Veritas InfoScale™ 7.3.1 Disaster Recovery Implementation Guide - Linux

VERITAS™

Last updated: 2017-11-04

Legal Notice

Copyright © 2017 Veritas Technologies LLC. All rights reserved.

Veritas and the Veritas Logo are trademarks or registered trademarks of Veritas Technologies LLC or its affiliates in the U.S. and other countries. Other names may be trademarks of their respective owners.

This product may contain third party software for which Veritas is required to provide attribution to the third party (“Third Party Programs”). Some of the Third Party Programs are available under open source or free software licenses. The License Agreement accompanying the Software does not alter any rights or obligations you may have under those open source or free software licenses. Refer to the third party legal notices document accompanying this Veritas product or available at:

https://www.veritas.com/about/legal/license-agreements

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 Veritas Technologies LLC and its licensors, if any.

THE 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. VERITAS TECHNOLOGIES LLC 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 as defined in FAR 12.212 and subject to restricted rights as defined in FAR Section 52.227-19 "Commercial Computer Software - Restricted Rights" and DFARS 227.7202, et seq. "Commercial Computer Software and Commercial Computer Software Documentation," as applicable, and any successor regulations, whether delivered by Veritas as on premises or hosted services. Any use, modification, reproduction release, performance, display or disclosure of the Licensed Software and Documentation by the U.S. Government shall be solely in accordance with the terms of this Agreement.

Veritas Technologies LLC
500 E Middlefield Road
Mountain View, CA 94043

http://www.veritas.com

Technical Support

Technical Support maintains support centers globally. All support services will be delivered in accordance with your support agreement and the then-current enterprise technical support policies. For information about our support offerings and how to contact Technical Support, visit our website:

https://www.veritas.com/support

You can manage your Veritas account information at the following URL:

https://my.veritas.com

If you have questions regarding an existing support agreement, please email the support agreement administration team for your region as follows:

Worldwide (except Japan)	CustomerCare@veritas.com
Japan	CustomerCare_Japan@veritas.com

Documentation

Make sure that you have the current version of the documentation. Each document displays the date of the last update on page 2. The latest documentation is available on the Veritas website:

https://sort.veritas.com/documents

Documentation feedback

Your feedback is important to us. Suggest improvements or report errors or omissions to the documentation. Include the document title, document version, chapter title, and section title of the text on which you are reporting. Send feedback to:

doc.feedback@veritas.com

You can also see documentation information or ask a question on the Veritas community site:

http://www.veritas.com/community/

Veritas Services and Operations Readiness Tools (SORT)

Veritas Services and Operations Readiness Tools (SORT) is a website that provides information and tools to automate and simplify certain time-consuming administrative tasks. Depending on the product, SORT helps you prepare for installations and upgrades, identify risks in your datacenters, and improve operational efficiency. To see what services and tools SORT provides for your product, see the data sheet:

https://sort.veritas.com/data/support/SORT_Data_Sheet.pdf

Contents

Section 1 Introducing Storage Foundation and High Availability Solutions for disaster recovery 10

Chapter 1 About supported disaster recovery scenarios 11

About disaster recovery scenarios 11
About campus cluster configuration 14
VCS campus cluster requirements 14
How VCS campus clusters work 15
Typical VCS campus cluster setup 19
About replicated data clusters 21
How VCS replicated data clusters work 22
About global clusters 23
How VCS global clusters work 23
User privileges for cross-cluster operations 24
VCS global clusters: The building blocks 25
Disaster recovery feature support for components in the Veritas InfoScale product suite 33
Virtualization support for Storage Foundation and High Availability Solutions 7.3.1 products in replicated environments 34

Chapter 2 Planning for disaster recovery 36

Planning for cluster configurations 36
Planning a campus cluster setup 36
Planning a replicated data cluster setup 37
Planning a global cluster setup 38
Planning for data replication 38
Data replication options 38
Data replication considerations 39

Section 2 Implementing campus clusters 40

Chapter 3 Setting up campus clusters for VCS and SFHA 41

About setting up a campus cluster configuration 41
Preparing to set up a campus cluster configuration 41
Configuring I/O fencing to prevent data corruption 42
Configuring VxVM disk groups for campus cluster configuration 42
Configuring VCS service group for campus clusters 44
Setting up campus clusters for VxVM and VCS using Veritas InfoScale Operations Manager 45
Fire drill in campus clusters 46
About the DiskGroupSnap agent 47
About running a fire drill in a campus cluster 47
Configuring the fire drill service group 47
Running a successful fire drill in a campus cluster 48

Chapter 4 Setting up campus clusters for SFCFSHA, SFRAC 50

About setting up a campus cluster for disaster recovery for SFCFSHA or SF Oracle RAC 50
Preparing to set up a campus cluster in a parallel cluster database environment 53
Configuring I/O fencing to prevent data corruption 54
Configuring VxVM disk groups for a campus cluster in a parallel cluster database environment 56
Configuring VCS service groups for a campus cluster for SFCFSHA and SF Oracle RAC 60
Tuning guidelines for parallel campus clusters 61
Best practices for a parallel campus cluster 61

Section 3 Implementing replicated data clusters 63

Chapter 5 Configuring a replicated data cluster using VVR 64

About setting up a replicated data cluster configuration 64
About typical replicated data cluster configuration 64

About setting up replication 65
Configuring the service groups 66
Configuring the service group dependencies 67
About migrating a service group 67
Switching the service group 68
Fire drill in replicated data clusters 68

Chapter 6 Configuring a replicated data cluster using third-party replication 69

About setting up a replicated data cluster configuration using third-party replication 69
About typical replicated data cluster configuration using third-party replication 70
About setting up third-party replication 70
Configuring the service groups for third-party replication 71
Fire drill in replicated data clusters using third-party replication 71

Section 4 Implementing global clusters 72

Chapter 7 Configuring global clusters for VCS and SFHA 73

Installing and Configuring Cluster Server 73
Setting up VVR replication 73
About configuring VVR replication 74
Best practices for setting up replication 74
Creating a Replicated Data Set 76
Synchronizing the Secondary and starting replication 92
Starting replication when the data volumes are zero initialized 98
Setting up third-party replication 99
Configuring clusters for global cluster setup 99
Configuring global cluster components at the primary site 100
Installing and configuring VCS at the secondary site 101
Securing communication between the wide-area connectors 101
Configuring remote cluster objects 103
Configuring additional heartbeat links (optional) 103
Configuring the Steward process (optional) 104
Configuring service groups for global cluster setup 107
Configuring VCS service group for VVR-based replication 108
Configuring a service group as a global service group 111
Fire drill in global clusters 112

Chapter 8 Configuring a global cluster with Storage Foundation Cluster File System High Availability, Storage Foundation for Oracle RAC, or Storage Foundation for Sybase CE 113

About global clusters 114
About replication for parallel global clusters using Storage Foundation and High Availability (SFHA) Solutions 114
About setting up a global cluster environment for parallel clusters 115
Configuring the primary site 116
Configuring the secondary site 119
Configuring the Sybase ASE CE cluster on the secondary site 125
Setting up replication between parallel global cluster sites 125
Testing a parallel global cluster configuration 132

Chapter 9 Configuring global clusters with VVR and Storage Foundation Cluster File System High Availability, Storage Foundation for Oracle RAC, or Storage Foundation for Sybase CE 134

About configuring a parallel global cluster using Volume Replicator (VVR) for replication 135
Setting up replication on the primary site using VVR 137
Creating the data and SRL volumes on the primary site 137
Setting up the Replicated Volume Group on the primary site 138
Setting up replication on the secondary site using VVR 140
Creating the data and SRL volumes on the secondary site 141
Editing the /etc/vx/vras/.rdg files 141
Setting up IP addresses for RLINKs on each cluster 142
Setting up the disk group on secondary site for replication 143
Starting replication of the primary site database volume to the secondary site using VVR 145
Configuring Cluster Server to replicate the database volume using VVR 147
Modifying the Cluster Server (VCS) configuration on the primary site 152
Modifying the VCS configuration on the secondary site 159
Configuring the Sybase ASE CE cluster on the secondary site 165

Replication use cases for global parallel clusters 167

Section 5 Configuring disaster recovery in cloud environments 174

Chapter 10 Replication scenarios 175

About data replication in cloud environments 175
About the supported replication scenarios 176

Chapter 11 Replicating in AWS cloud environment 177

Replication from on-premise to AWS cloud 177
Replication across Availability Zones within the same AWS region 181
Replication across AWS regions 183
Replication across multiple AWS Availability Zones and regions (campus cluster) 186

Chapter 12 Replicating in Azure cloud environment 193

Replication from on-premise to Azure cloud 193
Replication within same Azure region 197
Replication across Azure regions 200
Replication across multiple sites and regions (campus cluster) 202
About identifying a temporary resource disk 204

Chapter 13 Replicating across clouds 205

Replication in cross-cloud environment 205

Section 6 Reference 206

Appendix A Sample configuration files 207

Sample Storage Foundation for Oracle RAC configuration files 207
sfrac02_main.cf file 207
sfrac07_main.cf and sfrac08_main.cf files 208
sfrac09_main.cf and sfrac10_main.cf files 210
sfrac11_main.cf file 213
sfrac12_main.cf and sfrac13_main.cf files 214
Sample fire drill service group configuration 217

About sample main.cf files for Storage Foundation (SF) for Oracle RAC 219
Sample main.cf for Oracle 10g for CVM/VVR primary site 220
Sample main.cf for Oracle 10g for CVM/VVR secondary site 225
About sample main.cf files for Storage Foundation (SF) for Sybase ASE CE 230
Sample main.cf for a basic Sybase ASE CE cluster configuration under VCS control with shared mount point on CFS for Sybase binary installation 230
Sample main.cf for a basic Sybase ASE CE cluster configuration with local mount point on VxFS for Sybase binary installation 235
Sample main.cf for a primary CVM VVR site 239
Sample main.cf for a secondary CVM VVR site 246

Index 252

Section 1

Introducing Storage Foundation and High Availability Solutions for disaster recovery

- Chapter 1. About supported disaster recovery scenarios
- Chapter 2. Planning for disaster recovery

Chapter 1

About supported disaster recovery scenarios

This chapter includes the following topics:

- About disaster recovery scenarios
- About campus cluster configuration
- About replicated data clusters
- About global clusters
- Disaster recovery feature support for components in the Veritas InfoScale product suite
- Virtualization support for Storage Foundation and High Availability Solutions 7.3.1 products in replicated environments

About disaster recovery scenarios

Storage Foundation offers cost–effective, short-distance disaster recovery with active configurations and long distance replication solutions to effectively manage disaster recovery requirements.

This guide describes how to configure campus clusters, global clusters, and replicated data clusters (RDC) for disaster recovery failover using the following Storage Foundation and High Availability Solutions products:

- Storage Foundation Cluster File System High Availability (SFCFSHA)
- Storage Foundation™ for Oracle® RAC (SF Oracle RAC)
- Storage Foundation™ for Sybase® Adaptive Server Enterprise Cluster Edition (SF Sybase CE)

- Cluster Server (VCS)
- Volume Replicator (VVR)

Note: Disaster recovery scenarios for SF Sybase CE have only been tested and are only supported using Veritas Cluster Volume Manager and Volume Replicator.

See the *Storage Foundation Cluster File System High Availability Administrator's Guide* for more information on configuring SFCFSHA.

See the *Storage Foundation for Oracle RAC Administrator's Guide* for more information on configuring SF Oracle RAC.

See the *Storage Foundation for SF Sybase ASE CE Administrator's Guide* for more information on configuring SF Sybase CE.

See the *Cluster Server Administrator's Guide* for more information on configuring VCS.

See the *Veritas InfoScale™ Replication Administrator's Guide* for more information on configuring VVR.

Table 1-1 lists key use cases for campus cluster, global cluster, and replicated data cluster disaster recovery configurations.

Table 1-1 Key use cases for disaster recovery configurations

Use case description	Recommended disaster recovery configuration
Disaster Recovery of business-critical applications from the production site to a geographically distributed Disaster Recovery (DR) site. ■ Distance between the two sites exceeds 80 KM or 50 miles ■ Application data is made available at the DR site through replication ■ Application is expected to be active at only one site at any point of time (Active/Passive)	Cluster Server HA/DR with global clustering See " How VCS global clusters work" on page 23. SFRAC with global clustering

Table 1-1 Key use cases for disaster recovery configurations *(continued)*

Use case description	Recommended disaster recovery configuration
Disaster Recovery of business-critical applications from the production site to a geographically distributed Disaster Recovery (DR) site. ■ Distance between the two sites is less than 80 KM or 50 miles ■ Application data is made available at the DR site through replication ■ Application is expected to be active at only one site at any point of time (Active/Passive) ■ Automatic application failover within a site, automated failover across sites	Cluster Server HA/DR with Replicated Data Cluster (RDC) See " How VCS replicated data clusters work" on page 22.
Disaster Recovery of business-critical applications from the production site to a geographically distributed Disaster Recovery (DR) site. ■ Distance between the two sites is less than 80 KM or 50 miles ■ Application data is made available at the DR site through remote mirroring ■ Application is expected to be active at only one site at any point of time (Active/Passive) ■ Automatic application failover within a site, automated failover across sites	Cluster Server HA/DR with Campus Cluster See " How VCS campus clusters work" on page 15. Storage Foundation for remote mirroring
High Availability of business-critical applications across two geographically distributed sites. ■ Distance between the two sites is less than 80 KM or 50 miles ■ Application data is made available at the DR site through remote mirroring ■ Application is expected to be simultaneously active at both the sites (Active/Active)	Cluster Server with Campus Cluster See " How VCS campus clusters work" on page 15. Storage Foundation Cluster File System High Availability for remote mirroring and parallel cross-site access SFRAC with Campus Cluster for remote mirroring and parallel cross-site access

About campus cluster configuration

The campus cluster configuration provides local high availability and disaster recovery functionality in a single VCS cluster. This configuration uses data mirroring to duplicate data at different sites. There is no Host or Array base replication involved.

VCS supports campus clusters that employ disk groups mirrored with Veritas Volume Manager.

VCS campus cluster requirements

Review the following requirements for VCS campus clusters:

- You must install VCS.
- You must have a single VCS cluster with at least one node in each of the two sites, where the sites are separated by a physical distance of no more than 80 kilometers. When the sites are separated more than 80 kilometers, you can run Global Cluster Option (GCO) configuration.
- You must have redundant network connections between nodes. All paths to storage must also be redundant.

  Veritas recommends the following in a campus cluster setup:

  - A common cross-site physical infrastructure for storage and LLT private networks.

    Veritas recommends a common cross-site physical infrastructure for storage and LLT private networks
  - Technologies such as Dense Wavelength Division Multiplexing (DWDM) for network and I/O traffic across sites. Use redundant links to minimize the impact of network failure.
- You must install Veritas Volume Manager with the FMR license and the Site Awareness license.
- Veritas recommends that you configure I/O fencing to prevent data corruption in the event of link failures.

  See the Cluster Server Configuration and Upgrade Guide for more details.
- You must configure storage to meet site-based allocation and site-consistency requirements for VxVM.
  - All the nodes in the site must be tagged with the appropriate VxVM site names.
  - All the disks must be tagged with the appropriate VxVM site names.

- The VxVM site names of both the sites in the campus cluster must be added to the disk groups.
- The allsites attribute for each volume in the disk group must be set to on. (By default, the value is set to on.)
- The siteconsistent attribute for the disk groups must be set to on.

- Each host at a site must be connected to a storage switch. The switch must have access to storage arrays at all the sites..
- In environments where the Flexible Storage Sharing (FSS) feature is being used, nodes may not be connected to an external storage switch. Locally attached storage needs to be exported to enable network sharing with nodes within and across the site. For more details on FSS, see the *Storage Foundation Cluster File System High Availability Administrator's Guide*.
- SF Oracle RAC campus clusters require mirrored volumes with storage allocated from both sites.

How VCS campus clusters work

This topic describes how VCS works with VxVM to provide high availability in a campus cluster environment.

In a campus cluster setup, VxVM automatically mirrors volumes across sites. To enhance read performance, VxVM reads from the plexes at the local site where the application is running. VxVM writes to plexes at both the sites.

In the event of a storage failure at a site, VxVM detaches all the disks at the failed site from the disk group to maintain data consistency. When the failed storage comes back online, VxVM automatically reattaches the site to the disk group and recovers the plexes.

See the *Storage Foundation Cluster File System High Availability Administrator's Guide* for more information.

When service group or system faults occur, VCS fails over service groups based on the values you set for the cluster attribute SiteAware and the service group attribute AutoFailOver.

For campus cluster setup, you must define sites and add systems to the sites that you defined. A system can belong to only one site. Sit e definitions are uniform across VCS, You can define sites Veritas InfoScale Operations Manager, and VxVM. You can define site dependencies to restrict connected applications to fail over within the same site.

You can define sites by using:

- Veritas InfoScale Operations Manager

For more information on configuring sites, see the latest version of the *Veritas InfoScale Operations Manager User guide*.

Depending on the value of the AutoFailOver attribute, VCS failover behavior is as follows:

0	VCS does not fail over the service group.
1	VCS fails over the service group to another suitable node. By default, the AutoFailOver attribute value is set to 1.
2	VCS fails over the service group if another suitable node exists in the same site. Otherwise, VCS waits for administrator intervention to initiate the service group failover to a suitable node in the other site. This configuration requires the HA/DR license enabled. Veritas recommends that you set the value of AutoFailOver attribute to 2.

Sample definition for these service group attributes in the VCS main.cf is as follows:

```
cluster VCS_CLUS (
        PreferredFencingPolicy = Site
        SiteAware = 1
        )
site MTV (
        SystemList = { sys1, sys2 }
        )
site SFO (
        Preference = 2
        SystemList = { sys3, sys4 }
        )
```

The sample configuration for hybrid_group with AutoFailover = 1 and failover_group with AutoFailover = 2 is as following:

```
hybrid_group (
    Parallel = 2
    SystemList = { sys1 = 0, sys2 = 1, sys3 = 2, sys4 = 3 }
)

failover_group (
    AutoFailover = 2
    SystemList = { sys1 = 0, sys2 = 1, sys3 = 2, sys4 = 3 }
)
```

Table 1-2 lists the possible failure scenarios and how VCS campus cluster recovers from these failures.

Table 1-2 Failure scenarios in campus cluster

Failure	Description and recovery
Node failure	■ A node in a site fails. If the value of the AutoFailOver attribute is set to 1, VCS fails over the service group to another system within the same site defined for cluster or SystemZone defined by SystemZones attribute for the service group or defined by Veritas InfoScale Operations Manager. ■ All nodes in a site fail. If the value of the AutoFailOver attribute is set to 0, VCS requires administrator intervention to initiate a fail over in both the cases of node failure.
Application failure	The behavior is similar to the node failure.
Storage failure - one or more disks at a site fails	VCS does not fail over the service group when such a storage failure occurs. VxVM detaches the site from the disk group if any volume in that disk group does not have at least one valid plex at the site where the disks failed. VxVM does not detach the site from the disk group in the following cases: ■ None of the plexes are configured on the failed disks. ■ Some of the plexes are configured on the failed disks, and at least one plex for a volume survives at each site. If only some of the disks that failed come online and if the vxrelocd daemon is running, VxVM relocates the remaining failed disks to any available disks. Then, VxVM automatically reattaches the site to the disk group and resynchronizes the plexes to recover the volumes. If all the disks that failed come online, VxVM automatically reattaches the site to the disk group and resynchronizes the plexes to recover the volumes.
Storage failure - all disks at both sites fail	VCS acts based on the DiskGroup agent's PanicSystemOnDGLoss attribute value. See the *Cluster Server Bundled Agents Reference Guide* for more information.

Table 1-2 Failure scenarios in campus cluster *(continued)*

Failure	Description and recovery
Site failure	All nodes and storage at a site fail. Depending on the value of the AutoFailOver attribute, VCS fails over the service group as follows: ■ If the value is set to 1, VCS fails over the service group to a system. ■ If the value is set to 2, VCS requires administrator intervention to initiate the service group failover to a system in the other site. Because the storage at the failed site is inaccessible, VCS imports the disk group in the application service group with all devices at the failed site marked as NODEVICE. When the storage at the failed site comes online, VxVM automatically reattaches the site to the disk group and resynchronizes the plexes to recover the volumes.
Network failure (LLT interconnect failure)	Nodes at each site lose connectivity to the nodes at the other site The failure of all private interconnects between the nodes can result in split brain scenario and cause data corruption. Review the details on other possible causes of split brain and how I/O fencing protects shared data from corruption. Veritas recommends that you configure I/O fencing to prevent data corruption in campus clusters. When the cluster attribute PreferredFencingPolicy is set as Site, the fencing driver gives preference to the node with higher site priority during the race for coordination points. VCS uses the site-level attribute Preference to determine the node weight.

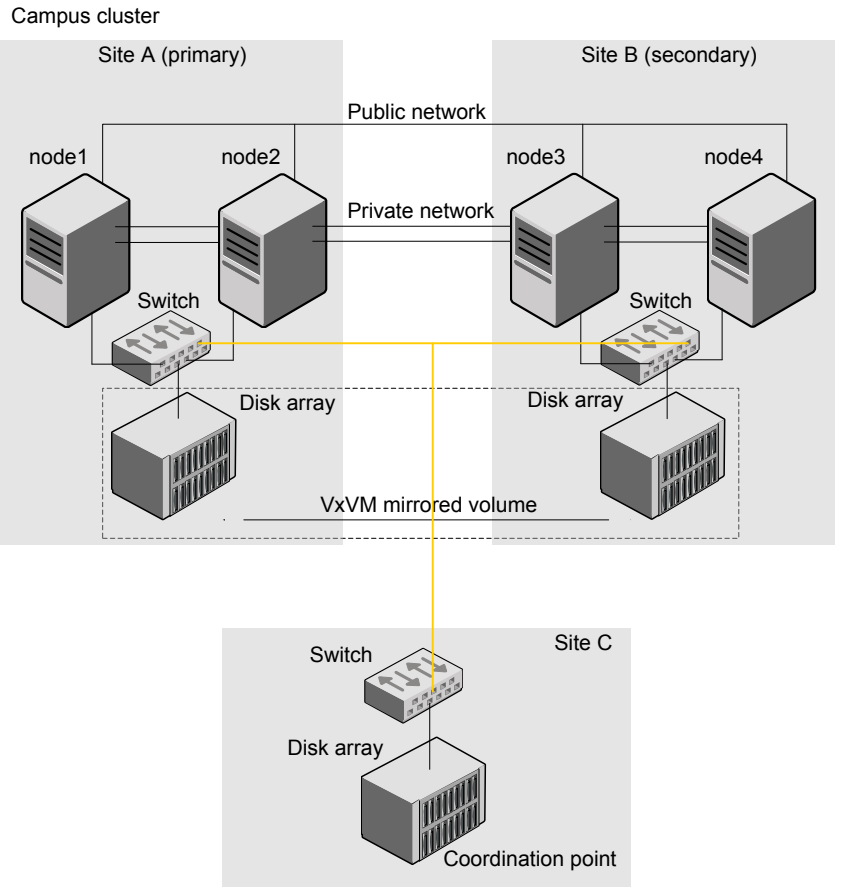
Table 1-2 Failure scenarios in campus cluster *(continued)*

Failure	Description and recovery
Network failure (LLT and storage interconnect failure)	Nodes at each site lose connectivity to the storage and the nodes at the other site Veritas recommends that you configure I/O fencing to prevent split brain and serial split brain conditions. ■ If I/O fencing is configured: The site that do not win the race triggers a system panic. When you restore the network connectivity, VxVM detects the storage at the failed site, reattaches the site to the disk group, and resynchronizes the plexes to recover the volumes. ■ If I/O fencing is not configured: If the application service group was online at site A during such failure, the application service group remains online at the same site. Because the storage is inaccessible, VxVM detaches the disks at the failed site from the disk group. At site B where the application service group is offline, VCS brings the application service group online and imports the disk group with all devices at site A marked as NODEVICE. So, the application service group is online at both the sites and each site uses the local storage. This causes inconsistent data copies and leads to a site-wide split brain. When you restore the network connectivity between sites, a serial split brain may exist. See the *Storage Foundation Administrator's Guide* for details to recover from a serial split brain condition.

Typical VCS campus cluster setup

Figure 1-1 depicts a typical VCS campus cluster setup.

Figure 1-1 Typical VCS campus cluster setup



VCS campus cluster typically has the following characteristics:

- Single VCS cluster spans multiple sites.
  In the sample figure, VCS is configured on four nodes: node 1 and node 2 are located at site A and node 3 and node 4 at site B.
- I/O fencing is configured with one coordinator disk from each site of the campus cluster and another coordinator disk from a third site.
  Figure 1-1 illustrates a typical setup with disk-based I/O fencing. You can also configure server-based I/O fencing.
  Mix mode fencing with two coordinator disks from each site and a CP server on third site is also supported.

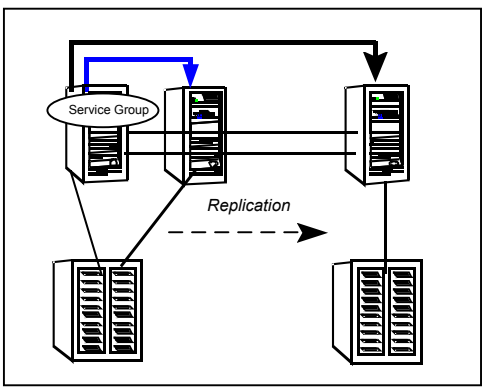
- The shared data is located on mirrored volumes on a disk group configured using Veritas Volume Manager.
- The volumes that are required for the application have mirrors on both the sites.
- All nodes in the cluster are tagged with the VxVM site name. All disks that belong to a site are tagged with the corresponding VxVM site name.
- The disk group is configured in VCS as a resource of type DiskGroup and is mounted using the Mount resource type.

About replicated data clusters

In a replicated data cluster no shared disks exist. Instead, a data replication product synchronizes copies of data between nodes or sites. Replication can take place at the application, host, and storage levels. Application-level replication products, such as Oracle DataGuard, maintain consistent copies of data between systems at the SQL or database levels. Host-based replication products, such as Veritas Volume Replicator, maintain consistent storage at the logical volume level. Storage-based or array-based replication maintains consistent copies of data at the disk or RAID LUN level.

Figure 1-2 shows a hybrid shared storage and replicated data cluster, in which different failover priorities are assigned to nodes according to particular service groups.

Figure 1-2 Shared storage replicated data cluster



You can also configure replicated data clusters without the ability to fail over locally, but this configuration is not recommended.

See “ How VCS replicated data clusters work” on page 22.

How VCS replicated data clusters work

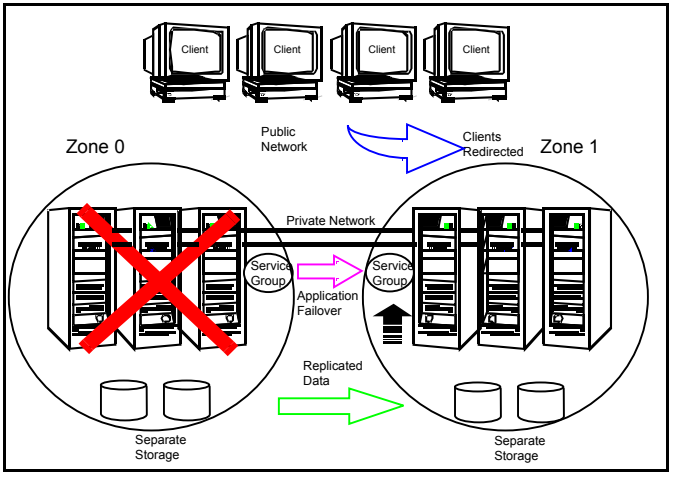
To understand how a replicated data cluster configuration works, let us take the example of an application configured in a VCS replicated data cluster. The configuration has two system zones:

- Primary zone (zone 0) comprising nodes located at the primary site and attached to the primary storage
- Secondary zone (zone 1) comprising nodes located at the secondary site and attached to the secondary storage

The application is installed and configured on all nodes in the cluster. Application data is located on shared disks within each RDC site and is replicated across RDC site to ensure data concurrency. The application service group is online on a system in the current primary zone and is configured to fail over in the cluster.

Figure 1-3 depicts an application configured on a VCS replicated data cluster.

Figure 1-3 A VCS replicated data cluster configuration



In the event of a system or application failure, VCS attempts to fail over the application service group to another system within the same RDC site. However, in the event that VCS fails to find a failover target node within the primary RDC site, VCS switches the service group to a node in the current secondary RDC site (zone 1).

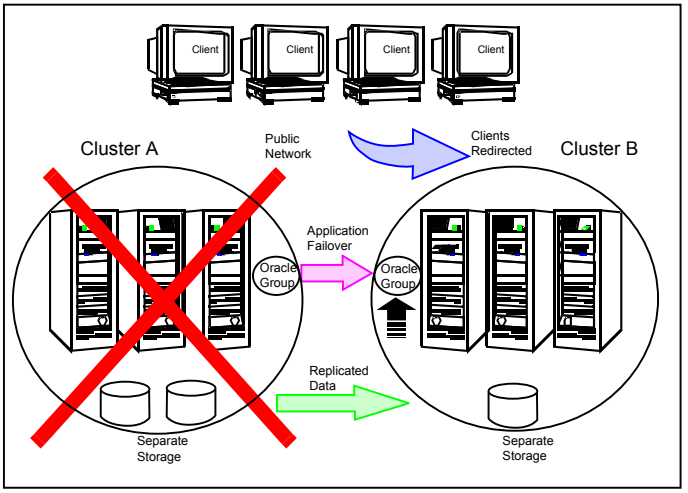
About global clusters

A global cluster links clusters at separate locations and enables wide-area failover and disaster recovery.

Local clustering provides local failover for each site or building. Campus and replicated cluster configurations offer protection against disasters that affect limited geographic regions. Large scale disasters such as major floods, hurricanes, and earthquakes can cause outages for an entire city or region. In such situations, you can ensure data availability by migrating applications to sites located considerable distances apart.

Figure 1-4 shows a global cluster configuration.

Figure 1-4 Global cluster



In a global cluster, if an application or a system fails, the application is migrated to another system within the same cluster. If the entire cluster fails, the application is migrated to a system in another cluster. Clustering on a global level also requires the replication of shared data to the remote site.

How VCS global clusters work

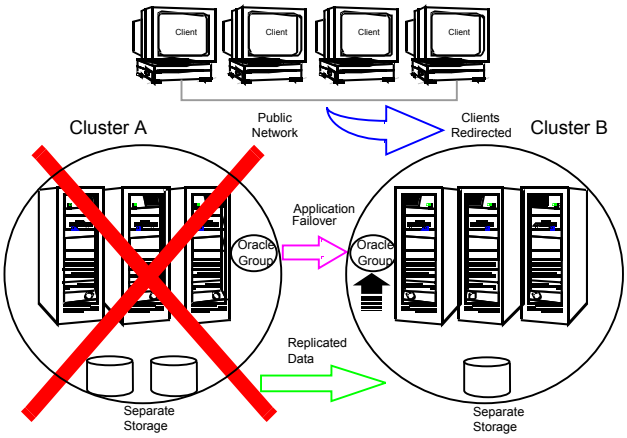
Local clustering provides local failover for each site or building. But, these configurations do not provide protection against large-scale disasters such as major floods, hurricanes, and earthquakes that cause outages for an entire city or region. The entire cluster could be affected by an outage.

In such situations, VCS global clusters ensure data availability by migrating applications to remote clusters located considerable distances apart.

Let us take the example of an Oracle database configured in a VCS global cluster. Oracle is installed and configured in both clusters. Oracle data is located on shared disks within each cluster and is replicated across clusters to ensure data concurrency. The Oracle service group is online on a system in cluster A and is configured to fail over globally, on clusters A and B.

Figure 1-5 shows a sample global cluster setup.

Figure 1-5 Sample global cluster setup



VCS continuously monitors and communicates events between clusters. Inter-cluster communication ensures that the global cluster is aware of the state of the service groups that are configured in the global cluster at all times.

In the event of a system or application failure, VCS fails over the Oracle service group to another system in the same cluster. If the entire cluster fails, VCS fails over the service group to the remote cluster, which is part of the global cluster. VCS also redirects clients once the application is online on the new location.

User privileges for cross-cluster operations

A user or user group can perform a cross-cluster online or offline operation only if they have one of the following privileges on the remote cluster:

- Group administrator or group operator privileges for the group
- Cluster administrator or cluster operator privileges

A user or user group can perform a cross-cluster switch operation only if they have one of the following privileges on both the clusters:

- Group administrator or group operator privileges for the group
- Cluster administrator or cluster operator privileges

VCS global clusters: The building blocks

VCS extends clustering concepts to wide-area high availability and disaster recovery with the following:

- Remote cluster objects
  See " Visualization of remote cluster objects" on page 25.
- Global service groups
  See "About global service groups" on page 25.
- Global cluster management
  See "About global cluster management" on page 26.
- Serialization
  See "About serialization–The Authority attribute" on page 27.
- Resiliency and right of way
  See "About resiliency and "Right of way"" on page 28.
- VCS agents to manage wide-area failover
  See " VCS agents to manage wide-area failover" on page 28.
- Split-brain in two-cluster global clusters
  See "About the Steward process: Split-brain in two-cluster global clusters" on page 31.
- Secure communication
  See " Secure communication in global clusters" on page 32.

Visualization of remote cluster objects

VCS enables you to visualize remote cluster objects using any of the supported components that are used to administer VCS such as VCS CLI and Veritas InfoScale Operations Manager

You can define remote clusters in your configuration file, `main.cf`. The Remote Cluster Configuration wizard provides an easy interface to do so. The wizard updates the `main.cf` files of all connected clusters with the required configuration changes.

About global service groups

A global service group is a regular VCS group with additional properties to enable wide-area failover. The global service group attribute ClusterList defines the list of clusters to which the group can fail over. The service group must be configured on

all participating clusters and must have the same name on each cluster. The Global Group Configuration Wizard provides an easy interface to configure global groups.

VCS agents manage the replication during cross-cluster failover.

See " VCS agents to manage wide-area failover" on page 28.

About global cluster management

VCS enables you to perform operations (online, offline, switch) on global service groups from any system in any cluster. You must log on with adequate privileges for cluster operations.

See "User privileges for cross-cluster operations" on page 24.

You can bring service groups online or switch them to any system in any cluster. If you do not specify a target system, VCS uses the FailOverPolicy to determine the system.

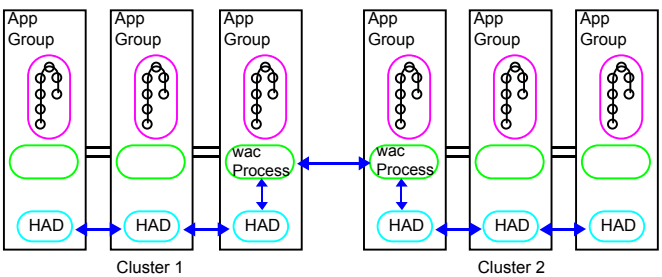
Management of remote cluster objects is aided by inter-cluster communication enabled by the wide-area connector (wac) process.

About the wide-area connector process

The wide-area connector (wac) is a failover Application resource that ensures communication between clusters.

Figure 1-6 is an illustration of the wide-area connector process.

Figure 1-6 Wide-area connector (wac) process



The wac process runs on one system in each cluster and connects with peers in remote clusters. It receives and transmits information about the status of the cluster, service groups, and systems. This communication enables VCS to create a consolidated view of the status of all the clusters configured as part of the global cluster. The process also manages wide-area heartbeating to determine the health of remote clusters. The process also transmits commands between clusters and returns the result to the originating cluster.

VCS provides the option of securing the communication between the wide-area connectors.

See " Secure communication in global clusters" on page 32.

About the wide-area heartbeat agent

The wide-area heartbeat agent manages the inter-cluster heartbeat. Heartbeats are used to monitor the health of remote clusters. VCS wide-area hearbeat agents include Icmp and IcmpS. While other VCS resource agents report their status to VCS engine, heartbeat agents report their status directly to the WAC process. The heartbeat name must be the same as the heartbeat type name. You can add only one heartbeat of a specific heartbeat type.

See "Sample configuration for the wide-area heartbeat agent" on page 27.

You can create custom wide-area heartbeat agents. For example, the VCS replication agent for SRDF includes a custom heartbeat agent for Symmetrix arrays.

You can add heartbeats using the `hahb -add heartbeatname` command and change the default values of the heartbeat agents using the `hahb -modify` command.

Sample configuration for the wide-area heartbeat agent

Following is a sample configuration for the wide-area heartbeat agent:

```
Heartbeat Icmp (
    ClusterList = {priclus
    Arguments @Cpriclus =
{"10.209.134.1"
    )
```

About serialization–The Authority attribute

VCS ensures that global service group operations are conducted serially to avoid timing problems and to ensure smooth performance. The Authority attribute prevents a service group from coming online in multiple clusters at the same time. Authority is a persistent service group attribute and it designates which cluster has the right to bring a global service group online. The attribute cannot be modified at runtime.

If two administrators simultaneously try to bring a service group online in a two-cluster global group, one command is honored, and the other is rejected based on the value of the Authority attribute.

The attribute prevents bringing a service group online in a cluster that does not have the authority to do so. If the cluster holding authority is down, you can enforce a takeover by using the command `hagrp -online -force service_group`. This

command enables you to fail over an application to another cluster when a disaster occurs.

Note: A cluster assuming authority for a group does not guarantee the group will be brought online on the cluster. The attribute merely specifies the right to attempt bringing the service group online in the cluster. The presence of Authority does not override group settings like frozen, autodisabled, non-probed, and so on, that prevent service groups from going online.

You must seed authority if it is not held on any cluster.

Offline operations on global groups can originate from any cluster and do not require a change of authority to do so, because taking a group offline does not necessarily indicate an intention to perform a cross-cluster failover.

About the Authority and AutoStart attributes

The attributes Authority and AutoStart work together to avoid potential concurrency violations in multi-cluster configurations.

If the AutoStartList attribute is set, and if a group's Authority attribute is set to 1, the VCS engine waits for the wac process to connect to the peer. If the connection fails, it means the peer is down and the AutoStart process proceeds. If the connection succeeds, HAD waits for the remote snapshot. If the peer is holding the authority for the group and the remote group is online (because of takeover), the local cluster does not bring the group online and relinquishes authority.

If the Authority attribute is set to 0, AutoStart is not invoked.

About resiliency and "Right of way"

VCS global clusters maintain resiliency using the wide-area connector process and the ClusterService group. The wide-area connector process runs as long as there is at least one surviving node in a cluster.

The wide-area connector, its alias, and notifier are components of the ClusterService group.

VCS agents to manage wide-area failover

VCS agents now manage external objects that are part of wide-area failover. These objects include replication, DNS updates, and so on. These agents provide a robust framework for specifying attributes and restarts, and can be brought online upon fail over.

DNS agent	The DNS agent updates the canonical name-mapping in the domain name server after a wide-area failover. See the *Cluster Server Bundled Agents Reference Guide* for more information.

VCS agents for Volume Replicator

You can use the following VCS agents for Volume Replicator in a VCS global cluster setup:

- RVG agent
  The RVG agent manages the Replicated Volume Group (RVG). Specifically, it brings the RVG online, monitors read-write access to the RVG, and takes the RVG offline. Use this agent when using Volume Replicator for replication.
- RVGPrimary agent
  The RVGPrimary agent attempts to migrate or take over a Secondary site to a Primary site following an application failover. The agent has no actions associated with the offline and monitor routines.
- RVGShared agent
  The RVGShared agent monitors the RVG in a shared environment. This is a parallel resource. The RVGShared agent enables you to configure parallel applications to use an RVG in a cluster. The RVGShared agent monitors the RVG in a shared disk group environment.
- RVGSharedPri agent
  The RVGSharedPri agent enables migration and takeover of a Volume Replicator replicated data set in parallel groups in a VCS environment. Bringing a resource of type RVGSharedPri online causes the RVG on the local host to become a primary if it is not already.
- RVGLogowner agent
  The RVGLogowner agent assigns and unassigns a node as the logowner in the CVM cluster; this is a failover resource. The RVGLogowner agent assigns or unassigns a node as a logowner in the cluster. In a shared disk group environment, currently only the cvm master node should be assigned the logowner role.
- RVGSnapshot agent
  The RVGSnapshot agent, used in fire drill service groups, takes space-optimized snapshots so that applications can be mounted at secondary sites during a fire drill operation.
  See the *Veritas InfoScale™ Replication Administrator's Guide* for more information.

In a CVM environment, the RVGShared agent, RVGSharedPri agent, RVGLogOwner agent, and RVGSnapshot agent are supported. For more information, see the *Cluster Server Bundled Agents Reference Guide*.

VCS agents for third-party replication technologies	VCS provides agents for other third-party array-based or application-based replication solutions. These agents are available in the High Availability Agent Pack software. See the agent pack documentation for a list of replication technologies that VCS supports.

About the Steward process: Split-brain in two-cluster global clusters

Failure of all heartbeats between any two clusters in a global cluster indicates one of the following:

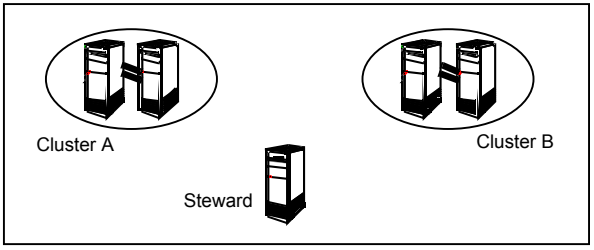
- The remote cluster is faulted.
- All communication links between the two clusters are broken.

In global clusters with three or more clusters, VCS queries the connected clusters to confirm that the remote cluster is truly down. This mechanism is called inquiry.

In a two-cluster setup, VCS uses the Steward process to minimize chances of a wide-area split-brain. The process runs as a standalone binary on a system outside of the global cluster configuration.

Figure 1-7 depicts the Steward process to minimize chances of a split brain within a two-cluster setup.

Figure 1-7 Steward process: Split-brain in two-cluster global clusters



When all communication links between any two clusters are lost, each cluster contacts the Steward with an inquiry message. The Steward sends an ICMP ping to the cluster in question and responds with a negative inquiry if the cluster is running or with positive inquiry if the cluster is down. The Steward can also be used in configurations with more than two clusters. VCS provides the option of securing communication between the Steward process and the wide-area connectors.

See “ Secure communication in global clusters” on page 32.

In non-secure configurations, you can configure the steward process on a platform that is different to that of the global cluster nodes. Secure configurations have not been tested for running the steward process on a different platform.

For example, you can run the steward process on a Windows system for a global cluster running on Linux systems. However, the VCS release for Linux contains the steward binary for Linux only. You must copy the steward binary for Windows from the VCS installation directory on a Windows cluster, typically `C:\Program Files\VERITAS\Cluster Server`.

