

Veritas Storage Foundation™ and High Availability Solutions Virtualization Guide

Solaris

6.0

Veritas Storage Foundation and High Availability Solutions Virtualization Guide

The software described in this book is furnished under a license agreement and may be used only in accordance with the terms of the agreement.

Product version: 6.0

Document version: 6.0.4

Legal Notice

Copyright © 2014 Symantec Corporation. All rights reserved.

Symantec, the Symantec Logo, the Checkmark Logo, Veritas, Veritas Storage Foundation, CommandCentral, NetBackup, Enterprise Vault, and LiveUpdate are trademarks or registered trademarks of Symantec Corporation or its affiliates in the U.S. and other countries. Other names may be trademarks of their respective owners.

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

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. SYMANTEC CORPORATION SHALL NOT BE LIABLE FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES IN CONNECTION WITH THE FURNISHING, PERFORMANCE, OR USE OF THIS DOCUMENTATION. THE INFORMATION CONTAINED IN THIS DOCUMENTATION IS SUBJECT TO CHANGE WITHOUT NOTICE.

The Licensed Software and Documentation are deemed to be commercial computer software 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, "Rights in Commercial Computer Software or Commercial Computer Software Documentation", as applicable, and any successor regulations. 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.

Symantec Corporation
350 Ellis Street
Mountain View, CA 94043

<http://www.symantec.com>

Technical Support

Symantec Technical Support maintains support centers globally. Technical Support's primary role is to respond to specific queries about product features and functionality. The Technical Support group also creates content for our online Knowledge Base. The Technical Support group works collaboratively with the other functional areas within Symantec to answer your questions in a timely fashion. For example, the Technical Support group works with Product Engineering and Symantec Security Response to provide alerting services and virus definition updates.

Symantec's support offerings include the following:

- A range of support options that give you the flexibility to select the right amount of service for any size organization
- Telephone and/or Web-based support that provides rapid response and up-to-the-minute information
- Upgrade assurance that delivers software upgrades
- Global support purchased on a regional business hours or 24 hours a day, 7 days a week basis
- Premium service offerings that include Account Management Services

For information about Symantec's support offerings, you can visit our website at the following URL:

www.symantec.com/business/support/index.jsp

All support services will be delivered in accordance with your support agreement and the then-current enterprise technical support policy.

Contacting Technical Support

Customers with a current support agreement may access Technical Support information at the following URL:

www.symantec.com/business/support/contact_techsupp_static.jsp

Before contacting Technical Support, make sure you have satisfied the system requirements that are listed in your product documentation. Also, you should be at the computer on which the problem occurred, in case it is necessary to replicate the problem.

When you contact Technical Support, please have the following information available:

- Product release level
- Hardware information

- Available memory, disk space, and NIC information
- Operating system
- Version and patch level
- Network topology
- Router, gateway, and IP address information
- Problem description:
 - Error messages and log files
 - Troubleshooting that was performed before contacting Symantec
 - Recent software configuration changes and network changes

Licensing and registration

If your Symantec product requires registration or a license key, access our technical support Web page at the following URL:

www.symantec.com/business/support/

Customer service

Customer service information is available at the following URL:

www.symantec.com/business/support/

Customer Service is available to assist with non-technical questions, such as the following types of issues:

- Questions regarding product licensing or serialization
- Product registration updates, such as address or name changes
- General product information (features, language availability, local dealers)
- Latest information about product updates and upgrades
- Information about upgrade assurance and support contracts
- Information about the Symantec Buying Programs
- Advice about Symantec's technical support options
- Nontechnical presales questions
- Issues that are related to CD-ROMs or manuals

Support agreement resources

If you want to contact Symantec regarding an existing support agreement, please contact the support agreement administration team for your region as follows:

Asia-Pacific and Japan customercare_apac@symantec.com

Europe, Middle-East, and Africa semea@symantec.com

North America and Latin America [supportolutions@symantec.com](mailto:supportsolutions@symantec.com)

Documentation

Your feedback on product documentation is important to us. Send suggestions for improvements and reports on errors or omissions. Include the title and document version (located on the second page), and chapter and section titles of the text on which you are reporting. Send feedback to:

doc_feedback@symantec.com

For information regarding the latest HOWTO articles, documentation updates, or to ask a question regarding product documentation, visit the Storage and Clustering Documentation forum on Symantec Connect.

<https://www-secure.symantec.com/connect/storage-management/forums/storage-and-clustering-documentation>

About Symantec Connect

Symantec Connect is the peer-to-peer technical community site for Symantec's enterprise customers. Participants can connect and share information with other product users, including creating forum posts, articles, videos, downloads, blogs and suggesting ideas, as well as interact with Symantec product teams and Technical Support. Content is rated by the community, and members receive reward points for their contributions.

<http://www.symantec.com/connect/storage-management>

Contents

Technical Support	4	
Chapter 1	Overview of Veritas Storage Foundation and High Availability Virtualization Solutions	13
	Overview	13
	Reference documentation	14
	About Veritas Storage Foundation and High Availability Virtualization Solutions	15
Chapter 2	Storage Foundation and High Availability Solutions support for Solaris Zones	16
	About Solaris Zones	17
	About VCS support for zones	17
	Overview of how VCS works with zones	17
	About the ContainerInfo service group attribute	18
	About the ContainerOpts resource type attribute	19
	Zone-aware resources	19
	About the Mount agent	20
	About networking agents	22
	About the Zone agent	22
	About configuring failovers among physical and virtual servers	22
	Configuring VCS in zones	22
	Prerequisites for configuring VCS in zones	23
	Deciding on the zone root location	24
	Creating a zone with root on local disk	25
	Creating a zone with root on shared storage	26
	Performing the initial internal zone configuration	27
	About installing applications in a zone	27
	Configuring the service group for the application	28
	Configuring a zone resource in a failover service group with the hazonesetup utility	30
	Configuring zone resource in a parallel service group with the hazonesetup utility	32
	Modifying the service group configuration	33

Verifying the zone configuration	35
Performing maintenance tasks	35
Troubleshooting zones	35
Configuring for physical to virtual and virtual to physical failovers—a typical setup	36
Adding VxFS file systems to a non-global zone	37
Direct mount of VxFS file systems from global zone	38
Mounting a VxFS file system in a non-global zone	38
Adding a direct mount to a zone's configuration	39
Creating VxFS file systems inside non-global zones	40
Veritas Storage Foundation Cluster File System mounts	42
Creating a non-global zone root on VxFS clustered file system configuration	42
Concurrent I/O access in non-global zones	44
Veritas extension for Oracle Disk Manager	44
Exporting VxVM volumes to a non-global zone	45
VxVM devices in Solaris 10 global zones	46
Removing a VxVM volume from a non-global zone	47
Configuring Solaris non-global zones for disaster recovery	47
Software limitations of Storage Foundation support of non-global zones	49
Cannot remove a volume configured in a non-global zone	50
Administration commands are not supported in non-global zone	50
A cluster-mounted VxFS file system is not supported as the root of a non-global zone	50
QIO and CQIO are not supported	50
Package and patch installation in non-global zones	50
Package and patch removal with non-global zone configurations	50
Root volume cannot be added to non-global zones	51
Some Veritas Volume Manager operations can cause volume device names to go out of sync	51
The <code>svc:/system/vxfs/vxfsldlic</code> service is not enabled in non-global zones	51

Chapter 3	Storage Foundation and High Availability Solutions support for Solaris Projects	53
	About Solaris Projects	53
	About VCS support for Solaris projects	54
	Overview of how VCS works with Solaris projects	54
	About the ContainerInfo service group attribute	54

	About the ContainerOpts resource type attribute	55
	Project-aware resources	56
	About the Project agent	56
	Configuring VCS in Solaris projects	56
	Prerequisites for configuring VCS in projects	56
Chapter 4	Storage Foundation and High Availability Solutions support for Oracle VM Server for SPARC	58
	Terminology for Oracle VM Server for SPARC	59
	Oracle VM Server for SPARC deployment models	60
	Split Storage Foundation stack	61
	Guest-based Storage Foundation stack	61
	Benefits of deploying Storage Foundation High Availability solutions in Oracle VM server for SPARC	61
	Standardization of tools	61
	Array migration	62
	Moving storage between physical and virtual environments	62
	Boot Image Management	62
	Features	62
	Storage Foundation features	62
	Oracle VM Server for SPARC features	65
	Split Storage Foundation stack model	66
	How Storage Foundation and High Availability Solutions works in the Oracle VM Server for SPARC	67
	Veritas Storage Foundation features restrictions	68
	Guest-based Storage Foundation stack model	69
	How Storage Foundation and High Availability Solutions works in the guest domains	70
	Supported configurations with SFCFS and multiple I/O Domains	72
	Veritas Storage Foundation features restrictions	75
	System requirements	76
	Solaris operating system requirements	76
	Solaris patch requirements	77
	Hardware requirements	78
	Veritas product release notes	78
	Veritas Storage Foundation and High Availability	78
	Veritas Storage Foundation Cluster File System	78
	Product licensing	79
	Installing Storage Foundation in a Oracle VM Server for SPARC environment	79

