file. The file
/etc/VRTSagents/ha/conf/Oracle/sample_main.cf can be used for reference:
 - Add an “include” line for the OracleTypes.cf file:

```
include "OracleTypes.cf"
```
 - Create the Oracle and Netlsnr resources. You can refer to the examples shown in the file /etc/VRTSagents/ha/conf/Oracle/sample_main.cf.
 - Edit the default attributes to match the parameters in your configuration.
See “[Resource type definitions](#)” on page 75.
 - Assign dependencies to the newly created resources. Refer to the sample file /etc/VRTSagents/ha/conf/Oracle/sample_main.cf. and the sample dependency graphs. See the *Veritas Cluster Server User’s Guide* for more information on assigning dependencies.
See “[Sample configurations](#)” on page 85.
- 7 Save and close the file.
- 8 Verify the syntax of the file /etc/VRTSvcs/conf/config/main.cf:

```
# cd /etc/VRTSvcs/conf/config/
# hacf -verify .
```
- 9 Start the VCS engine:

```
# hstart
```
- 10 Verify that all Oracle service group resources are brought online:

```
# hagr -state
```
- 11 Take the service group offline and verify that all resources are stopped:

```
# hagr -offline service_group -sys system_name
# hagr -state
```
- 12 Bring the service group online again and verify that all resources are available:

```
# hagr -online service_group -sys system_name
# hagr -state
```
- 13 Start the VCS engine on sysb:

```
# hstart
```
- 14 Switch the Oracle service group to sysb:

```
# hagr -switch service_group -to sysb
```

15 Verify that all Oracle service group resources are brought online on sysb:

```
# hagrps -state
```

16 On all the nodes, look at the following log files for any errors or status:

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

Encrypting passwords

VCS provides a utility to encrypt database user passwords and listener passwords. You must encrypt the Pword attribute in the Oracle agent and the LsnrPwd attribute in the Netlsnr agent before configuring these attributes.

Oracle provides the option of storing the listener password in the listener.ora file, in both clear text and encrypted formats. Irrespective of the format in which the password is stored in Oracle, you must encrypt the password using the `vcseencrypt` utility before configuring the LsnrPwd attribute. This password, when decrypted should be of the same format as stored in the listener.ora file.

Note: You need not encrypt passwords when using the configuration wizard or the VCS Cluster Manager (Java Console) to configure attributes. The user passwords that are used for detail monitoring of the Oracle database are encrypted. The listener password that is considered for querying the status of the listener and stopping the listener is also encrypted.

To encrypt passwords

- 1 From the path `$VCS_HOME/bin/`, run the `vcseencrypt` utility.
 - Type the following command:

```
# vcseencrypt -agent
```
 - Enter the password and confirm it by entering it again. Press Enter.

```
# Enter New Password:  
# Enter Again:
```
- 2 Review as the utility encrypts the password and displays the encrypted password.
- 3 Enter this encrypted password as the value for the attribute.
- 4 Copy the encrypted password for future reference.

Additional considerations for basic monitoring

Basic monitoring of Oracle processes is user-specific. As a result, an Oracle instance started under the context of another user cannot be detected as online.

For example, if an Oracle instance is started under the user “oraVRT” and the agent is configured for a user “oracle”, the agent will not detect the instance started by “oraVRT” as online.

This could lead to situations where:

- Issuing a command to online a resource on a node might online an already running instance on that node (or any other node).
- Within a failover service group, when the administrator onlines an Oracle resource on a node and if the Oracle instance is online on any other node within a cluster, the instance would come up. However, the database does not get mounted. In such circumstances, this failure is detected only by health check monitoring option of basic monitoring or detail monitoring. Detail monitoring updates the database table after detecting a failure whereas health check monitoring does not.

If health check monitoring option of basic monitoring or detail monitoring is not configured, then such a conflict would go undetected.

See “[Monitor options for Oracle agent](#)” on page 13.

So, Symantec recommends that instances started outside VCS control be configured with the correct Owner attribute corresponding to the OS user for that instance.

Setting up detail monitoring

The Veritas High Availability Agent for Oracle provides two levels of application monitoring: primary (basic monitoring) and secondary (detail monitoring). In the basic monitoring mode, the agent monitors the Oracle processes to verify that they are continuously active.

In the detail monitoring mode, the agent executes the script defined in the attribute MonScript of the Oracle and Netlsnr resources. If the script successfully executes, the agent considers the resource available. You can customize the default script according to your configuration.

You can use the agent’s detail monitoring capability to monitor the status of a database and listener and increase the confidence in their availability. Before setting up detail monitoring, you must have the agent running satisfactorily at the basic level of monitoring.

Note: Disable detail monitoring before undertaking any database maintenance that involves disabling database access to external users.

Setting up detail monitoring for Oracle

Detail monitoring for an Oracle resource verifies whether a database is ready for transactions by performing an update transaction against a table within the database. The update action is taken by the two scripts, `SqlTest.pl` and `SimpleTest.pl`, provided with the Veritas High Availability Agent for Oracle. The scripts are available under the directory `/opt/VRTSagents/ha/bin/Oracle/`. Both scripts update the timestamp to monitor the database.

The `SqlTest.pl` script checks whether the database is open before updating the timestamp. If the database is found to be in restricted mode, quiesced mode, or suspended mode, the monitor returns success. In such a case, only basic monitoring occurs. The `SimpleTest.pl` script does not perform database checks but only issues update statements against the table.

Before enabling detail monitoring for Oracle, you must create a test table (with a timestamp) in the Oracle database. The detail monitor script, `MonScript`, must exist and have execute permission for root. You can use a custom monitor script, or the scripts provided with the agent. In the monitor script, the return code 100 indicates failure. Return codes from 101 to 110 indicate success.

The example to set up detail monitoring, based on the use of the supplied script, shows how to create and test a table for use by detail monitoring, and how to enable detail monitoring.

To set up detail monitoring for Oracle

- 1 Make the VCS configuration writable:

```
haconf -makerw
```
- 2 Freeze the service group to avoid automated actions by VCS caused by an incomplete reconfiguration:

```
hagrp -freeze service_group
```
- 3 Log on as an Oracle user.

```
su - <Owner>
```
- 4 Set the environment variables for `ORACLE_HOME` and `ORACLE_SID`.

```
export ORACLE_HOME=<Home>
export ORACLE_SID=<Sid>
```
- 5 Start the `svrmgrl` or `sqlplus` utility to set up a database table:

```
$ORACLE_HOME/bin/svrmgrl
```

or

```
$ORACLE_HOME/bin/sqlplus /nolog
```
- 6 As the database administrator, issue the following statements at the `svrmgrl` or `sqlplus` prompt to create the test table:

```
connect / as sysdba

create user <User>
```

```
identified by <Pword>
default tablespace USERS
temporary tablespace TEMP
quota 100K on USERS;

grant create session to <User>;

create table <User>.<Table> ( tstamp date );
insert into <User>.<Table> (tstamp) values (SYSDATE);
```

- 7 To test the database table for use, do the following:

```
disconnect
connect <User>/<Pword>
update <User>.<Table> set ( tstamp ) = SYSDATE;

select TO_CHAR(tstamp, 'MON DD, YYYY HH:MI:SS AM')
from <User>.<Table>;
exit
```

- 8 Enable the detail monitoring for the Oracle resource using the following VCS commands:

```
hares -modify OracleResource User User
hares -modify OracleResource Pword Pword
hares -modify OracleResource Table Table
hares -modify OracleResource MonScript "./bin/Oracle/SqlTest.pl"
hares -modify OracleResource DetailMonitor 1
haconf -dump -makero
hagrp -unfreeze service_group
```

You can also use Cluster Manager (Java Console) to set these attributes.