A Steward is effective only if there are independent paths from each cluster to the host that runs the Steward. If there is only one path between the two clusters, you must prevent split-brain by confirming manually via telephone or some messaging system with administrators at the remote site if a failure has occurred. By default, VCS global clusters fail over an application across cluster boundaries with administrator confirmation. You can configure automatic failover by setting the ClusterFailOverPolicy attribute to Auto.

For more information on configuring the Steward process, see the *Cluster Server Administrator's Guide*.

The default port for the steward is 14156.

Secure communication in global clusters

In global clusters, VCS provides the option of making the following types of communication secure:

- Communication between the wide-area connectors.
- Communication between the wide-area connectors and the Steward process.

For secure authentication, the wide-area connector process gets a security context as an account in the local authentication broker on each cluster node.

The WAC account belongs to the same domain as HAD and Command Server and is specified as:

```
name = WAC
domain = VCS_SERVICES@cluster_uuid
```

You must configure the wide-area connector process in all clusters to run in secure mode. If the wide-area connector process runs in secure mode, you must run the Steward in secure mode.

Disaster recovery feature support for components in the Veritas InfoScale product suite

Table 1-3 lists high availability and disaster recovery features available for the components in the Veritas InfoScale product suite.

Table 1-3 Feature support for components in the Veritas InfoScale product suite

High availability and disaster recovery features	VCS	VCS HA/DR	SFHA	SFCFS HA	SFRAC	SF Syb CE
Clustering for high availability (HA)	Y	Y	Y	Y	Y	Y
Database and application/ISV agents	Y	Y	Y	Y	Y	Y
Advanced failover logic	Y	Y	Y	Y	Y	Y
Data integrity protection with I/O fencing	Y	Y	Y	Y	Y	Y
Advanced virtual machines support	Y	Y	Y	Y	Y	N
Virtual Business Services	Y	Y	Y	Y	Y	N
Replication agents for VVR	N	Y	O	O	O	O
Replication agents for third-party array-based replication	N	Y	O	O	O	N
Replicated Data Cluster	N	Y	O	O	N	N
Campus or stretch cluster	N	Y	O	O	O	N
Global clustering using VVR	N	Y	O	O	O	O
Global clustering using third-party array-based replication	N	Y	O	O	O	N
Fire Drill	N	Y	O	O	O	N

- Y=Feature is included in your license.

- O=Feature is not included in your license but may be licensed separately.
- N=Feature is not supported with your license.

The following components support multiple third-party replication options:

- Cluster Server (VCS)
- Storage Foundation High Availability (SFHA)
- Storage Foundation Cluster File System High Availability (SFCFSHA)
- Storage Foundation for Oracle RAC (SF Oracle RAC)

Storage Foundation for Sybase CE supports VVR replication only at this time.

For current information on third-party replication support:

See: https://sort.veritas.com/agents and select **Replication Agents** under **Agent type**.

Table 1-4 Replication support for databases across components in the Veritas InfoScale product suite

Database replication support	VCS	VCS HA/DR	SFHA	SFCFS HA	SFRAC	SF Syb CE
DB2	Y	Y	Y	Y	N	N
Single instance Oracle	Y	Y	Y	Y	Y	N
Oracle RAC	N	N	N	N	Y	N
Sybase	Y	Y	Y	Y	N	N
Syabase ASE CE	N	N	N	N	N	Y

Single instance Oracle and Oracle RAC replication support includes Storage Foundation for Databases (SFDB) tools replication support.

Virtualization support for Storage Foundation and High Availability Solutions 7.3.1 products in replicated environments

All VCS supported replication agents which are listed on SORT are supported inside the guest (virtual to virtual clustering). The in-guest support matrix for replication agent is the same as the VCS support matrix. If VCS is supported inside a guest on a virtualization platform, then the replication agent is also supported.

For Linux:

Supported Linux virtualization technologies:

- Kernel-based Virtual Machine (KVM) technology for Red Hat Enterprise Linux (RHEL) and SUSE Linux Enterprise Server (SLES)
- Red Hat Enterprise Virtualization (RHEV)
- Microsoft Hyper-V
- Oracle Virtual Machine (OVM)

Pre-requisite for replication agent support in virtual environments:

Data disks must be made visible to the guest as pass-through devices, such as VMware to RDM, LDOM to Physical Disk or Disk Lun without slice option.

Exception:

Firedrill functionality is not supported in virtual environments for the following replication agents:

- EMC MirrorView
- HP-UX EVA CA

Only Firedrill functionality is affected: these replication agents can be used to manage replication inside guests.

Chapter 2

Planning for disaster recovery

This chapter includes the following topics:

- Planning for cluster configurations
- Planning for data replication

Planning for cluster configurations

Storage Foundation and High Availability Solutions provides various disaster recovery configurations, such as campus clusters, global clusters for multi-site clusters. In multi-site clusters, the nodes can be placed in different parts of a building, in separate buildings, or in separate cities. The distance between the nodes depends on the type of disaster from which protection is needed and on the technology used to replicate data. Storage Foundation and High Availability supports various replication technologies for data replication.

To protect clusters against outages caused by disasters, the cluster components must be geographically separated.

Planning a campus cluster setup

A campus cluster is also known as a stretch cluster or remote mirror configuration. In a campus cluster, the hosts and storage of a cluster span multiple sites separated by a few miles.

Keep in mind the following best practices when you configure a Storage Foundation campus cluster:

- Campus cluster sites are typically connected using a redundant high-capacity network that provides access to storage and private network communication

between the cluster nodes. A single DWDM link can be used for both storage and private network communication.

- Tag the disks or enclosures that belong to a site with the corresponding VxVM site name. VxVM allocates storage from the correct site when creating or resizing a volume and when changing a volume's layout if the disks in the VxVM disk group that contain the volume are tagged with the site name.
- Tag each host with the corresponding VxVM site name. Make sure the read policy of the volumes is set to `SITEREAD`. This setting ensures that the reads on the volumes are satisfied from the local site's plex.
- Turn on the `allsites` attribute for all volumes that have data required by the application, to make sure they are evenly mirrored. Each site must have at least one mirror of all volumes hosting application data, including the FlashSnap log volume.
- Turn on the `siteconsistent` attribute for the disk groups and the volumes to enable site-aware plex detaches. Snapshot volumes need not be site-consistent.
- In the case of a two-site campus cluster, place the third coordinator disk on the third site. You may use iSCSI disk on the third site as an alternative to Dark Fiber connected FC-SAN or a Coordination Point Server (CPS), as a third coordination point.
- Make sure that a DCO log version 20 or higher is attached to the volumes to enable Fast Resync operations.
- Set the CVM disk detach policy as `global` or `local` for all disk groups containing data volumes.
  For OCR and voting disk, it is recommended to have the disk group policy as `local` detach policy.

Planning a replicated data cluster setup

The VCS replicated data cluster (RDC) configuration allows you to provide a robust and easy-to manage disaster recovery protection for your applications. For example you can convert a single instance database configured for local high availability in a VCS cluster to a disaster-protected RDC infrastructure using Volume Replicator or a supported third-party replication technology to replicate changed data.

Keep in mind the following best practicies when you configure an RDC:

- Make sure the sites and systems at each site are identified correctly for use when defining system zones in an RDC.
- Make sure there are dual dedicated LLT links between the replicated nodes.

- Since the sites used in the RDC configuration are within metro limits, synchronous replication is typically used. Make sure the replication technology that you plan to use supports synchronous replication mode.

The RDC can also be configured using supported third-party replication technologies.

See "Planning for data replication" on page 38.

Planning a global cluster setup

Global clusters provide the ability to fail over applications between geographically distributed clusters when a disaster occurs.

Global clustering involves two steps:

1. Replication of data between the sites
2. Configuring VCS clusters at the geographically distant sites and establishing a global cluster connection between them

The following aspects need to be considered when you design a disaster recovery solution:

- The amount of data lost in the event of a disaster (Recovery Point Objective)
- The acceptable recovery time after the disaster (Recovery Time Objective)

Planning for data replication

When planning for data replication, it is important to review the various hardware and software replication technologies and to review important considerations including the required level of data throughput.

Data replication options

Disaster recovery solutions support various hardware and software replication technologies.

Examples of hardware replication options	■ Hitachi True Copy ■ IBM Metro Mirror ■ IBM SVC ■ EMC Mirrorview
Examples of software replication options	■ Volume Replicator (VVR) ■ Oracle Data Guard

A complete list of supported replication technologies is listed on the Veritas Web site:

https://sort.veritas.com/agents

Data replication considerations

When you choose a replication solution, one of the important factors that you need to consider is the required level of data throughput. Data throughput is the rate at which the application is expected to write data. The impact of write operations on replication are of more significance than that of the read operations.

In addition to the business needs discussed earlier, the following factors need to be considered while choosing the replication options:

- Mode of replication
- Network bandwidth
- Network latency between the two sites
- Ability of the remote site to keep up with the data changes at the first site

Section 2

Implementing campus clusters

- Chapter 3. Setting up campus clusters for VCS and SFHA
- Chapter 4. Setting up campus clusters for SFCFSHA, SFRAC

Chapter 3

Setting up campus clusters for VCS and SFHA

This chapter includes the following topics:

- About setting up a campus cluster configuration
- Fire drill in campus clusters
- About the DiskGroupSnap agent
- About running a fire drill in a campus cluster

About setting up a campus cluster configuration

You must perform the following tasks to set up a campus cluster:

- Preparing to set up a campus cluster configuration
- Configuring I/O fencing to prevent data corruption
- Configuring VxVM disk groups for campus cluster configuration
- Configuring VCS service group for campus clusters

Preparing to set up a campus cluster configuration

Before you set up the configuration, review the VCS campus cluster requirements.

See " VCS campus cluster requirements" on page 14.

To prepare to set up a campus cluster configuration

1. Set up the physical infrastructure.
   - Set up access to the local storage arrays and to remote storage arrays on each node.
   - Set up private heartbeat network.

   See " Typical VCS campus cluster setup" on page 19.
2. Install VCS on each node to form a cluster with at least one node in each of the two sites.

   See the *Cluster Server Configuration and Upgrade Guide* for instructions.
3. Install VxVM on each node with the required licenses.

   See the *Storage Foundation and High Availability Configuration and Upgrade Guide* for instructions.

Configuring I/O fencing to prevent data corruption

Perform the following tasks to configure I/O fencing to prevent data corruption in the event of a communication failure.

See the *Cluster Server Configuration and Upgrade Guide* for more details.

To configure I/O fencing to prevent data corruption

1. Set up the storage at a third site.

   You can extend the DWDM to the third site to have FC SAN connectivity to the storage at the third site. You can also use iSCSI targets as the coordinator disks at the third site.
2. Set up I/O fencing.

Configuring VxVM disk groups for campus cluster configuration

Configure the campus cluster sites and configure VxVM disk groups for remote mirroring. You can also configure VxVM disk groups for remote mirroring using Veritas InfoScale Operations Manager.

Note: In cloud environments where one campus cluster is configured on one site and another on the second site, FSS volumes in campus cluster configurations can be used to replicate data for achieving high data availability across sites. In this FSS-campus cluster configuration, site tags can be added based on the name of the site to make data highly available during site failures.

See the *Storage Foundation Cluster File System High Availability Administrator's Guide* for more information on the VxVM commands.

To configure VxVM disk groups for campus cluster configuration

1 Set the site name for each host:

```
# vxdctl set site=sitename
```

The site name is stored in the /etc/vx/volboot file. Use the following command to display the site names:

```
# vxdctl list | grep siteid
```

2 Set the site name for all the disks in an enclosure:

```
# vxdisk settag site=sitename encl:enclosure
```

To tag specific disks, use the following command:

```
# vxdisk settag site=sitename disk
```

3 Verify that the disks are registered to a site.

```
# vxdisk listtag
```

4 Create a disk group with disks from both the sites.

```
# vxdg -s init diskgroup siteA_disk1 siteB_disk2
```

5 Configure site-based allocation on the disk group that you created for each site that is registered to the disk group.

```
# vxdg -g diskgroup addsite sitename
```

6 Configure site consistency on the disk group.

```
# vxdg -g diskgroup set siteconsistent=on
```

7 Create one or more mirrored volumes in the disk group.

```
# vxassist -g diskgroup make volume size nmirror=1/2
```

With the Site Awareness license installed on all hosts, the volume that you create has the following characteristics by default:

- The `allsites` attribute is set to `on`; the volumes have at least one plex at each site.
- The volumes are automatically mirrored across sites.
- The read policy `rdpol` is set to `siteread`.
- The volumes inherit the site consistency value that is set on the disk group.

Configuring VCS service group for campus clusters

Follow the procedure to configure the disk groups under VCS control and set up the VCS attributes to define failover in campus clusters.

To configure VCS service groups for campus clusters

1. Create a VCS service group (app_sg) for the application that runs in the campus cluster.

   ```
   hagrp -add app_sg
   hagrp -modify app_sg SystemList node1 0 node2 1 node3 2 node4 3
   ```

2. Set up the system zones or sites. Configure the SystemZones attribute for the service group. Skip this step when sites are configured through Veritas InfoScale Operations Manager.

   ```
   hagrp -modify app_sg SystemZones node1 0 node2 0 node3 1 node4 1
   ```

3. Set up the group fail over policy. Set the value of the AutoFailOver attribute for the service group.

   ```
   hagrp -modify app_sg AutoFailOver 2
   ```

4. For the disk group you created for campus clusters, add a DiskGroup resource to the VCS service group app_sg.

   ```
   hares -add dg_res1 DiskGroup app_sg
   hares -modify dg_res1 DiskGroup diskgroup_name
   hares -modify dg_res1 Enabled 1
   ```

5. Configure the application and other related resources to the app_sg service group.
6. Bring the service group online.

Setting up campus clusters for VxVM and VCS using Veritas InfoScale Operations Manager

Before configuring VxVM and VCS disk groups for campus clusters using Veritas InfoScale Operations Manager, you must remove the site tag from the disk group or cluster, as well as remove any tagging set by Veritas InfoScale Operations Manager.

See the *Veritas InfoScale Operations Manager User guide* for more information on using Multi Site Management in Veritas InfoScale Operations Manager to manage campus clusters across Storage Foundation and Cluster Server objects.

To remove VxVM site tag information

1. Determine which site a host belongs to, and execute the following command on each host:

   ```
   # vxdctl list | grep siteid
   ```

2. Remove the defined site name from each host:

   ```
   # vxdctl [-F] unset site
   ```

   The `-F` option is required if any imported disk groups are registered to the site.

3. Optionally, turn off auto-tagging on the disk group:

   ```
   # vxdg -g diskgroup set autotagging=off
   ```

4. Verify if site consistency has been enabled for the disk group:

   ```
   # vxdg list diskgroup | grep siteconsistent
   ```

5. Turn off the site consistency requirement for the disk group:

   ```
   # vxdg -g diskgroup set siteconsistent=off
   ```

6. Identify the site record for each site with the disk group:

   ```
   # vxprint -g diskgroup | grep ^SR
   ```

7. Unregister the site record for each site with the disk group:

   ```
   # vxdg -g diskgroup rmsite sitename
   ```

8. List the site tags for the disk group:

   ```
   # vxdg listtag [diskgroup]
   ```

9 Remove the site tag from the disk group:

```
# vxdg [-g diskgroup] rmtag [encl:enclosure] site=sitename
```

10 Check which disks or enclosures are registered to the site:

```
# vxdisk [-g diskgroup] listtag
```

11 Remove the site tag from the disk or enclosure:

```
# vxdisk rmtag site=sitename disk|encl:enclosure
```

To remove VCS site tag information

1 Make the cluster configuration writable.

```
# haconf -makerw
```

2 List the site tagging information on the cluster.

```
# hasite -list
```

3 Remove the site tagging information from the cluster.

```
# hasite -modify sitename SystemList -delete -keys

# hasite -delete sitename
```

4 Dump the configuration.

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

Fire drill in campus clusters

Fire drill tests the disaster-readiness of a configuration by mimicking a failover without stopping the application and disrupting user access.

The process involves creating a fire drill service group, which is similar to the original application service group. Bringing the fire drill service group online on the remote node demonstrates the ability of the application service group to fail over and come online at the site, should the need arise.

Fire drill service groups do not interact with outside clients or with other instances of resources, so they can safely come online even when the application service

group is online. Conduct a fire drill only at the remote site; do not bring the fire drill service group online on the node hosting the original application.

About the DiskGroupSnap agent

The DiskGroupSnap agent verifies the VxVM disk groups and volumes for site awareness and disaster readiness in a campus cluster environment. To perform a fire drill in campus clusters, you must configure a resource of type DiskGroupSnap in the fire drill service group.

Note: To perform fire drill, the application service group must be online at the primary site.

During fire drill, the DiskGroupSnap agent does the following:

- For each node in a site, the agent correlates the value of the SystemZones attribute for the application service group to the VxVM site names for that node.
- For the disk group in the application service group, the agent verifies that the VxVM site tags are defined for the disk group.
- For the disk group in the application service group, the agent verifies that the disks at the secondary site are not tagged with the same VxVM site name as the disks at the primary site.
- The agent verifies that all volumes in the disk group have a plex at each site.

See the *Cluster Server Bundled Agents Reference Guide* for more information on the agent.

About running a fire drill in a campus cluster

This topic provides information on how to run a fire drill in campus clusters.

Do the following tasks to perform fire drill:

- Configuring the fire drill service group
- Running a successful fire drill in a campus cluster

Configuring the fire drill service group

This topic provides information on how to configure the fire drill service group.

To configure the fire drill service group

1. Configure a fire drill service group similar to the application service group with the following exceptions:
   - The AutoFailOver attribute must be set to 0.
   - Network-related resources must not be configured.
   - The disk group names for the DiskGroup and the Mount resources in the fire drill service group must be appended with "_fd".
     For example, if the value of the DiskGroup attribute in the application service group is ccdg, then the corresponding value in the fire drill service group must be ccdg_fd.
     If the value of the BlockDevice attribute for the Mount resource in the application service group is /dev/vx/dsk/ccdg/ccvol, then the corresponding value in the fire drill service group must be /dev/vx/dsk/ccdg_fd/ccvol.
2. Add a resource of type DiskGroupSnap. Define the TargetResName and the FDSiteName attributes for the DiskGroupSnap resource.

   See the *Cluster Server Bundled Agent Reference Guide* for attribute descriptions.
3. Create a dependency such that the DiskGroup resource depends on the DiskGroupSnap resource.
4. Create a group dependency such that the fire drill service group has an offline local dependency on the application service group.

Running a successful fire drill in a campus cluster

Bring the fire drill service group online on a node within the system zone that does not have the application running. Verify that the fire drill service group comes online. This action validates that your solution is configured correctly and the production service group will fail over to the remote site in the event of an actual failure (disaster) at the local site.

You must take the fire drill service group offline before you shut down the node or stop VCS locally on the node where the fire drill service group is online or where the disk group is online. Otherwise, after the node restarts you must manually reattach the fire drill site to the disk group that is imported at the primary site.

Note: For the applications for which you want to perform fire drill, you must set the value of the FireDrill attribute for those application resource types to 1. After you complete fire drill, reset the value to 0.

To run a successful fire drill

1. Override the FireDrill attribute at the resource level for resources of the fire drill service group.
2. Set the FireDrill attribute for the application resources to 1. This prevents the agent from reporting a concurrency violation, when the application service group and the fire drill service group are online at the same time.
3. Bring the fire drill service group online.

   If the fire drill service group does not come online, review the VCS engine log to troubleshoot the issues so that corrective action can be taken as necessary in the production service group.

   Warning: You must take the fire drill service group offline after you complete the fire drill so that the failover behavior of the application service group is not impacted. Otherwise, when a disaster strikes at the primary site, the application service group cannot fail over to the secondary site due to resource conflicts.

   ---

4. After you complete the fire drill, take the fire drill service group offline.
5. Reset the Firedrill attribute for the resource to 0.
6. Undo or override the Firedrill attribute from the resource level.

Chapter 4

Setting up campus clusters for SFCFSHA, SFRAC

This chapter includes the following topics:

- About setting up a campus cluster for disaster recovery for SFCFSHA or SF Oracle RAC
- Preparing to set up a campus cluster in a parallel cluster database environment
- Configuring I/O fencing to prevent data corruption
- Configuring VxVM disk groups for a campus cluster in a parallel cluster database environment
- Configuring VCS service groups for a campus cluster for SFCFSHA and SF Oracle RAC
- Tuning guidelines for parallel campus clusters
- Best practices for a parallel campus cluster

About setting up a campus cluster for disaster recovery for SFCFSHA or SF Oracle RAC

Campus clusters:

- Are connected using a high speed cable that guarantees network access between the nodes

- Provide local high availability and disaster recovery functionality in a single cluster
- Employ shared disk groups mirrored across sites with Veritas Volume Manager (VxVM)
- Are supported for Storage Foundation and High Availability (SFHA) Solutions products including Storage Foundation Cluster File System High Availability (SFCFSHA) and Storage Foundation (SF) for Oracle RAC

Note: Campus clusters are not supported for Storage Foundation for Sybase CE at this time.

The following high-level tasks illustrate the setup steps for a campus cluster in a parallel cluster database environment. The example values are given for SF for Oracle RAC and should be adapted for an SFCFSHA cluster using another database application.

The following high-level tasks illustrate the setup steps for a parallel campus cluster in an SF for Oracle RAC environment.

Table 4-1 Tasks for setting up a parallel campus cluster for disaster recovery

Task	Description
Prepare to set up campus cluster configuration	See "Preparing to set up a campus cluster in a parallel cluster database environment" on page 53.
Configure I/O fencing to prevent data corruption	See "Configuring I/O fencing to prevent data corruption" on page 42.
Prepare to install Oracle RAC Clusterware and database binaries	See the *Storage Foundation for Oracle RAC Configuration and Upgrade Guide*.
Configure VxVM disk groups for campus cluster	See "Configuring VxVM disk groups for a campus cluster in a parallel cluster database environment" on page 56.
Install Oracle RAC Clusterware and database binaries	For Oracle RAC, see the *Storage Foundation for Oracle RAC Configuration and Upgrade Guide*. For SFCFSHA, see your database documentation.
Configure VCS service groups	See "Configuring VCS service groups for a campus cluster for SFCFSHA and SF Oracle RAC" on page 60.

The sample SF Oracle RAC configuration illustrates the configuration procedures with a four-node campus cluster with two nodes at each site. Each node is running SF Oracle RAC 7.3.1.

Figure 4-1 Sample SF Oracle RAC configuration

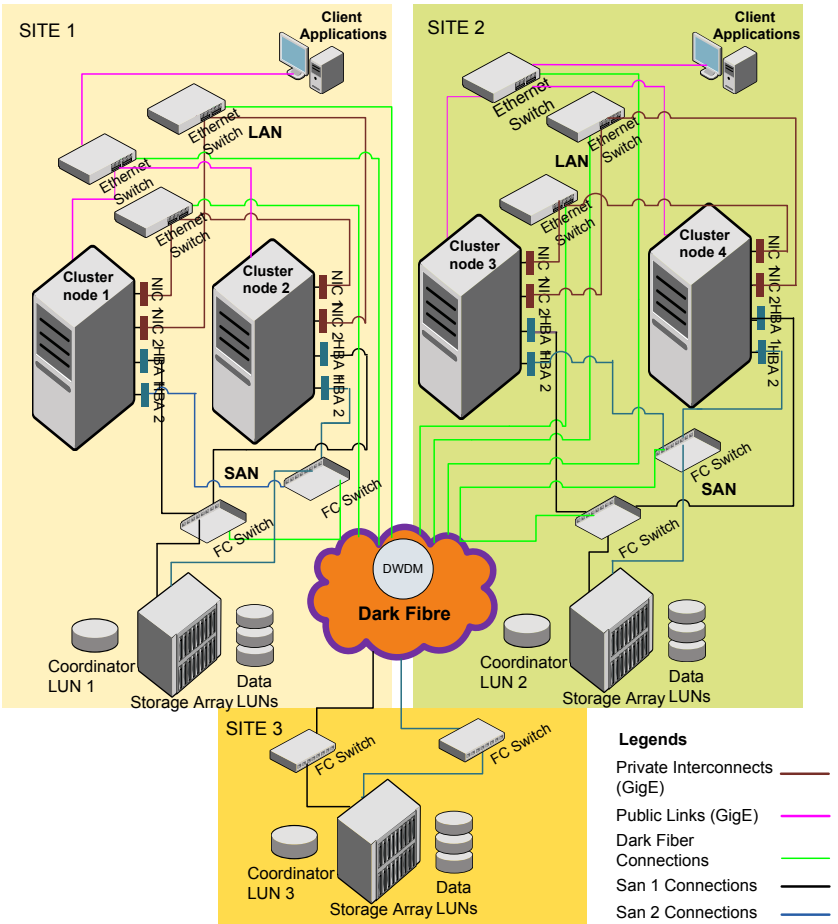


Table 4-2 Sample setup for an SF Oracle RAC campus cluster

Site	Hardware description
Site 1: site1 Cluster name: clus1	Servers: sys1 and sys2 Shared LUNs: disk01 disk02 disk03 disk04 (used as coordinator disk) disk05
Site 2: Site name: site2 Cluster name: clus1	Servers: sys3 and sys4 Shared LUNs: disk06 disk07 disk08 disk09 (used as coordinator disk)
Site 3: Site name: site3 Cluster name: clus1	Shared LUN disk10 (used as coordinator disk)

Although a Coordination Point (CP) server is not used in the current example, it can also be used instead of a third site for a coordinator disk.

Preparing to set up a campus cluster in a parallel cluster database environment

To verify your configuration is supported, review the product requirements and licensing information:

See the *Storage Foundation Cluster File System High Availability Configuration and Upgrade Guide*.

See the *Storage Foundation for Oracle RAC Configuration and Upgrade Guide*.

You will need to verify the following to setup a campus cluster:

- Hardware requirements for Storage Foundation Cluster File System High Availability (SFCFSHA) or Storage Foundation for Oracle RAC (SF Oracle RAC)

- License requirements: in addition to your product with HA/DR, you will need:
  - FlashSnap license
  - Site awareness license

With keyless licensing, your enterprise product keys enable all of the above features.

The following preparation must be completed before configuring the campus cluster.

To prepare to set up a campus cluster

1. Configure the physical infrastructure for campus cluster:
   - Set up access to the local storage arrays and to remote storage arrays on each node. The storage link will extend to the third site as well.
     In environments using the Flexible Storage Sharing (FSS) feature, this step is not mandatory. For more details on FSS, see the *Storage Foundation Cluster File System Administrator's Guide*.
   - Set up the private heartbeat network
     See " Typical VCS campus cluster setup" on page 19.
2. Install the operating system on all the nodes of the cluster.

   See your operating system documenation.
3. Install and configure either SFCFSHA or SF Oracle RAC on all nodes on both the sites.

   See the *Storage Foundation Cluster File System High Availability Configuration and Upgrade Guide*.

   See the *Storage Foundation for Oracle RAC Configuration and Upgrade Guide*.

   For a setup example, steps are provided to install and configure SF Oracle RAC 7.3.1 on all four nodes. Your installation and configuration steps will necessarily differ to reflect your configuration details.

Configuring I/O fencing to prevent data corruption

Perform the following tasks to configure I/O fencing to prevent data corruption in the event of a communication failure.

To configure I/O fencing to prevent data corruption

1. After installing and configuring SFCFSHA or SF Oracle RAC, configure I/O fencing for data integrity.

   See the *Storage Foundation Cluster File System High Availability Configuration and Upgrade Guide*.

See the *Storage Foundation for Oracle RAC Configuration and Upgrade Guide*.

2 Set up the storage at a third site.

You can extend the DWDM to the third site to have FC SAN connectivity to the storage at the third site. You can also use iSCSI targets as the coordinator disks at the third site.

For example:

Enable I/O fencing by using the coordinator disks from all the three sites.

```
# vxdisksetup -i disk04 format=cdsdisk


# vxdisksetup -i disk09 format=cdsdisk
# vxdisksetup -i disk10 format=cdsdisk
# hastop -all
# vxdg init fencedg disk10 disk04 disk09
# vxdg -g fencedg set coordinator=on
# vxdg deport fencedg
# vxdg -t import fencedg
# vxdg deport fencedg
```

Edit the main.cf to add "UseFence = SCSI3"

```
# vi /etc/VRTSvcs/conf/config/main.cf
# more /etc/vxfendg
fencedg
# more /etc/vxfentab
/dev/vx/rdmp/disk10
/dev/vx/rdmp/disk04
/dev/vx/rdmp/disk09
# cp /etc/vxfen.d/vxfenmode_scsi3_dmp /etc/vxfemode
```

For RHEL 7, SLES 12, and supported RHEL distributions:

```
# /opt/VRTSvcs/vxfen/bin/vxfen start
```

For earlier versions of RHEL, SLES, and supported RHEL distributions:

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

Starting vxfen..
Checking for /etc/vxfendg
Starting vxfen.. Done
```

On all nodes, start VCS:

```
# hastart
```

Set the site name for each host.

```
# vxdctl set site=site1
# vxdctl set site=site2
# vxdctl set site=site3
```

3 Start I/O fencing on all the sites.

Configuring VxVM disk groups for a campus cluster in a parallel cluster database environment

After configuring I/O fencing for data integrity, you must configure the VxVM disk groups for remote mirroring before installing your database.

Note: In cloud environments where one campus cluster is configured on one site and another on the second site, FSS volumes in campus cluster configurations can be used to replicate data for achieving high data availability across sites. In this FSS-campus cluster configuration, site tags can be added based on the name of the site to make data highly available during site failures.

For the example configuration, the database is Oracle RAC.

To configure VxVM disk groups for Oracle RAC on an SF for Oracle RAC campus cluster

1 Initialize the disks as CDS disks

```
# vxdisksetup -i disk01 format=cdsdisk
# vxdisksetup -i disk02 format=cdsdisk
# vxdisksetup -i disk03 format=cdsdisk
# vxdisksetup -i disk05 format=cdsdisk
# vxdisksetup -i disk06 format=cdsdisk
# vxdisksetup -i disk07 format=cdsdisk
# vxdisksetup -i disk08 format=cdsdisk
```

2 Set the site name for each host:

```
# vxdctl set site=sitename
```

The site name is stored in the /etc/vx/volboot file. To display the site names:

```
# vxdctl list | grep siteid
```

For example, for a four node cluster with two nodes at each site, mark the sites as follows:

On the nodes at first site:

```
# vxdctl set site=site1
```

On the nodes at second site:

```
# vxdctl set site=site2
```

3 Obtain the enclosure name using the following command:

```
# vxdmpadm listenclosure
ENCLR_NAME      ENCLR_TYPE    ENCLR_SNO STATUS    ARRAY_TYPE LUN_COUNT FIRMWARE
============================================================================
ams_wms0        AMS_WMS       75040638  CONNECTED A/A-A      35        -
hds9500-alua0   HDS9500-ALUA  D600145E  CONNECTED A/A-A       9        -
hds9500-alua1   HDS9500-ALUA  D6001FD3  CONNECTED A/A-A       6        -
disk            Disk          DISKS     CONNECTED Disk        2        -
```

4 Set the site name for all the disks in an enclosure.

```
# vxdisk settag site=sitename encl:ENCLR_NAME
```

5 Run the following command if you want to tag only the specific disks:

```
# vxdisk settag site=sitename disk
```

For example:

```
# vxdisk settag site=site1 disk01
# vxdisk settag site=site1 disk02
# vxdisk settag site=site1 disk03
# vxdisk settag site=site2 disk06
# vxdisk settag site=site2 disk08
```

6 Verify that the disks are registered to a site.

```
# vxdisk listtag
```

For example:

```
# vxdisk listtag
DEVICE          NAME                          VALUE
disk01           site                          site1
disk02           site                          site1
```

```
disk03          site                    site1
disk04          site                    site1
disk05          site                    site1
disk06          site                    site2
disk07          site                    site2
disk08          site                    site2
disk09          site                    site2
```

7 Create a disk group for OCR and Vote Disks and another for Oracle data, with disks picked from both the sites. While the example below shows a single disk group, you can create as many as you need.

```
# vxdg -s init ocrvotedg disk05 disk07

# vxdg -s init oradatadg disk01 disk06
```

8 Enable site-based allocation on the disk groups for each site.

```
# vxdg -g ocrvotedg addsite site1

# vxdg -g ocrvotedg addsite site2

# vxdg -g oradatadg addsite site1

# vxdg -g oradatadg addsite site2
```

9 If you are using an enclosure, set the tag on the enclosure for both sites.

```
# vxdg -o retain -g ocrvotedg settag encl:3pardata0 site=site1

# vxdg -o retain -g ocrvotedg settag encl:3pardata1 site=site2

# vxdg -o retain -g oradatadg settag encl:3pardata0 site=site1

# vxdg -o retain -g oradatadg settag encl:3pardata1 site=site2
```

10 Configure site consistency for the disk groups.

```
# vxdg -g ocrvotedg set siteconsistent=on

# vxdg -g oradatadg set siteconsistent=on
```

11 Create one or more mirrored volumes in the disk group.

```
# vxassist -g ocrvotedg make ocrvotevol 2048m nmirror=2
```

```
# vxassist -g oradatadg make oradatavol 10200m nmirror=2

# vxassist -g ocrvotedg make ocrvotevol 2048m nmirror=2

# vxassist -g oradatadg make oradatavol 10200m nmirror=2
```

12 To verify the site awareness license, use the `vxlicrep` command. The Veritas Volume Manager product section should indicate: Site Awareness = Enabled

With the Site Awareness license installed on all hosts, the volume created has the following characteristics by default.

- The all sites attribute is set to ON; the volumes have at least one mirror at each site.
- The volumes are automatically mirrored across sites.
- The read policy (rdpol) is set to siteread.
  The read policy can be displayed using the `vxprint -ht` command.
- The volumes inherit the site consistency value that is set on the disk group.

13 From the CVM master, start the volumes for all the disk groups.

```
# vxvol -g ocrvotedg startall
# vxvol -g oradatadg startall
```

14 Create a file system on each volume and mount the same.

```
# mkfs -t vxfs /dev/vx/rdsk/ocrvotedg/ocrvotevol


# mkfs -t vxfs /dev/vx/rdsk/oradatadg/oradatavol


# mount -t vxfs -o cluster /dev/vx/dsk/ocrvotedg/ocrvotevol /ocrvote


# mount -t vxfs -o cluster /dev/vx/dsk/oradatadg/oradatavol /oradata
```

15 Create seperate directories for OCR and Vote file as follows:

```
# mkdir -p /ocrvote/ocr

# mkdir -p /ocrvote/vote
```

16 After creating directories, change the ownership of these directories to Oracle or Grid user:

```
# chown -R user:group /ocrvote
```

Also change the ownership of /oradata to Oracle user:

```
# chown user:group /oradata
```

Note: One Vote Disk is sufficient since it is already mirrored by VxVM.

17 Install your database software.

For Oracle RAC:

- Insall Oracle Clusterware/GRID
- Install Oracle RAC binaries
- Perform library linking of Oracle binaries
- Create the database on /oradata. For detailed steps, See the *Storage Foundation for Oracle RAC Configuration and Upgrade Guide*.

Configuring VCS service groups for a campus cluster for SFCFSHA and SF Oracle RAC

Follow the procedure below to configure the disk groups under Storage Foundation (SF) for Oracle RAC control and set up the Cluster Server (VCS) attributes to define parallel applications in campus clusters. The Cluster Volume Manager (CVM) and Oracle service groups can be configured by editing the VCS configuration file, the main.cf, to define the service groups.

To configure the VCS service groups

1 Configure the disk groups under SFCFSHA or SF Oracle RAC control and set up the VCS attributes to define parallel applications in campus clusters. The CVM and Oracle service groups can be configured by editing the VCS configuration file, main.cf, to define the service groups.

2 Configure the SystemZones attribute in the service group definition as explained previously.

See "Configuring VCS service group for campus clusters" on page 44.

3 Group the hosts at each physical site into a single logical SystemZone. This will enable the failover applications to try to come up on local nodes before they try to come up on a remote site.

   Not all SFCFSHA or SF Oracle RAC service groups are parallel. In the sample configuration file, hosts sys1 and sys2 should be configured in zone 0 and hosts sys3 and sys4 in zone 1. In the event of a failure, this setting instructs VCS to failover the group first within the same site and then across the sites.

4 After configuring your service groups and before putting your configuration into production, you can verify your configuration resilience by means of testing various failure scenarios.

See "sfrac11_main.cf file" on page 213.

Tuning guidelines for parallel campus clusters

An important consideration while tuning a campus cluster in a Storage Foundation Cluster File System High Availability (SFCFS HA) or Storage Foundation (SF) for Oracle RAC environment is setting the LLT peerinact time. Follow the guidelines below to determine the optimum value of peerinact time:

- Calculate the roundtrip time using lltping (1M).
- Evaluate LLT heartbeat time as half of the round trip time.
- Set the LLT peer trouble time as 2-4 times the heartbeat time.
- LLT peerinact time should be set to be more than 4 times the heart beat time.

Best practices for a parallel campus cluster

The following best practices ensure a robust Storage Foundation Cluster File System High Availability (SFCFS HA) or Storage Foundation (SF) for Oracle RAC campus cluster:

- Tag all the mirrored volumes in the campus cluster with appropriate site names. VxVM allocates storage from the correct site when creating or resizing a volume and when changing a volume's layout if the volume is tagged with site name.
- All volumes that have data required by the application must be evenly mirrored. Each site must have at least one mirror of all volumes hosting application data, including the FlashSnap log volume.

  In Flexible Storage Sharing (FSS) environments, if possible, use multiple mirrors at each site since storage availability is tied to host availability due to local DAS or internal disks being used. For more details on FSS, see the *Storage Foundation Cluster File System Administrator's Guide*.

- Do not enable site consistency on VxVM snapshot volumes.
- Use redundant links for storage and private interconnects. DWDM can be used for storage and heartbeat together. Another redundant DWDM link can be used to prevent single point of failure. Separate switches and multiplexer / de-multiplexer devices should be used.
- Use Coordination Point Server as the third coordination point.
- Use the procedure for online replacement of coordination points, to replace disk based or Coordination Point Server based coordination points.

Section 3

Implementing replicated data clusters

- Chapter 5. Configuring a replicated data cluster using VVR
- Chapter 6. Configuring a replicated data cluster using third-party replication

Chapter 5

Configuring a replicated data cluster using VVR

This chapter includes the following topics:

- About setting up a replicated data cluster configuration
- About migrating a service group
- Fire drill in replicated data clusters

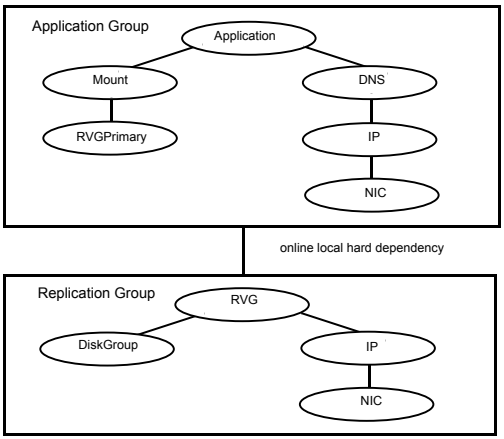
About setting up a replicated data cluster configuration

This topic describes the steps for planning, configuring, testing, and using the VCS RDC configuration to provide a robust and easy-to-manage disaster recovery protection for your applications. It describes an example of converting a single instance Oracle database configured for local high availability in a VCS cluster to a disaster-protected RDC infrastructure. The solution uses Volume Replicator to replicate changed data.

About typical replicated data cluster configuration

Figure 5-1 depicts a dependency chart of a typical RDC configuration.

Figure 5-1 Dependency chart of a typical RDC configuration



In this example, a single-instance application is configured as a VCS service group (DiskGroup) on a four-node cluster, with two nodes in the primary RDC system zone and two in the secondary RDC system zone. In the event of a failure on the primary node, VCS fails over the application to the second node in the primary zone.

The process involves the following steps:

- Setting Up Replication
- Configuring the Service Groups
- Configuring the Service Group Dependencies

About setting up replication

Veritas Volume Replicator (VVR) technology is a license-enabled feature of Veritas Volume Manager (VxVM), so you can convert VxVM-managed volumes into replicated volumes managed using VVR. In this example, the process involves grouping the Oracle data volumes into a Replicated Volume Group (RVG), and creating the VVR Secondary on hosts in another VCS cluster, located in your DR site.

When setting up VVR, it is a best practice to use the same DiskGroup and RVG name on both sites. If the volume names are the same on both zones, the Mount resources will mount the same block devices, and the same Oracle instance will start on the secondary in case of a failover.

Configuring the service groups

This topic describes how to configure service groups.

To configure the replication group

1. Create a hybrid service group (oragrp_rep) for replication.
2. Copy the DiskGroup resource from the application to the new group. Configure the resource to point to the disk group that contains the RVG.
3. Configure new resources of type IP and NIC.
4. Configure a new resource of type RVG in the service group.
5. Set resource dependencies as per the following information:
   - RVG resource depends on the IP resource
   - RVG resource depends on the DiskGroup resource
     IP resource depends on the NIC resource
6. Set the SystemZones attribute of the child group, oragrp_rep, such that all nodes in the primary RDC zone are in system zone 0 and all nodes in the secondary RDC zone are in system zone 1.

To configure the application service group

1. In the original Oracle service group (oragroup), delete the DiskGroup resource.
2. Add an RVGPrimary resource and configure its attributes.

   Set the value of the RvgResourceName attribute to the name of the RVG type resource that will be promoted and demoted by the RVGPrimary agent.

   Set the AutoTakeover and AutoResync attributes from their defaults as desired.
3. Set resource dependencies such that all Mount resources depend on the RVGPrimary resource. If there are a lot of Mount resources, you can set the TypeDependencies attribute for the group to denote that the Mount resource type depends on the RVGPRimary resource type.

4 Set the SystemZones attribute of the Oracle service group such that all nodes in the primary RDC zone are in system zone 0 and all nodes in the secondary RDC zone are in zone 1. The SystemZones attribute of both the parent and the child group must be identical.

5 If your setup uses BIND DNS, add a resource of type DNS to the oragroup service group. Set the Hostname attribute to the canonical name of the host or virtual IP address that the application uses on that cluster. This ensures DNS updates to the site when the group is brought online. A DNS resource would be necessary only if the nodes in the primary and the secondary RDC zones are in different IP subnets.

Configuring the service group dependencies

Set an online local hard group dependency from application service group to the replication service group to ensure that the service groups fail over and switch together.

1 In the Cluster Explorer configuration tree, select the cluster name.

2 In the view panel, click the **Service Groups** tab. This opens the service group dependency graph.

3 Click **Link**.

4 Click the parent group oragroup and move the mouse toward the child group, oragroup_rep.

5 Click the child group oragroup_rep.

6 On the Link Service Groups dialog box, click the online local relationship and the hard dependency type and click **OK**.

About migrating a service group

In the RDC set up for the Oracle database, consider a case where the primary RDC zone suffers a total failure of the shared storage. In this situation, none of the nodes in the primary zone see any device.

The Oracle service group cannot fail over locally within the primary RDC zone, because the shared volumes cannot be mounted on any node. So, the service group must fail over, to a node in the current secondary RDC zone.

The RVGPrimary agent ensures that VVR volumes are made writable and the DNS agent ensures that name services are resolved to the DR site. The application can be started at the DR site and run there until the problem with the local storage is corrected.

If the storage problem is corrected, you can switch the application to the primary site using VCS.

Switching the service group

Before switching the application back to the original primary RDC zone, you must resynchronize any changed data from the active DR site since the failover. This can be done manually through VVR or by running a VCS action from the RVGPrimary resource.

To switch the service group

1. In the **Service Groups** tab of the configuration tree, right-click the resource.
2. Click **Actions**.
3. Specify the details of the action as follows:
   - From the **Action** list, choose **fbsync**.
   - Click the system on which to execute the action.
   - Click **OK**.

   This begins a fast-failback of the replicated data set. You can monitor the value of the ResourceInfo attribute for the RVG resource to determine when the resynchronization has completed.
4. Once the resynchronization completes, switch the service group to the primary cluster. In the **Service Groups** tab of the of the Cluster Explorer configuration tree, right-click the service group.
5. Click **Switch To** and select the system in the primary RDC zone to switch to and click OK.

Fire drill in replicated data clusters

You can use fire drills to test the configuration's fault readiness by mimicking a failover without stopping the application in the primary data center. To set up a disaster recovery fire drill, you have the option to create and configure the fire drill service group manually, or using the Fire Drill Setup wizard.

See the *Cluster Server Administrator's Guide*.

Chapter 6

Configuring a replicated data cluster using third-party replication

This chapter includes the following topics:

- About setting up a replicated data cluster configuration using third-party replication
- About typical replicated data cluster configuration using third-party replication
- About setting up third-party replication
- Configuring the service groups for third-party replication
- Fire drill in replicated data clusters using third-party replication

About setting up a replicated data cluster configuration using third-party replication

The VCS replicated data cluster (RDC) configuration provides robust and easy-to manage disaster recovery protection for your applications. You can convert an application configured for local high availability in a VCS cluster to a disaster-protected RDC infrastructure. When configuring an RDC you can use a supported third-party replication technology to replicate application data.

Review the best practices and planning considerations prior to setting up an RDC using third-party replication.

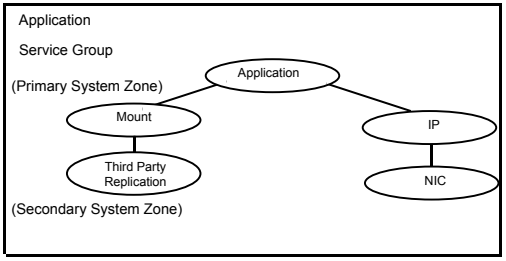
See “Planning for data replication” on page 38.

See “Planning a replicated data cluster setup” on page 37.

About typical replicated data cluster configuration using third-party replication

Figure 6-1 depicts a dependency chart of a typical RDC configuration using third-party replication.

Figure 6-1 Dependency chart of a typical RDC configuration using third-party replication



In this example, an application is configured as a VCS service group on a four-node cluster, with two nodes in the primary RDC system zone and two in the Secondary RDC system zone. In the event of a failure on the Primary node, VCS fails over the application to the second node in the Primary zone. When there is no system available for failover in the Primary system zone, VCS will failover the application to a system in the Secondary system zone.

Note: Some third-party replication software such as Oracle Dataguard require special configuration in RDC environment. Please refer to respective agent's Installation and Configuration Guide for more details.

The process involves the following steps:

- Setting up replication
- Configuring the service groups

About setting up third-party replication

A typical replicated data cluster configuration process involves setting up replication and configuring the service groups. You can set up replication for the RDC using various third-party replication technologies.

Review the best practices for setting up third-party replication.

See "Setting up third-party replication" on page 99.

Once the replication is configured, verify that the replicated storage is visible to all the nodes in the cluster according to the system zones.

Configuring the service groups for third-party replication

The original application service group will have the required application, storage, and network resources configured. You must set the SystemZones attribute of the application service group so that all the nodes in the Primary RDC zone are in system zone 0 and all nodes in the Secondary RDC zone are in zone 1.

To configure the service groups for third-party replication

1. Add the replication resource to the application service group
2. Configure the replication agent resources using guidelines from the replication agent's Installation and Configuration Guide
3. Localize the required resource attributes as per system-zone requirements.
4. Set the dependency between the storage (mount) resource and the replication resource.

Fire drill in replicated data clusters using third-party replication

A fire drill procedure verifies the fault-readiness of a disaster recovery configuration. This procedure is performed without stopping the application at the Primary site and disrupting user access.

A fire drill is performed at the Secondary site using a special service group for fire drills. The fire drill service group is identical to the application service group, but uses a fire drill resource in place of the replication agent resource. The fire drill service group uses a copy of the data that is used by the application service group.

Almost all third-party replication agents have a corresponding fire drill agent bundled with them. These fire drill agents are responsible for taking a snapshot inside the storage array and importing the snapshot disks on the hosts. The fire drill agents are named similar to the replication agents with the suffix “Snap” appended to the agent name. For example, the fire drill agent for EMC SRDF is SRDFSnap and the fire drill agent for Hitachi TrueCopy (HTC) agent is HTCSnap.

For more information about configuring fire drills, please refer the Installation and Configuration Guide of the appropriate third-party replication agent.

Section 4

Implementing global clusters

- Chapter 7. Configuring global clusters for VCS and SFHA
- Chapter 8. Configuring a global cluster with Storage Foundation Cluster File System High Availability, Storage Foundation for Oracle RAC, or Storage Foundation for Sybase CE
- Chapter 9. Configuring global clusters with VVR and Storage Foundation Cluster File System High Availability, Storage Foundation for Oracle RAC, or Storage Foundation for Sybase CE

Chapter 7

Configuring global clusters for VCS and SFHA

This chapter includes the following topics:

- Installing and Configuring Cluster Server
- Setting up VVR replication
- Setting up third-party replication
- Configuring clusters for global cluster setup
- Configuring service groups for global cluster setup
- Fire drill in global clusters

Installing and Configuring Cluster Server

To create a global cluster environment, you must first install and configure Cluster Server (VCS) at the primary site, and then set up the application for high availability on both the primary and secondary sites. You can install and configure VCS using the script-based installer, the Web-based installer, response files, or manually.

For more information on installing and configuring VCS, see the *Cluster Server Configuration and Upgrade Guide*.

Setting up VVR replication

After you have configured a VCS cluster at the primary site, and have set up the application for high availability on both the primary and secondary sites, you must set up data replication.

About configuring VVR replication

You can configure and administer Volume Replicator (VVR) using one of the following interfaces:

Command line interface (CLI)	You can use the command line interface of VVR to configure, administer, and monitor VVR in a distributed environment. This guide gives instructions on configuring, administering, and monitoring VVR using the Veritas product installer.

This topic explains how to set up a Replicated Data Set (RDS) using the command-line interface. VVR enables you to set up replication either when the data volumes are zero initialized or contain valid data. Make sure you follow the best practices or recommendations described to ensure successful configuration of VVR.

Detailed examples are also available on how to configure and set up a simple VVR configuration. Read this information before you start setting up replication.

Before setting up a Replicated Data Set, decide how you plan to lay out your VVR configuration.

To configure and set up replication, perform the following tasks in the order presented below.

See "Creating a Replicated Data Set" on page 76.

See "Synchronizing the Secondary and starting replication" on page 92.

Note: The procedure to set up replication is the same either when the application is running or stopped, unless noted otherwise.

Best practices for setting up replication

Set up replication according to the following best practices:

- Create one RVG for each application, rather than for each server. For example, if a server is running three separate databases that are being replicated, create three separate RVGs for each database. Creating three separate RVGs helps to avoid write-order dependency between the applications and provides three separate SRLs for maximum performance per application.
- Create one RVG per disk group. Creating one RVG per disk group enables you to efficiently implement application clustering for high availability, where only one RVG needs to be failed over by the service group. If the disk group contains more than one RVG, the applications using the other RVGs would have to be

stopped to facilitate the failover. You can use the Disk Group Split feature to migrate application volumes to their own disk groups before associating the volumes to the RVG.

- Plan the size and layout of the data volumes based on the requirement of your application.
- Plan the size of the network between the Primary and each Secondary host.
- Lay out the SRL appropriately to support the performance characteristics needed by the application. Because all writes to the data volumes in an RVG are first written to the SRL, the total write performance of an RVG is bound by the total write performance of the SRL. For example, dedicate separate disks to SRLs and if possible dedicate separate controllers to the SRL.
- Size the SRL appropriately to avoid overflow.
  The Volume Replicator Advisor (VRAdvisor), a tool to collect and analyze samples of data, can help you determine the optimal size of the SRL.
- Include all the data volumes used by the application in the same RVG. This is mandatory.
- Provide dedicated bandwidth for VVR over a separate network. The RLINK replicates data critical to the survival of the business. Compromising the RLINK compromises the business recovery plan.
- Use the same names for the data volumes on the Primary and Secondary nodes. If the data volumes on the Primary and Secondary have different names, you must map the name of the Secondary data volume to the appropriate Primary data volume.
- Use the same name and size for the SRLs on the Primary and Secondary nodes because the Secondary SRL becomes the Primary SRL when the Primary role is transferred.
- Mirror all data volumes and SRLs. This is optional if you use hardware-based mirroring.
- The `vradmin` utility creates corresponding RVGs on the Secondary of the same name as the Primary. If you choose to use the `vxmake` command to create RVGs, use the same names for corresponding RVGs on the Primary and Secondary nodes.
- Associate a DCM to each data volume on the Primary and the Secondary if the DCMs had been removed for some reason. By default, the `vradmin createpri` and `vradmin addsec` commands add DCMs if they do not exist.
- In a shared disk group environment, currently only the cvm master node should be assigned the logowner role. It is recommended to enable the PreOnline

trigger for the RVGLogOwner agent, so that the VVR logowner always resides on the CVM master node.

- In a shared disk group environment, currently only the cvm master node should be assigned the logowner role.
- The on-board write cache should not be used with VVR. The application must also store data to disk rather than maintaining it in memory. The takeover system, which could be a peer primary node in case of clustered configurations or the secondary site, must be capable of accessing all required information. This requirement precludes the use of anything inside a single system inaccessible by the peer. NVRAM accelerator boards and other disk caching mechanisms for performance are acceptable, but must be done on the external array and not on the local host.

Creating a Replicated Data Set

To create a Replicated Data Set (RDS), perform the following tasks in the order presented below:

- Create a Primary Replicated Volume Group (RVG) of the RDS
  You can also associate volume-set component volumes to an RDS.
- Add a Secondary to the RDS
- Change the Replication Settings for the Secondary

In a shared disk group environment, the `vradmin`, `vrstat`, and `vrnotify` commands can be issued on any node in the cluster. However, the `vradmin createpri`, `vxibc`, `vxrlink` (other than informational commands), and `vxrvg` commands must be issued from the CVM master node.

In a SAN disk group environment, the `vradmin` commands can be issued on the volume server or the volume client, provided the data volumes are attached to the host on which the commands are issued.

Creating a Primary RVG of an RDS

The first step in creating an RDS is creating its Primary RVG. VVR enables you to create a Primary RVG of an RDS using the `vradmin createpri` command.

The `vradmin createpri` command enables you to associate existing data volumes and the Storage Replicator Log (SRL) to the Primary RVG.

The `vradmin createpri` command performs the following operations:

- Creates the Primary RVG on the host on which the command is issued.
- Enables or starts the Primary RVG.

- Associates DCMs to the data volumes in the RVG.
- Associates the specified data volumes and SRL to the RVG.
- Associates the specified volume sets (if any) to the RVG.

Note: Specify the volume set name in the command, not the names of each component volume. Specifying the component volume name causes the command to fail.

VVR does not support RAID-5 volumes, that is, volumes with usage type `raid5` are not supported. Data volumes must be of usage type `gen` or `fsgen`. However, data volumes can be configured on hardware-based RAID-5 disks.

Dirty Region Logs (DRLs) are not needed with VVR because VVR uses the SRL to recover volumes, not the DRLs. If any of the data volumes or the SRL has a DRL, the `vradmin createpri` command removes the DRL before the data volume is associated to the RVG.

By default, the `vradmin createpri` command adds DCMs to the data volumes, if they have not already been added. The `vradmin createpri` command creates the DCM of an appropriate default size based on the size of the volume and mirrors the DCM by default. To create and add a DCM of a size that is different from the default, associate the DCM of the required size to the data volumes before running the `vradmin createpri` command.

Note: The `vradmin createpri` command will fail if there are not enough drives to mirror the DCM. You may need to associate the DCM to an appropriately sized data volume.

The `-nodcm` option when used with the `vradmin createpri` command associates data volumes to the RVG but does not add DCMs to the data volumes.

If you want to associate additional volumes to the RVG after creating the RVG, use the `vradmin addvol` command.

Prerequisites for creating a Primary RVG of an RDS

Before creating a Primary RVG of an RDS, the following prerequisites must be met:

- The data volumes and SRL must exist on the Primary. If the data volumes and SRL do not exist on the Primary, create them. To associate a volume set to the RVG, the volume set must exist on the Primary.
- The SRL cannot be a volume set or a component volume of a volume set.

- The data volumes and SRL must be started. If the data volumes and SRL are not started, start them. When a data volume is started, its state is active.
- The data volumes used by the application must exist in the same RVG. Include the data volumes used by the application in the same RVG.
- In a SAN disk group environment, if the application resides on the volume client, all the Primary data volumes must be attached to the volume client or unattached from the volume client.
- Make sure you include the appropriate loopback address(es) in the `/etc/hosts file`.
  - If your environment only uses IPv4, you must include an IPv4 loopback address in the `/etc/hosts` file. The following is a sample entry:

    ```
    127.0.0.1       localhost       loopback
    ```

  - If your environment only uses IPv6, you must include an IPv6 loopback address in the `/etc/hosts` file.

    ```
    ::1             localhost6.localdomain6      localhost6     loopback
    ```

  - If your environment uses both IPv4 and IPv6, the `/etc/hosts` file must include both loopback addresses.

    ```
    127.0.0.1       localhost       loopback
    ::1             localhost6.localdomain6      localhost6     loopback
    ```

To create a Primary RVG of an RDS

Issue the following command on the host on which you want to create the Primary RVG:

```
# vradmin -g diskgroup createpri rvgname \
      dv01_name,dv02_name... srl_name
```

The argument *rvgname* is the name of the RVG to be created.

The argument *dv01_name,dv02_name,...* is a comma-separated list of the names of the data volumes to be associated to the RVG. Each item can be an independent data volume name, or the name of a volume set. To associate a volume set to the RVG, specify the name of the volume set, not the names of the individual component volumes.

Note: In previous releases, component volumes could be associated directly to an RVG. Beginning in Release 5.0, the volume set itself is associated to the RVG, enabling VVR to verify consistency between the volume sets on the Primary and the Secondary RVGs. The `vradmin createpri` command fails if a component volume of the volume set and the volume set itself are each specified for an RVG.

The argument *`srl_name`* is the name of the SRL to be associated to the RVG.

Use `-nodcm` option if you do not want DCMs to be added to the data volumes. By default, DCMs are added automatically.

Example - Creating a Primary RVG containing a data volume

This example shows how to create a Primary RVG `hr_rvg` in the disk group `hrdg`, which contains the data volumes `hr_dv01` and `hr_dv02`, and the volume `hr_srl` that is to be used as the SRL. This example automatically adds DCMs to the data volumes.

```
# vradmin -g hrdg createpri hr_rvg hr_dv01,hr_dv02 hr_srl
```

Example - Creating a Primary RVG containing a volume set

This example shows how to create a Primary RVG `hr_rvg` in the disk group `hrdg`, which contains the volume set `hr_vset`, the data volumes `hr_dv01` and `hr_dv02`, and the volume `hr_srl` that is to be used as the SRL.

```
# vradmin -g hrdg createpri hr_rvg hr_dv01,hr_dv02,hr_vset \
hr_srl
```

If the volume set includes the component volumes `hr_vsetdv01` and `hr_vsetdv02`, these volumes are associated to the RVG `hr_rvg`. This example automatically adds DCMs to the data volumes, including the component volumes `hr_vsetdv01` and `hr_vsetdv02`.

Adding a Secondary to an RDS

After creating the Primary RVG of the RDS, go on to adding a Secondary. Use the `vradmin addsec` command to add a Secondary RVG to an RDS. This command can also be used to add additional Secondary RVGs. The `vradmin addsec` command can be issued from any host that is already in the RDS.

Note: Run the `vradmin addsec` from the Primary node. If you run this command from the node being added as the Secondary, the command fails.

The `vradmin addsec` command performs the following operations by default:

- Creates and adds a Secondary RVG of the same name as the Primary RVG to the specified RDS on the Secondary host. By default, the Secondary RVG is added to the disk group with the same name as the Primary disk group. Use the option `-sdg` with the `vradmin addsec` command to specify a different disk group on the Secondary.
- If any of the data volumes or the SRL on the Secondary has a DRL, the DRL is removed before the data volume is associated to the RVG. DRLs are not needed with VVR because VVR uses the SRL to recover volumes, not the DRLs.
- Automatically adds DCMs to the Primary and Secondary data volumes if they do not have DCMs. Use the `-nodcm` option to specify that DCMs are not to be added to the data volumes.

  The `vradmin addsec` command creates the DCM of an appropriate default size based on the size of the volume and mirrors the DCM by default. To create and add a DCM of a size that is different from the default, associate the DCM of the required size to the data volumes before running the `vradmin addsec` command.
- Associates to the Secondary RVG, existing data volumes of the same names and sizes as the Primary data volumes; it also associates an existing volume with the same name as the Primary SRL, as the Secondary SRL.
- If the Primary RVG includes a volume set, the `vradmin addsec` command associates the corresponding volume set to the Secondary, if the volume set exists on the Secondary. The volume set on the Secondary must include volumes of the same name, lengths and indices as the component volumes on the Primary. If the volume set exists on the Secondary and the volume set configuration is correct except that it does not include all of the component volumes corresponding to those in the volume set on the Primary, the `vradmin addsec` command attempts to add the remaining component volumes to the volume set on the Secondary and then associate the volume set to the Secondary RVG. This command succeeds if all of the remaining component volumes exist on the Secondary with the same names, lengths, and indices as the component volumes on the Primary. However, if any of the component volumes do not exist on the Secondary or have a mismatched name, length, or index, the `vradmin addsec` command fails with the appropriate error message.

  If the volume set does not exist on the Secondary, but the component volumes exist with the same names, lengths, and indices, the `vradmin addsec` command creates the volume set on the Secondary and then associates it to the Secondary RVG.
- Creates and associates to the Primary and Secondary RVGs respectively, the Primary and Secondary RLINKs with default RLINK names *`rlk_remotehost_rvgname`*. If you choose to use names other than the default,

use the `prlink` and `srlink` attributes of the `vradmin addsec` command to specify the Primary and Secondary RLINK names.
See "Example - Creating a Primary RVG containing a volume set" on page 79.

Note: For replication in asynchronous mode with secondary logging, the SRL size on both the Primary and Secondary must be the same.

Best practices for adding a Secondary to an RDS

When you add a Secondary to an RDS, we recommend the following best practices:

- Determine the network and IP addresses to use. Add all participating system names and IP addresses to the `/etc/hosts` files on each system or to the name server database of your name service. Make sure the IP addresses are available (that is, plumbed and up) on the appropriate hosts for your configuration.
- Plan ahead for application clustering by configuring the IP addresses used for replication as virtual IP addresses. For each replicated data set, the Primary and the Secondary cluster should each have one unique virtual IP address to use as the address for the RLINK. If you do this, you can place VVR under cluster control without having to modify the IP address of the RLINK later. Changing the IP address of an RLINK requires pausing replication.
- Plan the bandwidth of the network based on your requirement. You can choose to use either the UDP protocol or TCP protocol for network communication between the Primary and Secondary. Also, plan to operate in a firewall environment.
- We recommend that you use the following naming conventions for RLINKs. By default, VVR follows the following naming conventions for RLINKs:
  Primary RLINK: *`rlk_remotehost_rvgname`*. For example:
  `rlk_london_hr_rvg`
  Secondary RLINK: *`rlk_remotehost_rvgname`*. For example:
  `rlk_seattle_hr_rvg`
- If you have multiple secondaries in your RDS setup, VVR automatically creates RLINKs between ever pair of secondaries. By doing this, the additional secondaries will be automatically added to the RDS after the migrate operation has completed successfully.
- Associate a DCM to each data volume on the Primary and the Secondary to use the SRL Protection and Failback Logging features.

Prerequisites for adding a Secondary to an RDS

On the Secondary to be added, do the following:

- Create a disk group with the same name as the Primary disk group.
- Create data volumes of the same names and lengths as the Primary data volumes.
- Create an SRL of the same name as the Primary SRL. Note that the SRL cannot be a volume set or a component volume of a volume set.
- If the Primary RVG includes a volume set, make sure that the component volumes on the Secondary to be added have identical names, lengths, and indices as the component volumes on the Primary.
- Make sure the `/etc/vx/vras/.rdg` file on the Secondary host to be added to the RDS contains the Primary disk group ID. Ensure that each disk group ID entry in the `.rdg` file is on a separate line.

  Refer to the `.rdg` file for the sample format for the disk group ID entry.

  The `vradmin addsec` command checks whether the Primary RVG is authorized to create a corresponding Secondary RVG on the specified Secondary host. A Primary is determined as authorized if the `/etc/vx/vras/.rdg` file on the specified Secondary host contains the Primary disk group ID. If the Primary contains multiple RVGs in the same disk group, only one entry is required. A plus (+) sign in the `/etc/vx/vras/.rdg` file on the Secondary host indicates that all Primary RVGs on all hosts are authorized to create a Secondary RVG on the specified Secondary host.

  The `/etc/vx/vras/.rdg` file on the Secondary host is only used for authorization checking when a Secondary is added, or when remote data volumes are synchronized or verified. To perform these operations after a Secondary takes over from the Primary, the original Primary host should also have an `/etc/vx/vras/.rdg` file containing the disk group ID for the new Primary host.

  To display the Primary disk group ID, issue the following command on the Primary host:

  ```
  # vxprint -l diskgroup
  ```

  For example, to enable host `seattle` to create an RVG on Secondary host `london` the `.rdg` file on the host `london` must have the following entries, each on a new line.

  ```
  1083007373.10.seattle
  ```

- In a SAN disk group environment, if the application resides on the volume client, the Secondary data volumes must be attached to the corresponding volume client on the Secondary.
- In a SAN disk group environment, if the application resides on the volume client, the Primary volume server must have network connection to the Secondary volume server and Secondary volume client.

- In a SAN disk group environment, if the application resides on the volume client, the hostname or IP address for the Secondary must be available on the volume client on the Secondary.
- In a SAN disk group environment, if the application resides on the volume server, the hostname or IP address for the Secondary must be available on the volume server on the Secondary.

To add a Secondary to an RDS

```
# vradmin -g local_diskgroup addsec local_rvgname pri_hostname \
      sec_hostname
```

The argument *local_diskgroup* is the name of the disk group on the local host.

The argument *local_rvgname* is the name of the RVG on the local host.

The arguments *pri_hostname* and *sec_hostname* are either resolvable hostnames or IP addresses for the Primary and the Secondary hosts. These names are used as `local_host` and `remote_host` attributes while creating RLINKs. The `local_host` and `remote_host` specify the network connection to use for the Primary and Secondary RLINKs.

Use the `-nodcm` option if you do not want to add DCMs to the data volumes. By default, DCMs are automatically added unless the `-nodcm` option is specified.

Note: By default, SRL protection on the new Primary and Secondary RLINKs is set to `autodcm`. If you specify the `-nodcm` option, the `vradmin addsec` command disables SRL protection.

Note that the Secondary RVG is added to the disk group with the same name as the Primary disk group, unless specified otherwise using the `-sdg` option.

Example 1:

This example shows how to add a Secondary host `london_priv` to the RDS, which contains the RVG `hr_rvg`. For replication, this example uses a private network with the Primary hostname `seattle_priv`, Secondary hostname `london_priv`. On the Secondary, the RVG is added to the same disk group as the Primary, that is, `hrdg`. This example automatically adds DCMs to the data volumes.

```
# vradmin -g hrdg addsec hr_rvg seattle_priv london_priv
```

Example 2:

This example shows how to add the Secondary host `london_priv` to the RDS, which contains the RVG `hr_rvg`. It creates the Secondary with the specific Primary

and Secondary RLINK names `to_london` and `to_seattle`. The RLINK connects the Primary host `seattle_priv` and the Secondary host `london_priv`. On the Secondary, the RVG is added to the same disk group as the Primary, that is, `hrdg`.

```
# vradmin -g hrdg addsec hr_rvg seattle_priv london_priv \
prlink=to_london srlink=to_seattle
```

Example 3:

This example shows how to add a Secondary host `london-v6_priv` to the RDS, which contains the RVG `hr_rvg`. For replication, this example uses a private IPv6 network with the Primary hostname `seattle-v6_priv`, Secondary hostname `london-v6_priv`. Both hostnames `london-v6_priv` and `seattle-v6_priv` resolve to IPv6 addresses belonging to the private IPv6 network. On the Secondary, the RVG is added to the same disk group as the Primary, that is, `hrdg`. This example automatically adds DCMs to the data volumes.

```
# vradmin -g hrdg addsec hr_rvg seattle-v6_priv london-v6_priv
```

Example 4:

This example shows how to add a Secondary host `london-v6` to the RDS, which contains the RVG `hr_rvg`. It creates the Secondary with the specific Primary and Secondary RLINK names `to_london-v6` and `to_seattle-v6`. The RLINK connects the Primary host `seattle-v6` and the Secondary host `london-v6`, which resolve to IPv6 addresses `aaaa:bbbb:cccc:dddd:eeee:ffff:gggg:hhhh` and `pppp:qqqq:rrrr:ssss:wwww:xxxx:yyyy:zzzz` respectively. On the Secondary, the RVG is added to the same disk group as the Primary, that is, `hrdg`. This example also automatically adds DCMs to the data volumes.

```
# vradmin -g hrdg addsec hr_rvg aaaa:bbbb:cccc:dddd:eeee:ffff:gggg:hhhh \
pppp:qqqq:rrrr:ssss:wwww:xxxx:yyyy:zzzz prlink=to_london-v6 \
srlink=to_seattle-v6
```

Changing the replication settings for a Secondary

When you add a Secondary to an RDS, the default replication attributes of the Secondary are set to `synchronous=off`, `latencyprot=off`, `srlprot=autodcm`, `packet_size=8400` and `bandwidth_limit=none`.

To display the default replication settings for the Secondary, use the following form of the `vxprint` command:

```
vxprint -g diskgroup -Pl
```

If you are using the UDP protocol, this form of the `vxprint` command also shows the default packet size.

You can set up the replication mode, latency protection, SRL protection, transport protocol, packet size, and the bandwidth used by VVR using the replication attributes, such as `synchronous`, `latencyprot`, and `srlprot`. These attributes are of the form `attribute=value`. Each attribute setting could affect replication and must be set up with care.

The `vradmin set` command enables you to change the replication settings between the Primary and a Secondary. This command can be issued from any host in the RDS. It enables you to perform the following tasks:

- See "Setting the mode of replication for a Secondary" on page 85.
- See "Setting the latency protection for a Secondary" on page 86.
- See "Setting the SRL overflow protection for a Secondary" on page 88.
- See "Setting the network transport protocol for a Secondary" on page 89.
- See "Setting the packet size for a Secondary" on page 89.
- See "Setting the bandwidth limit for a Secondary" on page 90.

The `vradmin set` command changes the corresponding attributes on both the Primary and Secondary RLINK. The attributes `synchronous`, `latencyprot`, and `srlprot` are only active on the Primary RLINK; however, the Secondary attributes are already set up and ready for use if the Primary role is transferred to the Secondary.

Setting the mode of replication for a Secondary

You can set up VVR to replicate to a Secondary in synchronous or asynchronous mode by setting the `synchronous` attribute of the RLINK to `override`, or `off` respectively.

Setting the `synchronous` attribute to `override` puts the RLINK in synchronous mode. During normal operation, VVR replicates in synchronous mode, but if the RLINK becomes inactive due to a disconnection or administrative action, VVR switches temporarily to asynchronous mode and continues to receive updates from the application and store them in the SRL. After the connection is restored and the SRL is completely drained, the RLINK automatically switches back to synchronous mode. Most system administrators set the `synchronous` attribute to `override`.

The `vradmin` command does not allow you to set the `synchronous` attribute to `fail`. Use the `vxedit` command to set the attribute `synchronous=fail`. For more information on using the `vxedit` command, refer to the `vxedit` manual page.

Caution: if you use the `synchronous=fail` mode.

To enable asynchronous mode of replication

To set the replication to asynchronous mode, set the `synchronous` attribute to `off`.

```
# vradmin -g diskgroup set local_rvgname sec_hostname
synchronous=off
```

The argument *local_rvgname* is the name of the RVG on the local host and represents its RDS.

The argument *sec_hostname* is the name of the Secondary host displayed in the output of the *vradmin printrvg* command. If the RDS contains only one Secondary, the argument *sec_hostname* is optional.

Example - Setting the mode of replication to asynchronous for an RDS

To set the mode of replication to asynchronous for the RDS `hr_rvg` between the Primary `seattle` and the Secondary `london`, issue the following command on any host in the RDS:

```
# vradmin -g hrdg set hr_rvg london synchronous=off
```

To enable synchronous mode of replication

To set the synchronous attribute of the RLINK to `override`, use the following command:

```
# vradmin -g diskgroup set local_rvgname sec_hostname \
        synchronous=override
```

Example - Setting the mode of replication to synchronous for an RDS

To set the mode of replication to synchronous for the RDS `hr_rvg` between the Primary `seattle` and the Secondary `london`, issue the following command on any host in the RDS:

```
# vradmin -g hrdg set hr_rvg london synchronous=override
```

Setting the latency protection for a Secondary

The `vradmin set` command enables you to set the `latencyprot` attribute to `override`, `fail`, or `off`; it also enables you to specify a `latency_high_mark` and a `latency_low_mark`, which indicate when the protection becomes active or inactive.

Set the `latencyprot` attribute to enable latency protection between a Primary and a Secondary.

Note: Before enabling latency protection, be sure you understand how latency protection works when the Primary and Secondary are connected or disconnected.

To enable latency protection

1 Set the `latencyprot` attribute of the corresponding RLINKs on the Primary and Secondary.

To set the `latencyprot` attribute to `override`:

```
# vradmin -g diskgroup set local_rvgname sec_hostname \
        latencyprot=override
```

To set the `latencyprot` attribute to `fail`:

```
#    vradmin -g diskgroup set local_rvgname sec_hostname \
       latencyprot=fail
```

2 Set the `latency_high_mark` and the `latency_low_mark` attributes:

```
# vradmin -g diskgroup set local_rvgname sec_hostname \
        latency_high_mark=high_mark

# vradmin -g diskgroup set local_rvgname sec_hostname \
        latency_low_mark=low_mark
```

The argument *`local_rvgname`* is the name of the RVG on the local host and represents the RDS.

The argument *`sec_hostname`* is the name of the Secondary host as displayed in the output of the `vradmin printrvg` command.

Note that the value of `latency_high_mark` must be greater than the value of `latency_low_mark`. We recommend that the difference between the value of `latency_high_mark` and the value of `latency_low_mark` be a small number, for example, 50.

To disable latency protection

- Setting the `latencyprot` attribute to `off` disables latency protection. This does not limit the number of waiting updates in the SRL.

To set the `latencyprot` attribute to `off`:

```
# vradmin -g diskgroup set local_rvgname sec_hostname
    latencyprot=off
```

The argument *local_rvgname* is the name of the RVG on the local host and represents the RDS.

The argument *sec_hostname* is the name of the Secondary host as displayed in the output of the `vradmin printrvg` command.

Setting the SRL overflow protection for a Secondary

VVR provides the following modes of SRL overflow protection: `autodcm`, `dcm`, `override`, `fail`, and `off`.

To enable SRL overflow protection

- Set the `srlprot` attribute of the corresponding RLINK to either `autodcm`, `dcm`, `override`, or `fail`.
  - To set the `srlprot` attribute to `autodcm`, use the following command:

    ```
    # vradmin -g diskgroup set local_rvgname sec_hostname \
            srlprot=autodcm
    ```

  - To set the `srlprot` attribute to `dcm`, use the following command:

    ```
    # vradmin -g diskgroup set local_rvgname sec_hostname \
            srlprot=dcm
    ```

  - To set the `srlprot` attribute to `override`, use the following command:

    ```
    # vradmin -g diskgroup set local_rvgname sec_hostname \
            srlprot=override
    ```

  - To set the `srlprot` attribute to `fail`, use the following command:

    ```
    # vradmin -g diskgroup set local_rvgname sec_hostname \
            srlprot=fail
    ```

  - To set the `srlprot` attribute to `off`, use the following command:

    ```
    # vradmin -g diskgroup set local_rvgname sec_hostname \
            srlprot=off
    ```

  The argument *local_rvgname* is the name of the RVG on the local host and represents the RDS.

  The argument *sec_hostname* is the name of the Secondary host as displayed in the output of the `vradmin printrvg` command.

Setting the network transport protocol for a Secondary

The value specified for the protocol attribute determines the protocol that will be used to communicate between the hosts. You can specify one of the following values for the protocol attribute.

- UDP—The hosts communicate using the UDP/IP protocol.
  VVR automatically calculates the checksum for each data packet it replicates.
- TCP—The hosts communicate using the TCP/IP protocol, which is the default. If a protocol is not specified, then TCP is used as the protocol of communication between hosts.
  If you specify TCP, the VVR checksum is automatically disabled. VVR relies on the TCP checksum mechanism instead. Also, if a node in a replicated data set is using a version of VVR earlier than 5.1 SP1, VVR calculates the checksum regardless of the network protocol.
- STORAGE—Used for bunker replication. The Primary host and the bunker SRL communicate using STORAGE protocol. If the storage is directly accessible by the Primary, for example, DAS or NAS, set the protocol to STORAGE. If the bunker is replicating over IP, the protocol can be set to UDP or TCP.

Note: UDP, TCP, and STORAGE are case sensitive.

To set the network protocol

- To set the protocol for RDSs in disk group of version 110 or above, the following `vradmin` command can be used:

  ```
  # vradmin -g diskgroup set local_rvgname sec_hostname \
          protocol=protocol_name
  ```

  The argument *`protocol_name`* is the name of the protocol that the Primary will use to replicate to the Secondary. The protocol can be set to either TCP or UDP.

Setting the packet size for a Secondary

The packet size determines the number of bytes in a packet that are sent to the Secondary host. The packet size can be changed using the `packet_size` attribute for UDP mode only. If the protocol is set to TCP, the data is sent using the TCP stream.

for more information on the `packet_size` attribute.

To set the packet size

- # **vradmin -g *diskgroup* set *local_rvgname***
  **sec_hostname **
  packet_size=*n*

  The argument *local_rvgname* is the name of the RVG on the local host and represents the RDS.

  The argument *sec_hostname* is the name of the Secondary host as displayed in the output of the `vradmin printrvg` command.

  The argument *n* represents the packet size in bytes.

  The minimum value for the `packet_size` is 1300 bytes.

  The maximum value of the `packet_size` is 65464 bytes.

Example - Setting the packet size between the Primary and Secondary

To set the packet size between the Primary host `seattle` and the Secondary host `london` to 1400 bytes, issue the following command on any host in the RDS:

```
# vradmin -g hrdg set hr_rvg london packet_size=1400
```

Setting the bandwidth limit for a Secondary

Use the `bandwidth_limit` attribute of the `vradmin set` command to set the limit on the network bandwidth used to replicate from the Primary to the Secondary. If `bandwidth_limit` is set to `none`, then VVR uses the available network bandwidth. The default value is `none`. To limit the network bandwidth used by VVR when synchronizing volumes that are not part of an RDS, use the `bandwidth_limit` attribute of the `vradmin syncvol` command.

To control the network bandwidth used for replication

- To limit the bandwidth used for replication between the Primary and a Secondary in an RDS, issue the following command on any host in the RDS. In the command, you can either use the units of bandwidth `kbps`, `mbps`, or `gbps`, or abbreviate the units of bandwidth to `k`, `m`, `g`, respectively. The default unit of bandwidth is bits per second (bps).

  # **vradmin -g *diskgroup* set *local_rvgname sec_hostname* **
  bandwidth_limit=*value*

The argument *local_rvgname* is the name of the RVG on the local host and represents the RDS.

The argument *sec_hostname* is the name of the Secondary host as displayed in the output of the `vradmin printrvg` command.

Example: Limiting network bandwidth between the Primary and the Secondary

To limit the bandwidth to 30 mbps for the RDS `hr_rvg` between the Primary `seattle` and the Secondary `london`, issue the following command on any host in the RDS:

```
# vradmin -g hrdg set hr_rvg london bandwidth_limit=30mbps
```

To disable Bandwidth Throttling for a Secondary

- To disable Bandwidth Throttling for a Secondary in an RDS, issue the following command on any host in the RDS:

```
# vradmin -g diskgroup set local_rvgname sec_hostname \
     bandwidth_limit=none
```

The argument *local_rvgname* is the name of the RVG on the local host and represents the RDS.

The argument *sec_hostname* is the name of the Secondary host as displayed in the output of the `vradmin printrvg` command.

Example: Disabling Bandwidth Throttling between the Primary and the Secondary

To disable Bandwidth Throttling for replication between the Primary `seattle` and the Secondary `london` of RDS `hr_rvg`, issue the following command on any host in the RDS:

```
# vradmin -g hrdg set hr_rvg london bandwidth_limit=none
```

To control the network bandwidth used to synchronize volumes

- To limit the network bandwidth used by VVR when synchronizing volumes that are not part of an RDS, issue the following command:

```
# vradmin -g diskgroup syncvol local_vols_list \
        remote_hostname.... bandwidth_limit=value
```

The argument *local_vols_list* is a comma-separated list of volumes on the local host. The names of the volumes on the local and remote hosts are assumed to be the same.

The argument *remote_hostname* is a space-separated list of names of the remote hosts on which the volumes to be resynchronized reside. It must be possible for IP to resolve the remote host names.

Example: Limiting network bandwidth used by VVR when using full synchronization

This example shows how to limit the network bandwidth used by VVR when using full synchronization to synchronize the remote volumes on host `london` with the local volumes `hr_dv01`, `hr_dv02`, `hr_dv03` in the disk group `hrdg` on the local host `seattle`. The names of the disk group and the volumes on the remote host are the same as the names of the disk group and volumes on the local host.