Installing and configuring Oracle VM Server for SPARC and domains	80
Installing Storage Foundation in the control domain or guest DMP as a multi-pathing solution in an Oracle VM Server for SPARC environment	81
Installing Veritas File System in the guest domain using pkgadd	85
Verifying the configuration	86
Exporting a Veritas volume to a guest domain from the control domain	86
Provisioning storage for a guest domain	88
Provisioning Veritas Volume Manager volumes as data disks for guest domains	88
Provisioning Veritas Volume Manager volumes as boot disks for guest domains	90
Using Veritas Volume Manager snapshots for cloning logical domain boot disks	91
Configuring Oracle VM Server for SPARC guest domains for disaster recovery	96
Software limitations	99
Memory Corruption in the guest domain during SCSI commands	99
Exporting the raw volume device node fails	99
Resizing a Veritas Volume Manager volume (exported as a slice or full disk) does not dynamically reflect the new size of the volume in the guest	100
Known issues	100
Guest-based known issues	100
Split Storage Foundation stack known issues	102

Chapter 5	Veritas Cluster Server support for using CVM with multiple nodes in a Oracle VM Server for SPARC environment	104
	Clustering using Cluster Volume Manager	104
	Installing Storage Foundation on multiple nodes in a Logical Domain	105
	Reconfiguring the clustering agents for Cluster Volume Manager	105
	Cluster Volume Manager in the control domain for providing high availability	107

Chapter 6	Veritas Cluster Server: Configuring Oracle VM Server for SPARC for high availability	110
	About Veritas Cluster Server in a Oracle VM Server for SPARC environment	111
	Dynamic reconfiguration of memory and CPU of a guest domain	112
	Veritas Cluster Server prerequisites	112
	Veritas Cluster Server requirements	112
	Veritas Cluster Server limitations	112
	Veritas Cluster Server known issues	113
	About Veritas Storage Foundation Cluster File System in an Oracle VM Server for SPARC environment	114
	Veritas Storage Foundation Cluster File System limitations	114
	About Veritas Cluster Server configuration models in an Oracle VM Server for SPARC environment	115
	Veritas Cluster Server setup to fail over a logical domain on a failure	115
	Veritas Cluster Server installed in the control domain to monitor logical domains	116
	Veritas Cluster Server installed in the control domain to monitor the applications in the guest domain	117
	Veritas Cluster Server setup to fail over an application on a failure	118
	About configuring Veritas Cluster Server for Oracle VM Server for SPARC with multiple I/O domains	119
	About Alternate I/O domain	120
	Setting up the Alternate I/O domain	120
	Configuring VCS to manage a Logical Domain with multiple I/O domains	120
	Configuring Veritas Cluster Server to fail over a logical domain on a failure	120
	Configuration scenarios	122
	Creating the Veritas Cluster Server service groups for logical domains	125
	Configuring Veritas Cluster Server for application monitoring	125
	Verifying a logical domain service group failover	127
	Configuring Veritas Cluster Server to fail over an application on a failure	128
	Oracle VM Server for SPARC guest domain migration in VCS environment	128
	Overview of a warm migration	129
	Overview of a live migration	130

Prerequisites before you perform domain migration	131
Supported deployment models for Oracle VM Server for SPARC domain migration with VCS	132
Migrating Oracle VM guest when VCS is installed in the control domain that manages the guest domain	132
Migrating Oracle VM guest when VCS is installed in the control domain and single-node VCS is installed inside the guest domain to monitor applications inside the guest domain	133
Migrating Oracle VM guest when VCS cluster is installed in the guest domains to manage applications for Oracle VM Server for SPARC version 2.1 and above	135
Migrating Oracle VM guest when VCS cluster is installed in the guest domains to manage applications for Oracle VM Server for SPARC version 2.0	135
Configuring VCS to manage a Logical Domain using services from multiple I/O domains	137
A typical setup for a Logical Domain with multiple I/O services	139
Identify supported storage and network services	140
Determine the number of nodes to form VCS cluster	141
Install and configure VCS inside the control domain and alternate I/O domain	141
Configuring storage services	141
Configure storage service groups	144
Configure network service groups	147
Configure a service group to monitor services from multiple I/O domains	150
Configure the AlternateIO resource	151
Configure the service group for a Logical Domain	153
Failover scenarios	154
Recommendations while configuring VCS and Oracle VM Server for SPARC with multiple I/O domains	154
Sample VCS configuration for AlternateIO resource configured as a fail over type	155
Configuring VCS on logical domains to manage applications using services from multiple I/O domains	159

Overview of Veritas Storage Foundation and High Availability Virtualization Solutions

This chapter includes the following topics:

- [Overview](#)
- [About Veritas Storage Foundation and High Availability Virtualization Solutions](#)

Overview

This document provides information about Veritas Storage Foundation and High Availability Virtualization Solutions. Review this entire document before you install Veritas Storage Foundation and High Availability products in zones, branded zones, projects, and logical domains.

This book provides many high-level examples and information. As such, you should be a skilled user of Veritas products and knowledgeable concerning Oracle's virtualization technologies.

Each chapter in this guide presents information on using a particular Oracle virtualization technology with Veritas products. These chapters follow:

- Storage Foundation and High Availability Solutions support for Solaris Zones
- Storage Foundation and High Availability Solutions support for Solaris Projects
- Storage Foundation and High Availability Solutions support for Branded Zones

- Storage Foundation and High Availability Solutions support for Oracle VM Server for SPARC
- Using multiple nodes in an Oracle VM Server for SPARCn environment
- Configuring logical domains for high availability

Reference documentation

The following documentation provides information on installing, configuring, and using Veritas Cluster Server:

- *Veritas Cluster Server Release Notes*
- *Veritas Cluster Server Installation Guide*
- *Veritas Cluster Server Bundled Agents Reference Guide*
- *Veritas Cluster Server Agent for DB2 Installation and Configuration Guide*
- *Veritas Cluster Server Agent for Oracle Installation and Configuration Guide*
- *Veritas Cluster Server Agent for Sybase Installation and Configuration Guide*

The following documentation provides information on installing, configuring, and using Veritas Storage Foundation products:

- *Veritas Storage Foundation Release Notes*
- *Veritas Storage Foundation and High Availability Installation Guide*
- *Veritas Storage Foundation Administrator's Guide*

The following documentation provides information on installing, configuring, and using Veritas Storage Foundation Cluster File System:

- *Veritas Storage Foundation Cluster File System Release Notes*
- *Veritas Storage Foundation Cluster File System Installation Guide*
- *Veritas Storage Foundation Cluster File System Administrator's Guide*

Note: Storage Foundation Cluster File System does not support branded zones.

For Oracle VM Server for SPARC (formerly Solaris Logical Domains), Branded Zone, Projects, and Zone installation and configuration information, refer to the Oracle site: www.oracle.com.

Oracle provides regular updates and patches for Oracle VM Server for SPARC, Branded Zones, and Zone features. Contact Oracle for details.

About Veritas Storage Foundation and High Availability Virtualization Solutions

Veritas Storage Foundation and High Availability Virtualization Solutions includes support for non-global and Branded zones, Projects, and Oracle VM Server for SPARC.

Solaris Zones, also known as non-global zones is an operating system-level virtualization technology, which provides a means of virtualizing operating system services to create an isolated environment for running applications. Non-global zones function as completely isolated virtual servers with a single operating system instance.

Branded zones are an extension of the Solaris Zone infrastructure. A Branded zone is a non-native zone that allows individual zones to emulate an operating system environment other than the native environment of the global operating system.

Oracle VM Server for SPARC is a virtualization technology that enables the creation of independent virtual machine environments on the same physical system. Oracle VM Server for SPARC provides a virtualized computing environment abstracted from all physical devices, which allows you to consolidate and centrally manage your workloads on a system. The logical domains can be specified roles such as a control domain, service domain, I/O domain , and guest domain. Each domain is a full virtual machine where the operating systems can be started, stopped, and rebooted independently.

The Solaris operating system provides a facility called projects to identify workloads. The project serves as an administrative tag, which you can use to group useful and related work. You can for example create one project for a sales application and another project for a marketing application. By placing all processes related to the sales application in the sales project and the processes for the marketing application in the marketing project, you can separate and control the workloads in a way that makes sense to the business.

Storage Foundation and High Availability Solutions support for Solaris Zones

This chapter includes the following topics:

- [About Solaris Zones](#)
- [About VCS support for zones](#)
- [Configuring VCS in zones](#)
- [Adding VxFS file systems to a non-global zone](#)
- [Direct mount of VxFS file systems from global zone](#)
- [Veritas Storage Foundation Cluster File System mounts](#)
- [Creating a non-global zone root on VxFS clustered file system configuration](#)
- [Concurrent I/O access in non-global zones](#)
- [Veritas extension for Oracle Disk Manager](#)
- [Exporting VxVM volumes to a non-global zone](#)
- [Configuring Solaris non-global zones for disaster recovery](#)
- [Software limitations of Storage Foundation support of non-global zones](#)

About Solaris Zones

Solaris Zones is a software partitioning technology, which provides a means of virtualizing operating system services to create an isolated environment for running applications. This isolation prevents processes that are running in one zone from monitoring or affecting processes running in other zones.

You can configure non-global zones with a shared-IP address or an exclusive-IP address. The shared-IP zone shares a network interface with global-zone and the exclusive-IP zone does not share network interface with global-zone.

See the *System Administration Guide: Solaris Containers--Resource Management and Solaris Zones* Solaris operating environment document.