Enabling and disabling detail monitoring for Oracle

To enable detail monitoring

- ◆ Set the DetailMonitor attribute to 1.


```
hares -modify OracleResource DetailMonitor 1
```

To disable detail monitoring

- ◆ Set the DetailMonitor attribute to 0.


```
hares -modify OracleResource DetailMonitor 0
```

Setting up detail monitoring for Netlsnr

You can enable detail monitoring for Netlsnr by specifying a value for the MonScript attribute. The example to set up detail monitoring uses the supplied monitor script for Netlsnr, /opt/VRTSagents/ha/bin/Netlsnr/LsnrTest.pl. The detail monitoring script for the Netlsnr resource uses the Listener command `lsnrctl status $Listener` to test the Listener process.

To set up detail monitoring for Netlsnr

- 1 Make the VCS configuration writable:
haconf -makerw
- 2 Freeze the service group to avoid automated actions by VCS caused by an incomplete reconfiguration:
hagrp -freeze *service_group*
- 3 Enable detail monitoring by entering the following commands:
hares -modify *LsnrResource* MonScript *"/bin/Netlsnr/LsnrTest.pl"*
haconf -dump -makero
hagrp -unfreeze *service_group*
- 4 Disable detail monitoring by setting the MonScript attribute to an empty string:
haconf -makerw
hagrp -freeze *service_group*
hares -modify *SqlResource* MonScript ""
haconf -dump -makero
hagrp -unfreeze

Administering VCS service groups for Oracle

This chapter contains the following topics:

- [About administering VCS service groups](#)
- [Bringing the service group online](#)
- [Taking the service group offline](#)
- [Switching the service group](#)
- [Modifying the service group configuration](#)
- [Disabling the Veritas High Availability Agent for Oracle](#)
- [Removing the Veritas High Availability Agent for Oracle](#)

About administering VCS service groups

You can administer service groups in Cluster Server using the Cluster Manager or command-line. Review the procedure to administer the service groups using the Cluster Manager.

Bringing the service group online

Perform the following steps to bring the service group online.

To bring a 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 resources in the service 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

Perform the following steps to take the service group offline.

To take a service group offline

- 1 In 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 system and bringing it online on another system.

To switch a service group

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

Modifying the service group configuration

You can dynamically configure the Veritas High Availability Agent for Oracle in several ways, including the configuration wizard, the command-line interface,

Cluster Manager Java Console, and the Cluster Management Console. Refer to the *Veritas Cluster Server User's Guide* for more information.

To modify an Oracle service group using the cluster configuration wizard

- 1 Start the Oracle Configuration wizard.
`# hawizard oracle`
- 2 Read the information on the Welcome screen and click **Next**.
- 3 On the Wizard Options dialog box, select the **Modify service group** option, select the service group to be modified, and click **Next**.
- 4 Follow the wizard instructions and make modifications as per your configuration.
See “[Configuring the service group using the agent configuration wizard](#)” on page 47.

Disabling the Veritas High Availability Agent for Oracle

To disable the Veritas High Availability Agent for Oracle, you must change the Oracle service group to an OFFLINE state. You can stop the application completely or switch the agent to another system.

To disable the agent

- 1 To remove a system from the service group's SystemList, check if the service group is online:
`hagrp -state service_group -sys system_name`
- 2 If the service group is online, take it offline. Use one of the following commands:
 - To take the service group offline on one node and online it on another node, you can use the `-switch` option:
`hagrp -switch service_group -to system_name`
 - To take the service group offline without bringing it online on any other node in the cluster, enter:
`hagrp -offline service_group -sys system_name`
- 3 Stop the agent on the node:
`haagent -stop Oracle -sys system_name`
- 4 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 confirm the agent is stopped.

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

Removing the Veritas High Availability Agent for Oracle

Removing the agent involves removing the agent configuration wizard and the agent files *from each system*.

To remove the agent on a node

- 1 Remove the agent configuration wizard.
`# rpm -e VRTScsow`
- 2 Remove the agent. Answer prompts accordingly.
`# rpm -e VRTSwcsor`

Troubleshooting

This chapter contains the following topics:

- [Error messages common to the Oracle and Netlsnr agents](#)
- [Error messages specific to the Oracle agent](#)
- [Error messages specific to the Netlsnr agent](#)

Error messages common to the Oracle and Netlsnr agents

[Table 6-15](#) lists the Oracle enterprise agent error messages with the description and a recommended solution, if available.

Table 6-15 Oracle enterprise agent error messages

Message	Description and solution
No ORACLE_HOME specified	The Home attribute in the Oracle or Netlsnr type has not been set. Solution: Set the Home attribute to the correct full path name of the Oracle home.
Oracle home directory %s does not exist	The string specified for the Home attribute in the Oracle or Netlsnr type is incorrect. Solution: Set the Home attribute to the correct full path name of the Oracle home.
File %s is not a valid text file	The file specified by the EnvFile attribute for sourcing the environment variables is not present, not readable, or is not a text file. Solution: Set the EnvFile attribute to the correct full path name. Ensure that the file format is valid.

Table 6-15 Oracle enterprise agent error messages

Message	Description and solution
VCSAgExec returned failure when trying to execute in-depth test	Internal error. Solution: Contact Technical Support for further assistance.
Unable to open pipe from %s	Internal error. Solution: Contact Technical Support for further assistance.
Process %s restarted	Warning message to indicate that the PID for the Oracle process specified is different than the one registered by the previous monitor cycle.
Monitor procedure %s returned %s	MonScript failed to execute correctly. Solution: Debug MonScript to assess the exact cause of failure.
Monitor procedure %s did not exit, return value is %s	Internal error while executing MonScript. Solution: Contact Technical Support for further assistance.
No owner for Oracle executables was specified	The Owner attribute in the Oracle type has not been set. Solution: Set the Owner attribute to the correct owner of the database binaries.
Invalid owner %s for Oracle executables was specified	The OS user specified by the Owner attribute is invalid. Solution: Set the Owner attribute to the correct owner of the database binaries.
Access to Monscript %s denied. Detail Monitoring will not be enabled!! Please specify a valid file.	File specified by MonScript is not accessible or not found. Solution: Make sure that the file name indicates a valid and accessible file.
Encountered errors while decrypting password!	The agent cannot decrypt the password you specified. Solution: Use vcsencrypt utility to create a new encrypted password and supply the password.

Error messages specific to the Oracle agent

Table 6-16 lists the Oracle agent error messages with the description and a recommended solution, if available.

Table 6-16 Oracle agent error messages

Message	Description and solution
No SID specified	The Sid attribute in the Oracle type has not been set. Solution: Set the Sid attribute to the correct database instance.
sqlplus/svrmgrl not found in %s/bin	The client utilities svrmgrl or sqlplus are not found in the \$ORACLE_HOME/bin directory. Solution: Verify that the Oracle home has been correctly specified and that these executables are present.
srvctl not found in %s/bin	The client utility srvctl is not found in the \$ORACLE_HOME/bin directory. Solution: Verify that the Oracle home has been correctly specified and that this executable is present.
Oracle %s failed to stop	Warning message to indicate that Shutdown immediate or Shutdown abort commands were not successful in closing the Oracle instance, in the clean or offline entry point.
Oracle database %s not running	Warning message to indicate that the database instance was not running even before the clean or offline entry points were executed. Solution: No action required.
Oracle (%s) kill TERM %s	Warning message to indicate that the Oracle processes would be signaled with SIGTERM. Solution: No action required.
Oracle (%s) kill KILL %s	Warning message to indicate that the Oracle processes would be signaled with SIGKILL. Solution: No action required.
Database in QUIESCING/QUIESCED mode	Warning message to indicate that database is in QUIESCING or QUIESCED mode.