```
# vradmin -g hrdg -full syncvol hr_dv01,hr_dv02,hr_dv03 london  \
        bandwidth_limit=10mbps
```

Synchronizing the Secondary and starting replication

This section explains how to synchronize the Secondary and start replication.

Methods to synchronize the Secondary

You can synchronize the Secondary using the network, using block-level tape backup or by physically moving disks to the Secondary. Automatic synchronization is the recommended method to synchronize the Secondary. Use one of the following methods to synchronize the Secondary depending on your environment:

- Using the network
  - Automatic synchronization
  - Full synchronization with Storage Checkpoint
  - Difference-based synchronization with Storage Checkpoint
- Using block-level tape backup
  - Block-level tape backup and checkpointing
- Moving disks physically
  - Disk Group Split and Join

The following tables explain when and how to use the different synchronization methods:

Using the network to synchronize the Secondary

You can synchronize the Secondary over the network either when the application is active or inactive. SmartMove is supported with each of these methods.

Table 7-1 Synchronizing the Secondary using the network

To Synchronize the Secondary:	Perform:	Using This Command:
completely	automatic synchronization and start replication See "Using the automatic synchronization feature" on page 94.	`vradmin -a startrep`
completely	full synchronization with Storage Checkpoint	`vradmin -full -c checkpoint syncrvg`
when there is little difference between the data on the Primary and Secondary data volumes of the RDS	difference-based synchronization with Storage Checkpoint	`vradmin -c checkpoint syncrvg`

Using block-level tape backup to synchronize the Secondary

Table 7-2 shows how to synchronize the Secondary using block-level tape backup.

Table 7-2 Synchronizing the Secondary using block-level tape backup

To Synchronize the Secondary:	Do the following:	Using This Command:
completely and when a large amount of data must be moved from the Primary to the Secondary	1. Start a Primary Storage Checkpoint.	`vxrvg -c checkpoint checkstart` *`rvg_name`*
	2. Perform a block-level backup of the Primary.	
	3. End the Primary Storage Checkpoint.	`vxrvg -c checkpoint checkstart` *`rvg_name`*
	4. Restore the tapes on the Secondary and start replication to the Secondary using the Storage Checkpoint.	`vradmin -c checkpoint startrep`

Moving disks physically to synchronize the Secondary

Table 7-3 shows how to synchronize the Secondary by moving disks physically.

Table 7-3 Synchronizing the Secondary by moving disks physically

To Synchronize the Secondary:	Use This Feature	Using This Command:
completely by physically moving disks from the location of the Primary host to the location of Secondary host	Disk Group Split and Join	

Using the automatic synchronization feature

The Automatic Synchronization feature enables you to transfer the data on the Primary to the Secondary over the network. You can synchronize the Secondary using automatic synchronization either when the application is active or inactive.

The Automatic Synchronization procedure transfers data in the Primary data volumes to the Secondary by reading the Primary data volumes from start to finish and sending the data to the Secondary.

Note: Automatic Synchronization does not maintain the order of writes; therefore, the Secondary is inconsistent until the process is complete.

The Secondary becomes consistent after the automatic synchronization completes. To use Automatic Synchronization successfully, the network must be sized appropriately. Note that the synchronization will complete only if the Primary receives writes at a lesser rate than they can be sent to the Secondary. If the Primary receives writes at a faster rate than they can be sent to the Secondary, the synchronization might never complete, especially if the writes are dispersed widely in the volume.

This feature enables you to synchronize multiple Secondary hosts at the same time. When performing automatic synchronization to multiple Secondary hosts, synchronization proceeds at the rate of the slowest network.

VVR pauses synchronization if the Secondary fails or the network disconnects. If the Primary fails while synchronization is in progress, the synchronization continues from the point at which it had stopped when the Primary recovers.

Prerequisite for using Automatic Synchronization

- Each data volume in the Primary RVG must have a DCM associated to it. If data volumes do not have DCMs, an attempt to automatically synchronize a Secondary fails.

The `vradmin startrep` command when used with the option `-a` enables you to start replication and automatically synchronize the Secondary data volumes with

the Primary data volumes in an RDS; it brings the Secondary data volumes up-to-date with the Primary data volumes. You can use this command to synchronize the Secondary when the data volumes contain data and when the application is active or inactive. Replication to another Secondary can be started only after this automatic synchronization completes.

The `vradmin startrep` command can be issued from any host in the RDS. To check the status and progress of the automatic synchronization, use the `vxrlink status` command on the Primary RLINK.

To synchronize the Secondary and start replication using automatic synchronization, issue the following command:

```
# vradmin -g diskgroup -a startrep local_rvgname sec_hostname
```

The argument *`local_rvgname`* is the name of the RVG on the local host and represents its RDS.

The argument *`sec_hostname`* is the name of the Secondary host displayed in the output of the `vradmin printrvg` command. If the RDS contains only one Secondary, the *`sec_hostname`* is optional.

Example—Using the Automatic Synchronization Feature

In this example, the data volumes in the Primary RVG `hr_rvg` on host `seattle` contain valid data and the application is active. To start replication and synchronize the Secondary RVG `hr_rvg` on host `london`, issue the following command:

```
# vradmin -g hrdg -a startrep hr_rvg london
```

Notes on using automatic synchronization

Observe the following notes about using automatic synchronization:

- If you associate a new volume to an RDS while automatic synchronization is in progress, VVR does not automatically synchronize the newly associated data volume.
- In an RDS containing multiple Secondaries that have SRL overflow protection set to dcm, more than one Secondary may require the use of the DCM. If one Secondary is undergoing automatic synchronization and the RLINK of another Secondary is about to overflow, the Automatic Synchronization is abandoned and the DCM becomes active for the overflowing RLINK.
- If you try to automatically synchronize a new RLINK while an existing RLINK is using the DCM mechanism, the automatic synchronization fails.
- To remove a Secondary from a DCM resynchronization process, detach the corresponding Primary RLINK.

- If you try to dissociate a DCM from a data volume while the DCM is in use, the operation fails.
- If the DCM is detached because of I/O errors while the DCM is in use, the resynchronization is abandoned and the RLINKs that are being synchronized are detached.
- You can use automatic synchronization with SmartMove. This combination lets deploy your disaster recovery site must faster because you are replicating far less data.
  See "About SmartMove for VVR" on page 96.

Example for setting up replication using automatic synchronization

This example assumes that the RDS has been created using the example procedure.

You can synchronize the Secondary using automatic synchronization when the application is active or inactive.

To setup replication using automatic synchronization

- Start Secondary synchronization and replication using automatic synchronization by issuing the following command from any host in the RDS:

```
# vradmin -g hrdg -a startrep hr_rvg london
```

About SmartMove for VVR

The SmartMove for VVR feature enables VVR to leverage information from VxFS knowledge of the file system blocks in use to optimize the time and network bandwidth required for initial synchronization of replicated volumes. This feature is available if there is a VxFS file system mounted on top of the volume that is being synchronized.

If you use SmartMove with automatic synchronization, you can deploy the disaster recovery site faster because you are replicating far less data than the storage provisioned on the system. To use automatic synchronization with SmartMove in a cluster volume replication (CVR) environment, the file system must be mounted on the logowner.

The default behavior is to use the SmartMove for VVR feature for initial synchronization. The commands that use SmartMove during initial synchronization are `vradmin syncrvg/syncvol/startrep` and `vxrlink -a att`.

To turn off SmartMove

- Enter:

```
# vxtune usefssmartmove none
```

The `vradmin verifydata` command has also been enhanced to leverage VxFS knowledge of file system blocks in use for verification.

About thin storage reclamation and VVR

Thin storage helps you optimize your array capacity by allocating storage to applications only when it is needed. When files are created and written to in the file system, storage is allocated from a free storage pool on the array.

However, when you delete files on the host, the storage is not automatically returned to the pool. The result is large amounts of allocated storage on the array that is now unused. You must reclaim this storage manually.

See "Determining if a thin reclamation array needs reclamation" on page 98.

In a VVR environment, you can reclaim storage on volumes configured under an RVG. You can reclaim storage at the disk, disk group, or file system level.

Thin storage reclamation is only supported for LUNs that have the `thinrclm` attribute. VxVM automatically discovers LUNs that support thin reclamation from thin-capable storage arrays. On the host, you can list devices that have the `thinonly` or `thinrclm` attributes.

Thin storage reclamation is not supported on volumes in an RVG that has full instant or space-optimized snapshots that are associated to it. The reclaim command may complete without an error, but the storage space is not reclaimed. Thin storage reclamation is not supported as reclamation on the volume triggers a data transfer from the primary volume to the snapshot volume. Moving this data to the snapshot volume triggers storage allocation at the backend. If there are multiple snapshots, copies of the same data are maintained, which requires more storage than reclaimed. In the case of space-optimized snapshots, the cache object size increases as it copies all the reclaimable data to the space-optimized snapshot.

If you have used Storage Foundation thin storage reclamation in another context, the commands are identical when you use it in a VVR environment.

When you use thin reclamation with VVR, keep in mind the following:

- The VxFS file system must be mounted on the Primary site before you can perform thin reclamation on the volume.

- When you reclaim storage on the Primary site, it is automatically reclaimed on the Secondary site - unless the Primary site is in DCM mode or when autosync is running. The Primary site goes into DCM mode when its SRL overflows.
- You can reclaim storage on the Primary and Secondary sites even if the sites use different types of arrays. The arrays can have different fixed size physical storage allocation units.
- You can reclaim storage during a rolling upgrade with no impact on your systems.

For detailed information on thin storage, as well procedures for reclaiming thin storage, see *Veritas InfoScale™ Solutions Guide*.

Determining if a thin reclamation array needs reclamation

You can only perform thin storage reclamation on LUNs that have the `thinrclm` attribute. An array may be a good candidate for reclamation if:

- Your array-specific commands indicate that storage is nearing the maximum; for example, 90% is used.
- You receive an alert from the array that a certain storage level has been reached.

Even if storage capacity is reaching the maximum, that does not necessarily mean there is any data to reclaim; the data could still be needed. You should investigate further to determine if there are large blocks of deleted or unused data. If there are, run thin reclamation.

For more information reclaiming storage, see *Veritas InfoScale™ Solutions Guide*.

Starting replication when the data volumes are zero initialized

Use the option `-f` with the `vradmin startrep` command to start replication when the Primary and Secondary data volumes are zero initialized. The `vradmin startrep` command can be issued from any host in an RDS.

To start replication to a Secondary in an RDS when the data volumes are zero initialized:

```
# vradmin -g diskgroup -f startrep local_rvgname sec_hostname
```

The argument *`local_rvgname`* is the name of the RVG on the local host and represents its RDS.

The argument *`sec_hostname`* is the name of the Secondary host as displayed in the output of the `vradmin printrvg` command. If the RDS contains only one Secondary, the sec_hostname is optional.

Example: Starting replication when the data volumes are zero initialized

To start replication from the Primary RVG `hr_rvg` on `seattle` to the Secondary RVG on host `london` when the data volumes are zero initialized, issue the following command from any host in the RDS:

```
# vradmin -g hrdg -f startrep hr_rvg london
```

Note: The `-f` option can be used to stop the initial sync of the Primary and Secondary volumes if they already contain the same data.

Setting up third-party replication

Replication can be set up using VVR or various third-party replication technologies.

Set up replication according to the following best practices:

- Set up the replication. If you plan to use third-party software replication technology (Oracle Dataguard, IBM DB2HADR, etc.) or third-party hardware replication technology (EMC SRDF, Hitachi TrueCopy, IBM SVCCopyServices, etc.), review the vendor documentation for planning and configuring replication.
- Once the replication is configured, verify that the replicated storage is visible to all the nodes in the cluster(s).
- Verify that the storage vendor tools (CLIs or APIs) are installed on the VCS nodes as recommended in the respective replication agents Installation and Configuration Guide. For example, install RAID Manager CCI on each of the cluster nodes if you are using HITACHI TrueCopy replication.
- Download and install the required VCS replication agent from https://sort.veritas.com/agents.
- Configure the replication agent resources using guidelines from the replication agent's Installation and Configuration Guide.
- Using the scenarios provided in the Install and Configuration Guide, test failover for the configuration.

Configuring clusters for global cluster setup

Perform the following steps to configure the clusters for disaster recovery:

- Configure global cluster components at the primary site
  See “Configuring global cluster components at the primary site” on page 100.

- Install and configure VCS at the secondary site
  See "Installing and configuring VCS at the secondary site" on page 101.
- Secure communication between the wide-area connectors
  See "Securing communication between the wide-area connectors" on page 101.
- Configure remote cluster objects
  See "Configuring remote cluster objects" on page 103.
- Configure additional heartbeat links (optional)
  See "Configuring additional heartbeat links (optional)" on page 103.
- Configure the Steward process (optional)
  See "Configuring the Steward process (optional)" on page 104.

Configuring global cluster components at the primary site

Perform the following steps to configure global cluster components at the primary site.

If you have already completed these steps during the VCS cluster configuration at the primary site, then proceed to the next task to set up a VCS cluster at the secondary site.

See "Installing and configuring VCS at the secondary site" on page 101.

Run the GCO Configuration wizard to create or update the ClusterService group. The wizard verifies your configuration and validates it for a global cluster setup.

To configure global cluster components at the primary site

1. Start the GCO Configuration wizard.

   ```
   # gcoconfig
   ```

   The wizard discovers the NIC devices on the local system and prompts you to enter the device to be used for the global cluster.
2. Specify the name of the device and press Enter.
3. If you do not have NIC resources in your configuration, the wizard asks you whether the specified NIC will be the public NIC used by all systems.

   Enter **y** if it is the public NIC; otherwise enter **n**. If you entered n, the wizard prompts you to enter the names of NICs on all systems.
4. Enter the virtual IP to be used for the global cluster.

   You must use either IPv4 or IPv6 address. VCS does not support configuring clusters that use different Internet Protocol versions in a global cluster.

5 If you do not have IP resources in your configuration, the wizard does the following:

- For IPv4 address:
  The wizard prompts you for the netmask associated with the virtual IP. The wizard detects the netmask; you can accept the suggested value or enter another value.
- For IPv6 address:
  The wizard prompts you for the prefix associated with the virtual IP.

6 The wizard starts running commands to create or update the ClusterService group. Various messages indicate the status of these commands. After running these commands, the wizard brings the ClusterService group online.

7 Verify that the gcoip resource that monitors the virtual IP address for inter-cluster communication is online.

```
# hares -state gcoip
```

Installing and configuring VCS at the secondary site

Perform the following steps to set up a VCS cluster at the secondary site.

To install and configure VCS at the secondary site

1 At the secondary site, install and configure VCS cluster.

Note the following points for this task:

- During VCS configuration, answer the prompts to configure global cluster. This step configures the virtual IP for inter-cluster communication.

See the *Cluster Server Configuration and Upgrade Guide* for instructions.

2 Verify that the gcoip resource that monitors the virtual IP address for inter-cluster communication is online.

```
# hares -state gcoip
```

Securing communication between the wide-area connectors

Perform the following steps to configure secure communication between the wide-area connectors.

To secure communication between the wide-area connectors

1. Verify that security is configured in both the clusters. You can use the `installvcs -security` command to configure security.

   For more information, see the *Cluster Server Configuration and Upgrade Guide*.

2. Establish trust between the clusters.

   For example in a VCS global cluster environment with two clusters, perform the following steps to establish trust between the clusters:

   - On each node of the first cluster, enter the following command:

     ```
     # export EAT_DATA_DIR=/var/VRTSvcs/vcsauth/data/WAC;
     /opt/VRTSvcs/bin/vcsat setuptrust -b
     IP_address_of_any_node_from_the_second_cluster:14149 -s high
     ```

     The command obtains and displays the security certificate and other details of the root broker of the second cluster.
     If the details are correct, enter y at the command prompt to establish trust.
     For example:

     ```
     The hash of above credential is
       b36a2607bf48296063068e3fc49188596aa079bb
       Do you want to trust the above?(y/n) y
     ```

   - On each node of the second cluster, enter the following command:

     ```
     # export EAT_DATA_DIR=/var/VRTSvcs/vcsauth/data/WAC
     /opt/VRTSvcs/bin/vcsat setuptrust -b
     IP_address_of_any_node_from_the_first_cluster:14149 -s high
     ```

     The command obtains and displays the security certificate and other details of the root broker of the first cluster.
     If the details are correct, enter y at the command prompt to establish trust.
     Alternatively, if you have passwordless communication set up on the cluster, you can use the `installvcs -securitytrust` option to set up trust with a remote cluster.

3. - Skip the remaining steps in this procedure if you used the `installvcs -security` command after the global cluster was set up.
   - Complete the remaining steps in this procedure if you had a secure cluster and then used the `gcoconfig` command.

   On each cluster, take the wac resource offline on the node where the wac resource is online. For each cluster, run the following command:

```
# hares -offline wac -sys node_where_wac_is_online
```

4 Update the values of the StartProgram and MonitorProcesses attributes of the wac resource:

```
# haconf -makerw
hares -modify wac StartProgram \
"/opt/VRTSvcs/bin/wacstart -secure"
hares -modify wac MonitorProcesses \
"/opt/VRTSvcs/bin/wac -secure"
haconf -dump -makero
```

5 On each cluster, bring the wac resource online. For each cluster, run the following command on any node:

```
# hares -online wac -sys systemname
```

Configuring remote cluster objects

After you set up the VCS and replication infrastructure at both sites, you must link the two clusters. You must configure remote cluster objects at each site to link the two clusters. The Remote Cluster Configuration wizard provides an easy interface to link clusters.

To configure remote cluster objects

- Start the Remote Cluster Configuration wizard.

  From Cluster Explorer, click **Edit > Add/Delete Remote Cluster**.

  You must use the same IP address as the one assigned to the IP resource in the ClusterService group. Global clusters use this IP address to communicate and exchange ICMP heartbeats between the clusters.

Configuring additional heartbeat links (optional)

You can configure additional heartbeat links to exchange ICMP heartbeats between the clusters.

To configure an additional heartbeat between the clusters (optional)

1 On Cluster Explorer's **Edit** menu, click **Configure Heartbeats**.

2 In the Heartbeat configuration dialog box, enter the name of the heartbeat and select the check box next to the name of the cluster.

3 Click the icon in the **Configure** column to open the Heartbeat Settings dialog box.

4 Specify the value of the Arguments attribute and various timeout and interval fields. Click **+** to add an argument value; click **-** to delete it.

If you specify IP addresses in the Arguments attribute, make sure the IP addresses have DNS entries.

5 Click **OK**.

6 Click **OK** in the Heartbeat configuration dialog box.

Now, you can monitor the state of both clusters from the Java Console.

Configuring the Steward process (optional)

In case of a two-cluster global cluster setup, you can configure a Steward to prevent potential split-brain conditions, provided the proper network infrastructure exists.

See "About the Steward process: Split-brain in two-cluster global clusters" on page 31.

To configure the Steward process for clusters not running in secure mode

1 Identify a system that will host the Steward process.

2 Make sure that both clusters can connect to the system through a ping command.

3 Copy the file steward from a node in the cluster to the Steward system. The file resides at the following path:

```
/opt/VRTSvcs/bin/
```

4 In both clusters, set the Stewards attribute to the IP address of the system running the Steward process.

For example:

```
cluster cluster1938 (
UserNames = { admin = gNOgNInKOjOOmWOiNL }
ClusterAddress = "10.182.147.19"
Administrators = { admin }
CredRenewFrequency = 0
CounterInterval = 5
Stewards = {"10.212.100.165"}
}
```

5 On the system designated to host the Steward, start the Steward process:

```
# steward -start
```

To configure the Steward process for clusters running in secure mode

1 Verify that the prerequisites for securing Steward communication are met.

To verify that the wac process runs in secure mode, do the following:

- Check the value of the wac resource attributes:

```
# hares -value wac StartProgram
```

The value must be “/opt/VRTSvcs/bin/wacstart –secure.”

```
# hares -value wac MonitorProcesses
```

The value must be “/opt/VRTSvcs/bin/wac –secure.”

- List the wac process:

```
# ps -ef | grep wac
```

The wac process must run as “/opt/VRTSvcs/bin/wac –secure.”

2 Identify a system that will host the Steward process.

3 Make sure that both clusters can connect to the system through a ping command.

4 Perform this step *only* if VCS is not already installed on the Steward system. If VCS is already installed, skip to step 5.

- Install the VRTSvcs and VRTSperl rpms.

- If the cluster UUID is not configured, configure it by using `/opt/VRTSvcs/bin/uuidconfig.pl`.
- On the system that is designated to run the Steward process, run the `installvcs -securityonenode` command.
  The installer prompts for a confirmation if VCS is not configured or if VCS is not running on all nodes of the cluster. Enter **y** when the installer prompts whether you want to continue configuring security.
  For more information about the `-securityonenode` option, see the *Cluster Server Configuration and Upgrade Guide*.

5 Generate credentials for the Steward using `/opt/VRTSvcs/bin/steward_secure.pl` or perform the following steps:

```
# unset EAT_DATA_DIR

# unset EAT_HOME_DIR

# /opt/VRTSvcs/bin/vcsauth/vcsauthserver/bin/vssat createpd -d
VCS_SERVICES -t ab

# /opt/VRTSvcs/bin/vcsauth/vcsauthserver/bin/vssat addprpl  -t ab
-d VCS_SERVICES -p  STEWARD -s password

# mkdir -p /var/VRTSvcs/vcsauth/data/STEWARD

# export EAT_DATA_DIR=/var/VRTSvcs/vcsauth/data/STEWARD

# /opt/VRTSvcs/bin/vcsat setuptrust  -s high -b localhost:14149
```

6 Set up trust on all nodes of the GCO clusters:

```
# export EAT_DATA_DIR=/var/VRTSvcs/vcsauth/data/WAC

# vcsat setuptrust -b <IP_of_Steward>:14149 -s high
```

7 Set up trust on the Steward:

```
# export EAT_DATA_DIR=/var/VRTSvcs/vcsauth/data/STEWARD

# vcsat setuptrust -b <VIP_of_remote_cluster1>:14149 -s high

# vcsat setuptrust -b <VIP_of_remote_cluster2>:14149 -s high
```

8 In both the clusters, set the Stewards attribute to the IP address of the system running the Steward process.

For example:

```
cluster cluster1938 (
UserNames = { admin = gNOgNInKOjOOmWOiNL }
ClusterAddress = "10.182.147.19"
Administrators = { admin }
CredRenewFrequency = 0
CounterInterval = 5
Stewards = {"10.212.100.165"}
}
```

9 On the system designated to run the Steward, start the Steward process:

```
# /opt/VRTSvcs/bin/steward -start -secure
```

To stop the Steward process

- To stop the Steward process that is not configured in secure mode, open a new command window and run the following command:

```
# steward -stop
```

To stop the Steward process running in secure mode, open a new command window and run the following command:

```
# steward -stop -secure
```

Configuring service groups for global cluster setup

Perform the following steps to configure the service groups for disaster recovery.

To configure service groups

1 At the secondary site, set up the application for high availability.

Configure VCS service groups for the application. Create a configuration that is similar to the one in the first cluster.

- You can do this by either using Cluster Manager (Java Console) to copy and paste resources from the primary cluster, or by copying the configuration from the main.cf file in the primary cluster to the secondary cluster.

- Make appropriate changes to the configuration. For example, you must modify the SystemList attribute to reflect the systems in the secondary cluster.
- Make sure that the name of the service group (appgroup) is identical in both clusters.

2 To assign remote administration privileges to users for remote cluster operations, configure users with the same name and privileges on both clusters.

See "User privileges for cross-cluster operations" on page 24.

3 If your setup uses BIND DNS, add a resource of type DNS to the application service group at each site.

Refer to the *Cluster Server Bundled Agent's Reference Guide* for more details.

4 At each site, perform the following depending on the replication technology you have set up:

- Volume Replicator
  Configure VCS to manage and monitor Replicated Volume Groups (RVGs).
  See "Configuring VCS service group for VVR-based replication" on page 108.
- A supported third-party replication technology
  Install and configure the corresponding VCS agent for replication.
  See the Installation and Configuration Guide for the corresponding VCS replication agent for instructions.

Configuring VCS service group for VVR-based replication

Perform the following steps to configure VCS to monitor Volume Replicator (VVR). Then, set an online local hard group dependency from application service group (appgroup) to replication service group (appgroup_rep) to ensure that the service groups fail over and switch together.

To create the RVG resources in VCS

1 Create a new service group, say appgroup_rep.

2 Copy the DiskGroup resource from the appgroup to the new group.

3 Configure new resources of type IP and NIC in the appgroup_rep service group. The IP resource monitors the virtual IP that VVR uses for replication.

4 Configure a new resource of type RVG in the new (appgroup_rep) service group.

5 Configure the RVG resource.

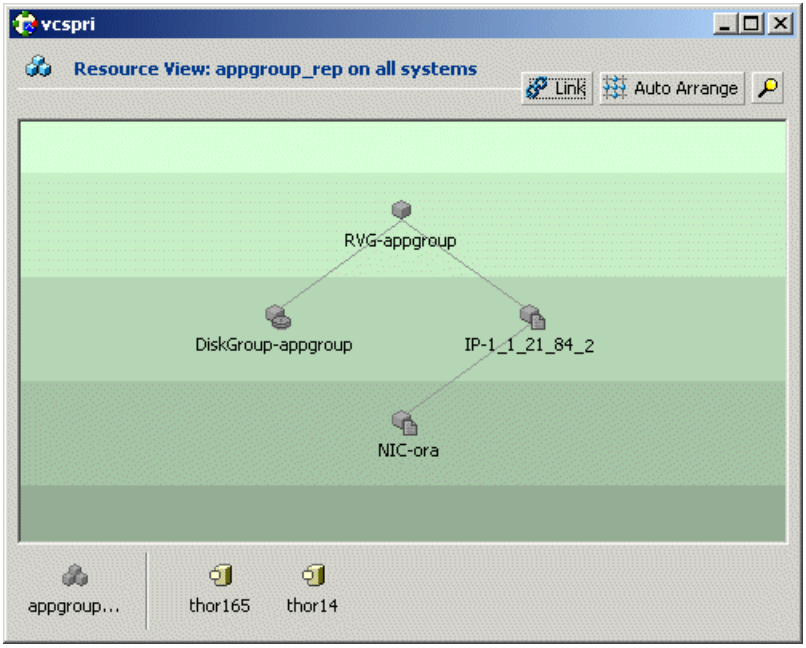
See the *Veritas InfoScale™ Replication Administrator's Guide* for more information about the resource.

Note that the RVG resource starts, stops, and monitors the RVG in its current state and does not promote or demote VVR when you want to change the direction of replication. That task is managed by the RVGPrimary agent.

6 Set dependencies as per the following information:

- RVG resource depends on the IP resource.
- RVG resource depends on the DiskGroup resource.
- IP resource depends on the NIC resource.

The service group now looks like:



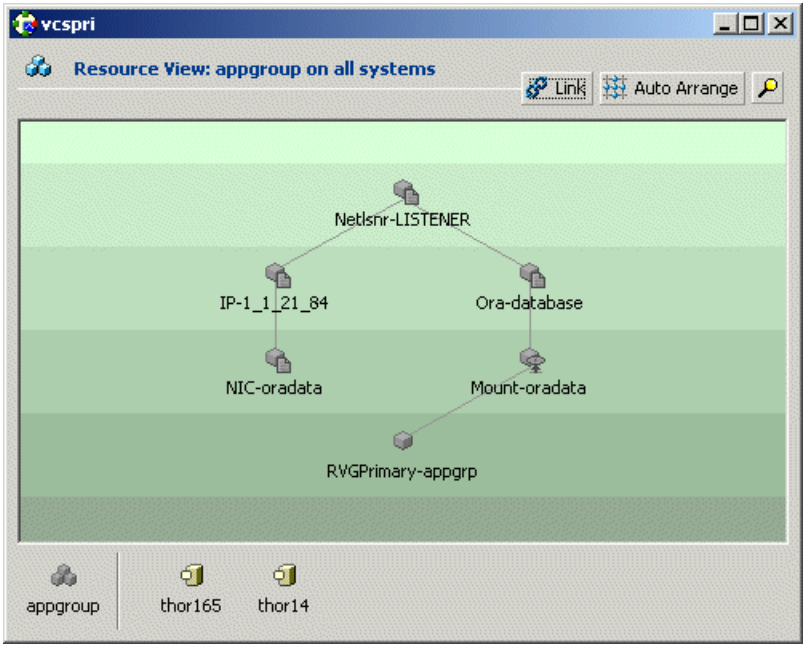
7 Delete the DiskGroup resource from the appgroup service group.

8 In the application service group (appgroup), add a resource of type RVGPrimary and configure its attributes.

See the *Veritas InfoScale™ Replication Administrator's Guide* for more information about the resource.

9 Set resource dependencies such that the Mount resource depends on the RVGPrimary resource.

The appgroup now looks like:



To link the application and replication service groups

1 In the Cluster Explorer configuration tree, click the cluster name.

2 In the view panel, click the **Service Groups** tab.

This opens the service group dependency graph.

3 Click **Link**.

4 Click the parent group, appgroup, and move the mouse toward the child group, appgroup_rep.

5 Click the child group appgroup_rep.

6 In the Link Service Groups dialog box, click the online local relationship and the hard dependency type and click **OK**.

Configuring a service group as a global service group

Run the Global Group Configuration wizard to configure the application service group (appgroup) as a global group.

To create the global service group

1 In the service group tree of Cluster Explorer, right-click the application service group (appgroup).

2 Select **Configure As Global** from the menu.

3 Enter the details of the service group to modify (appgroup).

4 From the **Available Clusters** box, click the clusters on which the group can come online. The local cluster is not listed as it is implicitly defined to be part of the ClusterList. Click the right arrow to move the cluster name to the **ClusterList** box.

5 Select the policy for cluster failover:

- **Manual** prevents a group from automatically failing over to another cluster.
- **Auto** enables a group to automatically fail over to another cluster if it is unable to fail over within the cluster, or if the entire cluster faults.
- **Connected** enables a group to automatically fail over to another cluster if it is unable to fail over within the cluster.

6 Click **Next**.

7 Enter or review the connection details for each cluster.

Click the **Configure** icon to review the remote cluster information for each cluster.

8 Enter the IP address of the remote cluster, the IP address of a cluster system, or the host name of a cluster system.

9 Enter the user name and the password for the remote cluster and click **OK**.

10 Click **Next**.

11 Click **Finish**.

12 Save the configuration.

The appgroup service group is now a global group and can be failed over between clusters.

Fire drill in global clusters

A fire drill procedure verifies the fault-readiness of a disaster recovery configuration. This procedure is performed without stopping the application at the Primary site and disrupting user access.

A fire drill is performed at the Secondary site using a special service group for fire drills. The fire drill service group is identical to the application service group, but uses a fire drill resource in place of the replication agent resource. The fire drill service group uses a copy of the data that is used by the application service group.

Almost all of the replication agents have a corresponding fire drill agent bundled with them. These fire drill agents are responsible for taking a snapshot inside the storage array and importing the snapshot disks on the hosts. The fire drill agents are named similar to the replication agents with the suffix “Snap” appended to the agent name. For example, the fire drill agent for EMC SRDF is SRDFSnap and the fire drill agent for Hitachi TrueCopy (HTC) agent is HTCSnap.

For more information about configuring fire drills, please refer the Installation and Configuration Guide of the appropriate replication agent.

Chapter 8

Configuring a global cluster with Storage Foundation Cluster File System High Availability, Storage Foundation for Oracle RAC, or Storage Foundation for Sybase CE

This chapter includes the following topics:

- About global clusters
- About replication for parallel global clusters using Storage Foundation and High Availability (SFHA) Solutions
- About setting up a global cluster environment for parallel clusters
- Configuring the primary site
- Configuring the secondary site
- Setting up replication between parallel global cluster sites
- Testing a parallel global cluster configuration

About global clusters

Global clusters provide the ability to fail over applications between geographically distributed clusters when disaster occurs. This type of clustering involves migrating applications between clusters over a considerable distance. You can set up HA/DR using hardware-based or software-based replication technologies.

You may add this license during the installation or at any time after the installation completes.

About replication for parallel global clusters using Storage Foundation and High Availability (SFHA) Solutions

Global clusters provide the ability to fail over applications between geographically distributed clusters when disaster occurs. This type of clustering involves migrating applications between clusters over a considerable distance. You can set up HA/DR using hardware-based or software-based replication technologies.

You can set up a primary cluster for replication to a secondary cluster by configuring global VCS service groups and using a replication technology. The database cluster at the secondary site can be a single node cluster. For example, you can have a two-node cluster on the primary site and a two-node or single-node cluster on the secondary site.

You can use one of the following replication technologies:

- Veritas Volume Replicator (VVR), which provides host-based volume replication. Using VVR you can replicate data volumes on a shared disk group in Storage Foundation Cluster File System High Availablity (SFCFS HA), Storage Foundation for Oracle RAC (SF Oracle RAC), or Storage Foundation for Sybase ASE CE (SF Sybase CE).
- Supported hardware-based replication technologies. Using hardware-based replication you can replicate data from a primary array to a secondary array.

To verify your configuration is supported, review the product requirements and licensing information:

See the *Storage Foundation Cluster File System High Availability Configuration and Upgrade Guide*.

See the *Storage Foundation for Oracle RAC Configuration and Upgrade Guide*.

See the *Storage Foundation for Sybase ASE CE Configuration and Upgrade Guide*.

To confirm the compatibility of your hardware, see the current compatibility list in the Veritas Technical Support website:

https://www.veritas.com/support/en_US/article.000126344

SFCFSHA and SF Oacle RAC support multiple third-party replication options.

Storage Foundation for Sybase CE supports VVR replication only at this time.

For current information on third-party replication support:

See: https://sort.veritas.com/agents and select **Replication Agents** under **Agent type**.

Notes:

- SF Sybase CE supports replication with VVR only.
- Check your vendor's compatibility list for the supported software versions. The support listed above only exists if the host, HBA, and array combination is in your vendor's hardware compatibility list. Check your array documentation.
- All arrays must support SCSI-3 persistent reservations.

The Veritas replication agents provide application failover and recovery support to your replication configuration in environments where data is replicated between clusters.

VCS replication agents control the direction of replication. They do not monitor the progress or status of replication. The replication agents manage the state of replicated devices that are attached to global cluster nodes. The agents make sure that the system which has the resource online also has safe and exclusive access to the configured devices.

For more current information on the replicated agents:

See the *Cluster Server Bundled Agents Guide*

About setting up a global cluster environment for parallel clusters

Configuring a global cluster for environment with parallel clusters requires the coordination of many component setup tasks. The procedures provided here are guidelines. Refer to your product installation guide for your product installation and basic configuration for each cluster. You will need this guide to configure a global cluster environment and replication between the two configured clusters.

Table 8-1 Tasks for setting up a global cluster environment with parallel clusters

Task	Description
Configure a parallel cluster at the primary site	See "Configuring the primary site" on page 116.
Configure an parallel cluster at the secondary site	See "Configuring the secondary site" on page 119.
Configure a global cluster environment	See "Setting up replication between parallel global cluster sites" on page 125.
Test the HA/DR configuration	See "Testing a parallel global cluster configuration" on page 132.

Upon successful testing, you can bring the environment into production

Some configuration tasks may require adjustments depending upon your particular starting point, environment, and configuration: details of your configuration may differ from the examples given in the procedures. Review the installation requirements and sample cluster configuration files for primary and secondary clusters.

For requirements, installation instructions, and sample configuration files:

See the *Storage Foundation Cluster File System High Availability Configuration and Upgrade Guide*.

See the *Storage Foundation for Oracle RAC Configuration and Upgrade Guide*.

See the *Storage Foundation for Sybase ASE CE Configuration and Upgrade Guide*.

Configuring the primary site

Table 8-2 Tasks for setting up a parallel global cluster at the primary site

Task	Description
Set up the cluster	See "To set up the cluster at the primary site" on page 117.
Set up the database	See "To set up the database at the primary site" on page 119.

Consult your product installation guide for planning information as well as specific configuration guidance for the steps below.

See the *Storage Foundation Cluster File System High Availability Configuration and Upgrade Guide*.

See the *Storage Foundation for Oracle RAC Configuration and Upgrade Guide*.

See the *Storage Foundation for Sybase ASE CE Configuration and Upgrade Guide*.

You can use an existing parallel cluster or you can install a new cluster for your primary site. If you are using an existing cluster as the primary and you want to set up a global cluster, skip the steps below and proceed to configure your secondary cluster.

See "Configuring the secondary site" on page 119.

If you do not have an existing cluster and you are setting up two new sites for a global cluster, follow the steps below.

To set up the cluster at the primary site

1. Install and configure servers and storage.
2. If you are using hardware-based replication, install the sofware for managing your array.
3. Verify that you have the correct installation options enabled, whether you are using keyless licensing or installing keys manually. You must have the GCO option enabled for a global cluster. If you are using VVR for replication, you must have it enabled.
4. Prepare, install, and configure your Storage Foundation and High Availability (SFHA) Solutions product according to the directions in your product's installation guide.

   For a multi-node cluster, configure I/O fencing.
5. Verify the CVM group is online on all nodes in the primary cluster.

   ```
   # hagrp -state cvm
   ```
6. Set storage connectivity for volume asymetry and I/O shiping for DCO logs policies for the disk group.
7. Prepare systems and storage for a global cluster. Identify the hardware and storage requirements before installing your database software.

   For SFCFSHA, you will need to set up:

   - Local storage for database software
   - Shared storage for resources which are not replicated as part of the hardware-based or host-based replication

- Replicated storage for database files

For SF Oracle RAC, you will need to set up:

- Local storage for Oracle RAC and CRS/GRID binaries
- Shared storage for OCR and Vote disk which is not replicated as part of the hardware-based or host-based replication
- Replicated shared storage for database files

For SF Sybase CE, you will need to set up:

- Shared storage for File sSystem and Cluster File System for Sybase ASE CE binaries which is not replicated
- Shared storage for the quorum device which is not replicated
- Replicated storage for database files
- Verify the configuration using procedures in the *Storage Foundation for Sybase ASE CE Configuration and Upgrade Guide*.

8 For SFCFSHA, install and configure your database binaries. Consult your database documentation.

Note: Resources which will not be replicated must be on non-replicated shared storage.

After successful database installation and configuration, verify that database resources are up on all nodes.

9 For Oracle RAC, see the instructions in the *Storage Foundation for Oracle RAC Configuration and Upgrade Guide* for installing, configuring, and linking Oracle database binaries:

- Oracle Clusterware/Grid Infrastructure software
- Oracle RAC database software
- The Oracle RAC binary versions must be exactly same on both sites.

Note: OCR and Vote disk must be on non-replicated shared storage.

After successful Oracle RAC installation and configuration, verify that CRS daemons and resources are up on all nodes.

```
$GRID_HOME/bin/crsctl stat res -t
```

10 For SF Sybase CE, see the instructions in the *Storage Foundation for Sybase ASE CE Configuration and Upgrade Guide* for installing and configuring Sybase ASE CE binaries.

Note the following configuration requirements:

- The quorum device must be on non-replicated shared storage.
- The Sybase binary versions must be exactly same on both sites, including the ESD versions.
- Configure Sybase binaries mounts/volumes under VCS control manually on the secondary site.

To set up the database at the primary site

1 Identify the disks that will be replicated, create the required CVM disk group, volume, and file system.

2 Create the database on the file system you created in the previous step.

3 Configure the VCS service groups for the database.

4 Verify that all VCS service groups are online.

Configuring the secondary site

The setup requirements for the secondary site parallel the requirements for the primary site with a few additions or exceptions as noted below.

Table 8-3 Tasks for setting up a parallel global cluster at the secondary site

Task	Description
Set up the cluster	See "To set up the cluster on secondary site" on page 120.
Set up the database	See "To set up the SFCFSHA database for the secondary site" on page 122. See "To set up the Oracle RAC database for the secondary site" on page 123. See "To set up the Sybase ASE CE database for the secondary site" on page 125.

Important requirements for parallel global clustering:

- Cluster names on the primary and secondary sites must be unique.

- You must use the same OS user and group IDs for your database for installation and configuration on both the primary and secondary clusters.
- For Oracle RAC, you must use the same directory structure, name, permissions for the CRS/GRID and database binaries.

You can use an existing parallel cluster or you can install a new cluster for your secondary site.

Consult your product installation guide for planning information as well as specific configuration guidance for the steps below.

See the *Storage Foundation Cluster File System High Availability Configuration and Upgrade Guide*.

See the *Storage Foundation for Oracle RAC Configuration and Upgrade Guide*.

See the *Storage Foundation for Sybase ASE CE Configuration and Upgrade Guide*.

To set up the cluster on secondary site

1. Install and configure servers and storage.
2. If you are using hardware-based replication, install the sofware for managing your array.
3. Verify that you have the correct installation options enabled, whether you are using keyless licensing or installing keys manually. You must have the GCO option for a global cluster. If you are using VVR for replication, you must have it enabled.
4. Prepare, install, and configure your Storage Foundation and High Availability (SFHA) Solutions product according to the directions in your product's installation guide.

   For a multi-node cluster, configure I/O fencing.
5. For a single-node cluster, do not enable I/O fencing. Fencing will run in disabled mode.
6. Prepare systems and storage for a global cluster. Identify the hardware and storage requirements before installing your database software.

   For SFCFSHA, you will need to set up:

   - Local storage for database software
   - Shared storage for resources which are not replicated as part of the hardware-based or host-based replication
   - Replicated storage for database files
   - You must use the same directory structure, name, permissions for the quorum and database binaries as on the primary.

For SF Oracle RAC, you will need to set up:

- Local storage for Oracle RAC and CRS binaries
- Shared storage for OCR and Vote disk which is not replicated as part of the hardware-based or host-based replication
- Replicated shared storage for database files
- You must use the same directory structure, name, permissions for the CRS/GRID and database binaries as on the primary.

For SF Sybase CE, you will need to set up:

- Shared storage for File sSystem and Cluster File System for Sybase ASE CE binaries which is not replicated
- Shared storage for the quorum device which is not replicated
- Replicated storage for database files
- You must use the same directory structure, name, permissions for the quorum and database binaries as on the primary.
- Verify the configuration using procedures in the *Storage Foundation for Sybase ASE CE Configuration and Upgrade Guide*.

Note: You must use the same directory structure, name, permissions for the CRS/GRID and database binaries.

---

7 For SFCFSHA, install and configure your database binaries. Consult your database documentation.

Note: Resources which will not be replicated must be on non-replicated shared storage.

After successful database installation and configuration, verify that database resources are up on all nodes.

8 For Oracle RAC, see the instructions in the *Storage Foundation for Oracle RAC Configuration and Upgrade Guide* for installing and configuring:

- Oracle Clusterware/Grid Infrastructure software
- Oracle RAC database software
- The Oracle RAC binary versions must be exactly same on both sites.

Note: OCR and Vote disk must be on non-replicated shared storage.

After successful Oracle RAC installation and configuration, verify that CRS daemons and resources are up on all nodes.

```
$GRID_HOME/bin/crsctl stat res -t
```

9 For SF Sybase CE, see the instructions in the *Storage Foundation for Sybase ASE CE Configuration and Upgrade Guide* for installing and configuring Sybase ASE CE binaries.

Note the following configuration requirements:

- The quorum device must be on non-replicated shared storage.
- The Sybase binary versions must be exactly same on both sites, including the ESD versions.
- Configure Sybase Binaries mounts/volumes under VCS control manually on the secondary site.

Do not create the database. The database will be replicated from the primary site.

To set up the SFCFSHA database for the secondary site

1 If you are using hardware-based replication, the database, disk group, and volumes will be replicated from the primary site.

Create the directory for the CFS mount point which will host the database data and control files.

2 If you are using VVR for replication, create an identical disk group and volumes for the replicated content with the same names and size as listed on the primary site.

Create the directories for the CFS mount points as they are on the primary site. These will be used to host the database and control files when the failover occurs and the secondary is promoted to become the primary site.

3 Create subdirectories for the database as you did on the primary site.

To set up the Oracle RAC database for the secondary site

1. If you are using hardware-based replication, the database, disk group, and volumes will be replicated from the primary site.

   Create the directory for the CFS mount point which will host the database data and control files.

2. If you are using VVR for replication, create an identical disk group and volumes for the replicated content with the same names and size as listed on the primary site.

   Create the directories for the CFS mount points as they are on the primary site. These will be used to host the database and control files when the failover occurs and the secondary is promoted to become the primary site.

3. On each node in the cluster, copy the initialization files (pfiles,spfiles) from the primary cluster to the secondary cluster maintaining the same directory path.

   For example, copy init$ORACLE_SID.ora and orapw$ORACLE_SID.ora from $ORACLE_HOME/dbs at the primary to $ORACLE_HOME/dbs at the secondary.

4. As Oracle user, create the following subdirectories on the secondary site to parallel the directories on the primary site:

   ```
   $ mkdir -p $ORACLE_BASE/diag
   $ mkdir -p $ORACLE_BASE/admin
   $ mkdir -p $ORACLE_BASE/admin/adump
   ```

   On both the primary and secondary sites, edit the file:

   ```
   $ORACLE_HOME/dbs/init$ORACLE_SID.ora
   ```

   as

   ```
   remote_listener = 'SCAN_NAME:1521'
   SPFILE=<SPFILE NAME>
   ```

5. Configure listeners on the secondary site with same name as on primary. You can do this by one of the following methods:

   - Copy the listener.ora and tnsnames.ora files from the primary site and update the names as appropriate for the secondary site.
   - Use Oracle's netca utility to to configure the listener.ora and tnsnames.ora files on the secondary site.

6 On the secondary site, register the database using the `srvctl` command as the database software owner.

Registering the database only has to be done once from any node in the secondary cluster.Use the following command as the Oracle database software owner

```
$ $ORACLE_HOME/bin/srvctl add database -d database_name -o oracle_home
```

7 To prevent automatic database instance restart, change the Management policy for the database (automatic, manual) to MANUAL using the `srvctl` command:

```
$ $ORACLE_HOME/bin/srvctl modify database -d database_name -y manual
```

You need only perform this change once from any node in the cluster.

8 Register the instances using srvctl command. Execute the following command on each node:

```
$ $ORACLE_HOME/bin/srvctl add instance -d database_name \
-i instance_name -n node-name
```

If the secondary cluster has more than one node, you must add instances using the `srvctl` command.

For example, if the database instance name is racdb, the instance name on sys3 is racdb1 and on sys4 is racdb2.