Oracle provides regular updates and patches for the Oracle Solaris Zones feature. Contact Oracle for more information.

About VCS support for zones

VCS provides application management and high availability to applications running in zones.

Note: All VxVM components run inside the global zone only. You can mount VxFS file systems as a loop back file system or as a direct mount from global zone by installing the VxFS package in the global zone. In addition, you can directly mount a VxFS file system from non-global zones.

The Zone agent is IMF-aware and uses the asynchronous monitoring framework (AMF) kernel driver for IMF notification. For more information about the Intelligent Monitoring Framework (IMF) and intelligent resource monitoring, refer to the *Veritas Cluster Server Administrator's Guide*. For more information about how to perform intelligent resource monitoring for the Zone agent, see the *Veritas Cluster Server Bundled Agents Reference Guide*.

Overview of how VCS works with zones

You can use VCS to perform the following:

- Start, stop, monitor, and fail over a non-global zone.
- Start, stop, monitor, and fail over an application that runs in a zone.

How VCS models containers

VCS and the necessary agents run in the global zone. For the applications that run in a zone, the agents can run some of their functions (entry points) inside the zone. If any resource faults, VCS fails over the service group with the zone to another node.

You can configure VCS to use Symantec Product Authentication Service to run in a secure environment. Communication from non-global zones to global zones is secure in this environment.

Installing and configuring zones in VCS environments

Install and configure the zone. Create the service group with the standard application resource types (application, storage, networking) and the Zone resource. VCS manages the zone as a resource. You then configure the service group's ContainerInfo attribute.

Configuring the ContainerInfo attribute

The service group attribute ContainerInfo specifies information about the zone. When you have configured and enabled the ContainerInfo attribute, you have enabled the zone-aware resources in that service group to work in the zone environment.

VCS defines the zone information at the level of the service group so that you do not have to define it for each resource. You need to specify a per-system value for the ContainerInfo attribute.

About the ContainerInfo service group attribute

The ContainerInfo attribute has the Name key, Type key, and Enabled key. The Name key defines the name of the container. The Type key lets you select the type of container that you plan to use. The Enabled key enables the Zone-aware resources within the service group. The ContainerInfo attribute specifies if you can use the service group with the container.

Assign the following values to the ContainerInfo attribute:

- Name
The name of the container.
- Type
The type of container. You can set this to Zone.
- Enabled

Specify the value as 0, if you want to disable the container. Specify the value as 1, if you want to enable the container. Specify the value as 2, to enable physical to virtual and virtual to physical failovers. When the value is 2, the Zone resource mimics a non-existent entity.

You can set a per-system value for this attribute.

About the ContainerOpts resource type attribute

The ContainerOpts resource attribute is pre-set for Zone-aware resource types. It determines the following:

- Whether the zone-aware resource can run in the zone.
- Whether the container information that is defined in the service group's ContainerInfo attribute is passed to the resource.

These values are only effective when you configure the ContainerInfo service group attribute.

attribute's keys follow:

The ContainerOpts resource type attribute's definitions for Zone-aware types contain the following values:

- RunInContainer (RIC)
When the value of the RunInContainer key is 1, the agent function (entry point) for that resource runs inside of the local container.
When the value of the RunInContainer key is 0, the agent function (entry point) for that resource runs outside the local container (in the global environment).
A limitation for the RunInContainer value is that only script agent functions (entry points) can run inside a container.
- PassCInfo (PCI)
When the value of the PassCInfo key is 1, the agent function receives the container information that is defined in the service group's ContainerInfo attribute.
An example use of this value is to pass the name of the container to the agent.

Zone-aware resources

[Table 2-1](#) lists the ContainerOpts attributes default values for resource types. Zone-aware resources have predefined values for the ContainerOpts attribute.

Note: Symantec recommends that you do not modify the value of the ContainerOpts attribute, with the exception of the Mount agent.

See [“About the Mount agent”](#) on page 20.

See “[About networking agents](#)” on page 22.

Table 2-1 ContainerOpts attribute default values for applications and resource types

Resource type	RunInContainer	PassCInfo
Apache	1	0
Application	1	0
ASMIInst	1	0
ASMDG	1	0
Db2udb	1	0
NIC	0	1
IP	0	1
IPMultiNIC	0	1
IPMultiNICB	0	1
Process	1	0
Zone	0	1
Oracle	1	0
Netlsnr	1	0
Sybase	1	0
SybaseBk	1	0

About the Mount agent

You may need to modify the ContainerOpts values for the Mount resource in certain situations.

In certain situations where the block device is not exported to zone, you can make the file system available inside local zone. Mount the block device on the directory that has a path that includes the zone root from global zone, for example:

```
BlockDevice = /dev/vx/dsk/dg/voll  
MountPoint = /zones/zone-test/root/mntpt
```

Where /zones/zone-test is the zone root of the local zone.

Bringing a Mount resource online in the zone

The Mount resource is brought online in the global zone by default (RunInContainer = 0). If you want to bring a mount resource online inside the non-global zone, perform the following:

- Export the block device to the zone through zone configuration. Ensure that the raw volume is used appropriately to prevent the possibility of data corruption.
- Modify the ContainerInfo attribute for the service group and set values for the Name, Type, and Enabled keys.

```
# hagrps -modify service_group ContainerInfo Name zone_name\  
Type Zone Enabled 1
```

- Override the ContainerOpts attribute at the resource level.
- Set the value of the RunInContainer key to 1, for example:

```
# hares -override Mountres ContainerOpts  
  
# hares -modify Mountres ContainerOpts\  
RunInContainer 1 PassCInfo 0
```

For information on overriding resource type static attributes, refer to the *Veritas Cluster Server Administrator's Guide*.

Selecting the proper attribute values for a Mount resource for NFS mounts

For NFS mounts, you must mount in the non-global zone.

- Modify the ContainerInfo attribute for the service group and set values for Name, Type and Enabled keys.
- Override the ContainerOpts attribute at the resource level.
- Set the value of the RunInContainer key to 1.
Set the RIC value to 1. When you set RIC=1, specify the value of the MountPoint attribute relative to the zone root, for example:

```
BlockDevice = abc:/fs1  
MountPoint = /mnt1
```

The file system is mounted on /zone root/mnt1.

About networking agents

Enable the attribute `ExclusiveIPZone` for resources of type IP and NIC when these resources are configured to manage the IP and the NIC inside an exclusive-IP zone. This attribute is disabled by default. The IP agent and the NIC agent assumes the native zone (shared-IP) by default.

VCS brings resources online in the global zone by default.

If you want to bring these resources online inside the exclusive-IP zone, perform the following tasks:

- Make sure that the resource is in a service group that has valid `ContainerInfo` attribute value configured.
- Set the value of the `ExclusiveIPZone` attribute to 1.

Note: The exclusive-IP zone supports the IP and NIC networking agents. For more information about these agents, see the *Veritas Cluster Server Bundled Agents Reference Guide*.

About the Zone agent

The Zone agent monitors zones, brings them online, and takes them offline. For more information about the agent, see the *Veritas Cluster Server Bundled Agents Reference Guide*.

Use `hazonesetup` utility to create user account with group administrative privileges. The `DeleteVCSZoneUser` attribute of zone resource controls removing the user account when the zone resource is taken offline. For more information, see the *Veritas Cluster Server Bundled Agents Reference Guide*.

About configuring failovers among physical and virtual servers

You can configure VCS to fail over from a physical system to a virtual system and vice versa. A physical to virtual failover gives an N + N architecture in an N + 1 environment. For example, several physical servers with applications can fail over to containers on another physical server.

See [“Configuring for physical to virtual and virtual to physical failovers—a typical setup”](#) on page 36.

Configuring VCS in zones

Configuring VCS in zones involves the following tasks:

First	Review the prerequisites. See “Prerequisites for configuring VCS in zones” on page 23.
Second	Decide on the location of the zone root, which is either on local storage or shared storage. See “Deciding on the zone root location” on page 24.
Third	Install the application in the zone. See “About installing applications in a zone” on page 27.
Fourth	Create the application service group and configure its resources. See “Configuring the service group for the application” on page 28.

Prerequisites for configuring VCS in zones

Review the following prerequisites for configuring VCS in zones:

- VCS supports CFS, UFS, ZFS, and VxFS mounts for the zone root.
- File system mounts must meet one of the following two conditions:
 - Use a loopback file system. All mounts that the application uses must be part of the zone configuration and must be configured in the service group. For example, you can create a zone, `z-ora`, and define the file system containing the application’s data to have the mount point as `/oradata`. When you create the zone, you can define a path in the global zone. An example is `/export/home/oradata`, which the mount directory in the non-global zone maps to. The `MountPoint` attribute of the `Mount` resource for the application is set to `/export/home/oradata`. Confirm that `/export/home/oradata` maps to `/oradata` with the command `zonecfg -z zone_name info`. You can also look into the zone configuration file `/etc/zones/zone_name.xml`. The `Zone` resource depends on the `Mount` resource.
 - Use a direct mount file system. All file system mount points that the application uses that run in a zone must be set relative to the zone’s root. For example, if the Oracle application uses `/oradata`, and you create the zone with the `zonpath` as `/z_ora`, then the mount must be `/z_ora/root/oradata`. The `MountPoint` attribute of the `Mount` resource must be set to this path. The `Mount` resource depends on the `Zone` resource.