Table 6-16 Oracle agent error messages

Message	Description and solution
Database in RESTRICTED mode	Warning message to indicate that database is in RESTRICTED mode.
Database in SUSPENDED state	Warning message to indicate that database is in SUSPENDED state.
Resource %s - monitor procedure did not complete within the expected time.	<p>Refer to Oracle's alert log for more information.</p> <p>When a monitor times out as many times as the value specified, the corresponding resource is brought down by calling the clean entry point. The resource is then marked FAULTED, or it is restarted, depending on the value set in the RestartLimit attribute.</p> <p>Solution: Set the FaultOnMonitorTimeouts attribute to 0 so that the monitor failures are not considered indicative of a resource fault.</p> <p>Another possible reason could be that automatic archiving was not enabled while setting up the database.</p> <p>Solution: Archive the database manually. If automatic archival is enabled, set the LOG_ARCHIVE_START parameter in the file init.ora to TRUE.</p>
Custom script /opt/VRTSagents/ha/bin/Oracle/start_custom_%s.sql does not exist. Will not be able to start the database.	<p>The agent could not find the custom script at the specified location to start the database.</p> <p>Solution: Make sure the custom file exists at the specified location and has valid permissions.</p>
Custom script /opt/VRTSagents/ha/bin/Oracle/shut_custom_%s.sql does not exist. Using default shutdown option.	<p>The agent could not find the custom script at the specified location to stop the database.</p> <p>Solution: Make sure the custom file exists and the specified location and has valid permissions.</p>
oraerror.dat did not have records that could be parsed	<p>The file oraerror.dat is not present or has records in an unsupported format.</p> <p>Solution: Make sure the file exists and has data in the supported format.</p>
Incorrect Monitor Option	<p>The MonitorOption value is less than 0 or greater than 1.</p> <p>Solution: Set the MonitorOption value to 0 or 1.</p>

Table 6-16 Oracle agent error messages

Message	Description and solution
MonitorOption value not applicable for this Oracle Version	The health check monitoring option is selected when Oracle version is not Oracle 10g or later. Solution: Set the MonitorOption value to 0 to select the process check monitoring option.
VCSAgExec returned failure when trying to execute health check monitor test	Internal error. Solution: Contact Technical Support for further assistance.
VCSAgExec returned failure while trying to find Oracle version	Internal error. Solution: Contact Technical Support for further assistance.
One or more of the attributes User:Pword:Table:MonScript are not set correctly. Detail monitoring will not be enabled!! Unset the DetailMonitor attribute if you want to disable DetailMonitoring.	Detail Monitoring has been enabled but the necessary attributes for detail monitoring have not been set correctly. Solution: Set the required attributes for detail monitoring correctly or set DetailMonitor attribute to 0 to disable detail monitoring.

Error messages specific to the Netlsnr agent

[Table 6-17](#) lists the Netlsnr agent error messages with the description and a recommended solution, if available.

Table 6-17 Netlsnr agent error messages

Message	Description and solution
Cannot open process directory.	The agent could not process the /proc entries in the particular monitor cycle. Solution: No action required.
Listener process %s not running	Warning message to indicate that the Listener process was not running even before the clean or offline entry points were executed. Solution: No action required.
Listener %s kill TERM %s	Warning message to indicate that the Listener process would be signaled with SIGTERM. Solution: No action required.

Table 6-17 Netlsnr agent error messages

Message	Description and solution
Listener %s kill KILL %s	Warning message to indicate that the Listener process would be signaled with SIGKILL. Solution: No action required.
lsnrctl not found in %s/bin	The client utility lsnrctl is not found in the \$ORACLE_HOME/bin directory. Solution: Verify that the Oracle home has been correctly specified and that this executable is present.
lsnrctl operation timed out	The tnslsnr process is not responding. Solution: Verify the underlying network protocol.

Resource type definitions

This appendix contains the following topics:

- [About the resource type and attribute definitions](#)
- [Oracle resource type](#)
- [Netlsnr resource type](#)

About the resource type and attribute definitions

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

Oracle resource type

The Oracle agent of the Veritas High Availability Agent for Oracle is represented by the Oracle resource type in VCS.

```
type Oracle (  
    static str AgentDirectory = "/opt/VRTSagents/ha/bin/Oracle"  
    static keylist SupportedActions = { VRTS_GetInstanceName,  
        VRTS_GetRunningServices, DBRestrict, DBUndoRestrict,  
        DBResume, DBSuspend, DBTbspBackup }  
    static str ArgList[] = { Sid, Owner, Home, Pfile, StartUpOpt,  
        ShutDownOpt, EnvFile, AutoEndBkup, MonitorOption,  
        DetailMonitor, User, Pword, Table, MonScript,  
        AgentDebug, Encoding }  
    str Sid  
    str Owner  
    str Home  
    str Pfile  
    str StartUpOpt = STARTUP_FORCE
```

```
str ShutDownOpt = IMMEDIATE
str EnvFile
boolean AutoEndBkup = 1
int MonitorOption = 0
int DetailMonitor = 0
str MonScript = "./bin/Oracle/SqlTest.pl"
str User
str Pword
str Table
boolean AgentDebug = 0
str Encoding
)
```

Oracle attribute definitions

Review the description of the Oracle agent attributes. The agent attributes are classified as required, optional, and internal.

Required attributes

[Table A-18](#) lists the required attributes. You must assign values to the required attributes.

Table A-18 Required attributes for Oracle agent

Required attributes	Type and dimension	Definition
Sid	string-scalar	The variable \$ORACLE_SID that represents the Oracle instance. The Sid is considered case-sensitive by the Oracle agent and by the Oracle database server.
Owner	string-scalar	The Oracle user, as the defined owner of executables and database files in /etc/passwd.
Home	string-scalar	The \$ORACLE_HOME path to Oracle binaries and configuration files. For example, you could specify the path as /opt/ora_home. Note: Do not append a slash (/) at the end of the path.

Optional attributes

[Table A-19](#) lists the optional attributes for Oracle agent. You can configure the optional attributes if necessary.

Table A-19 Optional attributes for Oracle agent

Optional Attributes	Type and Dimension	Definition
StartUpOpt	string-scalar	<p>Startup options for the Oracle instance. This attribute can take the following values:</p> <ul style="list-style-type: none">■ STARTUP■ STARTUP_FORCE■ RESTRICTED■ RECOVERDB■ SRVCTLSTART■ CUSTOM <p>Default is STARTUP_FORCE.</p> <p>See “Startup options” on page 12.</p>
ShutDownOpt	string-scalar	<p>Shut down options for the Oracle instance. This attribute can take the following values:</p> <ul style="list-style-type: none">■ IMMEDIATE■ TRANSACTIONAL■ SRVCTLSTOP■ CUSTOM <p>Default is IMMEDIATE.</p> <p>See “Shutdown options” on page 13.</p>
EnvFile	string-scalar	<p>The full path name of the file that is sourced by the entry point scripts. This file contains the environment variables set by the user for the Oracle database server environment such as LD_LIBRARY_PATH, NLS_DATE_FORMAT, and so on.</p> <p>The syntax for the contents of the file depends on the login shell of Owner. File must be readable by Owner. The file must not contain any prompts for user input.</p>