```
$ $ORACLE_HOME/bin/srvctl add instance -d racdb -i racdb1 -n sys3
```

```
$ $ORACLE_HOME/bin/srvctl add instance -d racdb -i racdb2 -n sys4
```

9 Register all other resources (for example listener, ASM, service) present in cluster/GRID at the primary site to the secondary site using the `srvctl` command or `crs_register`. For command details, see Oracle documentation at Metalink.

To set up the Sybase ASE CE database for the secondary site

1. Create the directories for the CFS mount points as they are on the primary site. These will be used to host the database files when the failover occurs and the secondary is promoted to become the primary site.
2. Create an identical disk group and volumes for the replicated content with the same names and size as listed on the primary site.

Configuring the Sybase ASE CE cluster on the secondary site

Before bringing Sybase ASE CE online at the secondary site, you must configure the Sybase ASE CE cluster on the secondary site.

- Modify the Sybase ASE CE configuration files to match the secondary site environment.
- Build the new quorum device for the secondary site.
- For configuration details: See "Configuring the Sybase ASE CE cluster on the secondary site" on page 165.

Setting up replication between parallel global cluster sites

You have configured Cluster Server (VCS) service groups for the database on each cluster. Each cluster requires an additional virtual IP address associated with the cluster for cross-cluster communication. The VCS installation and creation of the ClusterService group typically involves defining this IP address.

Configure a global cluster by setting:

- Heartbeat
- Wide area cluster (wac)
- GCO IP (gcoip)
- remote cluster resources

Table 8-4 Tasks for configuring a parallel global cluster

Task	Description
Prepare to configure global parallel clusters	Before you configure a global cluster, review the following requirements: ■ Cluster names on the primary and secondary sites must be unique. ■ Node and resource names must be unique within a cluster but not across clusters. ■ Each cluster requires a virtual IP address associated with the cluster. The VCS installation and creation of the ClusterService group typically involves defining this IP address. If you did not configure the ClusterService group when you installed your SFHA Solutions product, configure it when you configure global clustering. ■ One WAN (Wide Area Network) heartbeat must travel between clusters, assuming each cluster has the means to monitor the health of the remote cluster. Configure the heartbeat resource manually. ■ All database user and group IDs must be the same on all nodes. ■ The database, which is replicated from the storage on the primary site to the secondary site, must be defined in a global group having the same name on each cluster. Each resource in the group may differ from cluster to cluster, but clients redirected to a remote cluster after a wide-area failover must see the same application as the one in the primary cluster.
Configure a global cluster using the global clustering wizard.	See "To modify the ClusterService group for global clusters using the global clustering wizard" on page 127.
Define the remote global cluster and heartbeat objects	See "To define the remote cluster and heartbeat" on page 128.
Configure global service groups for database resources	See "To configure global service groups for database resources" on page 132.

Table 8-4 Tasks for configuring a parallel global cluster *(continued)*

Task	Description
Start replication between the sites.	For software-based replication using Volume Replicator (VVR): See "About configuring a parallel global cluster using Volume Replicator (VVR) for replication" on page 135. For replication using Oracle Data Guard see the Data Guard documentation by Oracle. For replication using hardware-based replication see the replicated agent guide for your hardware. See the *Cluster Server Bundled Agents Guide*
Test the HA/DR configuration before putting it into production	See "Testing a parallel global cluster configuration" on page 132.

The global clustering wizard completes the following tasks:

- Validates the ability of the current configuration to support a global cluster environment.
- Creates the components that enable the separate clusters, each of which contains a different set of GAB memberships, to connect and operate as a single unit.
- Creates the ClusterService group, or updates an existing ClusterService group.

To modify the ClusterService group for global clusters using the global clustering wizard

1 On the primary cluster, start the GCO Configuration wizard:

```
# /opt/VRTSvcs/bin/gcoconfig
```

2 The wizard discovers the NIC devices on the local system and prompts you to enter the device to be used for the global cluster. Specify the name of the device and press Enter.

3 If you do not have NIC resources in your configuration, the wizard asks you whether the specified NIC will be the public NIC used by all the systems. Enter **y** if it is the public NIC; otherwise enter **n**. If you entered **n**, the wizard prompts you to enter the names of NICs on all systems.

4 Enter the virtual IP address for the local cluster.

5 If you do not have IP resources in your configuration, the wizard prompts you for the netmask associated with the virtual IP. The wizard detects the netmask; you can accept the suggested value or enter another one.

The wizard starts running commands to create or update the ClusterService group. Various messages indicate the status of these commands. After running these commands, the wizard brings the ClusterService failover group online on any one of the nodes in the cluster.

After configuring global clustering, add the remote cluster object to define the IP address of the cluster on the secondary site, and the heartbeat object to define the cluster-to-cluster heartbeat. Heartbeats monitor the health of remote clusters. VCS can communicate with the remote cluster only after you set up the heartbeat resource on both clusters.

To define the remote cluster and heartbeat

1 On the primary site, enable write access to the configuration:

```
# haconf -makerw
```

2 On the primary site, define the remote cluster and its virtual IP address.

In this example, the remote cluster is clus2 and its IP address is 10.11.10.102:

```
# haclus -add clus2 10.11.10.102
```

3 Complete step 1 and step 2 on the secondary site using the name and IP address of the primary cluster.

In this example, the primary cluster is clus1 and its IP address is 10.10.10.101:

```
# haclus -add clus1 10.10.10.101
```

4 On the primary site, add the heartbeat object for the cluster. In this example, the heartbeat method is ICMP ping.

```
# hahb -add Icmp
```

5 Define the following attributes for the heartbeat resource:

- ClusterList lists the remote cluster.

- Arguments enable you to define the virtual IP address for the remote cluster.

For example:

```
# hahb -modify Icmp ClusterList clus2
# hahb -modify Icmp Arguments 10.11.10.102 -clus clus2
```

6 Save the configuration and change the access to read-only on the local cluster:

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

7 Complete step 4-6 on the secondary site using appropriate values to define the cluster on the primary site and its IP as the remote cluster for the secondary cluster.

8 It is advisible to modify "OnlineRetryLimit" & "OfflineWaitLimit" attribute of IP resource type to 1 on both the clusters:

```
# hatype -modify IP  OnlineRetryLimit  1

# hatype -modify IP  OfflineWaitLimit  1
```

9 Verify cluster status with the `hastatus -sum` command on both clusters.

```
# hastatus -sum
```

10 Display the global setup by executing haclus -list command.

```
# haclus -list
    clus1
    clus2
```

Example of heartbeat additions to the main.cf file on the primary site:

```
.
.
remotecluster clus2 (
Cluster Address = "10.11.10.102"
)
heartbeat Icmp (
    ClusterList = { clus2 }
    Arguments @clus2 = { "10.11.10.102" }
    )

system sys1 (
    )
.
.
```

Example heartbeat additions to the main.cf file on the secondary site:

```
.
.
remotecluster clus1 (
    Cluster Address = "10.10.10.101"
    )

heartbeat Icmp (
    ClusterList = { clus1 }
    Arguments @clus1 = { "10.10.10.101" }
    )

system sys3 (
    )
.
.
```

See the *Cluster Server Administrator's Guide* for more details for configuring the required and optional attributes of the heartbeat object.

To configure global service groups for database resources

1. Configure and enable global groups for databases and resources.
   - Configure VCS service groups at both sites.
   - Configure the replication agent at both sites.
   - For SF Oracle RAC, make the Oracle RAC service group a global service group, enabling failover across clusters.
   - For SF Sybase CE, make the database service group (sybasece) a global service group, enabling failover across clusters.
   - For example:
     See "Modifying the Cluster Server (VCS) configuration on the primary site" on page 152.
2. To test real data in an environment where HA/DR has been configured, schedule a planned migration to the secondary site for testing purposes.

   For example:

   See "To migrate the role of primary site to the remote site" on page 168.

   See "To migrate the role of new primary site back to the original primary site" on page 169.
3. Upon successful testing, bring the environment into production.

For more information about VCS replication agents:

See the *Cluster Server Bundled Agents Guide*

For complete details on using VVR in a shared disk environment:

See the *Veritas InfoScale™ Replication Administrator's Guide*.

Testing a parallel global cluster configuration

Veritas recommends testing before putting a global cluster configuration into production.

To test a planned failover

1. Offline the VCS service group for the database on the cluster at the primary site.
2. Online the VCS service group for the database on the cluster at the secondary site.

To test distaster recovery at the recovery site

1. Plan downtime to test the disaster recovery configuration.
2. Simulate a disaster at the primary site.

   For example:

   Shut down the hosts and storage arrays at the primary. If you can not shut down the storage arrays, disconnect the replication link between the sites.
3. Use VCS to fail over the database to the cluster at the secondary site.

To test failback on the primary site

1. Offline the VCS service group for the database on the cluster at the secondary site.
2. If the nodes and storage are down, restart the nodes and the storage array at the primary site.
3. Reconnect the replication link if it was broken.
4. Resynchronize the data from the secondary to make sure the data at the primary site is current.
5. Bring the VCS service group online at the primary site.

Chapter 9

Configuring global clusters with VVR and Storage Foundation Cluster File System High Availability, Storage Foundation for Oracle RAC, or Storage Foundation for Sybase CE

This chapter includes the following topics:

- About configuring a parallel global cluster using Volume Replicator (VVR) for replication
- Setting up replication on the primary site using VVR
- Setting up replication on the secondary site using VVR
- Starting replication of the primary site database volume to the secondary site using VVR
- Configuring Cluster Server to replicate the database volume using VVR
- Replication use cases for global parallel clusters

About configuring a parallel global cluster using Volume Replicator (VVR) for replication

Configuring a global cluster for environment with parallel clusters using Volume Replicator requires the coordination of many component setup tasks. The tasks listed below are guidelines.

Before configuring two clusters for global clustering, you must verify that:

- You have the correct installation options enabled for your Storage Foundations and High Availability (SFHA) Solutions product, whether you are using keyless licensing or installing keys manually. You must have the GCO option for a global cluster and VVR enabled.
  Review your product requirements and licensing information.
- Both clusters have your SFHA Solutions product software installed and configured.
  See "Configuring the primary site" on page 116.
  See "Configuring the secondary site" on page 119.

Note: You can install and configure both clusters at the same time, or you can configure the second cluster at a later time than the first.

You you can use this guide to configure a global cluster environment and replication between the two clusters. For installation and configuration information:

See the *Storage Foundation Cluster File System High Availability Configuration and Upgrade Guide*.

See the *Storage Foundation for Oracle RAC Configuration and Upgrade Guide*.

See the *Storage Foundation for Sybase ASE CE Configuration and Upgrade Guide*.

With two clusters installed and configured , you are ready to configure a global cluster environment using VVR. You must perform the following tasks to modify both cluster configurations to support replication in the global cluster environment.

Table 9-1 Tasks for configuring a parallel global cluster with VVR

Task	Description
Setting up replication on the primary site	■ Create the Storage Replicator Log (SRL) in the disk group for the database. ■ Create the Replicated Volume Group (RVG) on the primary site. See " Setting up replication on the primary site using VVR" on page 137.
Setting up replication on the secondary site	■ Create a disk group to hold the data volume, SRL, and RVG on the storage on the secondary site. You must match the names and sizes of these volumes with the names and sizes of the volumes on the primary site. ■ Edit the /etc/vx/vras/.rdg file on the secondary site. ■ Use resolvable virtual IP addresses that set network RLINK connections as host names of the primary and secondary sites. ■ Create the replication objects on the secondary site. See "Setting up replication on the secondary site using VVR" on page 140.
Starting replication of the database.	You can use either of the following methods to start replication: ■ Automatic synchronization ■ Full synchronization with Storage Checkpoint See "Starting replication of the primary site database volume to the secondary site using VVR" on page 145.
Configuring VCS for replication on clusters at both sites.	Configure Cluster Server (VCS) to provide high availability for the database: ■ Modify the VCS configuration on the primary site ■ Modify the VCS configuration on the secondary site See "Configuring Cluster Server to replicate the database volume using VVR" on page 147.

Once the global clusters and replication with VVR are configured, the following replication use cases are supported for it:

- Migration of the role of the primary site to the remote site
- Takeover of the primary site role by the secondary site

- Migrate the role of primary site to the secondary site
- Migrate the role of new primary site back to the original primary site
- Take over after an outage
- Resynchronize after an outage
- Update the rlink to reflect changes

See "Replication use cases for global parallel clusters" on page 167.

Setting up replication on the primary site using VVR

If you have not already done so, create a disk group to hold data volume, Storage Replicator Log (SRL), and Replicated Volume Group (RVG) on the storage on the primary site. For example, create the storage for your database.

See the *Storage Foundation Cluster File System High Availability Configuration and Upgrade Guide*.

See the *Storage Foundation for Oracle RAC Configuration and Upgrade Guide*.

See the *Storage Foundation for Sybase ASE CE Configuration and Upgrade Guide*.

Table 9-2 Tasks for setting up replication on the primary site using VVR

Task	Description
Create the Storage Replicator Log (SRL) in the disk group for the database.	See "Creating the data and SRL volumes on the primary site" on page 137.
Create the Replicated Volume Group (RVG) on the primary site.	See "Setting up the Replicated Volume Group on the primary site" on page 138.

Creating the data and SRL volumes on the primary site

Create the data volume if you do not have one already.

- The data volume on the secondary site has the same name and the same size as the data volume on the primary site.
  See "Configuring the secondary site" on page 119.
- The data volume and Storage Replicator Log (SRL) volume should exist in the same disk group.
- Mirror the data volume in the absence of hardware-based mirroring.

To create the data volume on the primary site

- In the disk group created for the database, create a data volume for the data on primary site. In the examples below, the dbdata_vol volume on the primary site is 12 GB:

```
# vxassist -g dbdatadg make dbdata_vol 12000M disk1 disk2
```

Create the SRL. The SRL is a volume in the Replicated Volume Group (RVG). The RVG also holds the data volumes for replication.

- The SRL on the secondary site has the same name and the same size as the SRL on the primary site.
- You must create SRLs on disks without other volumes.
- Mirror SRLs and in the absence of hardware-based mirroring.

In the example procedure below, dbdatadg is the disk group and dbdatavol is the data volume to be replicated.

To create the SRL volume on the primary site

1. On the primary site, determine the size of the SRL volume based on the configuration and amount of use.

   See the *Storage Foundation and High Availability Solutions Replication Administrator's Guide* for details.

2. Issue the following command:

```
# vxassist -g dbdatadg make dbdata_srl 6800M disk4 disk5
```

   Note: Assume that for the example setup that *disk4* and *disk5* are already added and are part of the same disk group. They are used in this step for mirroring and creation of the SRL.

3. If the SRL volume is not already started, start the SRL volume by starting all volumes in the disk group:

```
# vxvol -g dbdatadg startall
```

Setting up the Replicated Volume Group on the primary site

Before creating the Replicated Volume Group (RVG) on the primary site, make sure the volumes and Cluster Volume Manager (CVM) group are active and online.

To review the status of replication objects on the primary site

1. Verify the volumes you intend to include in the group are active.
2. Review the output of the `hagrp -state cvm` command to verify that the CVM group is online.
3. On each site, verify vradmin is running:

```
#  ps -ef |grep vradmin
    root  536594  598036   0 12:31:25      0  0:00 grep vradmin
```

If vradmin is not running start it:

```
# vxstart_vvr
VxVM VVR INFO V-5-2-3935 Using following ports:
heartbeat: 4145
vradmind: 8199
vxrsyncd: 8989
data: Anonymous-Ports
To change, see vrport(1M) command
#  ps -ef |grep vradmin
    root  536782       1   0 12:32:47      -  0:00 /usr/sbin/vradmind
    root 1048622  598036   0 12:32:55      0  0:00 grep vradmin
# netstat -an |grep 4145
tcp4       0      0  *.4145                 *.*          LISTEN
udp4       0      0  *.4145                 *.*
```

After reviewing the status of replication objects on the primary site, you can create the primary RVG.

The command to create the primary RVG takes the form:

```
vradmin -g disk_group createpri rvg_name data_volume srl_volume
```

where:

- disk_group is the name of the disk group containing the database
- rvg_name is the name for the RVG
- data_volume is the volume that VVR replicates
- srl_volume is the volume for the Storage Replicator Log (SRL)

To create the primary RVG

1 Determine which node is the CVM master node by entering:

```
# vxdctl -c mode
```

2 To create the dbdata_rvg RVG, you must run the following on the master node:

```
# vradmin -g dbdatadg createpri dbdata_rvg dbdata_vol dbdata_srl
```

The command creates the RVG on the primary site and adds a Data Change Map (DCM) for each data volume. In this case, a DCM exists for *dbdata_vol*.

Setting up replication on the secondary site using VVR

To create objects for replication on the secondary site, use the `vradmin` command with the `addsec` option. To set up replication on the secondary site, perform the following tasks:

Table 9-3 Tasks for setting up replication on the secondary site using VVR

Task	Description
Create a disk group to hold the data volume, Storage Replicator Log (SRL), and Replicated Volume Group (RVG) on the storage on the secondary site. You must match the names and sizes of these volumes with the names and sizes of the volumes on the primary site. For example, you must set up the data files. For example, create the disk group, volume, and mount point for the database datafiles.	See the *Storage Foundation Cluster File System High Availability Configuration and Upgrade Guide*. See the *Storage Foundation for Oracle RAC Configuration and Upgrade Guide*. See the *Storage Foundation for Sybase ASE CE Configuration and Upgrade Guide*. See "Creating the data and SRL volumes on the secondary site" on page 141.
Edit the /etc/vx/vras/.rdg file on the each site.	See "Editing the /etc/vx/vras/.rdg files" on page 141.
Use resolvable virtual IP addresses that set network RLINK connections as host names of the primary and secondary sites.	See "Setting up IP addresses for RLINKs on each cluster" on page 142.

Table 9-3 Tasks for setting up replication on the secondary site using VVR *(continued)*

Task	Description
Create the replication objects on the secondary site.	See "Setting up the disk group on secondary site for replication" on page 143.

Creating the data and SRL volumes on the secondary site

Note the following when creating volumes for the data and Storage Replicator Log (SRL):

- The sizes and names of the volumes must match the sizes and names of the corresponding volumes in the primary site before you create the disk group.
- The disk group must match the size and name of the disk group at the primary site.
- Create the data and SRL volumes on different disks in the disk group. Use the `vxdisk -g diskgroup list` command to list the disks in the disk group.
- Mirror the volumes.

To create the data and SRL volumes on the secondary site

1. In the disk group created for the database, create a data volume of same size as that in primary for data; in this case, the dbdata_vol volume on the primary site is 12 GB:

```
# vxassist -g dbdatadg make dbdata_vol 12000M disk11 disk12
```

2. Create the volume for the SRL, using the same name and size of the equivalent volume on the primary site. Create the volume on different disks from the disks for the database volume, but on the same disk group that has the data volume:

```
# vxassist -g dbdatadg make dbdata_srl 6800M disk14 disk16
```

See the *Veritas InfoScale™ Replication Administrator's Guide* for details.

Editing the /etc/vx/vras/.rdg files

Editing the /etc/vx/vras/.rdg file on the secondary site enables VVR to replicate the disk group from the primary site to the secondary site. On each node, VVR uses the /etc/vx/vras/.rdg file to check the authorization to replicate the Replicated Volume Group (RVG) on the primary site to the secondary site. The file on each node in

the secondary site must contain the primary disk group ID, and likewise, the file on each primary system must contain the secondary disk group ID (dgid).

To edit the /etc/vx/vras/.rdg files

1. On a node in the primary site, display the primary disk group ID:

```
# vxprint -l diskgroup

.....
```

2. On each node in the secondary site, edit the /etc/vx/vras/.rdg file and enter the primary disk group ID on a single line.
3. On each cluster node of the primary cluster, edit the /etc/vx/vras/.rdg file and enter the secondary disk group ID on a single line.

Setting up IP addresses for RLINKs on each cluster

Creating objects with the vradmin command requires resolvable virtual IP addresses that set network RLINK connections as host names of the primary and secondary sites.

To set up IP addresses for RLINKs on each cluster

1. Using the following command, determine whether a node is the CVM master or the slave:

```
# vxdctl -c mode
```

   You must configure RLINKs on the respective node.

2. For each Replicated Volume Group (RVG) running on each cluster, set up a virtual IP address on one of the nodes of the cluster. These IP addresses are part of the RLINK.

   The example assumes for the cluster on the primary site:

   - The public network interface is public_NIC1:2
   - The virtual IP address is 10.10.9.101
   - The net mask is 255.255.255.0

```
# ifconfig eth0:1 inet 10.10.9.101 netmask 255.255.255.0 up
```

3. Use the same commands with appropriate values for the interface, IP address, and net mask on the secondary site.

   The example assumes for the secondary site:

- The public network interface is public_NIC1:1
- virtual IP address is 10.11.9.102
- net mask is 255.255.255.0

4 Define the virtual IP addresses to correspond to a host name in the virtual cluster on the primary site and a host name in the virtual cluster on the secondary site.

Update the /etc/hosts file on all the nodes on both the primary and secondary sites.

The examples assume:

- clus1 has IP address 10.10.9.101
- clus2 has IP address 10.11.9.102

5 Use the ping command to verify the links are functional.

Setting up the disk group on secondary site for replication

Create the replication objects on the secondary site from the master node of the primary site, using the `vradmin` command.

To set up the disk group on the secondary site for replication

1 Issue the command in the following format from the cluster on the primary site:

```
# vradmin -g dg_pri addsec rvg_pri pri_host sec_host
```

where:

- dg_pri is the disk group on the primary site that VVR will replicate. For example: dbdata_vol
- rvg_pri is the Replicated Volume Group (RVG) on the primary site. For example: dbdata_rvg
- pri_host is the virtual IP address or resolvable virtual host name of the cluster on the primary site.
  For example: 10.10.9.101 or clus1
- sec_host is the virtual IP address or resolvable virtual host name of the cluster on the secondary site.
  For example: 10.11.9.102 or clus2

For example, the command to add the cluster on the primary site to the Replicated Data Set (RDS) is:

```
vradmin -g dbdatadg addsec dbdata_rvg clus1 clus2
```

or

```
vradmin -g dbdatadg addsec dbdata_rvg 10.10.9.101 10.11.9.102
```

On the secondary site, the above command performs the following tasks:

- Creates an RVG within the specified disk group using the same name as the one for the primary site
- Associates the data and Storage Replicator Log (SRL) volumes that have the same names as the ones on the primary site with the specified RVG
- Adds a data change map (DCM) for the data volume
- Creates cluster RLINKs for the primary and secondary sites with the default names; for example, the "primary" RLINK created for this example is rlk_dbdata_clus2_dbdata_rvg and the "secondary" RLINK created is rlk_dbdata_clus1_dbdata_rvg.
  If you use 10.10.9.101 and 10.11.9.102, creates cluster RLINKs for the primary and secondary sites with the default names; for example, the "primary" RLINK created for this example is rlk_10.11.9.102_dbdata_rvg and the "secondary" RLINK created is rlk_10.10.9.101_dbdata__rvg.

2 Verify the list of RVGs in the RDS by executing the following command.

```
# vradmin -g dbdatadg -l printrvg
```

For example:

```
Replicated Data Set: dbdata_rvg
Primary:
HostName: 10.180.88.187 <localhost>
RvgName: dbdata_rvg
DgName: dbdata_vol
datavol_cnt: 1
vset_cnt: 0
srl: dbdata_srl
RLinks:
name=rlk_clus2_dbdata_rvg, detached=on,
synchronous=off
Secondary:
HostName: 10.190.99.197
RvgName:dbdata_rvg
DgName: dbdatadg
datavol_cnt: 1
vset_cnt: 0
```

```
srl: dbdata_srl
RLinks:
name=rlk_clus1_dbdata_rvg, detached=on,
synchronous=off
```

Note: Once the replication is started the value of the detached flag will change the status from ON to OFF.

Starting replication of the primary site database volume to the secondary site using VVR

When you have both the primary and secondary sites set up for replication, you can start replication from the primary site to the secondary site.

Start with the default replication settings:

- Mode of replication: synchronous=off
- Latency Protection: latencyprot=off
- Storage Replicator Log (SRL) overflow protection: srlprot=autodcm
- Packet size: `packet_size=8400`
- Network protocol: `protocol=TCP`

Method of initial synchronization:

- Automatic synchronization
- Full synchronization with Storage Checkpoint

For guidelines on modifying these settings and information on choosing the method of replication for the initial synchronization:

See the *Veritas InfoScale™ Replication Administrator's Guide*

Use the `vradmin` command to start replication or the transfer of data from the primary site to the secondary site over the network. Because the cluster on the secondary site uses only one host name, the command does not require the *sec_host* argument.

To start replication using automatic synchronization

- From the primary site, use the following command to automatically synchronize the Replicated Volume Group (RVG) on the secondary site:

  ```
  vradmin -g disk_group -a startrep pri_rvg sec_host
  ```

where:

- disk_group is the disk group on the primary site that VVR will replicate
- pri_rvg is the name of the Replicated Volume Group (RVG) on the primary site
- sec_host is the virtual host name for the secondary site

For example:

```
# vradmin -g dbdatadg -a startrep dbdata_rvg clus2
```

Use the `vradmin` command with the Storage Checkpoint option to start replication using full synchronization with Storage Checkpoint.

To start replication using full synchronization with Storage Checkpoint

1 From the primary site, synchronize the RVG on the secondary site with full synchronization (using the `-c checkpoint` option):

```
vradmin -g disk_group -full -c ckpt_name syncrvg pri_rvg sec_host
```

where:

- disk_group is the disk group on the primary site that VVR will replicate
- ckpt_name is the name of the Storage Checkpoint on the primary site
- pri_rvg is the name of the RVG on the primary site
- *sec_host* is the virtual host name for the secondary site

For example:

```
# vradmin -g dbdatadg -c dbdata_ckpt syncrvg dbdata_rvg clus2
```

2 To start replication after full synchronization, enter the following command:

```
# vradmin -g dbdatadg -c dbdata_ckpt startrep dbdata_rvg clus2
```

Verify that replication is properly functioning.

To verify replication status

1 Check the status of VVR replication:

```
# vradmin -g disk_group_name repstatus rvg_name
```

2 Review the `flags` output for the status. The output may appear as `connected` and `consistent`. For example:

```
# vxprint -g dbdatadg -l rlk_clus2_dbdata_rvg
  Rlink: rlk_clus2_dbdata_rvg
  info: timeout=500 packet_size=8400 rid=0.1078
      latency_high_mark=10000 latency_low_mark=9950
      bandwidth_limit=none
  state: state=ACTIVE
      synchronous=off latencyprot=off srlprot=autodcm
  .
  .
  protocol: UDP/IP
  checkpoint: dbdata_ckpt
  flags: write enabled attached consistent connected
 asynchronous
```

Configuring Cluster Server to replicate the database volume using VVR

After configuring both clusters for global clustering and setting up the database for replication, configure Cluster Server (VCS) to provide high availability for the database. Specifically, configure VCS agents to control the cluster resources, including the replication resources.

Table 9-4 Tasks for configuring VCS to replicate the database volume using VVR

Task	Description
Modify the VCS configuration on the primary site	See "Modifying the Cluster Server (VCS) configuration on the primary site" on page 152.
Modifying the VCS configuration on the secondary site	See "Modifying the VCS configuration on the secondary site" on page 159.

The following resources must be configured or modified for replication:

Table 9-5 VCS resource modifications for replication with VVR

Resource	Modification
Log owner service group	Create a log owner service group including the RVGLogowner resources. The RVGLogowner resources are used by: ■ RLINKs for the RVG ■ RVGLogowner resource. The RVG and its associated disk group are defined as attributes for the RVGLogowner resource.
Replicated Volume Group (RVG) service group	Create an RVG group that includes the RVGShared resource replication objects. Define the RVGShared resource and CVMVolDg resource together within a parallel service group. The group is defined as parallel because it may be online at the same time on all cluster nodes. The RVG log owner service group has an online local firm dependency on the service group containing the RVG. VCS uses the following agents to control the following resources: ■ RVGLogowner agent to control the RVGLogowner resource ■ RVGShared agent to control the RVGShared resource
CVMVolDg resource	The CVMVolDg resource does not have replicated service volumes specified for the CVMVolume attribute; the volumes are contained in the RVG resource. The CVMVolume attribute for the CVMVolDg resource is empty because all volumes in the RVG are defined by the RVG attribute of the RVGShared resource. The RVG service group has an online local firm dependency on the CVM service group.
RVGSharedPri resource	Add the RVGSharedPri resource to the existing database service group. The CVMVolDg resource must be removed from the existing database service group.
Database service group	The existing database service group is a parallel group consisting of the database resource, CVMVolDg resource, and CFSMount resource (if the database resides in a cluster file system). Define the database service group as a global group by specifying the clusters on the primary and secondary sites as values for the ClusterList group attribute.

For more information on service replication resources:

See the *Cluster Server Bundled Agents Guide*

Review the following illustrations that display the changes to the VCS configuration, after setting up replication on the existing database. All of the dependencies between parent and child groups are online local firm. The CVM service group is the same in all illustrations because its definition requires no changes.

Configuration before modification for replication:

Figure 9-1 Dependencies before modification for replication of Oracle RAC

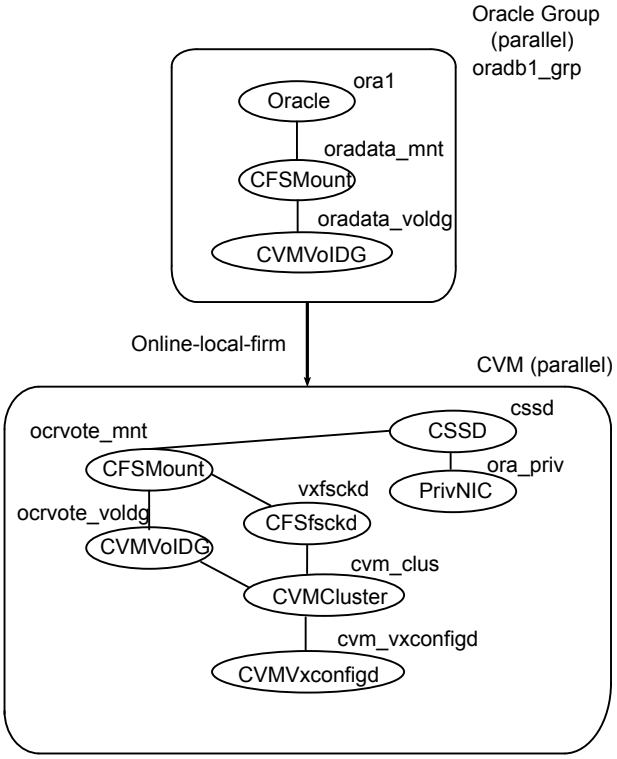
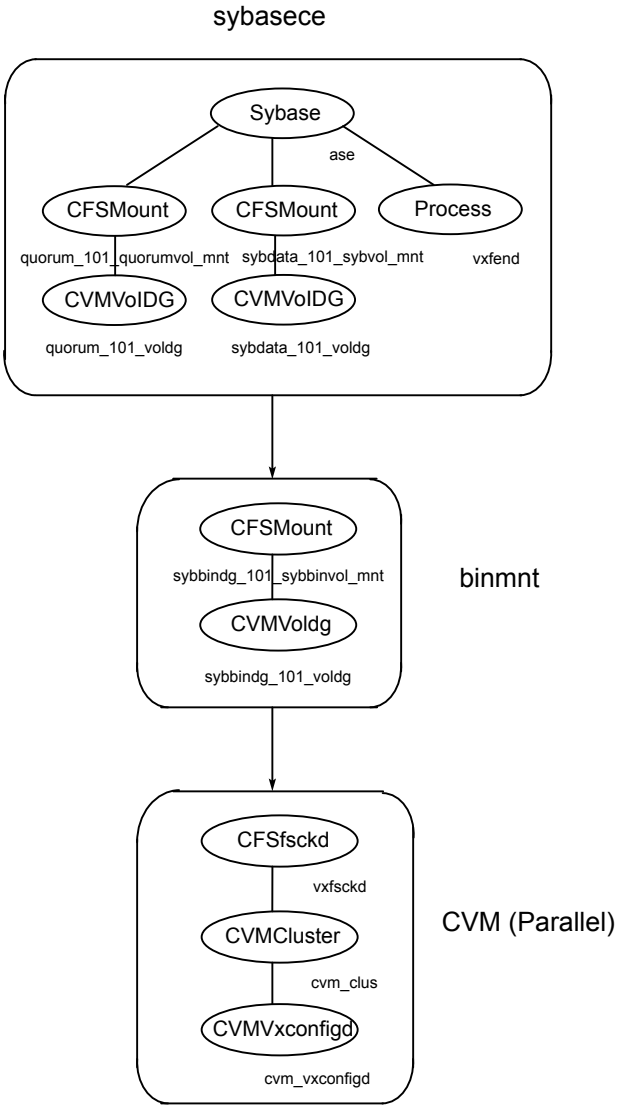


Figure 9-2 Dependencies before modification for replication of Sybase ASE CE



Configuration after modification for replication:

Figure 9-3 Dependencies after modification for replication of Oracle RAC

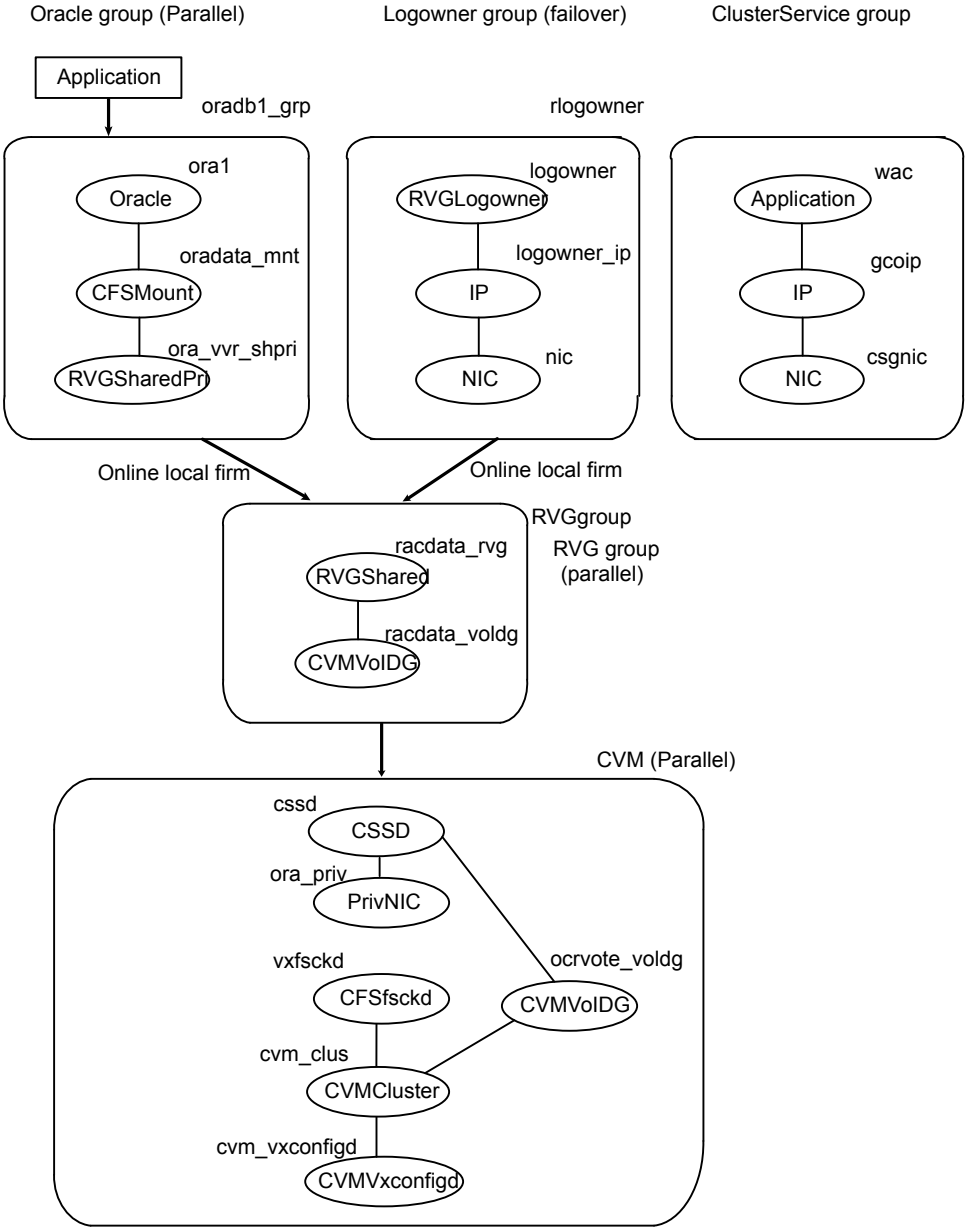
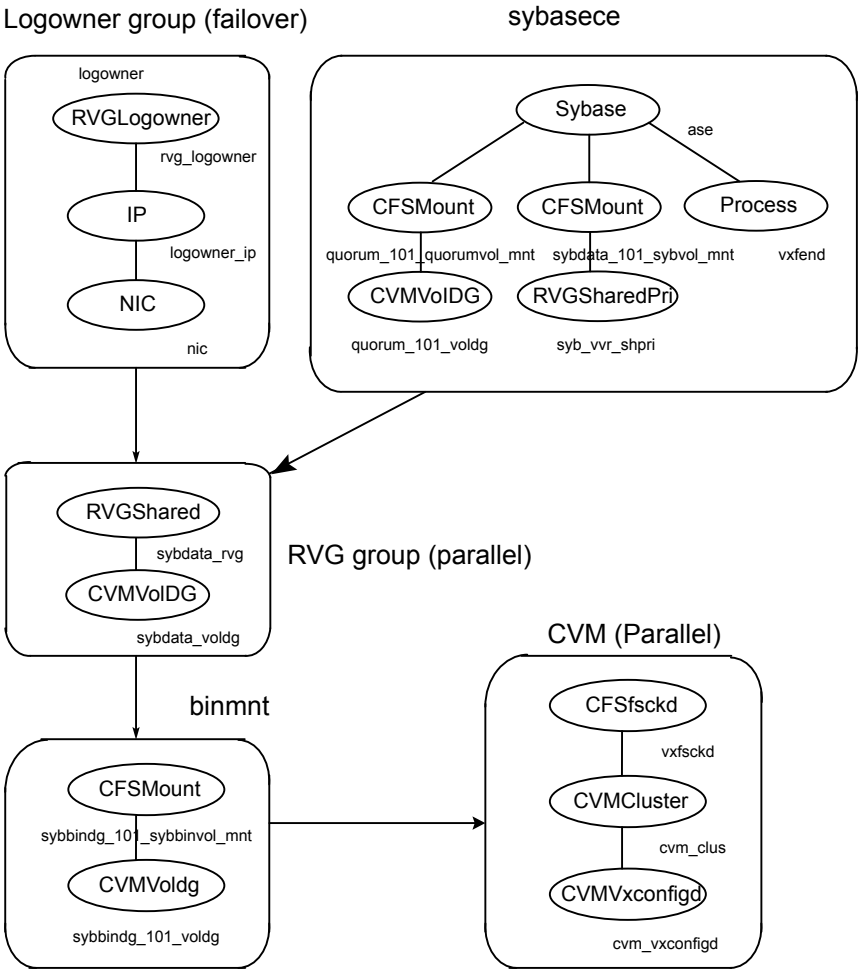


Figure 9-4 Dependencies after modification for replication of Sybase ASE CE



Modifying the Cluster Server (VCS) configuration on the primary site

The following are the tasks required to modify the existing VCS configuration on the primary site:

Table 9-6 Tasks for modifying the VCS configuration on the primary site

Task	Description
Configure two service groups: ■ A log owner group including the RVGLogowner resource. ■ A Replicated Volume Group (RVG) group including the RVGShared resource replication objects.	See "To modify VCS on the primary site" on page 154. See "Modifying the VCS configuration on the secondary site" on page 159.
Add the RVGSharedPri resource to the existing database service group and define this group as a global group by setting the ClusterList and ClusterFailOverPolicy attributes.	See "To modify VCS on the primary site" on page 154. See "Modifying the VCS configuration on the secondary site" on page 159.
Move the CVMVolDg resource from the existing database service group to the newly created RVG group.	See "To modify VCS on the primary site" on page 154. See "Modifying the VCS configuration on the secondary site" on page 159.
To view the sample main.cf files on your system:	See "To view sample configuration files for SF Oracle RAC" on page 153. See "To view sample configuration files for SF Sybase CE" on page 154.

To view sample configuration files for SF Oracle RAC

1 Change directories to the find the sample main.cfs:

```
# cd /etc/VRTSvcs/conf/sample_rac
```

2 Enter:

```
# ls *sfrac*
sfrac07_main.cf sfrac08_main.cf
```

The following files include CVM/VVR configuration examples:

- For the primary: sfrac_07_main.cf
- For the secondary: sfrac_08_main.cf

See "Sample Storage Foundation for Oracle RAC configuration files" on page 207.

To view sample configuration files for SF Sybase CE

1 Change directories to the find the sample main.cfs:

```
# cd /etc/VRTSagents/ha/conf/Sybase/
```

2 Enter:

```
# ls cvmvvr*
```

The following sample main.cf files illustrate the VCS configuration changes after setting up an existing database for replication:

See "About sample main.cf files for Storage Foundation (SF) for Sybase ASE CE" on page 230.

To modify VCS on the primary site

1 Log into one of the nodes on the primary cluster.

2 Use the following command to save the existing configuration to disk, and make the configuration read-only while you make changes:

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

3 Use the following command to make a backup copy of the main.cf file:

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

4 Use vi or another text editor to edit the main.cf file. Review the sample configuration file after your product installation.

Add a failover service group using the appropriate values for your cluster and nodes. Include the following resources:

- In a shared disk group environment, currently only the cvm master node should be assigned the logowner role.
- IP resource
- NIC resources

The following are examples of RVGLogowner service group for the different platforms.

Example for Oracle RAC:

Please see the section "CVM master node needs to assume the logowner role for VCS managed VVR resources" in the *Cluster Server Bundled Agents Reference Guide* for more info.