Note: The `Mount` agent does not support mounting VxFS file systems directly inside non-global zones.

Using custom agents in zones

If you use custom agents, review the following information for their use in zones:

- If you use custom agents to monitor the applications that run in zones, make sure that the agents use script-based entry points. VCS does not support running C++ entry points inside a zone.
- If you want the custom agent to monitor an application in the zone, for the custom agent type, set the following values for the ContainerOpts attribute:
RunInContainer = 1 and the PassCInfo = 0.
- If you don't want the custom agent to monitor an application in the zone, for the custom agent type, set the following values for the ContainerOpts attribute:
RunInContainer = 0 and the PassCInfo = 0.
- Two main use cases exist where you might want to use a RunInContainer = 0 and PassCInfo = 1, descriptions of these follow.
 - The first is the Zone agent's use of these values. The Zone agent's entry points cannot run inside of the non-global zone but the agent itself manages the zone. RunInContainer requires a value of 0 because the agent must run in the global zone. PassCInfo has a value of 1 because the Zone agent requires the name of the container from the ContainerInfo service group attribute.
 - The second case is how the IP agent uses RunInContainer and PassCInfo. The IP agent's entry points must run outside of the non-global zone because a shared-IP zone may cause the networking stack to not completely run in the non-global zone. You cannot perform an `ifconfig` command and then plumb the IP from inside of a non-global zone. When you run the `ifconfig` command in the global zone with the zone option - it plumbs the IP and makes it available to the zone that you specify. The need for the container's name comes from the use of this command, even though it cannot run in the container. This is applicable to all networking agents.

Deciding on the zone root location

Each zone has its own section of the file system hierarchy in the zone root directory. Processes that run in the zone can access files only within the zone root.

You can set the zone root in the following two ways:

- Zone root on local storage
In this configuration, you must configure and install a zone on each node in the cluster.
- Zone root on shared storage

In this configuration, configure and install a zone in shared storage from one system and duplicate the configuration on each node in the cluster. Setting the zone root on shared storage means you need to install the non-global zone on shared storage from one system only. The zone root can fail over to the other systems. To do this, the system software, including the patches, must be identical on each system during the existence of the zone.

Creating a zone with root on local disk

Create a zone root on the local disk on each node in the cluster. The file system for application data is on a shared device and is either the loopback type or the direct mount type. For a direct mount file system, run the mount command from the global zone with the mount point specified as the complete path that starts with the zone root. For a loopback file system, add it into the zone's configuration before you boot the zone.

To create a zone root on local disks on each node in the cluster

- 1 Configure the zone with the `zonectfg` command.

```
zonectfg -z newzone  
zonectfg:newzone> create
```

- 2 Set the `zonepath` parameter to specify a location for the zone root.

```
zonectfg:newzone> set zonepath=/export/home/newzone
```

- 3 If your application data resides on a loopback mount file system, create the loopback file system in the zone.

- 4 Exit the `zonectfg` configuration.

```
zonectfg> exit
```

- 5 Create the zone root directory.

```
mkdir zonepath
```

- 6 Set permissions for the zone root directory.

```
chmod 700 zonepath
```

- 7 Install the non-global zone.

```
zoneadm -z newzone install
```

- 8 Repeat step 1 to step 7 on each system in the service group's SystemList.
- 9 If the application data is on a loopback file system, mount the file system containing the application's data on shared storage.
- 10 Boot the zone.

```
zoneadm -z newzone boot
```

- 11 If the application data is on a direct mount file system, mount the file system from the global zone with the complete path that starts with the zone root.

Creating a zone with root on shared storage

Create a zone with root which points to the shared disk's location on each node in the cluster. The file system for application data is on a shared device and is either the loopback type or the direct mount type. For a direct mount file system, run the mount command from the global zone with the mount point specified as the complete path that starts with the zone root. For a loopback file system, add it into the zone's configuration before you boot the zone.

To create a zone root on shared disks on each node in the cluster

- 1 Create a file system on shared storage for the zone root. The file system that is to contain the zone root may be in the same disk group as the file system that contains the application data.
- 2 Configure the zone with the `zonecfg` command.

```
zonecfg -z newzone  
zonecfg:newzone> create
```

- 3 Set the `zonpath` parameter to specify a location for the zone root.

```
zonecfg:newzone> set zonpath=/export/home/newzone
```

- 4 If your application data resides on a loopback mount file system, create the loopback file system in the zone.
- 5 Exit the `zonecfg` configuration.

```
zonecfg> exit
```

- 6 Create the zone root directory.

```
mkdir zonpath
```

- 7 Set permissions for the zone root directory.

```
chmod 700 zonepath
```

- 8 Repeat step 2 to step 7 on each system in the service group's SystemList.
- 9 Mount the file system that contains the shared storage on one of the systems that share the storage to the directory specified in *zonepath*.
- 10 Run the following command to install the zone on the system where the zone path is mounted.

```
zoneadm -z newzone install
```

- 11 If the application data is on a loopback file system, mount the file system containing the application's data on shared storage.
- 12 Boot the zone.

```
zoneadm -z newzone boot
```

- 13 If the application data is on a direct mount file system, mount the file system from the global zone with the complete path that starts with the zone root.

Performing the initial internal zone configuration

When a zone is booted for the first time after installation, the zone is in an unconfigured state. The zone does not have an internal configuration for naming services. Its locale and time zone have not been set, and various other configuration tasks have not been performed. You need to perform the initial internal zone configuration after zone installation.

You can perform the internal zone configuration in the following ways:

- `sysidcfg` tool
- Zone console login

For more details refer to Oracle documentation about "Performing the Initial Internal Zone Configuration" section in the *System Administration Guide: Oracle Solaris Containers-Resource Management and Oracle Solaris Zones* guide.

About installing applications in a zone

Perform the following tasks to install the application in a zone:

- If you have created zones locally on each node in the cluster, install the application identically in all zones on all nodes. If you are installing an application

that is supported by a Veritas High Availability agent, see the installation and configuration guide for the agent.

- Install the agent packages on the global zone and the currently existing zones. Installs the agents in future zones when they are installed.
- You must define all the mount points that the application uses that are configured in the zone in the service group's configuration.

Configuring the service group for the application

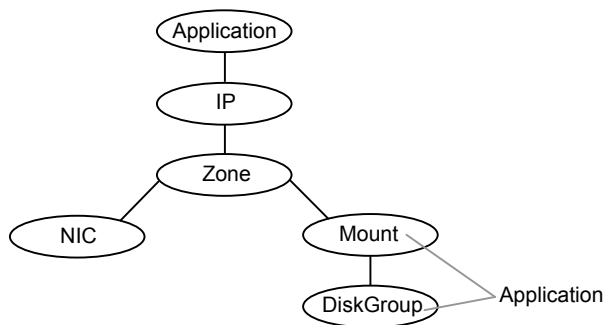
You need to configure the application service group and the required resource dependencies. The following diagrams illustrates different examples of resource dependencies. In one case the zone root is set up on local storage. In the other, zone root is set up on shared storage.

Resource dependency diagrams: zone root on local disks

The following resource dependency diagrams show zone configurations on local disks configured for loopback and direct mounted file systems.

[Figure 2-1](#) depicts the dependency diagram when the zone root is set up on local storage with the loopback file system for the application. You can replace the Mount resource with the CFSMount resource and the DiskGroup resources with the CVMVolDg resource in the following diagram. In this configuration, decide if you want the service group to be a parallel service group. If so, you may need to localize certain attributes for resources in the service group. For example, you have to change the IP resource's Address attribute for each node.

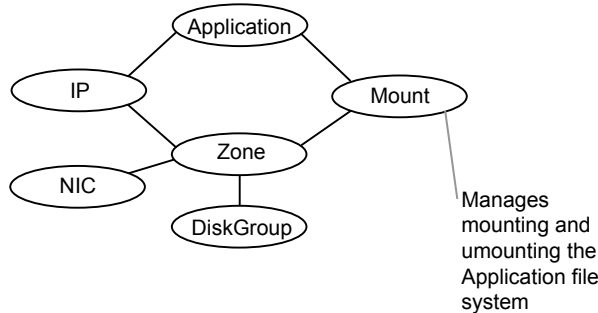
Figure 2-1 Zone root on local disks with loopback file system



[Figure 2-2](#) depicts the dependency diagram when the zone root is set up on local storage with a direct mount file system for the application. You can replace the Mount resource with the CFSMount resource and the DiskGroup resources with the CVMVolDg resource in the following diagram. In this configuration, decide if

you want the service group to be a parallel service group. If so, you may need to localize certain attributes for resources in the service group. For example, you have to change the IP resource's Address attribute for each node.

Figure 2-2 Zone root on local disks with direct mount file system



Resource dependency diagrams: zone root on shared disks

The following resource dependency diagrams show zone configurations on shared disks configured for loopback and direct mounted file systems.

Figure 2-3 depicts the dependency diagram when a zone root is set up on shared storage with the loopback file system. You can replace the Mount resource with the CFSMount resource and the DiskGroup resource with the CVMVoIDg resource in the following diagram. In this configuration, decide if you want the service group to be a parallel service group. If so, you may need to localize certain attributes for resources in the service group. For example, you have to change the IP resource's Address attribute for each node.