Table A-19 Optional attributes for Oracle agent

Optional Attributes	Type and Dimension	Definition
Pfile	string-scalar	<p>The name of the initialization parameter file with the complete path of the startup profile.</p> <p>You can also use the server parameter file. Create a one-line text initialization parameter file that contains only the SPFILE parameter. See the Oracle documentation for more information.</p> <p>See “Using the SPFILE in a VCS cluster” on page 104.</p>
AutoEndBkup	integer-scalar	<p>Setting the AutoEndBkup attribute to a non-zero value takes the datafiles in the database out of the backup mode, during Online.</p> <p>Default = 1</p> <p>See “Hot backup of Oracle database and VCS” on page 25.</p>
MonitorOption	integer-scalar	<p>Monitor options for the Oracle instance. This attribute can take values 0 or 1.</p> <ul style="list-style-type: none"> ■ 0 - Process check monitoring (recommended) ■ 1 - Health check monitoring <p>Default = 0</p> <p>See “Monitor options for Oracle agent” on page 13.</p>
DetailMonitor	integer-scalar	<p>Setting this flag to a non-zero enables detail monitoring for Oracle. The value indicates the number of monitor cycles after which the agent will monitor Oracle in detail. For example, the value 5 indicates that the agent will monitor Oracle in detail every five monitor intervals.</p> <p>Default = 0</p>

Table A-19 Optional attributes for Oracle agent

Optional Attributes	Type and Dimension	Definition
MonScript	string-scalar	<p>Pathname to the script provided for detail monitoring. The default (basic monitoring) is to monitor the database PIDs only.</p> <p>Note: Detail monitoring is disabled if the value of the attribute MonScript is invalid or is set to an empty string.</p> <p>The pathname to the supplied detail monitor script is /opt/VRTSagents/ha/bin/Oracle/SqlTest.pl.</p> <p>MonScript also accepts a pathname relative to /opt/VRTSagents/ha. A relative pathname should start with “./”, as in the path ./bin/Oracle/SqlTest.pl.</p>
User	string-scalar	<p>Internal database user. Connects to the database for detail monitoring.</p>
Pword	string-scalar	<p>Encrypted password for internal database-user authentication.</p> <p>Encrypt passwords only when entering them using the command-line. Passwords must be encrypted using the VCS Encrypt utility.</p> <p>See “Encrypting passwords” on page 59.</p>
Table	string-scalar	<p>Table for update by User / Pword.</p>
Encoding	string-scalar	<p>Specifies operating system encoding that corresponds to Oracle encoding for the displayed Oracle output. Default is “”.</p>
AgentDebug	boolean-scalar	<p>Additional debug messages are logged when this flag is set.</p> <p>Default = 0</p>

Internal attribute

Table A-20 lists the internal attribute for Oracle agent. This attribute is for internal use only. Symantec recommends not to modify the value of this attribute.

Table A-20 Internal attributes for Oracle agent

Optional Attributes	Type and Dimension	Definition
AgentDirectory	static-string	Specifies the location of binaries, scripts, and other files related to the Oracle agent. Default is /opt/VRTSagents/ha/bin/Oracle.

Netlsnr resource type

The Netlsnr agent of the Veritas High Availability Agent for Oracle is represented by the Netlsnr resource type in VCS.

```
type Netlsnr (
    static str AgentDirectory = "/opt/VRTSagents/ha/bin/Netlsnr"
    static keylist SupportedActions = { VRTS_GetInstanceName,
        VRTS_GetRunningServices }
    static str ArgList[] = { Owner, Home, TnsAdmin, Listener,
        EnvFile, MonScript, LsnrPwd, AgentDebug, Encoding }
    str Owner
    str Home
    str TnsAdmin
    str Listener
    str EnvFile
    str MonScript = "./bin/Netlsnr/LsnrTest.pl"
    str LsnrPwd
    boolean AgentDebug = 0
    str Encoding
)
```

Netlsnr attribute definitions

Review the description of the Netlsnr agent attributes. The agent attributes are classified as required, optional, and internal.

Required attributes

[Table A-21](#) lists the required attributes for Netlsnr agent. You must assign values to the required attributes.

Table A-21 Required attributes for Netlsnr agent

Required attributes	Type and dimension	Definition
Owner	string-scalar	Oracle user, as the defined owner of executables and database files in /etc/passwd.
Home	string-scalar	The \$ORACLE_HOME path to Oracle binaries and configuration files. For example, you could specify the path as /opt/ora_home. Do not append a slash (/) at the end of the path.

Optional attributes

[Table A-22](#) lists the optional attributes for Netlsnr agent. You can configure the optional attributes if necessary.

Table A-22 Optional attributes for Netlsnr agent

Optional attributes	Type and dimension	Definition
TnsAdmin	string-scalar	The \$TNS_ADMIN path to directory in which the Listener configuration file resides (listener.ora). Default is /var/opt/oracle.
Listener	string-scalar	Name of Listener. The name for Listener is considered case-insensitive by the Netlsnr agent and the Oracle database server. Default is LISTENER.
LsnrPwd	string-scalar	The VCS encrypted password used to stop and monitor the listener. This password is set in the Listener configuration file. Encrypt passwords only when entering them using the command-line. Passwords must be encrypted using the VCS Encrypt utility. See “Encrypting passwords” on page 59.

Table A-22 Optional attributes for Netlsnr agent

Optional attributes	Type and dimension	Definition
EnvFile	string-scalar	<p>Specifies the full path name of the file that is sourced by the entry point scripts. This file contains the environment variables set by the user for the Oracle listener environment such as LD_LIBRARY_PATH and so on.</p> <p>The syntax for the contents of the file depends on the login shell of Owner. This file must readable by Owner. The file must not contain any prompts for user input.</p>
MonScript	string-scalar	<p>Pathname to the script provided for detail monitoring. The default (basic monitoring) is to monitor the listener process only.</p> <p>Note: Detail monitoring is disabled if the value of the attribute MonScript is invalid or is set to an empty string.</p> <p>The pathname to the supplied detail monitoring script is /opt/VRTSagents/ha/bin/Netlsnr/LsnrTest.pl.</p> <p>MonScript also accepts a pathname relative to /opt/VRTSagents/ha. A relative pathname should start with “./”, as in the path ./bin/Netlsnr/LsnrTest.pl.</p>
Encoding	string-scalar	<p>Specifies operating system encoding that corresponds to Oracle encoding for the displayed Oracle output.</p> <p>Default is “”.</p>
AgentDebug	boolean	<p>Additional debug messages are logged when this flag is set.</p> <p>Default = 0</p>

Internal attribute

[Table A-23](#) lists the internal attribute for Netlsnr agent. This attribute is for internal use only. Symantec recommends not to modify the value of this attribute.

Table A-23 Internal attributes for Oracle agent

Optional Attributes	Type and Dimension	Definition
AgentDirectory	static-string	Specifies the location of binaries, scripts, and other files related to the Netlsnr agent. Default is /opt/VRTSagents/ha/bin/Netlsnr.

Sample configurations

This appendix contains the following topics:

- [Sample single Oracle instance configuration](#)
- [Sample multiple Oracle instances \(single listener\) configuration](#)
- [Sample multiple instance \(multiple listeners\) configuration](#)
- [Sample Oracle configuration with shared server support](#)

About the sample configurations for Oracle enterprise agent

The sample configuration include descriptions for typical service groups that are configured to monitor the state of Oracle in a VCS cluster.

See [“About Oracle configurations in VCS”](#) on page 39.