```
group rlogowner (
    SystemList = { sys1 = 0, sys2 = 1 }
    AutoStartList = { sys1, sys2 }
    )

    IP logowner_ip (
        Device = eth0
        Address = "10.10.9.101"
        NetMask = "255.255.255.0"
        )
    NIC nic (
        Device = eth0
        NetworkType = ether
        NetworkHosts = "10.10.8.1"
        )
RVGLogowner logowner (
     RVG = dbdata_rvg
     DiskGroup = dbdatadg
     )
requires group RVGgroup online local firm
logowner requires logowner_ip
logowner_ip requires nic
```

5 Add the RVG service group using the appropriate values for your cluster and nodes.

Example `RVGgroup` service group:

```
group RVGgroup (
     SystemList = { sys1 = 0, sys2 = 1 }
     Parallel = 1
     AutoStartList = { sys1, sys2 }
     )
RVGShared dbdata_rvg (
     RVG = dbdata_rvg
     DiskGroup = dbdatadg
     )
     CVMVolDg dbdata_voldg (
         CVMDiskGroup = dbdatadg
         CVMActivation = sw
         CVMVolume = { dbvol, dbdata_srl }
         )
requires group cvm online local firm
dbdata_rvg requires dbdata_voldg
```

6 Modify the database service group using the appropriate values for your cluster and nodes:

- Define the database service group as a global group by specifying the clusters on the primary and secondary sites as values for the ClusterList group attribute. See the bolded attribute in the example that follows.
- Add the ClusterFailOverPolicy cluster attribute. Veritas recommends using the Manual value. See the bolded attribute in the example.
- Add the RVGSharedPri resource to the group configuration.
- Remove the CVMVolDg resource, if it has been configured in your previous configuration. This resource is now part of the RVG service group.
- Specify the service group (online, local, firm) to depend on the RVG service group.
- Remove the existing dependency of the Database service group on the CVM service group. Remove the line:

  ```
  requires group CVM online local firm
  ```

- Remove the existing dependency between the CFSMount for the database and the CVMVoldg for the database. Remove the line:

  ```
  dbdata_mnt requires dbdata_voldg
  ```

  See configuration examples below.

7 Save and close the main.cf file.

8 It is advisible to modify "OnlineRetryLimit" & "OfflineWaitLimit" attribute of IP resource type to 1 on both the clusters:

```
# hatype -modify IP  OnlineRetryLimit  1

# hatype -modify IP  OfflineWaitLimit  1
```

9 Use the following command to verify the syntax of the /etc/VRTSvcs/conf/config/main.cf file:

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

10 Stop and restart VCS.

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

Wait for port h to stop on all nodes, and then restart VCS with the new configuration on all primary nodes:

```
# hastart
```

Storage Foundation Cluster File System High Availability (SFCFS HA) example of a database service group configured for replication:

```
group database_grp (
    SystemList = { sys1 = 0, sys2 = 1 }
    ClusterList = { clus1 = 0, clus2 = 1 }
    Parallel = 1
    ClusterFailOverPolicy = Manual
    Authority = 1
    AutoStartList = { sys1,sys2 }
    OnlineRetryLimit = 3
    TriggerResStateChange = 1
    OnlineRetryInterval = 120
    )

    CFSMount dbdata_mnt (
        MountPoint = "/dbdata"
        BlockDevice = "/dev/vx/dsk/dbdatadg/dbdata_vol"
        )

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

    RVGSharedPri dbdata_vvr_shpri (
       RvgResourceName = dbdata_rvg
       OnlineRetryLimit = 0
       )
```

```
requires group RVGgroup online local firm
oradata_mnt requires dbdata_vvr_shpri
```

Storage Foundation (SF) for Oracle RAC example of a database service group configured for replication:

```
group database_grp (
    SystemList = { sys1 = 0, sys2 = 1 }
    ClusterList = { clus1 = 0, clus2 = 1 }
    Parallel = 1
    ClusterFailOverPolicy = Manual
    Authority = 1
    AutoStartList = { sys1,sys2 }
    )

    CFSMount oradata_mnt (
        MountPoint = "/oradata"
        BlockDevice = "/dev/vx/dsk/oradatadg/racdbdata_vol"
        )

    RVGSharedPri ora_vvr_shpri (
       RvgResourceName = racdata_rvg
       OnlineRetryLimit = 0
       )

Oracle rac_db (
    Sid @sys1 = vrts1
    Sid @sys2 = vrts2
    Owner = Oracle
    Home = "/oracle/orahome"
    StartUpOpt = SRVCTLSTART
    ShutDownOpt = SRVCTLSTOP
    )
requires group RVGgroup online local firm
oradata_mnt requires ora_vvr_shpri
rac_db requires oradata_mnt
```

Storage Foundation (SF) for Sybase ASE CE example of a database service group configured for replication:

```
group database_group (
    SystemList = { sys1 = 0, sys2 = 1 }
    ClusterList = { clus1 = 0, clus2 = 1 }
    Parallel = 1
```

```
    ClusterFailOverPolicy = Manual
    Authority = 1
    AutoStartList = { sys1,sys2 }
    OnlineRetryLimit = 3
    TriggerResStateChange = 1
    OnlineRetryInterval = 120
    )

    CFSMount sybdata_101_syb_vol_mnt (
        MountPoint = "/sybdata"
        BlockDevice = "/dev/vx/dsk/sybdata_101/syb_vol"
        )

    RVGSharedPri syb_vvr_shpri (
       RvgResourceName = sybdata_rvg
       OnlineRetryLimit = 0
       )

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

Syabase ase (
    Sid @sys1 = ase1
    Sid @sys1 = ase2
    Owner = sybase
    Home = "/sybase"
    Version = 15
    SA = sa
    Quorum_dev = "/quorum/q.dat"
    )
requires group RVGgroup online local firm
sybdata_101_syb_vol_mnt requires syb_vvr_shpri
ase requires vxfend
ase requires sybdata_101_syb_vol_mnt
ase requires quorum_101_quorumvol_mnt
quorum_101_quorumvol_mnt requires quorum_101_voldg
```

Modifying the VCS configuration on the secondary site

The following are highlights of the procedure to modify the existing VCS configuration on the secondary site:

- Add the log owner and Replicated Volume Group (RVG) service groups.
- Add a service group to manage the database and the supporting resources.
- Define the replication objects and agents, such that the cluster at the secondary site can function as a companion to the primary cluster.

The following steps are similar to those performed on the primary site.

To modify VCS on the secondary site

1. Log into one of the nodes on the secondary site as root.
2. Use the following command to save the existing configuration to disk, and make the configuration read-only while making changes:

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

3. Use the following command to make a backup copy of the main.cf file:

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

4. Use vi or another text editor to edit the main.cf file. Edit the CVM group on the secondary site.

   Review the sample configuration file after the VCS installation to see the CVM configuration.

   See "To view sample configuration files for SF Oracle RAC" on page 153.

   See "To view sample configuration files for SF Sybase CE" on page 154.

   In our example, the secondary site has clus2 consisting of the nodes sys3 and sys4. To modify the CVM service group on the secondary site, use the CVM group on the primary site as your guide.

5. Add a failover service group using the appropriate values for your cluster and nodes. Include the following resources:

   - RVGLogowner resource. The node on which the group is online functions as the log owner (node connected to the second cluster for the purpose of replicating data).
   - IP resource
   - NIC resources

   Example `RVGLogowner` service group:

```
group rlogowner (
    SystemList = { sys3 = 0, sys4 = 1 }
```

```
    AutoStartList = { sys3, sys4 }
    )

    IP logowner_ip (
       Device = eth0
       Address = "10.11.9.102"
       NetMask = "255.255.255.0"
       )
NIC nic (
       Device = eth0
       NetworkHosts = { "10.10.8.1" }
       NetworkType = ether
       )
RVGLogowner logowner (
       RVG = dbdata_rvg
       DiskGroup = dbdatadg
       )
requires group RVGgroup online local firm
logowner requires logowner_ip
logowner_ip requires nic
```

6 Add the RVG service group using the appropriate values for your cluster and nodes.

The following is an example `RVGgroup` service group:

```
group RVGgroup (
    SystemList = { sys3 = 0, sys4 = 1 }
    Parallel = 1
    AutoStartList = { sys3, sys4 }
    )

RVGShared dbdata_rvg (
    RVG = dbdata_rvg
    DiskGroup = dbdatadg
    )

    CVMVolDg dbdata_voldg (
        CVMDiskGroup = dbdatadg
        CVMActivation = sw
        )

requires group cvm online local firm
dbdata_rvg requires dbdata_voldg
```

7 It is advisible to modify "OnlineRetryLimit" & "OfflineWaitLimit" attribute of IP resource type to 1 on both the clusters:

```
# hatype -modify IP  OnlineRetryLimit  1

# hatype -modify IP  OfflineWaitLimit  1
```

8 Add an database service group. Use the database service group on the primary site as a model for the database service group on the secondary site.

- Define the database service group as a global group by specifying the clusters on the primary and secondary sites as values for the ClusterList group attribute.
- Assign this global group the same name as the group on the primary site. For example, database_grp.
- Include the ClusterList and ClusterFailOverPolicy cluster attributes. Veritas recommends using the Manual value.
- Add the RVGSharedPri resource to the group configuration.
- Remove the CVMVolDg resource, if it has been configured in your previous configuration. This resource is now part of the RVG service group.
- Specify the service group to depend (online, local, firm) on the RVG service group.

See configuration examples below.

9 Save and close the `main.cf` file.

10 Use the following command to verify the syntax of the `/etc/VRTSvcs/conf/config/main.cf` file:

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

11 Stop and restart VCS.

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

Wait for port h to stop on all nodes, and then restart VCS with the new configuration on all primary nodes one at a time.

```
# hastart
```

12 Verify that VCS brings all resources online. On one node, enter the following command:

```
# hagrp -display
```

The database, RVG, and CVM groups are online on both nodes of the primary site. The RVGLogOwner and ClusterService groups are online on one node of the cluster. If either the RVG group or the RVGLogOwner group is partially online, manually bring the groups online using the `hagrp -online` command. This information applies to the secondary site, except for the database group which must be offline.

13 Verify the service groups and their resources that are brought online. On one node, enter the following command:

```
# hagrp -display
```

The database service group is offline on the secondary site, but the ClusterService, CVM, RVG log owner, and RVG groups are online.

This completes the setup for a global cluster using VVR for replication. Veritas recommends testing a global cluster before putting it into production.

Example of the Oracle RAC database group on the secondary site:

```
group database_grp (
    SystemList = { sys3 = 0, sys3 = 1 }
    ClusterList = { clus2 = 0, clus1 = 1 }
    Parallel = 1
    OnlineRetryInterval = 300
    ClusterFailOverPolicy = Manual
    Authority = 1
    AutoStartList = { sys3, sys4 }
    )

    RVGSharedPri dbdata_vvr_shpri (
        RvgResourceName = rdbdata_rvg
        OnlineRetryLimit = 0
        )

    CFSMount dbdata_mnt (
        MountPoint = "/dbdata"
        BlockDevice = "/dev/vx/dsk/dbdatadg/dbdata_vol"
        Critical = 0
        )
```

```
   RVGSharedPri dbdata_vvr_shpri (
        RvgResourceName = dbdata_rvg
        OnlineRetryLimit = 0
        )

Oracle rac_db (
     Sid @sys3 = vrts1
     Sid @sys4 = vrts2
     Owner = Oracle
     Home = "/oracle/orahome"
     Pfile @sys3 = "/oracle/orahome/dbs/initvrts1.ora"
     Pfile @sys4 = "/oracle/orahome/dbs/initvrts2.ora"
     StartUpOpt = SRVCTLSTART
     ShutDownOpt = SRVCTLSTOP
     )

requires group RVGgroup online local firm
dbdata_mnt requires dbdata_vvr_shpri
rac_db requires dbdata_mnt

RVGSharedPri dbdata_vvr_shpri (
        RvgResourceName = dbdata_rvg
        OnlineRetryLimit = 0
        )

requires group RVGgroup online local firm
dbdata_mnt requires dbdata_vvr_shpri
```

Example of the Sybase ASE CE database group on the secondary site:

```
.
group sybase (
    SystemList = { sys3 = 0, sys4 = 1 }
    ClusterList = { clus2 = 0, clus1 = 1 }
    Parallel = 1
    OnlineRetryInterval = 300
    ClusterFailOverPolicy = Manual
    Authority = 1
  # AutoStart = 0 here so faulting will not happen
    AutoStartList = { sys3, sys4 }
    )

    CFSMount dbdata_mnt (
```

```
        MountPoint = "/dbdata"
        BlockDevice = "/dev/vx/dsk/dbdatadg/dbdata_vol"
        )

RVGSharedPri dbdata_vvr_shpri (
        RvgResourceName = dbdata_rvg
        OnlineRetryLimit = 0
        )

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

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

Sybase ase (
     Sid @sys3 = ase1
     Sid @sys4 = ase2
     Owner = sybase
     Home = "/sybase"
     Version = 15
     SA = sa
     Quorum_dev = "/quorum/q.dat"
     )

requires group RVGgroup online local firm
dbdata_mnt requires dbdata_vvr_shpri
ase requires vxfend
ase requires dbdata_mnt
ase requires quorum_101_quorumvol_mnt
quorum_101_quorumvol_mnt requires quorum_101_voldg
```

Configuring the Sybase ASE CE cluster on the secondary site

Before bringing Sybase ASE CE online at the secondary site, you must configure the Sybase ASE CE cluster on the secondary site.

To configure Sybase ASE CE cluster on the secondary site

1. Modify the the Sybase ASE CE configuration files to match the secondary site environment.
2. When replication is complete, suspend replication.
3. Mount the corresponding file system at the secondary site.
4. Install and configure the Sybase ASE CE binaries:

   See the *Storage Foundation for Sybase ASE CE Configuration and Upgrade Guide*

   Note: The quorum device must be on non-replicated shared storage.

5. Copy the the following from the primary to the secondary site:
   - interfaces
   - configuration file (ase.cfg)
   - inp files (QIase*****.inp)
   - RUN files

   After copying, edit them to reflect the secondary site environment.
6. Start the database once manually before proceeding to bring it under VCS control. You must rebuild the quorum as well as copy the cluster ID of the datafile to the quorum on secondary site. This is needed only for the first time you bring the database online on the secondary site.

   Use the following command:

   ```
   # su - sybase; source $SYBASE/SYBASE.sh; /$SYBASE/ASE-15_0/bin/ \
   dataserver --instance=dbdata1 -Q /qrmmnt/qrm -F \
   /sybase_home/QIdbdata19442.inp \
   --buildquorum=force --create-cluster-id=quorum
   ```

   If you do not perform this task, you will receive the following error on the secondary site:

   ```
   Unique cluster id on quorum device does not match master device.
   You may be using the wrong master device. If this is the correct
   master, pass 'create-cluster-id' on the command line to pair the
   devices.
   ```

   The buildquorum is required to reflect the secondary site settings for the Sybase database.

7 Configure the VCS service groups for the database.

Verify that all VCS service groups are online.

For sample configurations:

See "About sample main.cf files for Storage Foundation (SF) for Sybase ASE CE" on page 230.

Note: The database service group will be online only at one site at a time. Hence if it is online on the primary site, it will be offline on secondary site and vice versa.

8 Stop the Sybase ASE service group on the secondary site, unmount the file system, and establish replication.

Note: Make sure the private interconnects which are used for the Sybase ASE CE cluster on the secondary site are also plumbed and pingable amongst the nodes.

Replication use cases for global parallel clusters

For information on the VCS commands for global clusters:

See the *Cluster Server Administrator's Guide*.

If you have two clusters configured to use VVR for replication, the following replication use cases are supported:

Table 9-7 Replication use cases for global parallel clusters

Management option	Description
Migration of the role of the primary site to the remote site	Migration is a planned transfer of the role of primary replication host from one cluster to a remote cluster. This transfer enables the application on the remote cluster to actively use the replicated data. The former primary cluster becomes free for maintenance or other activity.
Takeover of the primary site role by the secondary site	Takeover occurs when an unplanned event (such as a disaster) causes a failure, making it necessary for the applications using the replicated data to be brought online on the remote cluster.
Migrate the role of primary site to the secondary site	See "To migrate the role of primary site to the remote site" on page 168.

Table 9-7 Replication use cases for global parallel clusters *(continued)*

Management option	Description
Migrate the role of new primary site back to the original primary site	See "To migrate the role of new primary site back to the original primary site" on page 169.
Take over after an outage	See "To take over after an outage" on page 171.
Resynchronize after an outage	See "To resynchronize after an outage" on page 172.
Update the rlink	See "To update the rlink" on page 173.

After configuring the replication objects within VCS, you can use VCS commands to migrate the role of the cluster on the primary site to the remote cluster. In the procedure below, VCS takes the replicated database service group, *database_grp*, offline on the primary site and brings it online on the secondary site; the secondary site now assumes the role of the primary site.

Note: The `hagrp -switch` command cannot migrate a parallel group within a cluster or between clusters in a global cluster environment.

To migrate the role of primary site to the remote site

1. From the primary site, use the following command to take the database service group offline on all nodes.

   ```
   # hagrp -offline database_grp -any
   ```

   Wait for VCS to take all database service groups offline on the primary site.

2. Verify that the RLINK between the primary and secondary is up to date. Use the `vxrlink -g` command with the status option and specify the RLINK for the primary cluster. You can use the command from any node on the primary cluster.

   For example:

   ```
   # vxrlink -g data_disk_group status rlk_clus2_dbdata_rvg
   ```

   Where rlk_clus1_dbdata_rvg is the RLINK.

3. On the secondary site, which is now the new primary site, bring the database service group online on all nodes:

   ```
   # hagrp -online database_grp -any
   ```

After migrating the role of the primary site to the secondary site, you can use VCS commands to migrate the role of the cluster on the new primary site to the original primary site. In the procedure below, VCS takes the replicated database service group, database_grp, offline on the new primary (former secondary) site and brings it online on the original primary site; the original primary site now resumes the role of the primary site.

Note: The `hagrp -switch` command cannot migrate a parallel group within a cluster or between clusters in a global cluster environment.

To migrate the role of new primary site back to the original primary site

1. Make sure that all database resources are online, and switch back the group database_grp to the original primary site.

   Issue the following command on the remote site:

   ```
   # hagrp -offline database_grp -any
   ```

2. Verify that the RLINK between the primary and secondary is up to date. Use the `vxrlink -g` command with the status option and specify the RLINK for the primary cluster. You can use the command from any node on the current primary cluster.

   For example:

   ```
   # vxrlink -g data_disk_group status rlk_clus1_dbdata_rvg
   ```

   Where rlk_clus1_dbdata_rvg is the RLINK.

3. Make sure that database_grp is offline on the new primary site. Then, execute the following command on the original primary site to bring the database_grp online:

   ```
   # hagrp -online database_grp -any
   ```

Takeover occurs when the remote cluster on the secondary site starts the application that uses replicated data. This situation may occur if the secondary site perceives the primary site as dead, or when the primary site becomes inaccessible (perhaps for a known reason). For a detailed description of concepts of taking over the primary role:

See the *Veritas InfoScale™ Replication Administrator's Guide*.

Before enabling the secondary site to take over the primary role, the administrator on the secondary site must "declare" the type of failure at the remote (primary, in

this case) site and designate the failure type using one of the options for the `haclus` command.

Takeover options are:

Table 9-8 Takeover options on global parallel clusters

Takeover option	Description
Disaster	When the cluster on the primary site is inaccessible and appears dead, the administrator declares the failure type as "disaster." For example, fire may destroy a data center, including the primary site and all data in the volumes. After making this declaration, the administrator can bring the service group online on the secondary site, which now has the role as "primary" site.
Outage	When the administrator of a secondary site knows the primary site is inaccessible for a known reason, such as a temporary power outage, the administrator may declare the failure as an "outage." Typically, an administrator expects the primary site to return to its original state. After the declaration for an outage occurs, the RVGSharedPri agent enables DCM logging while the secondary site maintains the primary replication role. After the original primary site becomes alive and returns to its original state, DCM logging makes it possible to use fast fail back resynchronization when data is resynchronized to the original cluster. Before attempting to resynchronize the data using the fast fail back option from the current primary site to the original primary site, take the precaution at the original primary site of making a snapshot of the original data. This action provides a valid copy of data at the original primary site for use in the case the current primary site fails before the resynchronization is complete.
Disconnect	When both clusters are functioning properly and the heartbeat link between the clusters fails, a split-brain condition exists. In this case, the administrator can declare the failure as "disconnect," which means no attempt will occur to take over the role of the primary site at the secondary site. This declaration is merely advisory, generating a message in the VCS log indicating the failure results from a network outage rather than a server outage.

Table 9-8 Takeover options on global parallel clusters *(continued)*

Takeover option	Description
Replica	In the rare case where the current primary site becomes inaccessible while data is resynchronized from that site to the original primary site using the fast fail back method, the administrator at the original primary site may resort to using a data snapshot (if it exists) taken before the start of the fast fail back operation. In this case, the failure type is designated as "replica".

The examples illustrate the steps required for an outage takeover and resynchronization.

To take over after an outage

1 From any node of the secondary site, issue the `haclus` command:

```
# haclus -declare outage -clus clus1
```

2 After declaring the state of the remote cluster, bring the *database_grp* service group online on the secondary site. For example:

```
# hagrp -online -force database_grp -any
```

To resynchronize after an outage

1. On the original primary site, create a snapshot of the Replicated Volume Group (RVG) before resynchronizing it in case the current primary site fails during the resynchronization. Assuming the disk group is data_disk_group and the RVG is *dbdata1_rvg*, type:

   ```
   # vxrvg -g data_disk_group -F snapshot dbdata_rvg1
   ```

   See the *Veritas InfoScale™ Replication Administrator's Guide* for details on RVG snapshots.

2. Resynchronize the RVG. From any node of the current primary site, issue the `hares` command and the `-action` option with the fbsync action token to resynchronize the RVGSharedPri resource. For example:

   ```
   # hares -action dbdata_vvr_shpri fbsync -sys sys3

   # vxdctl -c mode
   ```

3. Perform one of the following commands, depending on whether the resynchronization of data from the current primary site to the original primary site is successful:

   - If the resynchronization of data is successful, use the `vxrvg` command with the `snapback` option to reattach the snapshot volumes on the original primary site to the original volumes in the specified RVG:

     ```
     # vxrvg -g data_disk_group snapback dbdata_rvg1
     ```

   - A failed attempt at the resynchronization of data (for example, a disaster hits the primary RVG when resynchronization is in progress) could generate inconsistent data.
     You can restore the contents of the RVG data volumes from the snapshot taken in step 1:

     ```
     # vxrvg -g data_disk_group snaprestore dbdata_rvg1
     ```

If the rlink is not up to date, use the `hares -action` command with the `resync` action token to synchronize the RVG.

To update the rlink

- The following command example is issued on any node (`sys1`, in this case) in the primary cluster, specifying the RVGSharedPri resource, `dbdata_vvr_shpri`:

```
# hares -action dbdata_vvr_shpri resync -sys sys1
```

Section 5

Configuring disaster recovery in cloud environments

- Chapter 10. Replication scenarios
- Chapter 11. Replicating in AWS cloud environment
- Chapter 12. Replicating in Azure cloud environment
- Chapter 13. Replicating across clouds

Chapter 10

Replication scenarios

This chapter includes the following topics:

- About data replication in cloud environments
- About the supported replication scenarios

About data replication in cloud environments

Veritas storage sharing and replication technologies in tandem with cloud services offer scalable, cost-effective disaster recovery options for your business. With Veritas volume replication and Flexible Storage Sharing (FSS) technologies, you can leverage the cloud as a DR site and replicate application data to or within the cloud without incurring the infrastructural costs needed to maintain a second physical site.

You can replicate data across Availability Zones and regions for both standalone and clustered systems. You can form clusters across Availability Zones within a region using FSS, a technology that enables seamless cluster-wide sharing of local storage over the network.

In the event of failures—node, storage, or Availability Zone— data will always be available.

Managing AWS resources

Automated IP configuration services by the Veritas AWSIP agent enables you to easily configure and manage your AWS IP resources—private IP (PIP), elastic IP (EIP), and overlay IP (OIP)— in the cloud.

The agent performs the following functions:

- Associates and disassociates the private IP address from the Elastic Network Interface (ENI)

- Associates public IP address and configures the private IP along with the elastic IP
- Manages the route table for overlay IP address for fail over across subnets

About the supported replication scenarios

In a cloud environment, you can use the Veritas storage sharing and replication technologies to replicate data in any of the following scenarios:

Replication from on-premise to cloud

In this scenario, replication is set between an on-premises data center to an on-cloud data center. The on-premise data center can be configured as a primary site and the on-cloud data center as a secondary site.

Replication within a region

In this scenario, replication is set between two on-cloud data centers that are located in different or same Availability Zones/VNETs (terminology varies with the vendor), within a same region. In the event of an Availability Zone/VNET failure, VVR/CVR replicates data from one Availability Zone/VNET to the second Availability Zone/VNET.

Replication across regions

In this scenario, replication is set between two on-cloud data centers that are located in two different regions. You can configure a primary site in one region and a secondary site in another region. With CVR, the cluster can span across Availability Zones, protecting your data from Availability Zone failures.

Replication across cloud

In this scenario, replication is set between Azure and AWS cloud. You can configure one of the cloud as a primary site and the other as a disaster recovery site.

Chapter 11

Replicating in AWS cloud environment

This chapter includes the following topics:

- Replication from on-premise to AWS cloud
- Replication across Availability Zones within the same AWS region
- Replication across AWS regions
- Replication across multiple AWS Availability Zones and regions (campus cluster)

Replication from on-premise to AWS cloud

In this scenario, data is replicated from on premise systems to cloud.

Pre-requisites for replication

You must meet the following requirements before setting up replication from on-premise to cloud. This configuration assumes a clustered environment that uses CVR for replication.

- Open ports for the communication between on premise and AWS cloud clusters in those subnets where the primary and secondary clusters are configured.
- The virtual private IP addresses must be plumbed on both master nodes.
- The virtual private IP addresses must be configured within the subnet.

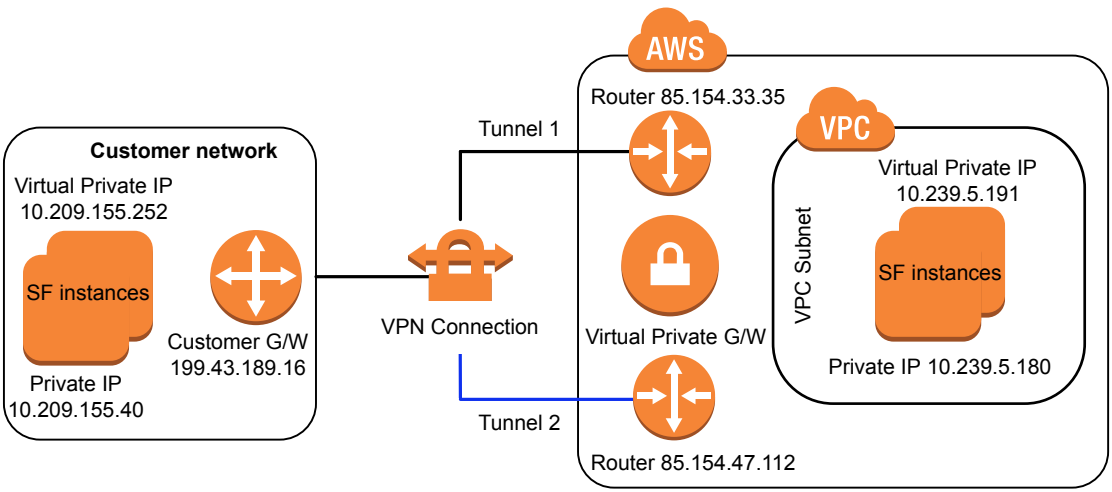
Sample configuration

Figure 11-1 illustrates the configuration.

- The primary site is configured on premise.

- The secondary site is configured on the cloud.
- Private IP addresses used for replication in standalone environments OR Virtual private IP addresses used for replication in clustered environments.
- A customer gateway on premise and a virtual private gateway on the cloud
- 2 VPN tunnels connecting the customer gateway and the virtual private gateway. This provides increased availability for the Amazon VPC service. If there's a device failure within AWS, the VPN connection automatically fails over to the second tunnel to provide uninterrupted access.
- 2 routers connecting the customer and cloud networks.

Figure 11-1 Replication from on-premise to cloud



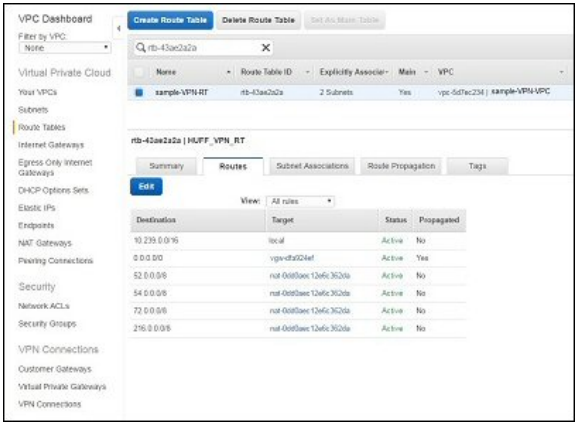
Setting up replication from on-premise to cloud environments

Perform the steps in the following procedure to set up replication from on-premise to cloud environments.

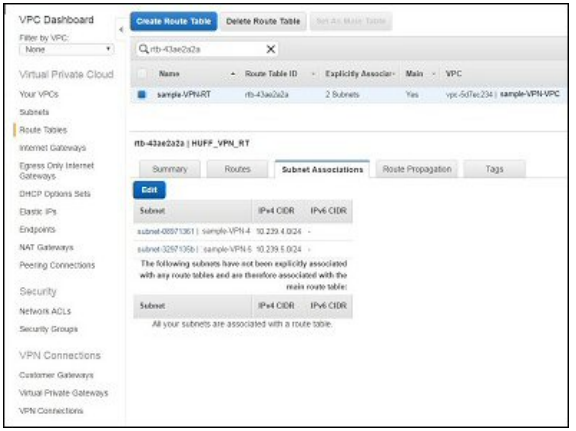
To set up replication from on-premise to cloud environments

1. Create a VPC with a valid CIDR block, for example 10.239.0.0/16.
2. Add a subnet in the Availability Zone.
3. Create the EC2 instance.
4. Configure the virtual private gateway and attach it to the VPC.
5. Configure the customer gateway.

6 Create route table entries.

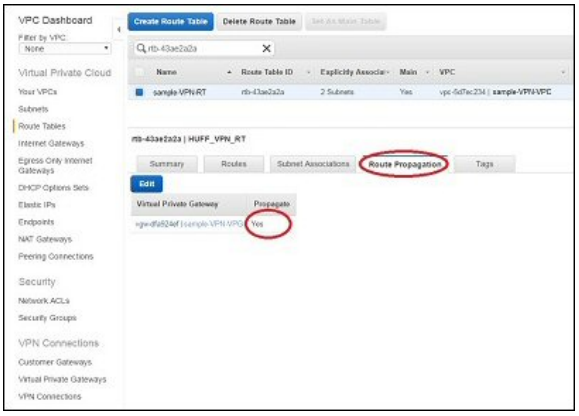


7 Associate the subnet with the route table.



8 Enable route propagation to automatically propagate the routes to the table.

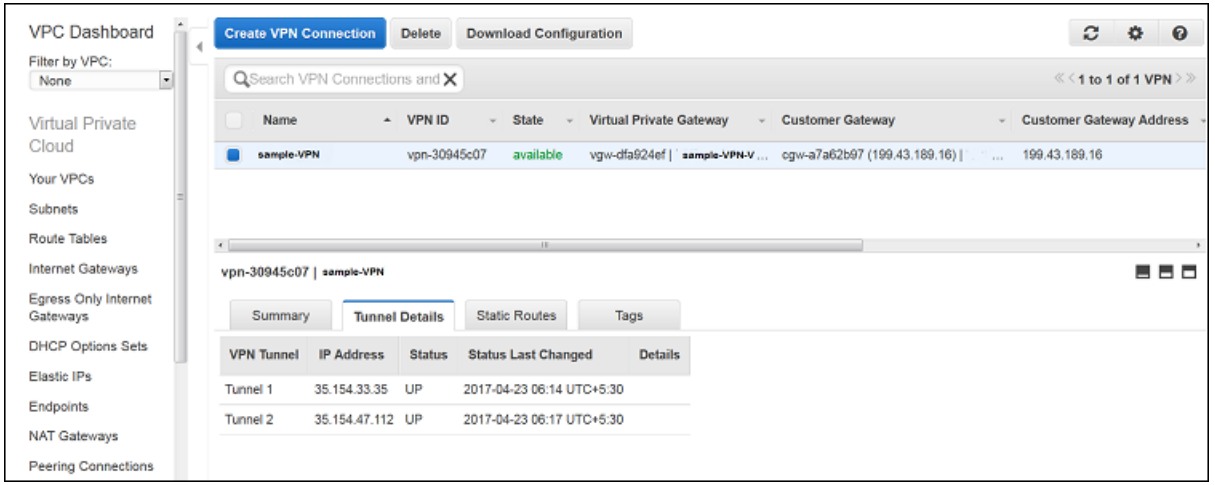
On the **Route Propagation** tab in the details pane, choose **Edit**, select the virtual private gateway that you created.



9 Create a VPN connection.

10 Download the VPN configuration file.

11 Create a VPN tunnel between the customer network and the cloud network.



12 Set up replication between the on-premise and cloud instances.

For instructions, see the chapter *Setting up replication* in the *Veritas InfoScale Replication Administrator's Guide*.

13 Verify the status of replication.

```
# vradmin -g dg_name repstatus rvg_name
```

Ensure that the replication status shows:

```
Replication status: replicating (connected)
```

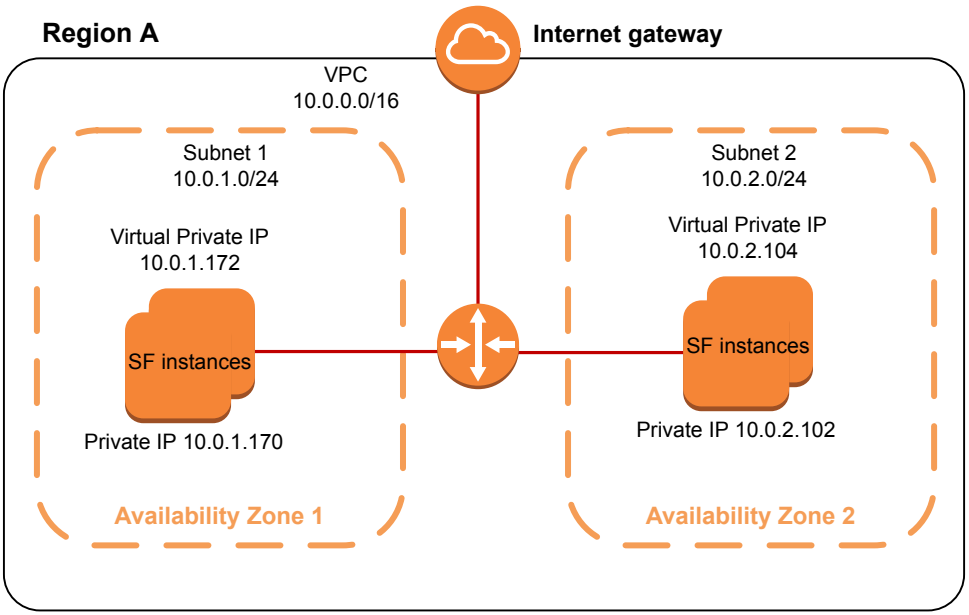
Replication across Availability Zones within the same AWS region

In this scenario, replication is set up across Availability Zones within the same region.

Figure 11-2 illustrates the configuration.

- A VPC with a valid CIDR block (for example, 10.0.0.0/16) that spans two Availability Zones (AZs) within the same region.
- The primary instance is configured in AZ1 and the secondary instance is configured in AZ2.
- Veritas InfoScale instances in the primary site and secondary site subnets.
- An Internet gateway to allow access to the Internet.
- Private IP addresses used for replication in standalone environments OR Virtual private IP addresses used for replication in clustered environments.

Figure 11-2 Replication across Availability Zones



Setting up replication across AZs within the same region

Perform the steps in the following procedure to set up replication across AZs within the same region.

To set up replication in cloud environments

1. Create a VPC with a valid CIDR block, for example, 10.0.0.0/16.
2. Create the internet gateway and attach it to the VPC.
3. Modify the VPC route table such that the two instances across availability zones can communicate with each other using private IP addresses.
4. Create two subnets—one subnet for the primary site in AZ1 and the second subnet for the secondary site in AZ2 with valid CIDR range, for example 10.0.1.0/24 and 10.0.2.0/24 respectively.
5. Launch the EC2 instances in the primary and secondary subnets. Install Veritas InfoScale on the instances.

Note: For standalone environments, private IP addresses are used for replication.

For clustered environments, virtual private IP addresses are used for replication.

---

6 Set up the AWSIP agent resource to manage the virtual private IP address (VPIP) configuration.

For detailed instructions, see the *Cluster Server Administrator's Guide*.

7 Verify connectivity between the virtual private IP addresses of the instances.

```
# ping PIP
```

```
# ping VPIP
```

8 Set up replication between the instances using private IP address or virtual private IP address.

For instructions, see the chapter *Setting up replication* in the *Veritas InfoScale Replication Administrator's Guide*.

9 Verify the status of replication.

```
# vradmin -g dg_name repstatus rvg_name
```

Ensure that the status shows:

```
Replication status: replicating (connected)
```

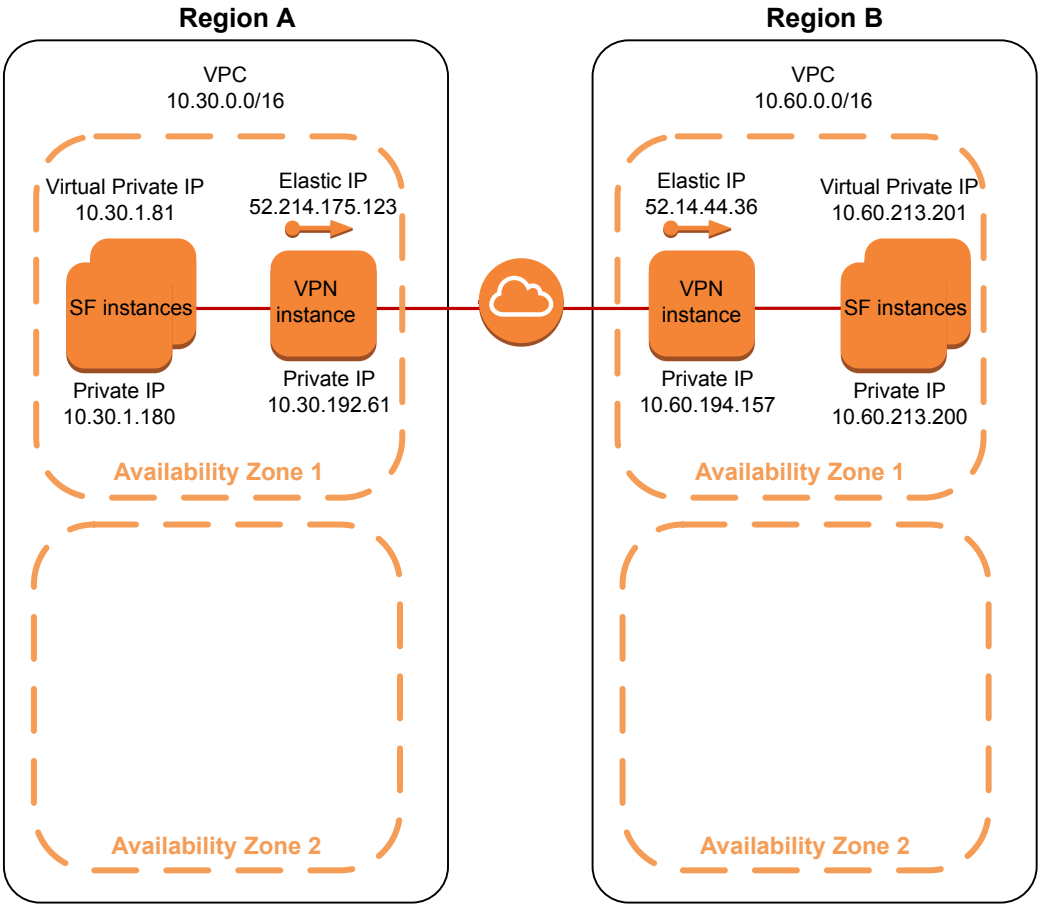
Replication across AWS regions

In this scenario, replication is set up across Availability Zones configured in different regions. The configuration uses software VPN `Openswan` to connect the VPCs across different regions.

Figure 11-3 illustrates the configuration for replication across regions.

- Two VPCs with valid CIDR blocks (for example, 10.30.0.0/16 and 10.60.0.0/16 respectively) that are located in two different regions.
- The primary instance belongs to AZ1 of region A and the secondary instance belongs to AZ1 of region B.
- Veritas InfoScale instances in each AZ.
- A VPN tunnel is established between two VPCs in two different regions using software VPN. Here, we have used OpenSwan software VPN. This is a secure IPSec tunnel.
- Elastic IP addresses (EIP) to connect the two VPN instances
- Private IP addresses used for replication in standalone environments OR Virtual private IP addresses used for replication in clustered environments.

Figure 11-3 Replication across regions



Setting up replication across regions

Perform the steps in the following procedure to set up replication across regions.

To set up replication across regions

1. Create two VPCs with valid CIDR blocks in different regions, for example, 10.30.0.0/16 and 10.60.0.0/16 respectively.
2. Create the primary site EC2 instance.
3. Create the primary site VPN instance, which belongs to the same VPC as that of the primary EC2 instance.

4. Modify the route table on the primary site. Ensure that the route table entry directs the secondary site traffic through the primary site VPN instance.
5. Create the secondary site EC2 instance.
6. Create the secondary site VPN instance, which belongs to the same VPC as that of the secondary EC2 instance.
7. Modify the route table on the secondary site. Ensure that the route table entry directs the primary site traffic through the secondary site VPN instance.
8. Set up connectivity across regions using software VPN. The sample configuration uses Openswan.

   Perform the following steps:

   - Install the Openswan packages on the primary and secondary VPN instances.
   - Configure the `/etc/ipsec.conf` and `/etc/ipsec.secrets` files.

     Note: The `/etc/ipsec.conf` file contains information about the private IP address of the VPN instance, the subnet range of the left subnet, elastic IP address of the destination VPN, the subnet range of the destination right subnet.

     The `/etc/ipsec.secrets` file contains the secret key. This key must be the same on both VPN sites.

   - Restart the IPSec service.

     ```
     # service ipsec restart
     ```

   - Add the IPSec connection.

     ```
     # ipsec auto --add vpc2vpcConnection
     # ipsec auto --up vpc2vpcConnection
     ```

   - Enable IPSec forwarding.

     ```
     # sysctl -w net.ipv4.ip_forward=1
     ```

9 Set up replication between the instances using the private IP address or virtual private IP address.

For instructions, see the chapter *Setting up replication* in the *Veritas InfoScale Replication Administrator's Guide*.

10 Verify the status of replication.