Figure 2-3 Zone root on shared storage with loopback file system

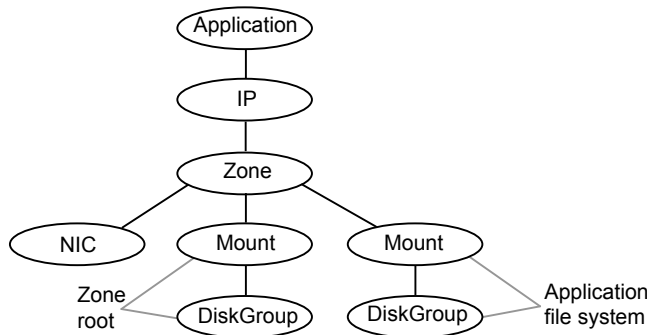
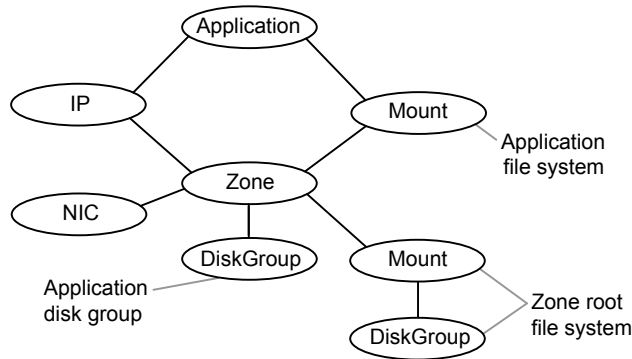


Figure 2-4 depicts the dependency diagram when a zone root is set up on shared storage with the direct mount file system for the application. You can replace the

Mount resource with the CFSSMount resource and the DiskGroup resource with the CVMVolDg resource in the following diagram. In this configuration, decide if you want the service group to be a parallel service group. If so, you may need to localize certain attributes for resources in the service group. For example, you have to change the IP resource's Address attribute for each node.

Figure 2-4 Zone root on shared storage a direct mounted file system



Use the following principles when you create the service group:

- Set the MountPoint attribute of the Mount resource to the mount path.
- If the application requires an IP address, configure the IP resource in the service group.
- If the zone root file system is on shared storage, you can configure separate mounts for the zone and the application (as shown in the illustration), but you can configure the same disk group for both.

Configuring a zone resource in a failover service group with the hazonesetup utility

The hazonesetup utility helps you configure zone under VCS. This section covers typical scenarios based on where the zone root is located.

Two typical setups for zone configuration in a failover scenario follow:

- Zone root on local storage
To configure a zone under VCS control using the hazonesetup utility when the zone root is on local storage
- Zone root on shared storage
To configure a zone under VCS control using the hazonesetup utility when the zone root is on shared storage

Consider an example in a two-node cluster (sysA and sysB). Zone local-zone is configured on both the nodes.

To configure a zone under VCS control using the hazonesetup utility when the zone root is on local storage

- 1 Boot the local zone on first node outside VCS.

```
sysA# zoneadm -z local-zone boot
```

- 2 Run hazonesetup with correct arguments on the first node. This adds failover zone service group and zone resource in VCS configuration.

```
sysA# hazonesetup -g zone_grp -r zone_res -z local-zone\  
-p password -a -s sysA,sysB
```

- 3 Switch the zone service group to next node in the cluster.

```
sysA# hagr -switch zone_grp -sys sysB
```

- 4 Run the hazonesetup utility with correct arguments on the node. The hazonesetup utility detects that the zone service group and zone resource are already present in VCS configuration and update the configuration accordingly.

```
sysB# hazonesetup -g zone_grp -r zone_res -z local-zone\  
-p password -a -s sysA,sysB
```

- 5 Repeat step 3 and step 4 for all the remaining nodes in the cluster.

To configure a zone under VCS control using the hazonesetup utility when the zone root is on shared storage

- 1 Configure a failover service group with required storage resources (DiskGroup, Volume, Mount, etc.) to mount the zone root on the node. Set the required dependency between storage resources (DiskGroup->Volume->Mount).

```
sysA# hagr -add zone_grp  
sysA# hares -add zone_dg DiskGroup zone_grp  
sysA# hares -add zone_vol Volume zone_grp  
sysA# hares -add zone_mnt Mount zone_grp  
sysA# hares -link zone_mnt zone_vol  
sysA# hares -link zone_vol zone_dg
```

- 2 Bring the service group online on first node. This mounts the zone root on first node.

```
sysA# hagr -online zone_grp -sys sysA
```

- 3 Boot the local zone on first node outside VCS.

```
sysA# zoneadm -z local-zone boot
```

- 4 Run hazonesetup with correct arguments on the first node. Use the service group configured in step 1. This adds the zone resource to VCS configuration.

```
sysB# hazonesetup -g zone_grp -r zone_res -z local-zone \  
-p password -a -s sysA,sysB
```

- 5 Set the proper dependency between the Zone resource and other storage resources. The Zone resource should depend on storage resource (Mount->Zone).

```
sysA# hares -link zone_res zone_mnt
```

- 6 Switch the service group to next node in the cluster.

```
sysA# hagrps -switch zone_grp -sys sysB
```

- 7 Run the hazonesetup utility with correct arguments on the node. The hazonesetup utility detects that the service group and the zone resource are already present in VCS configuration and update the configuration accordingly.

```
sysB# hazonesetup -g zone_grp -r zone_res -z local-zone \  
-p password -a -s sysA,sysB
```

- 8 Repeat step 6 and step 7 for all the remaining nodes in the cluster

Configuring zone resource in a parallel service group with the hazonesetup utility

The hazonesetup utility helps you configure a zone under VCS. This section covers typical scenarios based on the location of the zone root.

In the case of a zone resource in parallel service group, the zone root can be on local or shared storage that the node owns.

Consider an example in a two-node cluster (sysA and sysB). Zone local-zone is configured on both the nodes.

To configure a zone under VCS control using hazonesetup utility when the zone root is on local storage

- 1 Boot the local zone on all the nodes outside VCS.

```
sysA# zoneadm -z local-zone boot sysB# zoneadm -z local-zone boot
```

- 2 Run the hazonesetup utility with correct arguments on all the nodes successively.

```
sysA# hazonesetup -g zone_grp -r zone_res -z local-zone\  
-p password -a -l -s sysA,sysB  
sysB# hazonesetup -g zone_grp -r zone_res -z local-zone\  
-p password -a -l -s sysA,sysB
```

- 3 Running hazonesetup on first node adds parallel zone service group and zone resource in VCS configuration. Running hazonesetup on other nodes detect that the zone service group and zone resource are already present in VCS configuration and update the configuration accordingly.

Note: Run hazonesetup on all the nodes in the cluster that have a zone running on that node. This is required as hazonesetup runs the `halogin` command inside the local zone that enables password-less communication between local zone and global zone.

You can use the same user for multiple zones across systems. Specify the same user name while running hazonesetup utility for different zones on different systems. When you do not specify a user name while running hazonesetup utility, the utility creates a user with the default user name `z_resname_hostname` for a non-secure cluster and `z_resname_clustername` for a secure cluster.

Modifying the service group configuration

Perform the following procedure to modify a service group's configuration.

To create the configuration to manage a zone

- 1 Run the `hazonesetup` script to set up the zone configuration.

```
# hazonesetup [-t] -g sg_name -r res_name -z zone_name\
[-u] user_name -p password [-a] [-l] -s systems
```

Where the values are:

<code>-t</code>	Updates the password for the VCS zone user.
<code>-g <i>sg_name</i></code>	Name of the zone service group to be created in VCS configuration.
<code>-r <i>res_name</i></code>	Name of the zone resource to be created in VCS configuration.
<code>-z <i>zone_name</i></code>	Name of the zone that is configured on the system.
<code>-u <i>user_name</i></code>	Name of the VCS user used for password less communication between the local zone and the global zone. If no username is specified the default username is used.
<code>-p <i>password</i></code>	Password for the VCS user used for password less communication. If Symantec Authentication Service is enabled, the password should be at least six characters long.
<code>-a</code>	Populate <code>AutoStartList</code> for the group.
<code>-l</code>	Configure a parallel service group. If you do not specify the <code>-l</code> option, a failover service group is created by default.
<code>-s <i>systems</i></code>	A comma separated list of systems where the zone service group need to be configured, for example: <code>sys1,sys2,sys3</code> .

If the application service group does not exist, the script creates a service group.

The script adds a resource of type `Zone` to the application service group. The script logs in to the zone and runs the `halogin` command. It also creates a user account with group administrative privileges to enable password less communication between global zone and local zone for VCS.

- 2 Modify the resource dependencies to reflect your zone configuration. See the resource dependency diagrams for more information.
See [“Configuring the service group for the application”](#) on page 28.
- 3 Save the service group configuration and bring the service group online.

Verifying the zone configuration

Run the `hazoneverify` command to verify the zone configuration.

The command verifies the following requirements:

- The systems hosting the service group have the required operating system to run zones.
- The service group does not have more than one resource of type Zone.
- The dependencies of the Zone resource are correct.

To verify the zone configuration

- 1 If you use custom agents make sure the resource type is added to the `APP_TYPES` or `SYS_TYPES` environment variable.

See “Using custom agents in zones” on page 24.