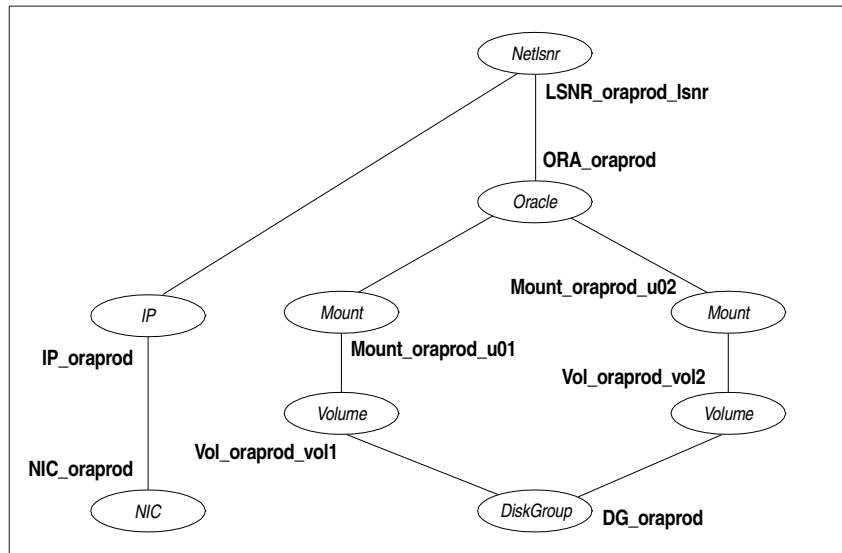
The sample dependency graphs depict the resource types, resources, and resource dependencies within the service group. The sample configuration file (main.cf) is also included for your reference.

Review these dependencies carefully before configuring the agent. For more information about VCS resource types, see the *Veritas Cluster Server Bundled Agents Reference Guide*.

Sample single Oracle instance configuration

[Figure B-4](#) describes a typical service group configured to monitor the state of an Oracle instance in a VCS cluster.

Figure B-4 Dependency graph for single Oracle instance



The shared disk groups and volumes in the cluster are configured as resources of type **DiskGroup** and **Volume** respectively. The volumes are mounted using the **Mount** agent. The virtual IP address for the service group is configured using the **IP** and **NIC** resource types. The Oracle server can be started after each of these resources is brought online.

If your configuration does not use Veritas Volume Manager, use the **DiskReservation** resource type to configure shared storage instead of the **DiskGroup** and **Volume** resource types.

Sample configuration file for single Oracle instance

Review the sample configuration with a resource of type **Oracle** that is configured as follows in **main.cf** file:

```

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

cluster vcs (
)

system vcslnx1 (
)

system vcslnx2 (
)

group ORA_PROD_Group (
```

```
SystemList = { vcslnx1 = 0, vcslnx2 = 1 }
AutoStartList = { vcslnx1 }
)

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

IP IP_oraprod (
    Device = eth0
    Address = "10.212.102.13"
)

Mount Mount_oraprod_u01 (
    MountPoint = "/prod/u01"
    BlockDevice = "/dev/vx/dsk/ora_prod_dg/u01-vol"
    FSType = vxfs
    FsckOpt = "-n"
)

Mount Mount_oraprod_u02 (
    MountPoint = "/prod/u02"
    BlockDevice = "/dev/vx/dsk/ora_prod_dg/u02-vol"
    FSType = vxfs
    FsckOpt = "-n"
)

NIC NIC_oraprod (
    Device = eth0
)

Netlsnr LSNR_oraprod_lsnr (
    Owner = oraprod
    Home = "/orahome/Oracle"
    TnsAdmin = "/orahome/Oracle/network/admin"
    Listener = LISTENER_PROD
    MonScript = "./bin/Netlsnr/LsnrTest.pl"
    LsnrPwd = cqf0do0ol0o
)

Oracle ORA_oraprod (
    Sid = PROD
    Owner = oraprod
    Home = "/orahome/Oracle"
    EnvFile = "/tmp/env.sh"
    DetailMonitor = 1
    MonScript = "./bin/Oracle/SqlTest.pl"
    User = thor
    Pword = hvlTptWvj
    Table = thor
```

Sample multiple Oracle instances (single listener) configuration

```

        MonitorOption = 0
    )

    Volume Vol_oraprod_vol1 (
        Volume = u01-vol
        DiskGroup = ora_prod_dg
    )

    Volume Vol_oraprod_vol2 (
        Volume = u02-vol
        DiskGroup = ora_prod_dg
    )

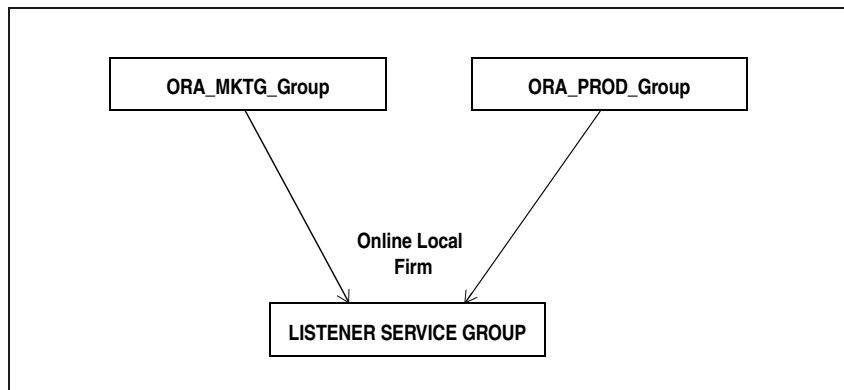
    IP_oraprod requires NIC_oraprod
    LSNR_oraprod_lsnr requires IP_oraprod
    LSNR_oraprod_lsnr requires ORA_oraprod
    Mount_oraprod_u01 requires Vol_oraprod_vol1
    Mount_oraprod_u02 requires Vol_oraprod_vol2
    ORA_oraprod requires Mount_oraprod_u01
    ORA_oraprod requires Mount_oraprod_u02
    Vol_oraprod_vol1 requires DG_oraprod
    Vol_oraprod_vol2 requires DG_oraprod

```

Sample multiple Oracle instances (single listener) configuration

Figure B-5 describes a typical VCS configuration to monitor two Oracle instances sharing a listener. This configuration has a service group for each Oracle instance. The Listener too is configured in a separate service group.

Figure B-5 Two Oracle instances sharing a listener



The Oracle service groups are made dependent on the Listener service group using an Online Local Firm dependency.

Figure B-6 shows the dependency graph for one of the Oracle instances in the VCS configuration. In the Oracle service group, the shared disk groups and volumes in the cluster are configured as resources of type DiskGroup and Volume respectively. The volumes are mounted using the Mount agent.

Figure B-6 Dependency graph for one of the Oracle instances

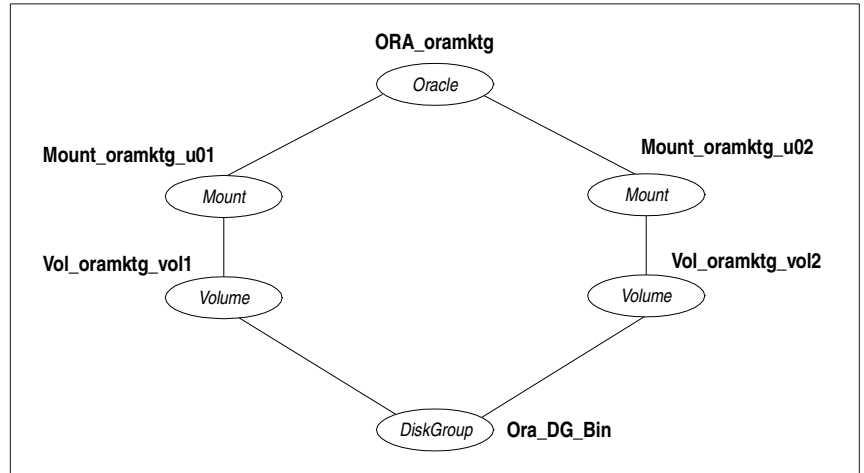
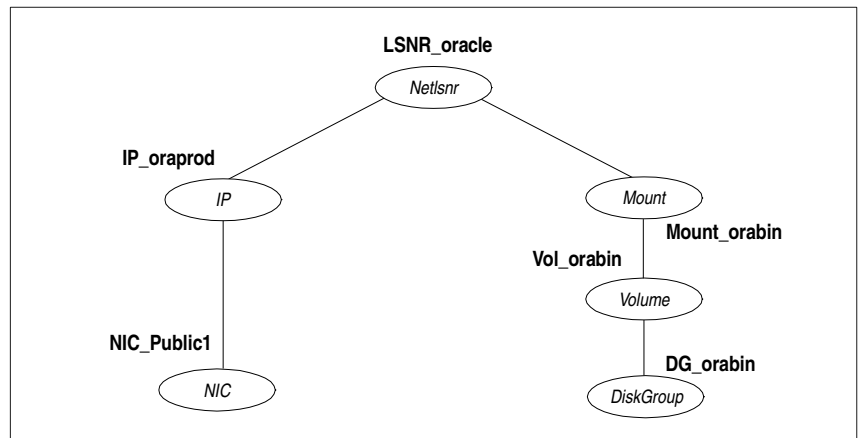


Figure B-7 shows the dependency graph for the listener that the two Oracle instances share in the VCS configuration. In the Listener service group, the virtual IP address is configured using the IP and NIC resource types. The Listener can be started after the IP and NIC resources are brought online.