```
# vradmin -g dg_name repstatus rvg_name
```

Ensure that the replication status shows:

```
Replication status: replicating (connected)
```

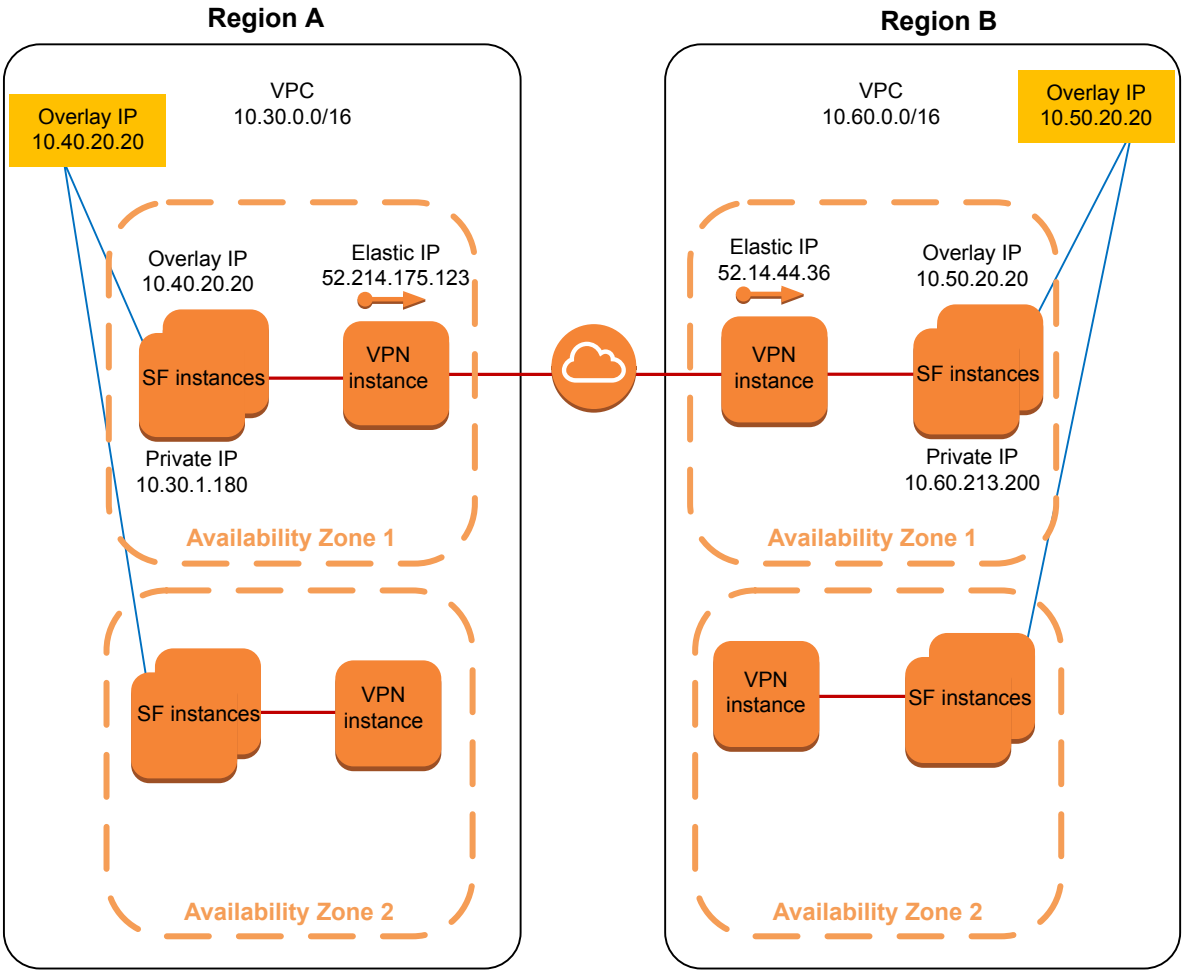
Replication across multiple AWS Availability Zones and regions (campus cluster)

In this scenario, data is replicated across multiple Availability Zones and regions. The configuration uses software VPN `Openswan` to connect the VPCs across different regions.

Figure 11-4 illustrates the configuration for replication across multiple Availability Zones and regions.

- Two VPCs with valid CIDR blocks (for example, 10.30.0.0/16 and 10.60.0.0/16 respectively) that are located in two different regions.
- The primary instance belongs to AZ1 of region 1 and the secondary instance belongs to AZ2 of region 2.
- Veritas InfoScale instances in each AZ.
- The primary Instance communicates with the secondary Instance using VPN instances at both ends.
- A VPN tunnel to secure communication between the instances across VPCs.
- Elastic IP addresses (EIP) to connect the two VPN instances
- Private IP addresses used for replication in standalone environments OR Overlay IP addresses used for replication in clustered environments.

Figure 11-4 Replication across multiple Availability Zones and regions



Setting up replication across multiple Availability Zones and regions (campus cluster)

Perform the steps in the following procedure to set up replication across regions.

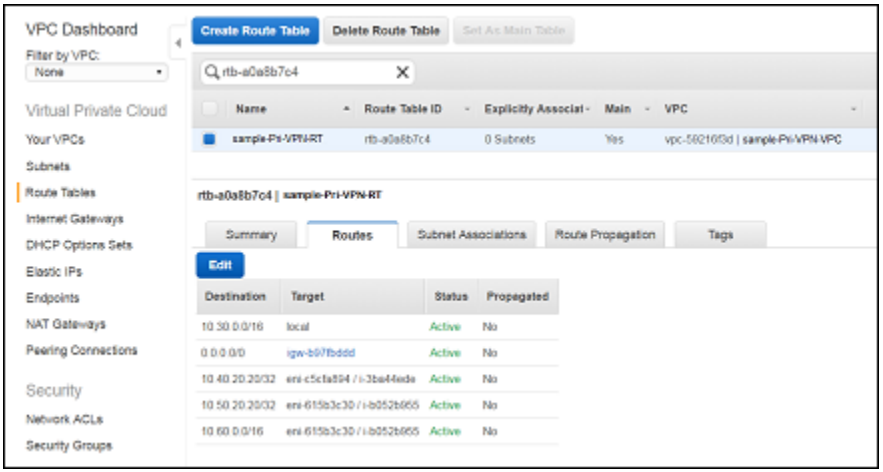
To set up replication across regions

1. Create two VPCs with valid CIDR blocks, for example, 10.30.0.0/16 and 10.60.0.0/16 respectively.
2. Create the primary site EC2 instances in the respective Availability Zones of the region.

3 Create the primary site VPN instances in the respective Availability Zones of the region. The VPN instances belong to the same VPC as that of the primary EC2 instance.

4 Choose a valid overlay IP address as the replication IP address for the primary site. The overlay IP address is a private IP address outside of the primary site's VPC CIDR block. Plumb the overlay IP address on the master node of the primary site cluster.

5 Modify the route table on the primary site to include the overlay IP address. Ensure that the route table entry directs any traffic destined for the primary site to be routed through the secondary VPN instance and the traffic destined for the secondary site overlay IP to be routed through the secondary InfoScale instance.

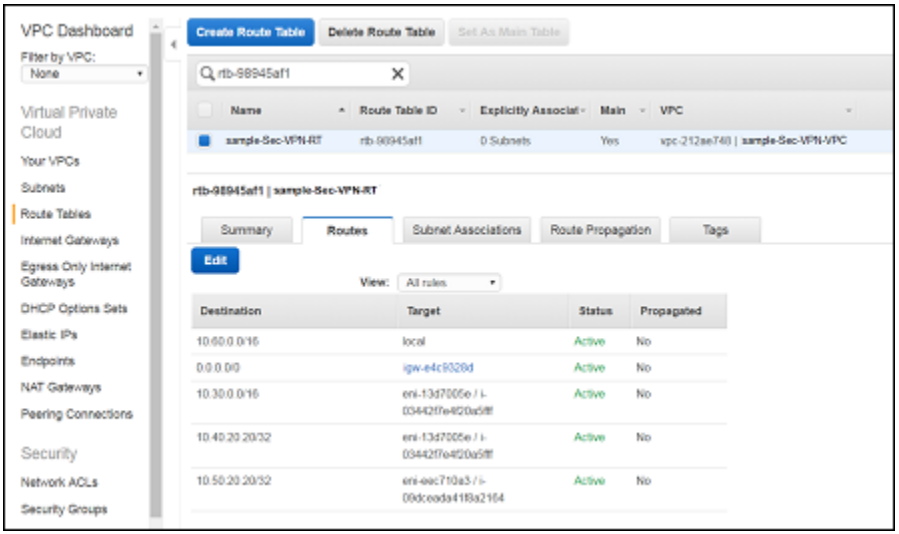


6 Create the secondary site EC2 instances in the respective Availability Zones of the second region.

7 Create the secondary site VPN instances in the respective Availability Zones of the second region. The VPN instances belong to the same VPC as that of the secondary EC2 instance.

8 Choose a valid overlay IP address as the replication IP address for the secondary site. The overlay IP address is a private IP address outside of the secondary site's VPC CIDR block. Plumb the overlay IP address on the master node of the secondary site cluster.

9 Modify the route table on the secondary site. Ensure that the route table entries direct traffic destined for the secondary site to be routed through the primary VPN instance and the traffic destined for the primary site overlay IP to be routed through the primary InfoScale instance.



10 Set up connectivity across regions using software VPN. The sample configuration uses Openswan.

Perform the following steps:

- Install the Openswan packages on the primary and secondary VPN instances.
- Configure the `/etc/ipsec.conf` and `/etc/ipsec.secrets` files.

Note: The `/etc/ipsec.conf` file contains information about the private IP address of the VPN instance, the subnet range of the left subnet, elastic IP address of the destination VPN, the subnet range of the destination right subnet.

The `/etc/ipsec.secrets` file contains the secret key. This key must be the same on both VPN sites.

- Restart the IPSec service.

```
# service ipsec restart
```

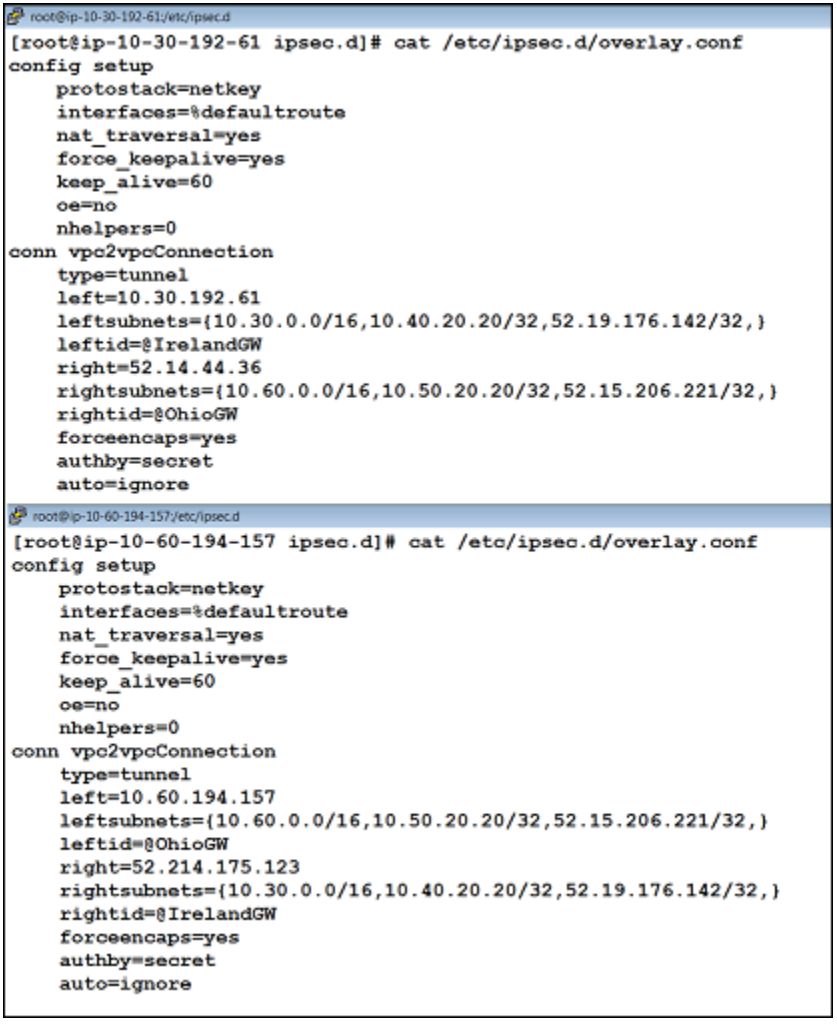
- Add the IPSec connection.

```
# ipsec auto -add vpc2vpcConnection
# ipsec auto -up vpc2vpcConnection
```

- Enable IPSec forwarding.

```
# sysctl -w net.ipv4.ip_forward=1
```

11 Modify the `ipsec.conf` file to add the overlay IP address for both primary and secondary site VPN instances.



```
root@ip-10-30-192-61:/etc/ipsec.d
[root@ip-10-30-192-61 ipsec.d]# cat /etc/ipsec.d/overlay.conf
config setup
    protostack=netkey
    interfaces=%defaultroute
    nat_traversal=yes
    force_keepalive=yes
    keep_alive=60
    oe=no
    nhelpers=0
conn vpc2vpcConnection
    type=tunnel
    left=10.30.192.61
    leftsubnets={10.30.0.0/16,10.40.20.20/32,52.19.176.142/32,}
    leftid=@IrelandGW
    right=52.14.44.36
    rightsubnets={10.60.0.0/16,10.50.20.20/32,52.15.206.221/32,}
    rightid=@OhioGW
    forceencaps=yes
    authby=secret
    auto=ignore
root@ip-10-60-194-157:/etc/ipsec.d
[root@ip-10-60-194-157 ipsec.d]# cat /etc/ipsec.d/overlay.conf
config setup
    protostack=netkey
    interfaces=%defaultroute
    nat_traversal=yes
    force_keepalive=yes
    keep_alive=60
    oe=no
    nhelpers=0
conn vpc2vpcConnection
    type=tunnel
    left=10.60.194.157
    leftsubnets={10.60.0.0/16,10.50.20.20/32,52.15.206.221/32,}
    leftid=@OhioGW
    right=52.214.175.123
    rightsubnets={10.30.0.0/16,10.40.20.20/32,52.19.176.142/32,}
    rightid=@IrelandGW
    forceencaps=yes
    authby=secret
    auto=ignore
```

12 Verify whether or not the master nodes on the primary and secondary site can reach each other using the overlay IP address.

13 Set up replication between the primary and secondary sites.

For instructions, see the chapter *Setting up replication* in the *Veritas InfoScale Replication Administrator's Guide*.

14 Verify the status of replication.

```
# vradmin -g dg_name repstatus rvg_name
```

Ensure that the RLINK is in CONNECT state and the replication status shows:

```
Replication status: replicating (connected)
```

Chapter 12

Replicating in Azure cloud environment

This chapter includes the following topics:

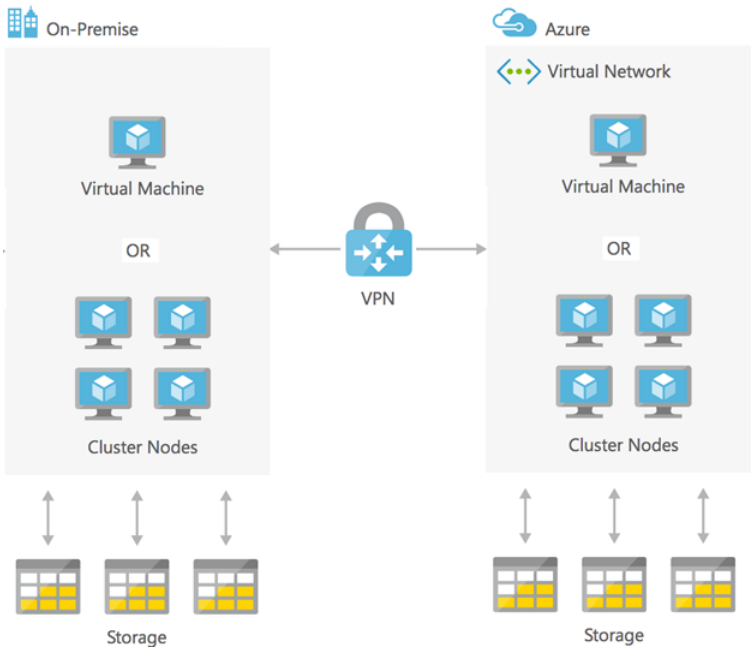
- Replication from on-premise to Azure cloud
- Replication within same Azure region
- Replication across Azure regions
- Replication across multiple sites and regions (campus cluster)
- About identifying a temporary resource disk

Replication from on-premise to Azure cloud

The following diagram illustrates the sample configuration for setting up replication between an on-premise data center to Azure cloud (on-cloud data center):

Note: For ease of use, the machines, whether virtual or physical are commonly mentioned as virtual machines. Your on-premise data center may include physical machines instead of the virtual machines. In any case, the steps to set up replication from an on-premise data center to an on-cloud data center remain the same for physical and well as virtual machines.

Figure 12-1 Sample configuration for setting up replication between an on-premise data center to on-cloud data center



About setting up replication between an on-premise data center to on-cloud data center

Replication between an on-premise data center to on-cloud data center involves the following high-level steps:

1. Prepare the setup at on-premise data center
2. Prepare the setup at on-cloud data center
3. Establish a tunnel from on-premise data center to cloud data center
4. Deploy setup

The following sections provide details about performing each of these steps.

Preparing the setup at on-premise data center

Perform the following steps to prepare the setup at on-premise data center:

1. Enable the ports that are used for inbound and outbound communication.

   For a list of required ports and services, refer to, *Veritas InfoScale™ Replication Administrator's Guide - Linux*.

2. Create a subnet and a local VPN gateway.
3. Note the address space that is allotted for the subnet and the public IP address that is allotted for the local VPN gateway.

Preparing the setup at on-cloud data center

Perform the following steps to prepare the setup at on-cloud data center:

1. Using Microsoft Azure portal, create a resource group.
2. Create a VNet in the resource group created and specify an IP address space for the VNet.

   The IP address range must be diff on the on-premise subnet and on the on-cloud subnet.

3. Create a gateway subnet.
4. Create a VPN gateway and associate it with the created VNet.

   Note the public IP address that is allotted for the on-cloud VPN gateway.

5. Create a local network gateway.

   When you create the local network gateway, you must provide the on-premise subnet IP address range and the public IP address of the on-premise local VPN gateway.

6. Establish a tunnel from on-cloud to on-premise network.

   To establish the tunnel, create a connection of type **Site-to-Site (IPSec)** and choose the on-cloud VPN gateway and the local network gateway.

7. Provide a shared key (alpha-numeric key).

   A shared key is a pass-phrase. This pass-phrase is required when you establish a tunnel from on-premise data center to cloud data center.

Establishing a tunnel from on-premise data center to on-cloud data center

To establish a tunnel from on-premise data center to on-cloud data center, use the following parameters:

- Public IP address that is allotted for the on-cloud VPN gateway

- Shared key (alpha-numeric key) that was provided while establishing a tunnel from on-cloud data center to on-premise data center
- On-cloud VPN gateway configuration type (Policy based or Route based)

Deploying the setup

Perform the following steps to deploy the setup (in both the data centers):

1. Create virtual machines in the subnets created.
2. Provision storage.
3. Install InfoScale Storage/InfoScale Enterprise.
4. Create VxVM disk groups, VxVM volumes, Storage Replicator Log (SRL), Replicated Volume Group (RVG), and RLinks.

   For details refer to, *Storage Foundation Cluster File System High Availability Administrator's Guide*.

   Note: In Azure environment, by default, in addition to the storage disks that you have attached, every virtual machine that is provisioned contains a temporary resource disk, that serves as an ephemeral storage. Do not use the temporary resource as a data disk (VxVM disk) to store persistent data. The disk may change after a machine is redeployed or is restarted, and the data will be lost. For more information about how Azure uses a temporary disk, see Microsoft documentation.

   For details about how to identify a temporary resource disk:

   See "About identifying a temporary resource disk" on page 204.

   ---

5. Flush the iptables on both the virtual machines.

   ```
   # iptable -F
   ```

6. Set up replication between the virtual machines using the private IP address or the virtual IP address.

   For details about setting up replication, see, *Setting up replication* in the *Veritas InfoScale Replication Administrator's Guide*.

7. Verify the replication status.

   ```
   # vradmin -g dg_name repstatus rvg_name
   ```

   Ensure that the replication status shows:

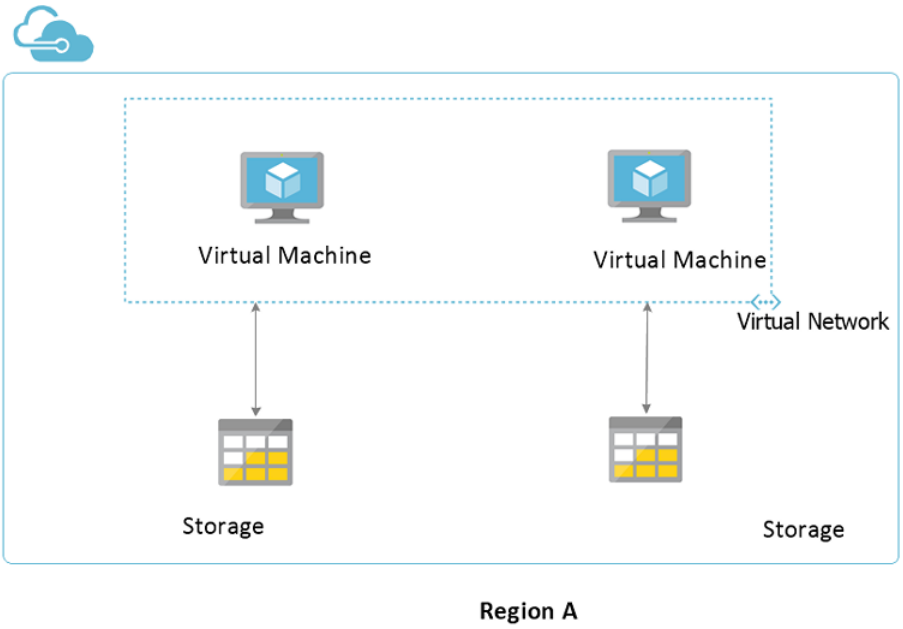
   ```
   Replication status: replicating (connected)
   ```

Replication within same Azure region

In an Azure cloud environment, in a single region, you can provision your setup across virtual networks (VNets) or within a VNet.

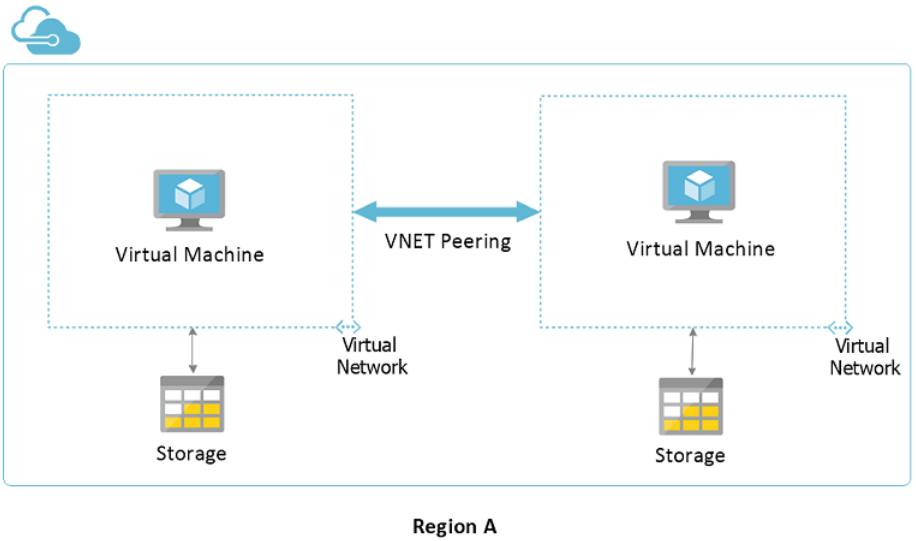
The following diagram illustrates the sample configuration for setting up replication between the same VNet:

Figure 12-2 Sample configuration for setting up replication between same VNet



The following diagram illustrates the sample configuration for setting up replication across VNets:

Figure 12-3 Sample configuration for setting up replication across VNets



Setting up replication in the same VNet

Perform the following steps to set up replication in the same VNet, within the same region

1. Enable the ports that are used for inbound and outbound communication.

   For a list of required ports and services, refer to, *Veritas InfoScale™ Replication Administrator's Guide - Linux*.

2. Using Microsoft Azure portal, create a VNet and specify an IP address space for the VNet.
3. Create a subnet in the VNet created.

   For details about creating a VNet, specifying an IP address space, and creating a subnet, refer to Microsoft documentation.

4. Create two virtual machines within the subnets and provision storage.
5. Install InfoScale Storage/InfoScale Enterprise on both the virtual machines.

6 Create VxVM disk groups, VxVM volumes, Storage Replicator Log (SRL), Replicated Volume Group (RVG), and RLinks.

For details refer to, *Storage Foundation Cluster File System High Availability Administrator's Guide*.

Note: In Azure environment, by default, in addition to the storage disks that you have attached, every virtual machine that is provisioned contains a temporary resource disk, that serves as an ephemeral storage. Do not use the temporary resource as a data disk (VxVM disk) to store persistent data. The disk may change after a machine is redeployed or is restarted, and the data will be lost. For more information about how Azure uses a temporary disk, see Microsoft documentation.

For details about how to identify a temporary resource disk:

See "About identifying a temporary resource disk" on page 204.

---

7 Flush the iptables on both the virtual machines.

```
# iptable -F
```

8 Set up replication between the virtual machines using the private IP address or the virtual IP address.

For details about setting up replication see *Setting up replication* in the *Veritas InfoScale Replication Administrator's Guide*.

9 Verify the replication status.

```
# vradmin -g dg_name repstatus rvg_name
```

Ensure that the replication status shows:

```
Replication status: replicating (connected)
```

Setting up replication across VNet

Perform the following steps to set up replication across the VNets, within the same region

1 Using Microsoft Azure portal, create two VNets and specify an IP address space for each VNet.

2 Set up VNet Peering between the two VNets.

3 Create a subnet in each VNet.

4 Create a virtual machine in each subnet and provision storage.

5 Install InfoScale Storage/InfoScale Enterprise on both the virtual machines.

6 Create VxVM disk groups, VxVM volumes, Storage Replicator Log (SRL), Replicated Volume Group (RVG), and RLinks.

For details refer to, *Storage Foundation Cluster File System High Availability Administrator's Guide*.

Note: In Azure environment, by default, in addition to the storage disks that you have attached, every virtual machine that is provisioned contains a temporary resource disk, that serves as an ephemeral storage. Do not use the temporary resource as a data disk (VxVM disk) to store persistent data. The disk may change after a machine is redeployed or is restarted, and the data will be lost. For more information about how Azure uses a temporary disk, see Microsoft documentation.

For details about how to identify a temporary resource disk:

See "About identifying a temporary resource disk" on page 204.

---

7 Flush the iptables on both the virtual machines.

```
# iptable -F
```

8 Set up replication between the virtual machines using the private IP address or the virtual IP address.

For details about setting up replication see *Setting up replication* in the *Veritas InfoScale Replication Administrator's Guide*.

9 Verify the replication status.

```
# vradmin -g dg_name repstatus rvg_name
```

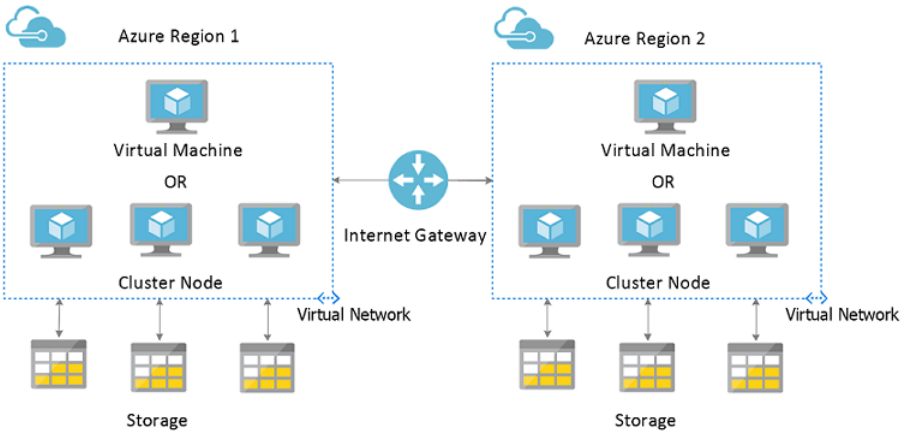
Ensure that the replication status shows:

```
Replication status: replicating (connected)
```

Replication across Azure regions

The following diagram illustrates the sample configuration for setting up replication across regions:

Figure 12-4 Sample configuration for setting up replication across regions



Setting up replication across regions

Perform the following steps to set up replication across region

1. Enable the ports that are used for inbound and outbound communication.

   For a list of required ports and services, refer to, *Veritas InfoScale™ Replication Administrator's Guide - Linux*.

2. Using Azure portal, create a resource group (RG) in both the regions.
3. Create a VNet in each region and specify a non-overlapping IP address space for each VNet.
4. Create a subnet and a gateway subnet under the VNets created in both the regions.
5. Create a virtual network gateway in both the regions.
6. Establish a connection in between the two virtual network gateways.
7. Create virtual machines in the subnets and provision storage.
8. Install InfoScale Storage/InfoScale Enterprise.

9 Create VxVM disk groups, VxVM volumes, Storage Replicator Log (SRL), Replicated Volume Group (RVG), and RLinks.

For details refer to, *Storage Foundation Cluster File System High Availability Administrator's Guide*

Note: In Azure environment, by default, in addition to the storage disks that you have attached, every virtual machine that is provisioned contains a temporary resource disk, that serves as an ephemeral storage. Do not use the temporary resource as a data disk (VxVM disk) to store persistent data. The disk may change after a machine is redeployed or is restarted, and the data will be lost. For more information about how Azure uses a temporary disk, see Microsoft documentation.

For details about how to identify a temporary resource disk:

See "About identifying a temporary resource disk" on page 204.

---

10 Flush the iptables on both the virtual machines.

```
# iptable -F
```

11 Set up replication using the private IP address or the virtual IP address.

For details about setting up replication see *Setting up replication* in the *Veritas InfoScale Replication Administrator's Guide*.

12 Verify the replication status.

```
# vradmin -g dg_name repstatus rvg_name
```

Ensure that the replication status shows:

```
Replication status: replicating (connected)
```

Replication across multiple sites and regions (campus cluster)

To set up replication across regions, in a campus cluster, perform the following steps at both the regions:

1 Enable the ports that are used for inbound and outbound communication.

For a list of required ports and services, refer to, *Veritas InfoScale™ Replication Administrator's Guide - Linux*.

2 Using Azure portal, create a resource group (RG) in both the regions.

3. Create a VNet in each region and specify a non-overlapping IP address space for each VNet.
4. Create a subnet and a gateway subnet under the VNets created in both the regions.
5. Create a virtual network gateway in both the regions.
6. Establish a connection in between the two virtual network gateways.
7. Create virtual machines in the subnets and provision storage.
8. Install InfoScale Storage/InfoScale Enterprise.
9. Create FSS disk groups, VxVM volumes, Storage Replicator Log (SRL), Replicated Volume Group (RVG), and RLinks.

   For details refer to, *Storage Foundation Cluster File System High Availability Administrator's Guide*.

   Note: In Azure environment, by default, in addition to the storage disks that you have attached, every virtual machine that is provisioned contains a temporary resource disk, that serves as an ephemeral storage. Do not use the temporary resource as a data disk (VxVM disk) to store persistent data. The disk may change after a machine is redeployed or is restarted, and the data will be lost. For more information about how Azure uses a temporary disk, see Microsoft documentation.

   For details about how to identify a temporary resource disk:

   See "About identifying a temporary resource disk" on page 204.

   ---

10. Create clusters in both the regions.
11. Flush the iptables on all the cluster nodes.

    ```
    # iptable -F
    ```

12. Set up replication using the private IP address or the virtual IP address.

    For details about setting up replication see *Setting up replication* in the *Veritas InfoScale Replication Administrator's Guide*.

13. Verify the replication status.

    ```
    # vradmin -g dg_name repstatus rvg_name
    ```

    Ensure that the replication status shows:

    ```
    Replication status: replicating (connected)
    ```

About identifying a temporary resource disk

Typically, a temporary resource disk is named as /dev/sdb and is mounted at /mnt. However, the location may change depending on whether or not it is utilized for swap space or is unmounted by a user.

To identify whether or not swap space is configured and a disk utilized for swap space

1. Identify the swap space configuration and check the swap file:

   ```
   # swapon
   ```

   Following is the sample output of this command:

   ```
   NAME TYPE SIZE USED PRIO /mnt/resource/swapfile file 2G 0B -1
   ```

   Where, /mnt/resource is the default location where a temporary disk is mounted when used for swap space.

2. Identify the disk used for swap space:

   ```
   # mount | grep "/mnt/resource"
   ```

   Following is the sample output of this command:

   ```
   /dev/sdb on /mnt/resource type filesystem
   ```

   Where /dev/sdb is the temporary disk.

3. Identify a VxVM disk that corresponds to a temporary disk.

   ```
   # vxdisk -e list | grep sdb (If sdb is the OS device name for
   temporary disk found in the earlier step)
   ```

   Following is the sample output for the command:

   ```
   10-0-15-6_disk_490 auto:none - - online invalid sdb -
   ```

Chapter 13

Replicating across clouds

This chapter includes the following topics:

- Replication in cross-cloud environment

Replication in cross-cloud environment

Replication across cloud is similar to setting up replication from an on-premise data center to an on-cloud datacenter, were one of the cloud can be considered as an on-premise data center and the second as an on-cloud data center.

It involves the following high-level steps:

1. Prepare the deployment setup in each cloud.

   See “Replication from on-premise to AWS cloud” on page 177.

   See “Replication from on-premise to Azure cloud” on page 193.

2. Set up a communication channel between the clouds. For example, you can set up a virtual network gateway or can configure a VPN-based connection.
3. Install InfoScale Storage/InfoScale Enterprise and configure VVR on the virtual machines created.

   For details about installing InfoScale Storage/InfoScale Enterprise, refer to *Veritas InfoScale Installation Guide*.

   For details about configuring replication, refer to *Disaster Recovery Implementation Guide*.

Section 6

Reference

- Appendix A. Sample configuration files

Appendix A

Sample configuration files

This appendix includes the following topics:

- Sample Storage Foundation for Oracle RAC configuration files
- About sample main.cf files for Storage Foundation (SF) for Oracle RAC
- About sample main.cf files for Storage Foundation (SF) for Sybase ASE CE

Sample Storage Foundation for Oracle RAC configuration files

Storage Foundation for Oracle RAC provides several sample configuration files illustrating various scenarios. You may use the sample files as a guideline for setting up your cluster environment. These sample files are located at `/etc/VRTSvcs/conf/sample_rac/`.

This section briefly describes each of the sample files and illustrates the service group configuration for each of them. The section does not include a copy of the main.cf files.

The following sample files are useful references for disaster recovery use cases:

- sfrac02_main.cf file
- sfrac11_main.cf file
- See “Sample fire drill service group configuration” on page 217.

sfrac02_main.cf file

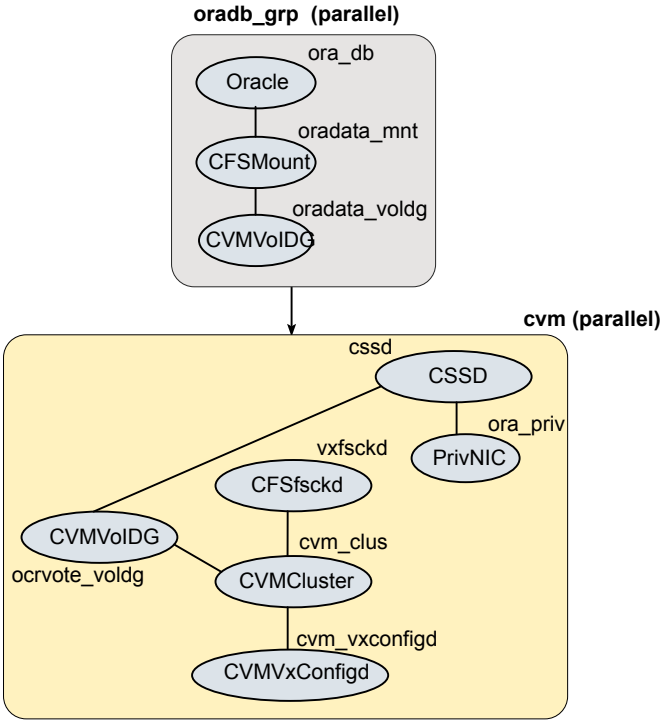
This sample file describes the following configuration:

- A two node Storage Foundation for Oracle RAC cluster.
- The Oracle database is stored on CFS.

- The database is managed by the VCS agent for Oracle.
  The agent starts, stops, and monitors the database.
- The database uses the Oracle UDP IPC for database cache fusion.
- A common IP address is used by Oracle Clusterware and database cache fusion. The private IP address is managed by the PrivNIC agent for high availability.
- The Oracle Cluster Registry (OCR) and voting disk are stored on CVM raw volumes.

Figure A-1 illustrates the configuration.

Figure A-1 Service group configuration for sfrac02_main.cf file



sfrac07_main.cf and sfrac08_main.cf files

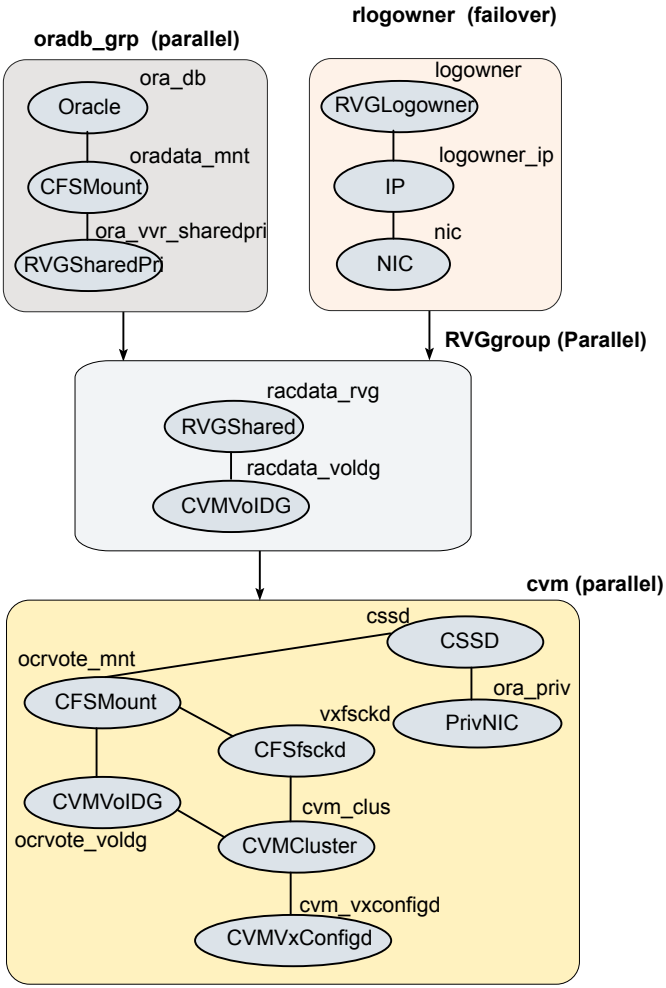
The sample configuration, sfrac07_main.cf, describes a disaster recovery configuration for the primary site. The sample configuration, sfrac08_main.cf, describes a disaster recovery configuration for the secondary site. The configuration uses VVR for replicating data between the sites.

This sample file describes the following configuration:

- Two Storage Foundation for Oracle RAC clusters, comprising two nodes each, hosted at different geographical locations.
- A single Oracle database that is stored on CFS.
- The database is managed by the VCS agent for Oracle.
  The agent starts, stops, and monitors the database.
- The database uses the Oracle UDP IPC for database cache fusion.
- A common IP address is used by Oracle Clusterware and database cache fusion. The private IP address is managed by the PrivNIC agent for high availability.
- One virtual IP address must be configured under the `ClusterService` group on each site for inter-cluster communication.
- The Oracle Cluster Registry (OCR) and voting disk are stored on CFS.
- Volume Replicator (VVR) is used to replicate data between the sites.
- The shared volumes replicated across the sites are configured under the RVG group.
- The replication link used by VVR for communicating log information between sites are configured under the rlogowner group. This is a failover group, and remains online from the cvm master node. The PreOnline trigger for the RVGLogOwner agent should be enabled.
- The database group will be online on the primary cluster. The RVGSharedPri resource determines where the database group will be brought online.
- The database group is configured as a global group by specifying the clusters on the primary and secondary sites as values for the ClusterList group attribute.

Figure A-2 illustrates the configuration. The service group configuration is the same on the primary and secondary site. The availability of groups (online/offline) differ between the sites.

Figure A-2 Service group configuration for sfrac07_main.cf and sfrac08_main.cf files



sfrac09_main.cf and sfrac10_main.cf files

The sample configuration, sfrac09_main.cf, describes a disaster recovery configuration for the primary site. The sample configuration, sfrac10_main.cf, describes a disaster recovery configuration for the secondary site. The sample configuration uses EMC SRDF technology for replicating data between the sites.

Note: You can use other supported hardware-based replication technologies with this configuration.

This sample file describes the following configuration:

- Two Storage Foundation for Oracle RAC clusters, comprising two nodes each, hosted at different geographical locations.
- A single Oracle database that is stored on CFS.
- The database is managed by the VCS agent for Oracle.
  The agent starts, stops, and monitors the database.
- The database uses the Oracle UDP IPC for database cache fusion.
- A common IP address is used by Oracle Clusterware and database cache fusion. The private IP address is managed by the PrivNIC agent for high availability.
- One virtual IP address must be configured under the `ClusterService` group on each site for inter-cluster communication.
- The Oracle Cluster Registry (OCR) and voting disk are stored on CFS.
- EMC SRDF is used to replicate data between the sites.
- The SRDF disk groups that are replicated across the sites using SRDF technology and the replication mode are specified under the SRDF resource in the database group. The CVM disk group that comprises the SRDF disk group must be configured under the CVMVolDg resource in the database group.
- The database group will be online on the primary cluster. The SRDF resource determines where the database group will be brought online.
- The database group is configured as a global group by specifying the clusters on the primary and secondary sites as values for the ClusterList group attribute.

Figure A-3 illustrates the configuration on the primary site.

Figure A-3 Service group configuration for sfrac09_main.cf file

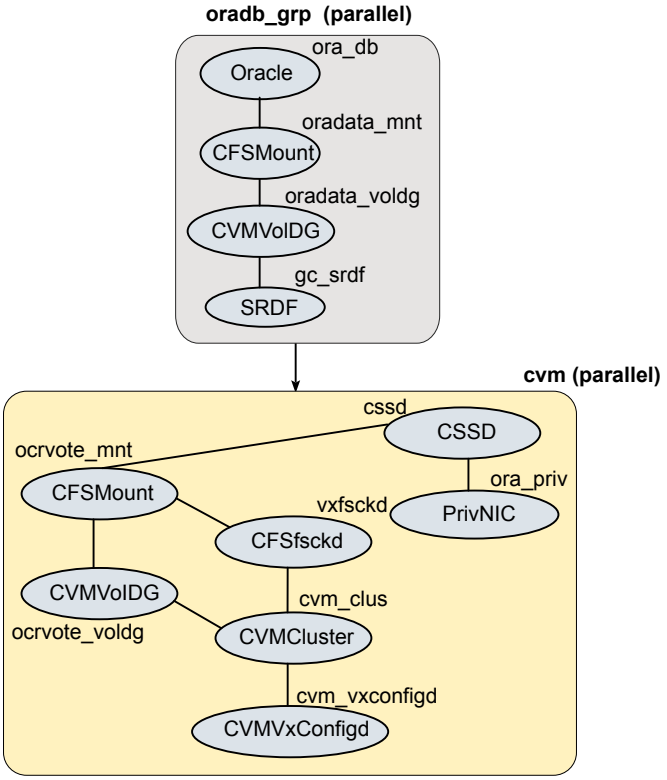
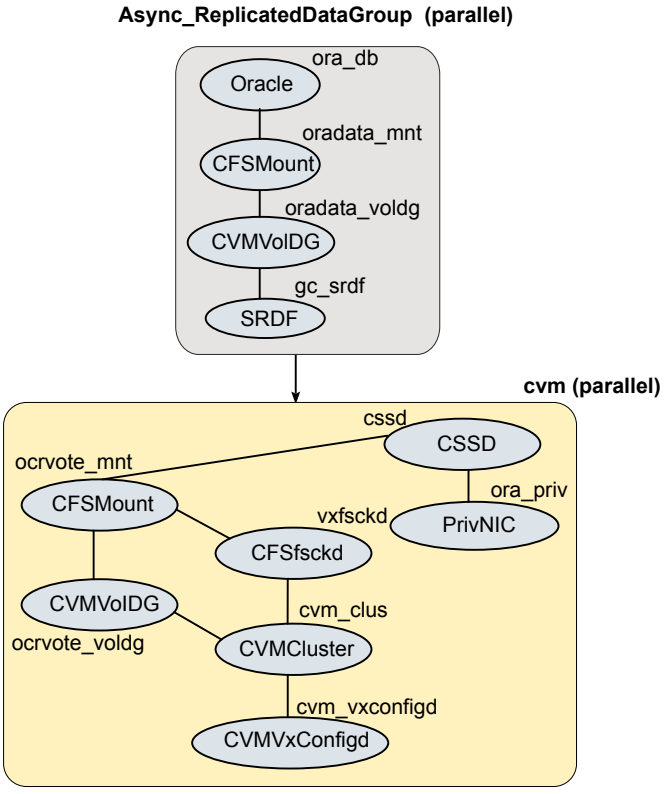


Figure A-4 illustrates the configuration on the secondary site.

Figure A-4 Service group configuration for sfrac10_main.cf file



sfrac11_main.cf file

This sample file describes the following configuration:

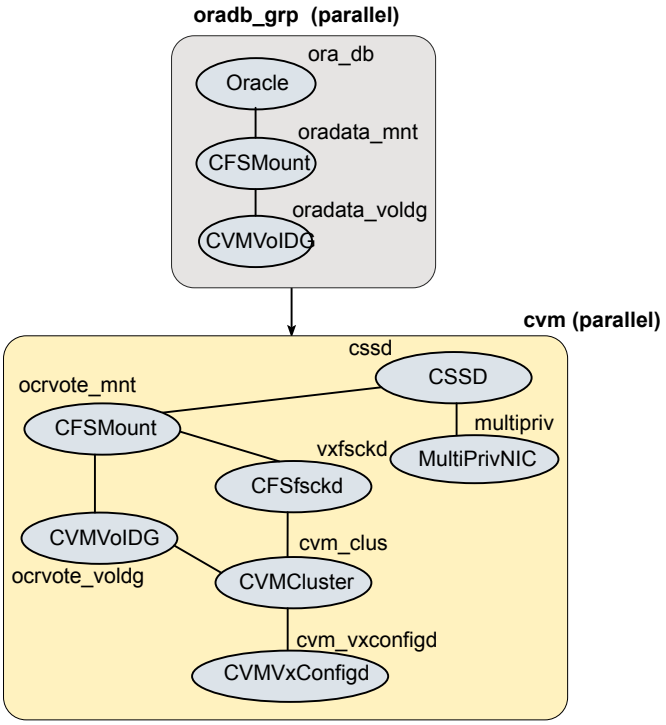
- An Storage Foundation for Oracle RAC campus cluster with four nodes hosted across two sites.
- Each site comprises two nodes of the cluster hosting a single database.
- The Oracle database is stored on CFS.
- The database is managed by the VCS agent for Oracle.
  The agent starts, stops, and monitors the database.
- The database uses the Oracle UDP IPC for database cache fusion.
- The IP address on NIC1 is used by Oracle Clusterware.
  The second IP address on NIC2 is used for Oracle database cache fusion.

The private IP addresses are managed by the MultiPrivNIC agent for high availability.

- The Oracle Cluster Registry (OCR) and voting disk are stored on CFS.
- Group the hosts at each physical site into separate logical system zones using the SystemZones attribute.

Figure A-5 illustrates the configuration.

Figure A-5 Service group configuration for sfrac11_main.cf file



sfrac12_main.cf and sfrac13_main.cf files

The sample configuration, sfrac12_main.cf, describes a disaster recovery configuration for the primary site. The sample configuration, sfrac13_main.cf, describes a disaster recovery configuration for the secondary site with fire-drill capability. The sample configuration uses Hitachi True Copy technology for replicating data between the sites.

Note: You can use other supported hardware-based replication technologies with this configuration.

This sample file describes the following configuration:

- Two Storage Foundation for Oracle RAC clusters, comprising two nodes each, hosted at different geographical locations.
- A single Oracle database that is stored on CFS.
- The database is managed by the VCS agent for Oracle.
  The agent starts, stops, and monitors the database.
- The database uses the Oracle UDP IPC for database cache fusion.
- A common IP address is used by Oracle Clusterware and database cache fusion. The private IP address is managed by the PrivNIC agent for high availability.
- One virtual IP address must be configured under the `ClusterService` group on each site for inter-cluster communication.
- The Oracle Cluster Registry (OCR) and voting disk are stored on CFS.
- Hitachi True Copy is used to replicate data between the sites.
- The HTC disk groups that are replicated across the sites using HTC technology and the replication mode are specified under the HTC resource in the database group. The CVM disk group that comprises the HTC disk group must be configured under the CVMVolDg resource in the database group.
- The database group will be online on the primary cluster. The HTC resource determines where the database group will be brought online.
- The database group is configured as a global group by specifying the clusters on the primary and secondary sites as values for the ClusterList group attribute.
- The database group oradb_grp_fd on the secondary is configured for fire drill.
- When the group oradb_grp_fd is brought online, the HTCSnap creates a snapshot of the disk group configured under the HTC resource in the database group oradg_grp.
  Further, the Oracle database and the associated volumes and mount points configured under the service group oradb_grp_fd are brought online using the snapshots created by HTCSnap.

Figure A-6 illustrates the configuration on the primary site.

Figure A-6 Service group configuration for sfrac12_main.cf file

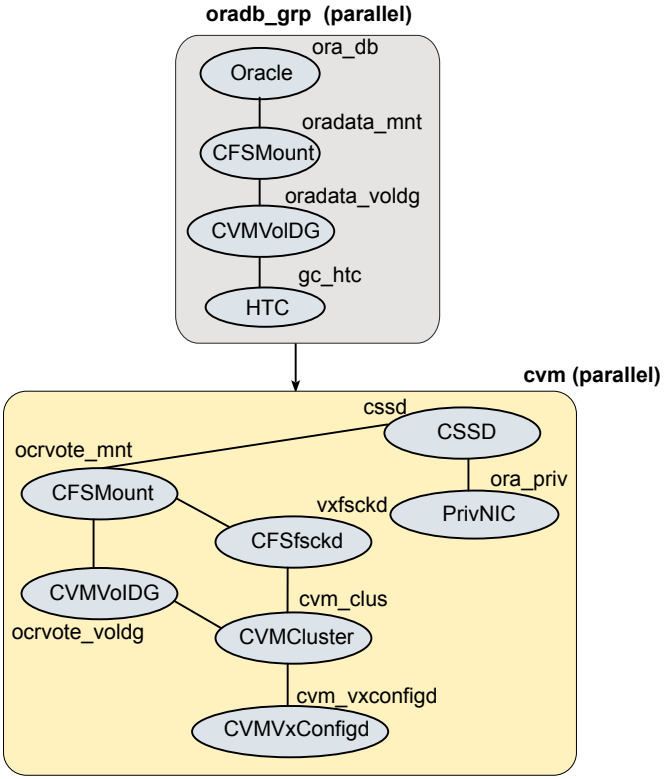
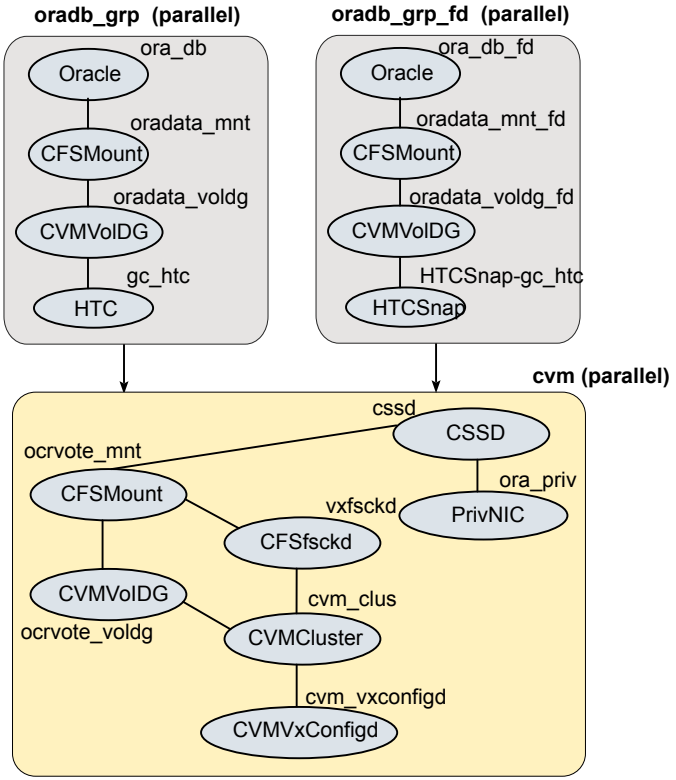


Figure A-7 illustrates the configuration on the secondary site.

Figure A-7 Service group configuration for sfrac13_main.cf file



Sample fire drill service group configuration

The sample configuration in this section describes a fire drill service group configuration on the secondary site. The configuration uses VVR for replicating data between the sites.

The sample service group describes the following configuration:

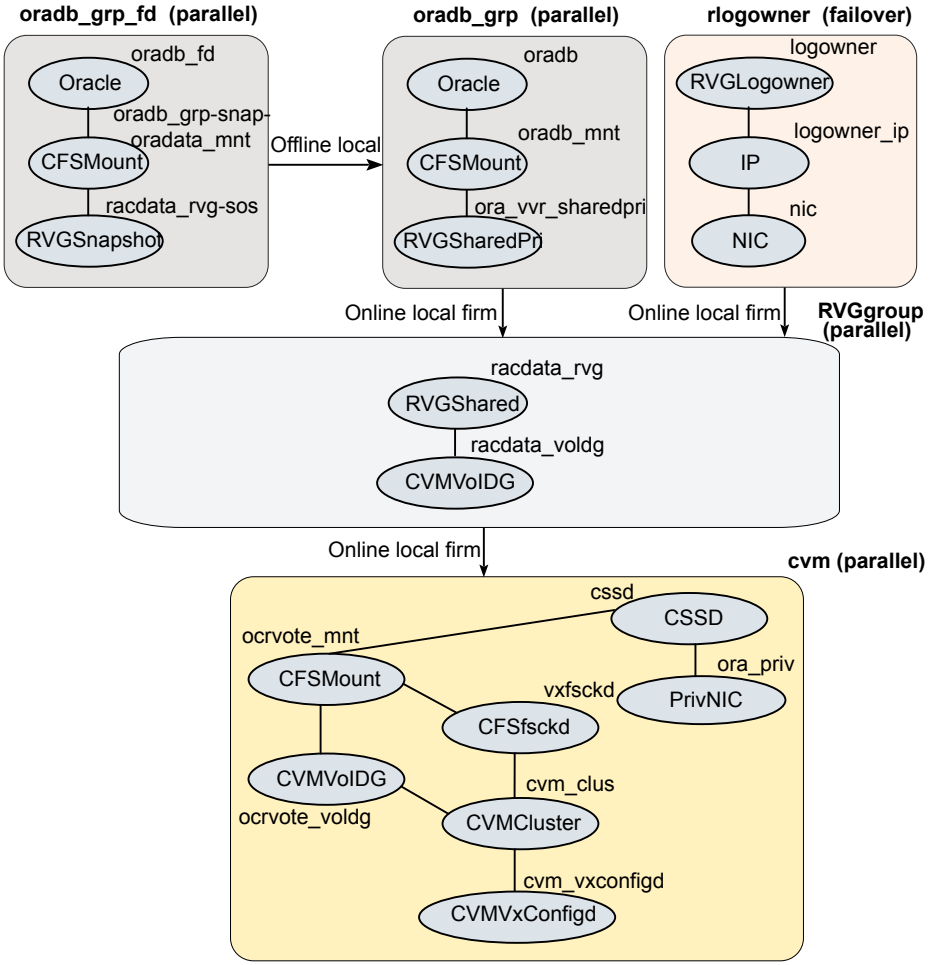
- Two Storage Foundation for Oracle RAC clusters, comprising two nodes each, hosted at different geographical locations.
- A single Oracle database that is stored on CFS.
- The database is managed by the VCS agent for Oracle.
  The agent starts, stops, and monitors the database.
- The database uses the Oracle UDP IPC for database cache fusion.

- A common IP address is used by Oracle Clusterware and database cache fusion. The private IP address is managed by the PrivNIC agent for high availability.
- One virtual IP address must be configured under the `ClusterService` group on each site for inter-cluster communication.
- The Oracle Cluster Registry (OCR) and voting disk are stored on CFS.
- Volume Replicator (VVR) is used to replicate data between the sites.
- The shared volumes replicated across the sites are configured under the RVG group.
- The replication link used by VVR for communicating log information between sites are configured under the rlogowner group. This is a failover group, and remains online from the cvm master node. The PreOnline trigger for the RVGLogOwner agent should be enabled.

  Please see the section "CVM master node needs to assume the logowner role for VCS managed VVR resources" in the *Cluster Server Bundled Agents Reference Guide* for more info.
- The database group is configured as a global group by specifying the clusters on the primary and secondary sites as values for the ClusterList group attribute.
- The fire drill service group `oradb_grp_fd` creates a snapshot of the replicated data on the secondary site and starts the database using the snapshot. An offline local dependency is set between the fire drill service group and the application service group to make sure a fire drill does not block an application failover in case a disaster strikes the primary site.

Figure A-8 illustrates the configuration.

Figure A-8 Service group configuration for fire drill



About sample main.cf files for Storage Foundation (SF) for Oracle RAC

You can examine the VCS configuration file, main.cf, to verify SF Oracle RAC installation and configuration.

Sample main.cf file examples are provided for the following Oracle RAC configurations:

- Replicating data between two clusters

- For a primary site in a CVM VVR configuration
- For a secondary site in a CVM VVR configuration

Sample main.cf for Oracle 10g for CVM/VVR primary site

The following are the configuration details for this sample main.cf:

- Configuration file name: cvmvvr_primary_main.cf
- More general purpose, can have multiple Oracle databases

```
include "types.cf"
include "types.cf"
include "CFSTypes.cf"
include "CVMTypes.cf"
include "OracleTypes.cf"
include "PrivNIC.cf"
include "VVRTypes.cf"
include "VVRTypes.cf"
include "VVRTypes.cf"
include "/etc/VRTSvcs/conf/config/VVRTypes.cf"

cluster rac_cluster101 (
    UserNames = { admin = bopHo }
    ClusterAddress = "10.10.10.101"
    Administrators = { admin }
    UseFence = SCSI3
    )

remotecluster rac_cluster102 (
    ClusterAddress = "10.11.10.102"
    )

heartbeat Icmp (
    ClusterList = { rac_cluster102 }
    Arguments @rac_cluster102 = { "10.11.10.102" }
    )

system galaxy (
    )

system nebula (
    )

group ClusterService (
    SystemList = { galaxy = 0, nebula = 1 }
```

```
    AutoStartList = { galaxy, nebula }
    OnlineRetryLimit = 3
    OnlineRetryInterval = 120
    )

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

    IP gcoip (
        Device = eth0
        Address = "10.10.10.101"
        NetMask = "255.255.240.0"
        )

    NIC csgnic (
       Device = eth0
       NetworkHosts = { "10.10.12.2", "10.10.12.3" }
       )

    gcoip requires csgnic
    wac requires gcoip

group RVGgroup (
    SystemList = { galaxy = 0, nebula = 1 }
    Parallel = 1
    AutoStartList = { galaxy, nebula }
    )

    CVMVolDg racdata_voldg (
        CVMDiskGroup = oradatadg
        CVMActivation = sw
        )

    RVGShared racdata_rvg (
        RVG = rac1_rvg
        DiskGroup = oradatadg
        )

    requires group cvm online local firm
    racdata_rvg requires racdata_voldg
```

```
group cvm (
    SystemList = { galaxy = 0, nebula = 1 }
    AutoFailOver = 0
    Parallel = 1
    AutoStartList = { galaxy, nebula }
    )

    CSSD cssd (
        Critical = 0
        CRSHOME="/u01/app/grid/product/11.2.0/gridhome"
        )

    CFSfsckd vxfsckd (
        )

    CVMCluster cvm_clus (
        CVMClustName = rac_cluster101
        CVMNodeId = { galaxy = 1, nebula = 2 }
        CVMTransport = gab
        CVMTimeout = 200
        )

CVMVolDg ocrvote_voldg (
    Critical = 0
    CVMDiskGroup = ocrvotedg
    CVMVolume = { ocrvol, votevol }
    CVMActivation = sw
    )

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

PrivNIC ora_priv (
    Critical = 0
    Device = { eth0 = 0, eth1 = 1}
    Address@galaxy = "192.168.12.1"
    Address@nebula = "192.168.12.2"
    NetMask = "255.255.240.0"
    )


    cssd requires ocrvote_voldg
    cssd requires ora_priv
```

```
    cssd requires ora_priv
    ocrvote_voldg requires cvm_clus
    vxfsckd requires cvm_clus
    cvm_clus requires cvm_vxconfigd

CFSMount oradata_mnt (
    MountPoint = "/oradata"
    BlockDevice = "/dev/vx/dsk/oradatadg/oradatavol"
    )

Oracle ora1 (
     Critical = 0
    Sid @galaxy = vrts1
    Sid @nebula = vrts2
    Owner = oracle
    Home = "/app/oracle/orahome"
    StartUpOpt = SRVCTLSTART
    ShutDownOpt = SRVCTLSTOP
    )

RVGSharedPri ora_vvr_sharedpri (
    RvgResourceName = racdata_rvg
    OnlineRetryLimit = 0
    )

requires group RVGgroup online local firm
 ora1 requires oradata_mnt
 oradata_mnt requires ora_vvr_sharedpri

group rlogowner (
    SystemList = { galaxy = 0, nebula = 1 }
    AutoStartList = { galaxy, nebula }
    OnlineRetryLimit = 2
    )

    IP logowner_ip (
        Device = eth0
        Address = "10.10.9.101"
        NetMask = "255.255.240.0"
    )

    NIC nic (
        Device = eth0
        NetworkHosts = { "10.10.12.2", "10.10.12.3" }
        )
```

```
RVGLogowner logowner (
    RVG = rac1_rvg
    DiskGroup = oradatadg
    )

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

group VxSS (
SystemList = { north = 0, south = 1 }
Parallel = 1
OnlineRetryLimit = 3
OnlineRetryInterval = 120
)
Phantom phantom_vxss (
)
ProcessOnOnly vxatd (
IgnoreArgs = 1
PathName = "/opt/VRTSat/bin/vxatd"
)

group CMC (
SystemList = { north, south }
AutoStartList = { north, south }
OnlineRetryLimit = 3
OnlineRetryInterval = 120
)
ClusterConnectorConfig CMC_ClusterConfig (
MSAddress = "mgmtserver1.veritasexample.com"
MSPort = 14145
ClusterId = "1145613636"
ClusterType = "vcs"
ClusterPort = 14141
VCSLoggingLevel = "TAG_A"
Logging = "/opt/VRTScmccc/conf/cc_logging.properties"
ClusterConnectorVersion = "5.0.1000.0"
)
Process CMC_ClusterConnector (
PathName = "/bin/sh"
Arguments = "/opt/VRTScmccc/bin/cluster_connector.sh"
)
CMC_ClusterConnector requires CMC_ClusterConfig
```

Sample main.cf for Oracle 10g for CVM/VVR secondary site

The following are the configuration details for this sample main.cf:

- Configuration file name: cvmvvr_secondary_main.cf
- More general purpose, can have multiple Oracle databases

```
include "types.cf"
include "CFSTypes.cf"
include "CVMTypes.cf"
include "OracleTypes.cf"
include "PrivNIC.cf"
include "VVRTypes.cf"

cluster rac_cluster102 (
    UserNames = { admin = bopHo }
    ClusterAddress = "10.11.10.102"
    Administrators = { admin }
    UseFence = SCSI3
    )

remotecluster rac_cluster101 (
    ClusterAddress = "10.10.10.101"
    )

heartbeat Icmp (
    ClusterList = { rac_cluster101 }
    Arguments @rac_cluster101 = { "10.10.10.101" }
    )

system mercury (
    )

system jupiter (
    )

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

    Application wac (
        StartProgram = "/opt/VRTSvcs/bin/wacstart"
        StopProgram = "/opt/VRTSvcs/bin/wacstop"
```

```
        MonitorProcesses = { "/opt/VRTSvcs/bin/wac" }
        RestartLimit = 3
        )

    IP gcoip (
        Device = eth0
        Address = "10.11.10.102"
        NetMask = "255.255.240.0"
        )

    NIC csgnic (
        Device = eth0
        NetworkHosts = { "10.10.12.2", "10.10.12.3" }
        )

    gcoip requires csgnic
    wac requires gcoip

group RVGgroup (
    SystemList = { mercury = 0, jupiter = 1 }
    Parallel = 1
    AutoStartList = { mercury, jupiter }
    )

    CVMVolDg racdata_voldg (
        CVMDiskGroup = oradatadg
        CVMActivation = sw
        )

    RVGShared racdata_rvg (
        RVG = rac1_rvg
        DiskGroup = oradatadg
        )

    requires group cvm online local firm
    racdata_rvg requires racdata_voldg

group cvm (
    SystemList = { mercury = 0, jupiter = 1 }
    AutoFailOver = 0
    Parallel = 1
    AutoStartList = { mercury, jupiter }
    )

    CSSD cssd (
        Critical = 0
```

```
        CRSHOME="/u01/app/grid/product/11.2.0/gridhome"
        )

    CFSfsckd vxfsckd (
        )

    CVMCluster cvm_clus (
        CVMClustName = rac_cluster102
        CVMNodeId = { mercury = 1, jupiter = 2 }
        CVMTransport = gab
        CVMTimeout = 200
        )

    CVMVolDg ocrvote_voldg (
        Critical = 0
        CVMDiskGroup = ocrvotedg
        CVMVolume = { ocrvol, votevol }
        CVMActivation = sw
        )

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

    PrivNIC ora_privnic (
        Critical = 0
        Device = { eth0 = 0, eth1 = 1}
        Address@galaxy = "192.168.12.1"
        Address@nebula = "192.168.12.2"
        NetMask = "255.255.240.0"
        )

    cssd requires ocrvote_voldg
    cssd requires ora_priv
    ocrvote_voldg requires cvm_clus
    vxfsckd requires cvm_clus
    cvm_clus requires cvm_vxconfigd

group oradb1_grp (
    SystemList = { mercury = 0, jupiter = 1 }
    Parallel = 1
    ClusterList = { rac_cluster101 = 0, rac_cluster102 = 1 }
    OnlineRetryInterval = 300
    ClusterFailOverPolicy = Manual
```

```
    Authority = 1
    AutoStartList = { mercury, jupiter }
    )

    CFSMount oradata_mnt (
        MountPoint = "/oradata"
        BlockDevice = "/dev/vx/dsk/oradatadg/oradatavol"
        )

    Oracle ora1 (
        Critical = 0
        Sid @mercury = vrts1
        Sid @jupiter = vrts2
        Owner = oracle
        Home = "/app/oracle/orahome"
        StartUpOpt = SRVCTLSTART
        ShutDownOpt = SRVCTLSTOP
        )

    RVGSharedPri ora_vvr_sharedpri (
        RvgResourceName = racdata_rvg
        OnlineRetryLimit = 0
        )

    requires group RVGgroup online local firm
    ora1 requires oradata_mnt
    oradata_mnt requires ora_vvr_sharedpri

group rlogowner (
    SystemList = { mercury = 0, jupiter = 1 }
    AutoStartList = { mercury, jupiter }
    OnlineRetryLimit = 2
    )

    IP logowner_ip (
        Device = eth0
        Address = "10.11.9.102"
        NetMask = "255.255.240.0"
        )

    NIC nic (
        Device = eth0
        NetworkHosts = { "10.10.12.2", "10.10.12.3" }
        )
```

```
    RVGLogowner logowner (
        RVG = rac1_rvg
        DiskGroup = oradatadg
        )

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

group VxSS (
SystemList = { north = 0, south = 1 }
Parallel = 1
OnlineRetryLimit = 3
OnlineRetryInterval = 120
)
Phantom phantom_vxss (
)
ProcessOnOnly vxatd (
IgnoreArgs = 1
PathName = "/opt/VRTSat/bin/vxatd"
)

group CMC (
SystemList = { north, south }
AutoStartList = { north, south }
OnlineRetryLimit = 3
OnlineRetryInterval = 120
)
ClusterConnectorConfig CMC_ClusterConfig (
MSAddress = "mgmtserver1.veritasexample.com"
MSPort = 14145
ClusterId = "1145613636"
ClusterType = "vcs"
ClusterPort = 14141
VCSLoggingLevel = "TAG_A"
Logging = "/opt/VRTScmccc/conf/cc_logging.properties"
ClusterConnectorVersion = "5.0.1000.0"
)
Process CMC_ClusterConnector (
PathName = "/bin/sh"
Arguments = "/opt/VRTScmccc/bin/cluster_connector.sh"
)
CMC_ClusterConnector requires CMC_ClusterConfig
```

About sample main.cf files for Storage Foundation (SF) for Sybase ASE CE

You can examine the VCS configuration file, main.cf, to verify your installation and configuration.

- The main.cf file is located in the folder /etc/VRTSvcs/conf/config.
- After installation, several sample main.cf file types can be viewed in the following directory: `/etc/VRTSagents/ha/conf/Sybase`
- All sample configurations assume that the High Availability Agent for Sybase binaries are installed on local disks and that they are managed by the operating system. These file systems must be specified in the `/etc/fstab` file.
- For the following configuration samples, please note the "cluster" definition in all of the configurations should specify UseFence=SCSI3.

Sample main.cf file examples are provided for the following SF Sybase CE configurations:

- Basic cluster configuration
  - With shared mount point on CFS for Sybase binary installation
  - With local mount point on VxFS for Sybase binary installation
- Replicating data between two clusters
  - For a primary site in a CVM VVR configuration
  - For a secondary site in a CVM VVR configuration

Sample main.cf for a basic Sybase ASE CE cluster configuration under VCS control with shared mount point on CFS for Sybase binary installation

This sample main.cf is for a single site with a basic cluster configuration with shared mount point on CFS for Sybase binary installation.

The following are the configuration details for this SF Sybase CE configuration sample main.cf:

- File name: sybasece_cfs_main.cf
- File location: /etc/VRTSagents/ha/conf/Sybase/

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

```
include "SybaseTypes.cf"

cluster clus1 (
    UserNames = { admin = HopHojOlpKppNxpJom }
    Administrators = { admin }
    HacliUserLevel = COMMANDROOT
    UseFence=SCSI3
    )

system system1 (
    )

system system2 (
    )

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

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

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

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

 requires group cvm online local firm
 sybbindg_101_sybbinvol_mnt requires sybbindg_101_voldg


 // resource dependency tree
 //
 // group binmnt
```

```
 // {
 // CFSMount sybbindg_101_sybbinvol_mnt
 //     {
 //     CVMVolDg sybbindg_101_voldg
 //     }
 // }

// cvm group for CVM and CFS specific agents.

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

    CFSfsckd vxfsckd (
        )

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

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

 cvm_clus requires cvm_vxconfigd
 vxfsckd requires cvm_clus


 // resource dependency tree
 //
 // group cvm
 // {
 // CFSfsckd vxfsckd
 //     {
 //     CVMCluster cvm_clus
 //         {
```

```
 //          CVMVxconfigd cvm_vxconfigd
 //          }
 //      }
 // }

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

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

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

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

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

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

   Process vxfend (
        PathName = "/sbin/vxfend"
```

```
        Arguments = "-m sybase -k /tmp/vcmp_socket"
        )

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

requires group binmnt online local firm
ase requires quorum_101_quorumvol_mnt
ase requires sybdata_101_sybvol_mnt
ase requires vxfend
quorum_101_quorumvol_mnt requires quorum_101_voldg
sybdata_101_sybvol_mnt requires sybdata_101_voldg


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

Sample main.cf for a basic Sybase ASE CE cluster configuration with local mount point on VxFS for Sybase binary installation

This sample main.cf is for a single site with a basic cluster configuration with local mount point on VxFS for Sybase binary installation.

Please see the section "CVM master node needs to assume the logowner role for VCS managed VVR resources" in the *Cluster Server Bundled Agents Reference Guide* for more info.

The following are the configuration details for this SF Sybase CE configuration sample main.cf:

- File name: sybasece_vxfs_main.cf
- File location: /etc/VRTSagents/ha/conf/Sybase/

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

cluster clus1 (
    UserNames = { admin = HopHojOlpKppNxpJom }
    Administrators = { admin }
    HacliUserLevel = COMMANDROOT
    UseFence=SCSI3
    )

system system1 (
    )

system system2 (
    )

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

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

```
    DiskGroup sybbindg_101_voldg (
         DiskGroup = sybbindg
         )

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

    Volume sybbindg_101_vol (
         DiskGroup = sybbindg
         Volume = sybbinvol
         )

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


 // resource dependency tree
 //
 // group binlocalmnt
 // {
 // Mount sybbindg_101_sybbinvol_mnt
 //     {
 //     Volume sybbindg_vol
 //         {
 //         DiskGroup sybbindg_101_voldg
 //         }
 //     }
 // }

// cvm group for CVM and CFS specific agents.

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

```
    CFSfsckd vxfsckd (
        )

    CVMCluster cvm_clus (
        CVMClustName = clus1
        CVMNodeId = { system1 = 0, system2 = 1 }
        CVMTransport = gab
        CVMTimeout = 200
        )

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

 cvm_clus requires cvm_vxconfigd
 vxfsckd requires cvm_clus


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

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

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

```
    OnlineRetryInterval = 120
    )

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

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

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

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

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

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

requires group binlocalmnt online local firm
ase requires quorum_101_quorumvol_mnt
ase requires sybdata_101_sybvol_mnt
```

```
ase requires vxfend
quorum_101_quorumvol_mnt requires quorum_101_voldg
sybdata_101_sybvol_mnt requires sybdata_101_voldg


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

Sample main.cf for a primary CVM VVR site

This sample main.cf is for a primary site in a CVM VVR configuration. It is one of two sample main.cfs for replicating data between two clusters.

The following are the configuration details for this SF Sybase CE configuration sample main.cf:

- File name: sybasece_cvmvvr_primary_main.cf
- File location: /etc/VRTSagents/ha/conf/Sybase

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

cluster clus1 (
        UserNames = { admin = HopHojOlpKppNxpJom }
        ClusterAddress = "10.180.88.188"
```

```
        Administrators = { admin }
        HacliUserLevel = COMMANDROOT
        UseFence=SCSI3
        )

remotecluster clus2 (
        ClusterAddress = "10.190.99.199"
        )

heartbeat Icmp (
        ClusterList = { clus2 }
        Arguments @clus2 = { "10.190.99.199" }

        )
system system1 (
        )

system system2 (
        )


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

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

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

        NIC csgnic (
                Device = Public_NIC
```

```
                )

        gcoip requires csgnic
        wac requires gcoip


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


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

        CVMVolDg sybdata_voldg (
                CVMDiskGroup = sybdata_101
                CVMActivation = sw
                )

        RVGShared sybdata_rvg (
                RVG = syb_rvg
                DiskGroup = sybdata_101
                )

        requires group binmnt online local firm
        sybdata_rvg requires sybdata_voldg



group binmnt (
        SystemList = { system1 = 0, system2 = 1 }
```

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

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

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

        CFSfsckd vxfsckd (
                )

        CVMCluster cvm_clus (
                CVMClustName = clus1
                CVMNodeId = { system1 = 0, system2 = 1 }
                CVMTransport = gab
                CVMTimeout = 200
                )

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

        cvm_clus requires cvm_vxconfigd
        vxfsckd requires cvm_clus
```

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


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

        IP logowner_ip (
                Device = public_NIC
                Address = "10.10.9.101"
                NetMask = "255.255.255.0"
                )

        NIC nic (
                Device = public_NIC
                )

        RVGLogowner rvg_logowner (
                RVG = syb_rvg
                DiskGroup = sybdata_101
                )

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


        // resource dependency tree
        //
        //      group logowner
```

```
        //      {
        //      RVGLogowner rvg_logowner
        //          {
        //          IP logowner_ip
        //              {
        //              NIC nic
        //              }
        //          }
        //      }

group sybasece (
        SystemList = { system1 = 0, system2 = 1 }
        Parallel = 1
        ClusterList = { clus1 = 0, clus2 = 1 }
        AutoStartList = { system1, system2 }
        ClusterFailOverPolicy = Manual
        Authority = 1
        OnlineRetryLimit = 3
        TriggerResStateChange = 1
        OnlineRetryInterval = 120
        )

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

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

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

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

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

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

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

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

Sample main.cf for a secondary CVM VVR site

This sample main.cf is for a secondary site in a CVM VVR configuration. It is the second of two sample main.cfs for replicating data between two clusters.

The following are the configuration details for this SF Sybase CE configuration sample main.cf:

- File name: sybasece_cvmvvr_secondary_main.cf
- File location: /etc/VRTSagents/ha/conf/Sybase

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

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

cluster clus2 (
        UserNames = { admin = HopHojOlpKppNxpJom }
        ClusterAddress = "10.190.99.199"
        Administrators = { admin }
        HacliUserLevel = COMMANDROOT
        UseFence=SCSI3
        )

remotecluster clus1 (
        ClusterAddress = "10.180.88.188"
        )

heartbeat Icmp (
        ClusterList = { clus1 }
        Arguments @clus1 = { "10.180.88.188" }
        )

system system3 (
        )

system system4 (
        )

group ClusterService (
        SystemList = { system3 = 0, system4 = 1 }
```

```
        AutoStartList = { system3, system4 }
        OnlineRetryLimit = 3
        OnlineRetryInterval = 120
        )

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

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

        NIC csgnic (
                Device = public_NIC
                )

        gcoip requires csgnic
        wac requires gcoip

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


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

```
        )

        CVMVolDg sybdata_voldg (
                CVMDiskGroup = sybdata_101
                CVMActivation = sw
                )

        RVGShared sybdata_rvg (
                RVG = syb_rvg
                DiskGroup = sybdata_101
                )

        requires group binmnt online local firm
        sybdata_rvg requires sybdata_voldg


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

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

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

```
        CFSfsckd vxfsckd (
                )

        CVMCluster cvm_clus (
                CVMClustName = clus2
                CVMNodeId = { system3 = 0, system4 = 1 }
                CVMTransport = gab
                CVMTimeout = 200
                )

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

        cvm_clus requires cvm_vxconfigd
        vxfsckd requires cvm_clus


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


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

        IP logowner_ip (
                Device = public_NIC
                Address = "10.11.9.102"
                NetMask = "255.255.255.0"
                )
```

```
        NIC nic (
                Device = public_NIC
                )

        RVGLogowner rvg_logowner (
                RVG = syb_rvg
                DiskGroup = sybdata_101
                )

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


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

group sybasece (
        SystemList = { system3 = 0, system4 = 1 }
        Parallel = 1
        ClusterList = { clus2 = 0, clus1 = 1 }
        AutoStartList = { system3, system4 }
        OnlineRetryLimit = 3
        OnlineRetryInterval = 120
        )

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

        CVMVolDg quorum_101_voldg (
                CVMDiskGroup = quorum_101
```

```
                CVMVolume = { quorumvol }
                CVMActivation = sw
                )

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

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

        RVGSharedPri syb_vvr_shpri (
                RvgResourceName = sybdata_rvg
                OnlineRetryLimit = 0
                )

        Sybase ase (
                Server @system3 = ase1
                Server @system4 = ase2
                Owner = sybase
                Home = "/sybase"
                Version = 15
                SA = sa
                Quorum_dev = "/quorum/q.dat"
                )

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

Index

Symbols

database replication 145

A

adding a Secondary
best practices 81
using command line 79
agents
Heartbeat 27
Wide-Area Heartbeat 27
associating SRL to a Primary RVG
using command line 76
asynchronous replication
setting up 86
Authority attribute
about 27
autodcm
srlprot setting 88
Automatic Synchronization
synchronizing Secondary example 96

B

backup
block-level tape 93
bandwidth limit
controlling
used for replication. *See* using command line
used to synchronize volumes. *See* using command line
Bandwidth Throttling 90
See also bandwidth limit
about 90
disabling
using command line 91
block-level tape backup 93

C

campus cluster
using Veritas™ InfoScale Operations Manager 45
commands
gcoconfig 127
vradmin 146
vxassist 138, 141
vxvol 138
configurations
global cluster 23
replicated data 21
shared storage/replicated data 21
Configuring global clusters
tasks 115
creating a Primary RVG
using command line 76
creating Replicated Data Set
using command line 74
creating RVGs
using vradmin vs. vxmake 75

D

dcm
srlprot setting
changing 88
disabling Bandwidth Throttling 91
disaster recovery
scenarios 11
use cases 11
disk groups
displaying disk group ID 82

E

examples
using command line
setting up replication 96

F

fail
srlprot setting 88
firewalls
using VVR in 81

G

gcoconfig command 127
global cluster configuration 23
global clusters
    about 114
    configuration
        using VVR 125
    configuration examples 147
    configuring VCS service groups 125
    fire drill 112
    illustration of dependencies 147
    migration 167
    operation 23
    overview 115
    replicating database volumes 147
    requirements 114
    reverse migration 167
    takeover 167
groups
    log owner 147
    RVG 147
GUIs, VVR
    about 74

H

Heartbeat agent 27
high mark
    latency
        setting. *See* using command line

I

interfaces of VVR
    about 74

L

latency low and high mark
    setting
        using command line 86
latency protection
    disabling
        using command line 87
    enabling
        using command line 87
low mark
    latency
        setting. *See* using command line

M

modes of replication
    setting 85

N

network transport protocol
    attribute
        STORAGE 89
        TCP 89
        UDP 89
    planning 81
    setting 89

O

off
    mode of replication 85
    srlprot setting 88
overflow protection
    SRL 88
override
    srlprot setting 88

P

packet size
    setting 89
Primary RVG
    creating
        using command line 76
primary site
    creating SRL volume 137
    setting up 116
    setting up replication objects 138
    VCS configuration 152, 159
protocol
    network 81

R

RAID-5 volumes and VVR 77
RDS
    creating
        using command line 76
Recovering After a Disaster 67
replicated data clusters
    about 21
    fire drill 68
    planning 37

replicated data clusters *(continued)*
third-party replication
about 69
configuring service groups 71
fire drill 71
setting up 70
replicated data configuration 21
replication
automatic synchronization 145
full synchronization with Storage Checkpoint 145
modifying VCS configuration 147
setting up primary site 116
setting up secondary site 119
starting 92
supported hardware 114
supported software 114
supported technologies 114
using VVR 137
verifying status 145
replication settings
defaults 85
latency protection 86
packet size 89
SRL overflow protection 88
resources
CVMVolDg 147
RVGSharedPri 147
RLINK
configuring 84

S

sample SF Sybase CE configuration files. *See* main.cf samples
Secondary 92
See also synchronizing Secondary
adding 79
secondary site
configuring replication 140
creating SRL volume 141
setting up 119
setting up disk groups 143
setting up RLINKs 142
service groups
VCS, for global clusters 125
set
vradmin keyword 85
setting bandwidth limit
using command line 90
setting network transport protocol 89
setting packet size 89
shared storage/replicated data configuration 21
SmartMove for VVR 96
Turn off 97
split-brain
in global clusters 31
SRL
associating to a Primary RVG
using command line 76
SRL overflow protection
explanation 88
modes of 88
srlprot
srlprot=dcm
override. *See* autodcm
starting replication 92
startrep
vradmin keyword 95
steward process
about 31
synchronizing Secondary
automatic synchronization
example 96
synchronous replication
setting up 86
synchronous=off 86
synchronous=override 86

T

tape backups
block-level 93
TCP protocol 81
Turn off
SmartMove for VVR 97

U

UDP protocol 81
usage types
VVR volumes 77

V

VCS
configuration, for database volume replication 147
configuring service groups 125
illustration of global cluster dependencies 147
VCS configuration
for replication 147

VCS Global cluster option. *See* GCO
Veritas Volume Replicator. *See* VVR
vradmin command 146
vradmin utility
    addsec 79
    createpri 76–77
    creating RVGs 75
    set 85
    startrep 95
VVR
    configuration examples 147
    configuring global clusters 125
    database replication 145
    database volume replication
        configuring VCS 147
    defining heartbeat cluster objects 128
    defining remote clusters 128
    global cluster overview 135
    primary site
        creating SRL volume 137
        setting up replication objects 138
    replication
        using automatic synchronization 145
        using full synchronization with Storage Checkpoint 145
    secondary site
        configuring replication 140
        creating SRL volume 141
        setting up disk groups 143
        setting up RLINKs 142
    setting up primary site 116
    setting up replication 73, 137
    setting up secondary site 119
    setting up third-party replication 99
    VCS configuration
        CVMoIDG resource 147
        database service group 147
        log owner group 147
        primary site 152
        RVG group 147
        RVGSharedPri resource 147
        secondary site 159
    verifying replication 145
VVR GUIs
    about 74
vxassist command 138, 141
vxedit command
    set keyword
        setting synchronous=fail 85
vxmake command
    creating RVGs 75
vxvol command 138

W

wac 26
wide-area connector 26
wide-area failover 23
    VCS agents 28
Wide-Area Heartbeat agent 27