- 2 Run the `hazoneverify` command to verify the zone configuration.

```
# hazoneverify servicegroup_name
```

Performing maintenance tasks

Perform the following maintenance tasks as necessary:

- Make sure that the zone configuration files are consistent on all the nodes at all times. The file is located at `/etc/zones/zone_name.xml`.
- When you add a patch or upgrade the operating system on one node, make sure to upgrade the software on all nodes.
- Make sure that the application configuration is identical on all nodes. If you update the application configuration on one node, apply the same updates to all nodes.

Troubleshooting zones

Use following information to troubleshoot VCS and zones:

- VCS HA commands do not work.

Recommended actions:

- Verify the VCS packages are installed.
- Run the `halogin` command from the zone.

For more information on the `halogin` command, refer to the *Veritas Cluster Server Administrator's Guide*.

- Verify your VCS credentials. Make sure the password is not changed.
- Verify the VxSS certificate is not expired.
- Resource does not come online in the zone.

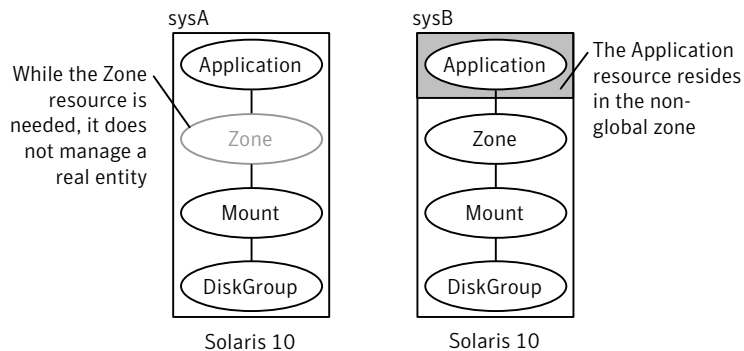
Recommended actions:

- Verify VCS and the agent packages are installed correctly.
- Verify the application is installed in the zone.
- Verify the configuration definition of the agent.

Configuring for physical to virtual and virtual to physical failovers—a typical setup

In this configuration, you have two physical nodes. One node runs Solaris 10 without zones configured (sysA) and another node runs Solaris 10 with zones configured (sysB).

Figure 2-5 An application service group that can fail over into a zone and back



In the main.cf configuration file, define the container name, type of container, and whether it is enabled or not in the service group definition.

```
ContainerInfo@sysA = {Name = Z1, Type = Zone, Enabled = 2}
ContainerInfo@sysB = {Name = Z1, Type = Zone, Enabled = 1}
```

On sysA, set the value of Enabled to 2 to ignore zones so that the application runs on the physical system. When the service group fails over to sysB, the application runs inside the zone after the failover because Enabled is set to 1 on sysB. The application can likewise fail over to sysA from sysB.

When ContainerInfo::Enabled is set to 2, the Zone agent reports resource state based on state of the corresponding group. Currently, the Zone agent check the

state of the group on all the systems. If group is online on any of the nodes in the cluster Zone resource is reported as online. This triggers concurrency violation issue. The Zone agent should look for group state on local system.

Note: IMF must be disabled on the node where Enable is set to 2 (sysA in this example). To disable IMF, set the mode to 0.

Adding VxFS file systems to a non-global zone

VxFS file systems that were previously created in the global zone can be made available in the non-global zone using a loopback file system mount. This functionality is especially useful when the sole purpose of making the file system available in the non-global zone is to share access of this file system with one or more non-global zones. For example, if a configuration file is available in a particular file system and this configuration file is required by the non-global zone, then the file system can be shared with the non-global zone using a loopback file system mount.

The following commands share access of file system `/mnt1` as a loopback file system mount with an existing non-global zone `myzone`:

```
# zonecfg -z myzone
zonecfg:myzone> add fs
zonecfg:myzone:fs> set dir=/mnt1
zonecfg:myzone:fs> set special=/mnt1
zonecfg:myzone:fs> set type=lofs
zonecfg:myzone:fs> end
zonecfg:myzone> verify
zonecfg:myzone> commit
zonecfg:myzone> exit
```

The value of `dir` is a directory in the non-global zone. The value of `special` is a directory in the global zone to be mounted in the non-global zone.

Caution: Sharing file systems with non-global zones through a loopback file system mount makes the file system available for simultaneous access from all the non-global zones. This method should be used only when you want shared read-only access to the file system.

The loopback file system mount mode of sharing file systems in the non-global zones is supported in Veritas File System 4.1 and later.

Direct mount of VxFS file systems from global zone

Exclusive access of a VxFS file system can be delegated to a non-global zone by direct mounting the file system in the non-global zone. Using direct mounts limits the visibility of and access to the file system to only the non-global zone that has direct mounted this file system.

See [“Mounting a VxFS file system in a non-global zone”](#) on page 38.

See [“Adding a direct mount to a zone's configuration”](#) on page 39.

See [“Creating VxFS file systems inside non-global zones”](#) on page 40.

Mounting a VxFS file system in a non-global zone

To direct mount a VxFS file system in a non-global zone, the directory to mount must be in the non-global zone and the mount must take place from the global zone. The following procedure mounts the directory `dirmnt` in the non-global zone `newzone` with a mount path of `/zonedir/newzone/root/dirmnt`.

Note: VxFS entries in the global zone `/etc/vfstab` file for non-global zone direct mounts are not supported, as the non-global zone may not yet be booted at the time of `/etc/vfstab` execution.

Once a file system has been delegated to a non-global zone through a direct mount, the mount point will be visible in the global zone through the `mount` command, but not through the `df` command.

To direct mount a VxFS file system in a non-global zone

- 1 Log in to the zone and make the mount point:

```
global# zlogin newzone
newzone# mkdir dirmnt
newzone# exit
```

- 2 Mount the file system from the global zone:

- Non-cluster file system:

```
global# mount -F vxfs /dev/vx/dsk/dg/voll /zonedir/zone1\
/root/dirmnt
```

- Cluster file system:

```
global# mount -F vxfs -o cluster /dev/vx/dsk/dg/vol1 \  
/zonedir/zone1/root/dirmnt
```

- 3 Log in to the non-global zone and ensure that the file system is mounted:

```
global# zlogin newzone  
newzone# df | grep dirmnt  
/dirmnt (/dirmnt):142911566 blocks 17863944 files
```

Adding a direct mount to a zone's configuration

A non-global zone can also be configured to have a VxFS file system direct mount automatically when the zone boots using `zonecfg`. The `fsck` command is run, before the file system is mounted. If the `fsck` command fails, the zone fails to boot.

To add a direct mount to a zone's configuration

- 1 Check the status and halt the zone:

```
global# zoneadm list -cv  
ID NAME STATUS PATH BRAND IP  
0 global running / native shared  
2 myzone running /zones/myzone native shared  
global# zoneadm -z myzone halt
```

- 2 Add the file system to zones configuration.

```
global# zonecfg -z myzone  
zonecfg:myzone> add fs  
zonecfg:myzone:fs> set dir=/dirmnt  
zonecfg:myzone:fs> set special=/dev/vx/dsk/dg_name/vol_name  
zonecfg:myzone:fs> set raw=/dev/vx/rdisk/dg_name/vol_name  
zonecfg:myzone:fs> set type=vxfs  
zonecfg:myzone:fs> end  
zonecfg:myzone> verify  
zonecfg:myzone> commit  
zonecfg:myzone> exit
```

3 Boot the zone:

```
global# zoneadm -z myzone boot
```

4 Log in to the non-global zone and ensure that the file system is mounted:

```
global# zlogin myzone  
myzone# df | grep dirmnt  
/dirmnt (/dirmnt):142911566 blocks 17863944 files
```

Creating VxFS file systems inside non-global zones

You can create a VxFS file system inside of a non-global zone.

To create the VxFS file system inside of a non-global zone

1 Check the zone status and halt the zone:

```
global# zoneadm list -cv
ID NAME STATUS PATH BRAND IP
0 global running / native shared
2 myzone running /zones/myzone native shared
global# zoneadm -z myzone halt
```

2 Add devices to the zone's configuration:

```
global# zonecfg -z myzone
zonecfg:myzone:fs> add device
zonecfg:myzone:fs> set match=/dev/vxportal
zonecfg:myzone:fs> end
zonecfg:myzone:fs> add device
zonecfg:myzone:fs> set match=/dev/fdd
zonecfg:myzone:fs> end
zonecfg:myzone:fs> add device
zonecfg:myzone:fs> set match=/dev/vx/rdisk/dg_name/vol_name
zonecfg:myzone:fs> end
zonecfg:myzone:fs> add device
zonecfg:myzone:fs> set match=/dev/vx/dsk/dg_name/vol_name
zonecfg:myzone:fs> end
zonecfg:myzone:fs> add fs
zonecfg:myzone:fs> set dir=/etc/vx/licenses/lic
zonecfg:myzone:fs> set special=/etc/vx/licenses/lic
zonecfg:myzone:fs> set type=lofs
zonecfg:myzone:fs> end
zonecfg:myzone> verify
zonecfg:myzone> commit
zonecfg:myzone> exit
```

3 Boot the zone:

```
global# zoneadm -z myzone boot
```

Veritas Storage Foundation Cluster File System mounts

Veritas Storage Foundation Cluster File System (SFCFS) provides support for the same file system to be made available from multiple nodes that have been grouped together as a cluster. VxFS supports the sharing or delegation of cluster-mounted file systems in the non-global zone.

Note: Creating a non-global zone root on VxFS and CFS is not supported.