Figure B-7 Dependency graph for the single listener



Sample multiple Oracle instances (single listener) configuration

The Oracle server can be started after the Listener service group and the resources in the Oracle service group are brought online.

If your configuration does not use Veritas Volume Manager, use the DiskReservation resource type to configure shared storage instead of the DiskGroup and Volume resource types.

Note: In this case, make sure you have modified all proper system files, such as /etc/system, /etc/passwd, /etc/group, and /etc/shadow to support multiple databases. Pay particular attention to system requirements like physical memory and shared memory segment availability. Also ensure a single system is capable of sustaining a multiple instance load in the event of a server failure and extended operation on the backup server.

Sample configuration file for multiple Oracle instances (single listener)

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

cluster vcs (
)

system vcslnx1 (
)

system vcslnx2 (
)

group ORA_MKTG_Group (
    SystemList = { vcslnx1 = 0, vcslnx2 = 1 }
    AutoStartList = { vcslnx1 }
)

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

    Mount Mount_oramktg_u01 (
        MountPoint = "/mktg/u01"
        BlockDevice = "/dev/vx/dsk/ora_mktg_dg/u01-vol1"
        FSType = vxfs
        FsckOpt = "-n"
    )

    Mount Mount_oramktg_u02 (
        MountPoint = "/mktg/u02"
```

```
        BlockDevice = "/dev/vx/dsk/ora_mktg_dg/u02-vol"
        FSType = vxfs
        FckOpt = "-n"
    )

    Oracle ORA_oramktg (
        Sid = MKTG
        Owner = oramktg
        Home = "/orahome/Oracle"
        EnvFile = "/tmp/env.sh"
        DetailMonitor = 1
        MonScript = "./bin/Oracle/SqlTest.pl"
        User = thor
        Pword = hvlTptWvj
        Table = thor
        MonitorOption = 0
    )

    Volume Vol_oramktg_vol1 (
        Volume = u01-vol
        DiskGroup = ora_mktg_dg
    )

    Volume Vol_oramktg_vol2 (
        Volume = u02-vol
        DiskGroup = ora_mktg_dg
    )

    requires group Common_Service online firm
    Mount_oramktg_u01 requires Vol_oramktg_vol1
    Mount_oramktg_u02 requires Vol_oramktg_vol2
    ORA_oramktg requires Mount_oramktg_u01
    ORA_oramktg requires Mount_oramktg_u02
    Vol_oramktg_vol1 requires DG_oramktg
    Vol_oramktg_vol2 requires DG_oramktg

    group ORA_PROD_Group (
        SystemList = { vcslnx1 = 0, vcslnx2 = 1 }
        AutoStartList = { vcslnx1 }
    )

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

    Mount Mount_oraprod_u01 (
        MountPoint = "/prod/u01"
        BlockDevice = "/dev/vx/dsk/ora_prod_dg/u01-vol"
        FSType = vxfs
```

Sample multiple Oracle instances (single listener) configuration

```

        FsckOpt = "-n"
    )

Mount Mount_oraprod_u02 (
    MountPoint = "/prod/u02"
    BlockDevice = "/dev/vx/dsk/ora_prod_dg/u02-vol"
    FSType = vxfs
    FsckOpt = "-n"
)

Oracle ORA_oraprod (
    Sid = PROD
    Owner = oraprod
    Home = "/orahome/Oracle"
    EnvFile = "/tmp/env.sh"
    DetailMonitor = 1
    MonScript = "../bin/Oracle/SqlTest.pl"
    User = thor
    Pword = hv1TptWvj
    Table = thor
    MonitorOption = 0
)

Volume Vol_oraprod_vol1 (
    Volume = u01-vol
    DiskGroup = ora_prod_dg
)

Volume Vol_oraprod_vol2 (
    Volume = u02-vol
    DiskGroup = ora_prod_dg
)

requires group Common_Service online firm
Mount_oraprod_u01 requires Vol_oraprod_vol1
Mount_oraprod_u02 requires Vol_oraprod_vol2
ORA_oraprod requires Mount_oraprod_u01
ORA_oraprod requires Mount_oraprod_u02
Vol_oraprod_vol1 requires DG_oraprod
Vol_oraprod_vol2 requires DG_oraprod

group Common_Service (
    SystemList = { vcslnx1 = 0, vcslnx2 = 1 }
    AutoStartList = { vcslnx1 }
)

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

```

```

IP IP_oraprod (
    Device = eth0
    Address = "10.212.102.13"
)

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

NIC NIC_Public1 (
    Device = eth0
)

Netlsnr LSNR_oracle (
    Owner = oracle
    Home = "/orahome/Oracle"
    TnsAdmin = "/orahome/Oracle/network/admin"
    Listener = LISTENER_ORACLE
    MonScript = "./bin/Netlsnr/LsnrTest.pl"
    LsnrPwd = cqf0do0o10o
)

Volume Vol_orabin (
    Volume = u01-vol
    DiskGroup = ora_bin_dg
)

IP_oraprod requires NIC_Public1
LSNR_oracle requires IP_oraprod
LSNR_oracle requires Mount_orabin
Mount_orabin requires Vol_orabin
Vol_orabin requires DG_orabin

```

Sample multiple instance (multiple listeners) configuration

This configuration has several single-instance configurations. Each Oracle instance is configured in a separate service group. The resource dependencies are similar to those described in the [“Sample single Oracle instance configuration”](#) on page 85.

Sample configuration file

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

Sample multiple instance (multiple listeners) configuration

```

include "OracleTypes.cf"

cluster vcs (
)

system vcslnx1 (
)

system vcslnx2 (
)

group ORA_MKTG_Group (
    SystemList = { vcslnx1 = 0, vcslnx2 = 1 }
    AutoStartList = { vcslnx1 }
)

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

IP IP_oramktg (
    Device = eth0
    Address = "10.212.102.14"
)

Mount Mount_oramktg_u01 (
    MountPoint = "/mktg/u01"
    BlockDevice = "/dev/vx/dsk/ora_mktg_dg/u01-vol"
    FSType = vxfs
    FsckOpt = "-n"
)

Mount Mount_oramktg_u02 (
    MountPoint = "/mktg/u02"
    BlockDevice = "/dev/vx/dsk/ora_mktg_dg/u02-vol"
    FSType = vxfs
    FsckOpt = "-n"
)

Netlsnr LSNR_oramktg_lsnr (
    Owner = oramktg
    Home = "/orahome/Oracle"
    TnsAdmin = "/orahome/Oracle/network/admin"
    Listener = LISTENER_MKTG
    MonScript = "../bin/Netlsnr/LsnrTest.pl"
    LsnrPwd = cqf0do0o10o
)

Oracle ORA_oramktg (
    Sid = MKTG

```

```
    Owner = oramktg
    Home = "/orahome/Oracle"
    EnvFile = "/tmp/env.sh"
    DetailMonitor = 1
    MonScript = "./bin/Oracle/SqlTest.pl"
    User = thor
    Pword = hvltPtWvj
    Table = thor
    MonitorOption = 0
)

Proxy NICProxy_oramktg (
    TargetResName = NIC_Public1
)

Volume Vol_oramktg_vol1 (
    Volume = u01-vol
    DiskGroup = ora_mktg_dg
)

Volume Vol_oramktg_vol2 (
    Volume = u02-vol
    DiskGroup = ora_mktg_dg
)

IP_oramktg requires NICProxy_oramktg
LSNR_oramktg_lsnr requires IP_oramktg
LSNR_oramktg_lsnr requires ORA_oramktg
Mount_oramktg_u01 requires Vol_oramktg_vol1
Mount_oramktg_u02 requires Vol_oramktg_vol2
ORA_oramktg requires Mount_oramktg_u01
ORA_oramktg requires Mount_oramktg_u02
Vol_oramktg_vol1 requires DG_oramktg
Vol_oramktg_vol2 requires DG_oramktg

group ORA_PROD_Group (
    SystemList = { vcslnx1 = 0, vcslnx2 = 1 }
    AutoStartList = { vcslnx1 }
)

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

IP IP_oraprod (
    Device = eth0
    Address = "10.212.102.13"
)
```

Sample multiple instance (multiple listeners) configuration

```

Mount Mount_oraprod_u01 (
    MountPoint = "/prod/u01"
    BlockDevice = "/dev/vx/dsk/ora_prod_dg/u01-vol"
    FSType = vxfs
    FsckOpt = "-n"
)

Mount Mount_oraprod_u02 (
    MountPoint = "/prod/u02"
    BlockDevice = "/dev/vx/dsk/ora_prod_dg/u02-vol"
    FSType = vxfs
    FsckOpt = "-n"
)

Netlsnr LSNR_oraprod_lsnr (
    Owner = oraprod
    Home = "/orahome/Oracle"
    TnsAdmin = "/orahome/Oracle/network/admin"
    Listener = LISTENER_PROD
    MonScript = "../bin/Netlsnr/LsnrTest.pl"
    LsnrPwd = cqf0do0ol0o
)

Oracle ORA_oraprod (
    Sid = PROD
    Owner = oraprod
    Home = "/orahome/Oracle"
    EnvFile = "/tmp/env.sh"
    DetailMonitor = 1
    MonScript = "../bin/Oracle/SqlTest.pl"
    User = thor
    Pword = hvlTptWvj
    Table = thor
    MonitorOption = 0
)

Proxy NICProxy_oraprod (
    TargetResName = NIC_Public1
)

Volume Vol_oraprod_vol1 (
    Volume = u01-vol
    DiskGroup = ora_prod_dg
)

Volume Vol_oraprod_vol2 (
    Volume = u02-vol
    DiskGroup = ora_prod_dg
)

IP_oraprod requires NICProxy_oraprod
LSNR_oraprod_lsnr requires IP_oraprod

```



```

LSNR_oraprod_lsnr requires ORA_oraprod
Mount_oraprod_u01 requires Vol_oraprod_vol1
Mount_oraprod_u02 requires Vol_oraprod_vol2
ORA_oraprod requires Mount_oraprod_u01
ORA_oraprod requires Mount_oraprod_u02
Vol_oraprod_vol1 requires DG_oraprod
Vol_oraprod_vol2 requires DG_oraprod

group Parallel_Service (
  SystemList = { vcslnx1 = 0, vcslnx2 = 1 }
  Parallel = 1
  AutoStartList = { vcslnx1 }
)

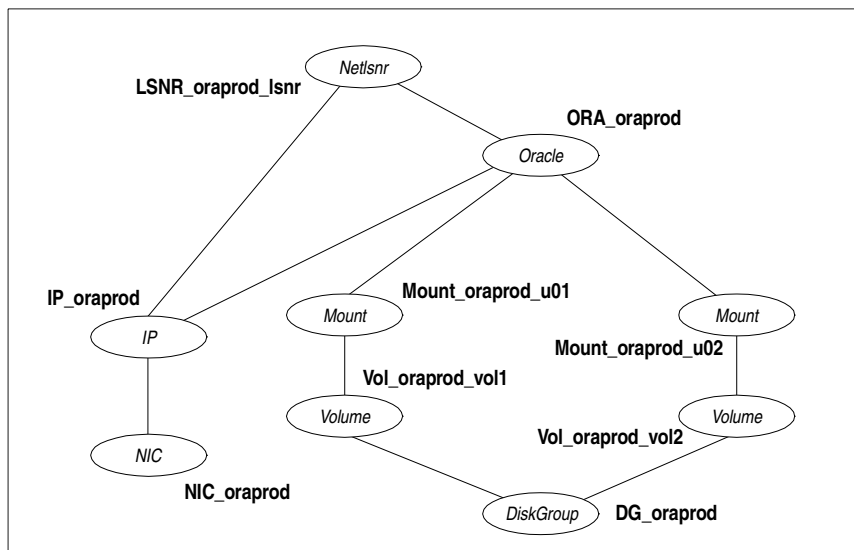
NIC NIC_Public1 (
  Device = eth0
)

```

Sample Oracle configuration with shared server support

Figure B-8 describes a typical service group configured to monitor Oracle with shared server support.

Figure B-8 Dependency for Oracle configured with shared server support



The shared disk groups and volumes in the cluster are configured as resources of type DiskGroup and Volume respectively. The volumes are mounted using the Mount agent. The virtual IP address for the service group is configured using the IP and NIC resource types. The Oracle server can be started after each of these resources is brought online.

If your configuration does not use Veritas Volume Manager, use the DiskReservation resource type to configure shared storage instead of the DiskGroup and Volume resource types.

Sample configuration file for Oracle instance configured with shared server support

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

cluster vcs (
)

system vcslnx1 (
)

system vcslnx2 (
)

group ORA_PROD_Group (
    SystemList = { vcslnx1 = 0, vcslnx2 = 1 }
    AutoStartList = { vcslnx1 }
)

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

IP IP_oraprod (
    Device = eth0
    Address = "10.212.102.13"
)

Mount Mount_oraprod_u01 (
    MountPoint = "/prod/u01"
    BlockDevice = "/dev/vx/dsk/ora_prod_dg/u01-vol1"
    FSType = vxfs
    FsckOpt = "-n"
)

Mount Mount_oraprod_u02 (
    MountPoint = "/prod/u02"
```

```

        BlockDevice = "/dev/vx/dsk/ora_prod_dg/u02-vol"
        FSType = vxfs
        FsckOpt = "-n"
    )

    NIC NIC_ORAPROD (
        Device = eth0
    )

    Netlsnr LSNR_oraprod_lsnr (
        Owner = oraprod
        Home = "/orahome/Oracle"
        TnsAdmin = "/orahome/Oracle/network/admin"
        Listener = LISTENER_PROD
        MonScript = "./bin/Netlsnr/LsnrTest.pl"
        LsnrPwd = cqfOdoOolOo
    )

    Oracle ORA_oraprod (
        Sid = PROD
        Owner = oraprod
        Home = "/orahome/Oracle"
        EnvFile = "/tmp/env.sh"
        DetailMonitor = 1
        MonScript = "./bin/Oracle/SqlTest.pl"
        User = thor
        Pword = hvLTptWvj
        Table = thor
        MonitorOption = 0
    )

    Volume Vol_oraprod_vol1 (
        Volume = u01-vol
        DiskGroup = ora_prod_dg
    )

    Volume Vol_oraprod_vol2 (
        Volume = u02-vol
        DiskGroup = ora_prod_dg
    )

    IP_oraprod requires NIC_ORAPROD
    LSNR_oraprod_lsnr requires IP_oraprod
    Mount_oraprod_u01 requires Vol_oraprod_vol1
    Mount_oraprod_u02 requires Vol_oraprod_vol2
    ORA_oraprod requires IP_oraprod
    ORA_oraprod requires Mount_oraprod_u01
    ORA_oraprod requires Mount_oraprod_u02
    Vol_oraprod_vol1 requires DG_oraprod
    Vol_oraprod_vol2 requires DG_oraprod

```


Best practices

This appendix contains the following topic:

- [Best practices for multiple Oracle instance configurations](#)

Best practices for multiple Oracle instance configurations

Review some of the best practices for using multiple Oracle instances in a VCS environment.