See [“Adding VxFS file systems to a non-global zone”](#) on page 37.

See [“Direct mount of VxFS file systems from global zone”](#) on page 38.

The requirements to support SFCFS in non-global zones are similar to those of SFCFS support in global zones. Some key points are as follows:

- Both lofs and direct mount are supported; Symantec recommends direct mount
- The device must be visible and shared on all nodes
- The mount point path must be the same on all nodes
- The zone name and configuration should be the same on all nodes

Support for SFCFS in a non-global zone is available in Veritas File System 5.0 Maintenance Pack 1 and later.

Creating a non-global zone root on VxFS clustered file system configuration

Symantec provides the following steps to make each system in a cluster aware of a non-global zone that resides on a VxFS clustered file system:

To create a non-global zone root on a VxFS clustered file system configuration

- 1 On the first system, configure, install, boot, halt the non-global zone and remove its zone root.
 - Configure the non-global zone:

```
# zonecfg -z zonename
```

where *zonename* is the name of the zone.
Refer to Oracle documentation for more information.
 - Install the non-global zone:

```
# zoneadm -z zonename install
```

- Boot the non-global zone:

```
# zoneadm -z zonename boot
```

- Halt the non-global zone:

```
# zoneadm -z zonename halt
```

- Remove its zone root:

```
# rm -rf zonepath/zonename
```

where *zonepath* is the location where you created the zone.

- 2 Ensure that the non-global zone is in the installed state:

```
# zoneadm list -cv
      ID NAME      STATUS      PATH                      BRAND      IP
      0 global    running    /                        native    shared
      - zonename  installed  /zonepath/zonename  native    shared
```

If the non-global zone is not in the installed state, then you might have not configured the non-global zone correctly. Refer to Sun Microsystems' documentation.

- 3 Repeat steps 1 and 2 for each system in the cluster, except the last system.
- 4 On the last system in the cluster, configure, install, and boot the non-global zone.

- Configure the non-global zone:

```
# zonecfg -z zonename
```

Refer to Oracle documentation for more information.

- Install the non-global zone:

```
# zoneadm -z zonename install
```

- Boot the non-global zone:

```
# zoneadm -z zonename boot
```

- On the systems where the zone is halted in step 1, boot the non-global zone:

```
# zoneadm -z zonename boot
```

Concurrent I/O access in non-global zones

Concurrent I/O allows multiple processes to read from or write to the same file without blocking other `read(2)` or `write(2)` calls. POSIX semantics requires `read` and `write` calls to be serialized on a file with other `read` and `write` calls. Concurrent I/O is generally used by applications that require high performance for accessing data and do not perform overlapping writes to the same file.

Veritas Storage Foundation supports concurrent I/O for applications running in the non-global zones as well. This implies that a process in a non-global zone can access the file concurrently with other processes in the global or non-global zone. The application or running threads are responsible for coordinating the write activities to the same file when using Concurrent I/O.

An application must perform the following activities to enable the concurrent I/O advisory on a file:

```
fd=open(filename, oflag)
ioctl(fd, VX_SETCACHE, VX_CONCURRENT)
write(fd, buff, numofbytes)
```

Veritas extension for Oracle Disk Manager

The Veritas extension for Oracle Disk Manager (ODM) is specifically designed for Oracle9i or later to enhance file management and disk I/O throughput. The features of Oracle Disk Manager are best suited for databases that reside in a Veritas File System. Oracle Disk Manager allows Oracle9i or later users to improve database throughput for I/O intensive workloads with special I/O optimization.

The Veritas extension for Oracle Disk Manager is supported in non-global zones. To run Oracle 10g on a non-global zone and use Oracle Disk Manager, the Oracle software version must be 10.1.0.3 or higher.

Care must be taken when installing and removing packages and patches when working with the `VRTSodm` package, for more information refer to the following:

- See [“Package and patch installation in non-global zones”](#) on page 50.
- See [“Package and patch removal with non-global zone configurations”](#) on page 50.

The following procedure enables Oracle Disk Manager file access from non-global zones with Veritas File System.

To enable Oracle Disk Manager file access from non-global zones with Veritas File System

- 1 Make global zone licenses visible to the non-global zone by exporting the `/etc/vx/licenses/lic` directory to the non-global zone as a lofs:

```
global# zonecfg -z myzone
zonecfg:myzone> add fs
zonecfg:myzone:fs> set dir=/etc/vx/licenses/lic
zonecfg:myzone:fs> set special=/etc/vx/licenses/lic
zonecfg:myzone:fs> set type=lofs
zonecfg:myzone:fs> end
zonecfg:myzone> commit
```

- 2 Create the `/dev/odm` directory in the non-global zonepath from the global zone:

```
global# mkdir -p myzonepath/dev/odm
```

- 3 Log in to the non-global zone and mount `/dev/odm` either manually or use the startup script. Use one of the following:

- `global# zlogin myzone`
`myzone# mount -F odm /dev/odm /dev/odm`

Or:

- `global# zlogin myzone`
`myzone# /lib/svc/method/odm start`

Exporting VxVM volumes to a non-global zone

A volume device node can be exported for use in non-global zone using the `zonecfg` command. The following procedure makes a volume `vol1` available in the non-global zone `myzone`.

Caution: Exporting raw volumes to non-global zones has implicit security risks. It is possible for the zone administrator to create malformed file systems that could later panic the system when a mount is attempted. Directly writing to raw volumes, exported to non-global zones, and using utilities such as `dd` can lead to data corruption in certain scenarios.

To export VxVM volumes to a non-global zone**1 Create a volume `voll` in the global zone:**

```
global# ls -l /dev/vx/rdisk/rootdg/voll
crw----- 1 root    root    301, 102000 Jun  3
12:54 /dev/vx/rdisk/rootdg/vollcrw----- 1 root  sys    301, 10200
0 Jun  3 12:54 /devices/pseudo/vxio@0:rootdg,voll,102000,raw
```

2 Add the volume device `voll` to the non-global zone `myzone`:

```
global# zonecfg -z myzone
zonecfg:myzone> add device
zonecfg:myzone:device> set match=/dev/vx/rdisk/mydg/voll
zonecfg:myzone:device> end
zonecfg:myzone> commit
```

3 Ensure that the devices will be seen in the non-global zone:

```
global# zoneadm -z myzone halt
global# zoneadm -z myzone boot
```

4 Verify that `/myzone/dev/vx` contains the raw volume node and that the non-global zone can perform I/O to the raw volume node.

The exported device can now be used for performing I/O or for creating file systems.

VxVM devices in Solaris 10 global zones

On the Solaris 10 operating environment, there are two physical nodes corresponding to each volume node entry, `/devices` and `/dev`, respectively, with the same major and minor number. The physical nodes appear as follows:

```
/devices raw volume node : /devices/pseudo/vxio@0:
  dname,volname,minor_number,raw
/devices block volume node : /devices/pseudo/vxio@0:
  dname,volname,minor_number,blk
/dev raw volume node : /dev/vx/rdisk/dname/volumename
/dev block volume node : /dev/vx/dsk/dname/volumename
```

The following example provides sample values in `/devices`:

```
ls -l /devices/pseudo/vxio*voll*
brw----- 1 root    sys    302, 66000 Mar 25
```

```
17:21 /devices/pseudo/vxio@0:mydg,voll,66000,blk
crw----- 1 root    sys      302, 66000 Mar 25
17:21 /devices/pseudo/vxio@0:mydg,voll,66000,raw
```

The following example provides sample values in `/dev`:

```
ls -l /dev/vx/*dsk/mydg/voll
brw----- 1 root    root     302, 66000 Mar 25 17:21 /dev/vx/dsk/mydg/voll
crw----- 1 root    root     302, 66000 Mar 25 17:21 /dev/vx/rdsk/mydg/voll
```

Removing a VxVM volume from a non-global zone

The following procedure removes a VxVM volume from a non-global zone.

To remove a VxVM volume from a non-global zone

- ◆ Remove the volume device `voll` from the non-global zone `myzone`:

```
global# zonecfg -z myzone
zonecfg:myzone> remove device match=/dev/vx/rdsk/rootdg/voll
zonecfg:myzone> end
zonecfg:myzone> commit
```

Configuring Solaris non-global zones for disaster recovery

Solaris Zones can be configured for disaster recovery by replicating the zone root using replication methods like Hitachi TrueCopy, EMC SRDF, Veritas Volume Replicator, and so on. The network configuration for the Zone in the primary site may not be effective in the secondary site if the two sites are in different IP subnets. Hence, you need to make these additional configuration changes to the Zone resource.

To configure the non-global zone for disaster recovery, configure VCS on both the sites in the global zones with GCO option.

Refer to the *Veritas Cluster Server Administrator's Guide* for more information about global clusters, their configuration, and their use.

To set up the non-global zone for disaster recovery

- 1 On the primary site, create the non-global Zone and configure its network parameters.
 - Create the non-global zone on the primary site using the `zonecfg` command.