- For each SID to be configured, create Linux accounts with DBA privileges.
- Make sure that each Oracle instance has a separate disk group and is configured as a separate service group.
- Define the system parameters such that the allocation of semaphore and shared memory is appropriate on all systems.
- Use a dedicated set of binaries for each Oracle instance, even if each instance uses the same Oracle version.
- If your configuration uses the same Oracle version for all instances, install a version on the root disk or preferably on a secondary disk. Locate the pfiles in the default location and define several listener processes to ensure clean failover.
- If your configuration has different versions of Oracle, create a separate \$ORACLE_HOME for each Oracle version.
- Follow the Optimal Flexible Architecture (OFA) standard (/uwx/<SID>). In cluster configurations, you could adapt the standard to make it more application-specific. For example, /app/uwx/<SID>.
- Listeners accompanying different versions of Oracle may not be backward-compatible. So, if you want to create a single listener.ora file, you

must verify that the listener supports the other versions of Oracle in the cluster. You must also create a separate Envfile for each version of Oracle.

- Make sure that each listener listens to a different virtual address. Also, assign different names to listeners and make sure that they do not listen to the same port.
- The pfiles must be co-ordinated between systems. For the same instance of a database, ensure that the pfiles referenced are identical across the nodes.

Using the SPFILE in a VCS cluster for Oracle

This appendix contains the following topics:

- [About the Oracle initialization parameter files](#)
- [Starting an Oracle instance](#)
- [Using the SPFILE in a VCS cluster](#)

About the Oracle initialization parameter files

Oracle versions prior to Oracle9i used an initialization file `initSID.ora`, a text file, to start database instances. Changes applied to instance parameters during a database session were not saved to the file; you had to manually apply them to the initialization file.

Oracle9i introduced the SPFILE, which is a binary file stored on the database server. With this feature, changes to the instance parameters can be set to be persistent across all startup and shutdown procedures.

Starting an Oracle instance

For versions above Oracle9i, you can start an Oracle instance in the following ways:

- Using the default SPFILE `spfileSID.ora`
- Using the default init.ora file `initSID.ora`
- By specifying an initialization file `init.ora`
- By specifying an SPFILE in the initialization file `init.ora`

When you run the `startup` command without a PFILE clause, Oracle reads the initialization parameters from the SPFILE. On Linux platforms, the default location for the SPFILE or PFILE is `$ORACLE_HOME/dbs`.

Oracle locates the initialization parameter file by examining file names in the following order:

- `SPFILESID.ora`
- `SPFILE.ora`
- `initSID.ora`

Using the SPFILE in a VCS cluster

When using the Veritas High Availability Agent for Oracle, you can start a database instance by specifying a PFILE. If you do not specify the PFILE, the database instance starts up using the default SPFILE.

The agent attribute `Pfile` must specify the location of the PFILE. If your configuration uses the SPFILE, the contents of the PFILE must specify the location of the SPFILE, which must be created from the PFILE.

Note: If you want the changes to the session parameters in the SPFILE be persistent across an instance failover, then Symantec recommends you to save the SPFILE on shared storage.

Specifying the SPFILE location in the PFILE

To specify the location of the SPFILE in a PFILE, create a PFILE and specify the following entry in the PFILE:

```
SPFILE = spfile_location
```

The variable *spfile_location* represents the complete path of the SPFILE. For example:

```
SPFILE = /database/startup/spfileoral.ora
```

In this case, to start the database use the following command:

```
startup pfile=location_of_pfile
```

Creating the SPFILE from a PFILE

The SPFILE must be created from the PFILE. You must have the `sysdba` or the `sysoper` system privileges to create an SPFILE.

You can run the following command to create the SPFILE:

```
CREATE SPFILE [= spfile_name] FROM PFILE [= pfile_name ];
```


If you do not specify the complete path for the SPFILE, this commands creates an SPFILE at the default location (\$ORACLE_HOME/db on Linux).

Index

Symbols

`$ORACLE_HOME`
location 23
on local disk 24
on shared disk 23

A

agent operations
 Netlsnr agent 16
 Oracle agent 11
attribute definitions
 Netlsnr agent 80
 Oracle agent 76
attributes
 editing 57
 modifying 57

B

basic monitoring 14
 health check 14
 process 14

C

cluster manager 55
configuration
 modifying 66
 typical setup 19
configuration files 85
configuration wizard 47
cssd daemon 29
 disabling 30

D

database tablespaces 24
detail monitoring 14
 about 60
 disabling for Netlsnr 63
 disabling for Oracle 62
 enabling for Netlsnr 63

enabling for Oracle 62
setting up 60

E

editing attributes 57
encrypting passwords 59
enterprise agent
 about 9
 configuring using cluster manager 55
 configuring using command-line 57
 configuring using wizard 47
 disabling 67
 error messages 69
 installing 35
 modifying configuration 66
 removing 68
 troubleshooting 69
 typical setup 19
 upgrading 36
error messages 69

F

failover 17

H

health check APIs 14
health check monitoring 14
hot backup 25

M

message log 69
monitoring
 basic 14
 detail 14
monitoring options 17

N

Netlsnr agent

- attribute definitions 80
 - operations 16
 - resource type 80
- Netlsnr agent attributes
 - AgentDebug 82
 - Encoding 82
 - EnvFile 82
 - Home 81
 - Listener 81
 - LsnrPwd 81
 - MonScript 82
 - Owner 81
 - TnsAdmin 81
- O**
 - offlining service group 66
 - onlining service group 65
 - operations
 - Netlsnr agent 16
 - Oracle agent 11
 - Oracle
 - database tablespaces 24
 - error handling 18
 - installing 21
 - multiple instances 24
 - shutdown options 13
 - startup options 12
 - supported configurations 39
 - Oracle agent
 - attribute definitions 76
 - operations 11
 - resource type 75
 - Oracle agent attributes
 - AgentDebug 79
 - AutoEndBkup 78
 - DetailMonitor 78
 - Encoding 79
 - EnvFile 77
 - Home 76
 - MonitorOption 78
 - MonScript 79
 - Owner 76
 - Pfile 78
 - Pword 79
 - ShutDownOpt 77
 - Sid 76
 - StartUpOpt 77
 - Table 79
 - User 79

- Oracle configuration
 - multiple instances (multiple listeners) 42
 - multiple instances (single listener) 88
 - single instance 85
- Oracle database, hot backup 25

P

- passwords, encrypting 59
- process monitoring 14

R

- removing agent 65
- resource attributes, editing 57
- resource type
 - Netlsnr 80
 - Oracle 75

S

- sample configuration files 85
- sample Oracle configuration 19
- service group
 - bringing online 65
 - switching 66
 - taking offline 66
- shutdown options 13
- SPFILE
 - creating 104
 - location of 104
 - using in VCS 103
- startup options 12
- switching service group 66

T

- troubleshooting 69

V

- vsencrypt utility 59
- virtual fire drill 16

W

- wizard 47