- Add the network adapters to the non-global zone's configuration if the zone is configured as an exclusive IP zone. Assign an IP address to the network adapter along with the Netmask and Gateway.
 - After the zone boots, set up the other network-related information such as the HostName, DNS Servers, DNS Domain, and DNS Search Path in the appropriate files (/etc/hostname, /etc/resolve.conf).
- 2 On the primary site, shut down the zone.
 - 3 Use replication-specific commands to failover the replication to the secondary site from primary site.
 - 4 Repeat step 1 on the secondary site.
 - 5 Perform step 6, step 7, step 8, and step 9 on the primary cluster and secondary clusters.
 - 6 Create a VCS service group with a VCS Zone resource for the non-global zone. Configure the DROpts association attribute on the Zone resource with the following keys and site-specific values for each: HostName, DNSServers, DNSSearchPath, and DNSDomain. If the non-global zone is an exclusive IP zone in this site, configure the following keys in the DROpts association attribute on the Zone resource: Device (network adapter name), IPAddress, Netmask, and Gateway.
 - 7 Add the appropriate Mount resources and DiskGroup resources for the File System and DiskGroup on which the non-global zone's zoneroot resides. Add a resource dependency from the Zone resource to the Mount resource and another dependency from the Mount resource to the Diskgroup resource.
 - 8 Add one of the following VCS replication resources to the service group for managing the replication.
 - A hardware replication agent
Examples of these agents include SRDF for EMC SRDF, HTC for Hitachi TrueCopy, MirrorView for EMC MirrorView, etc. Refer to the appropriate VCS replication agent guide for configuring the replication resource.
 - The VVRPrimary agent

For VVR-based replication, add the RVGPrimary resource to the service group.

Refer to the following manuals for more information:

- For information about configuring VVR-related resources, see the *Veritas Storage Foundation and High Availability Solutions Replication Administrator's Guide*.

- For information about the VVR-related agents, see the *Veritas Cluster Server Bundled Agents Reference Guide*.

9 Add a dependency from the DiskGroup resource to the replication resource.

Figure 2-6 Sample resource dependency diagram for hardware replication based non-global zones

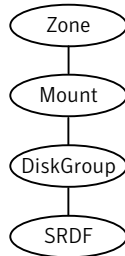
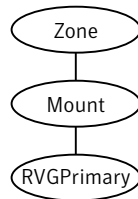


Figure 2-7 Sample resource dependency diagram for VVR replication-based non-global zones



When the resource is online in a site, the replication resource makes sure of the following:

- The underlying replicated devices are in primary mode, and the underlying storage and eventually the zone root goes into read-write mode.
- The remote devices are in secondary mode.

Thus, when the Zone resource goes online the resource modifies the appropriate files inside the non-global zone to apply the disaster recovery-related parameters to the non-global zone.

Software limitations of Storage Foundation support of non-global zones

This section describes the software limitations of Storage Foundation support of non-global zones in this release.

Cannot remove a volume configured in a non-global zone

After a volume node is configured in a Solaris non-global zone, the node cannot be removed in certain Solaris releases. Solaris patch 122660-10 resolves this issue.

Administration commands are not supported in non-global zone

All administrative tasks, such as resizing a volume, adding a volume to a volume set, and file system reorganization, are supported only in the global zone. Consequently, administrative commands, such as `fsadm`, `fsvoladm`, and `vxassist`, and administrative `ioctl`s are not supported in the non-global zone by both VxFS and VxVM.

A cluster-mounted VxFS file system is not supported as the root of a non-global zone

The root of a non-global zone cannot currently be on a cluster-mounted VxFS file system.

QIO and CQIO are not supported

Quick I/O and Cached Quick I/O are not supported by VxFS in non-global zones.

Package and patch installation in non-global zones

To ensure that the package and patch updates applied to Veritas products in the global zone are also propagated to the non-global zone, ensure that the non-global zones are in a bootable state (installed/running). For example, to update the software installed in a system, the disk group containing the root file system of the non-global zone should be imported and the file system should be mounted at the time that you run `patchadd`, `pkgadd`, or the CPI installation scripts.

Package and patch removal with non-global zone configurations

If non-global zones are part of the system configuration and the `VRTSodm` package is installed, ensure that `/dev/odm` is unmounted in each non-global zone prior to `VRTSodm` package removal or product uninstallation. This ensures there are no non-global zone `odm` module references that might prevent the global zone from unloading the `odm` module.

You can unmount `/dev/odm` in the non-global zone with the following commands:

```
global# zlogin myzone
myzone# umount /dev/odm
```

Root volume cannot be added to non-global zones

The root volume cannot be added to non-global zones.

Some Veritas Volume Manager operations can cause volume device names to go out of sync

If a volume is exported to a non-global zone, some Veritas Volume Manager operations can cause the global and non-global volume device names to go out of sync, which can create data corruption. This is because the Solaris operating environment zone support is not aware of the `devfsadm(1M)` command, and thus the zone configuration is not updated with any changes to the `/dev` or `/devices` namespaces.

The following operations can cause device names to go out of sync:

- Removing a volume
- Importing a disk group
- Deporting a disk group
- Renaming a disk group or volume
- Reminoring a disk group
- Restarting `vxconfigd` or resetting the kernel

To prevent device names from to going out of sync, if a volume is exported to a non-global zone and an operation that can cause device names to go out of sync occurs on that volume, remove the volume from the zone configuration using the `zonecfg` command and reboot the zone using the `zoneadm` command.

See the `zonecfg(1M)` and `zoneadm(1M)` manual pages.

Note: This issue applies to any Solaris device for which the `/dev` or `/devices` device node is changed and has been configured in the non-global zone before the change.

The `svc:/system/vxfs/vxfsldlic` service is not enabled in non-global zones

When the `VRTSvxfs` package is installed on a host where the zone is configured, the `svc:/system/vxfs/vxfsldlic` service does not start automatically for the configured or installed non-global zones. As a result, the `svc:/milestone/multi-user` service may enter the maintenance mode, after the `VRTSvxfs` package is installed.

Workaround:The following commands are used to enable the `svc:/system/vxfs/vxfsldlic` service in the non-global zones:

```
svcadm enable svc:/system/vxfs/vxfsldlic
```

```
svcadm clear svc:/milestone/multi-user
```

```
svcadm enable svc:/milestone/multi-user
```

Storage Foundation and High Availability Solutions support for Solaris Projects

This chapter includes the following topics:

- [About Solaris Projects](#)
- [About VCS support for Solaris projects](#)
- [Configuring VCS in Solaris projects](#)

About Solaris Projects

The Solaris operating system provides the projects facility to identify workloads. The project serves as an administrative tag that you use to group related work in a useful manner. You can create one project for a sales application and another project for a marketing application. By placing all processes related to the sales application in the sales project and the processes for the marketing application in the marketing project, you can separate and control the workloads in a way that makes sense to the business.

A user that is a member of more than one project can run processes in multiple projects at the same time. This multiple project approach makes it possible for users to participate in several workloads simultaneously. All processes that a process starts inherits the project of the parent process. As a result, switching to a new project in a startup script runs all child processes in the new project.

For more information, refer to the Solaris operating environment document *System Administration Guide: Solaris Containers--Resource Management and Solaris Zones*

About VCS support for Solaris projects

VCS provides application management and high availability to applications that run in Solaris projects.

Overview of how VCS works with Solaris projects

You can use VCS to perform the following:

- Start, stop, monitor, and fail over a Solaris project.
- Start, stop, monitor, and fail over an application that runs inside a Solaris project.

How VCS models containers

VCS and necessary agents run in the global zone. For the applications that run in a Solaris project, the agents can run online entry point inside the project. If any resource faults, VCS fails over the service group.

Installing and configuring projects in a VCS environment

Install and configure the project. Create the service group with the standard application resource types (application, storage, networking) and the Project resource. VCS manages the project as a resource. You then configure the service group's ContainerInfo attribute.

Configuring the ContainerInfo attribute

The service group attribute ContainerInfo specifies information about the Solaris project. When you have configured and enabled the ContainerInfo attribute, you have enabled the project-aware resources in that service group to work in the project environment. VCS defines the project information at the level of the service group so that you do not have to define it for each resource. You need to specify a per-system value for the ContainerInfo attribute.

About the ContainerInfo service group attribute

The ContainerInfo attribute has the Name key, Type key, and Enabled key. The Name key defines the name of the container. The Type key lets you select the type of container that you plan to use. The Enabled key enables the Project-aware resources within the service group. The ContainerInfo attribute specifies if you can use the service group with the container.

Assign the following values to the ContainerInfo attribute:

- Name

The name of the container.

- **Type**
The type of container. You can set this to Project.
- **Enabled**
Specify the value as 0, if you want to disable the container. Specify the value as 1, if you want to enable the container. Specify the value as 2, to enable physical to virtual and virtual to physical failovers. When the value is 2, the Project resource mimics a non-existent entity.

You can set a per-system value for this attribute.

About the ContainerOpts resource type attribute

The ContainerOpts resource attribute is pre-set for project-aware resource types. It determines the following:

- Whether the project-aware resource can run in the project.
- Whether the container information that is defined in the service group's ContainerInfo attribute is passed to the resource.

These values are only effective when you configure the ContainerInfo service group attribute.

attribute's keys follow:

The ContainerOpts resource type attribute's definitions for project-aware types contain the following values:

- **RunInContainer**
When the value of the RunInContainer key is 1, only online agent function (entry point) for that resource runs inside of the project.
When the value of the RunInContainer key is 0, the agent function (entry point) for that resource runs outside the local container (in the global environment).
A limitation for the RunInContainer value is that only script agent functions (entry points) can run inside a container.
- **PassCInfo**
When the value of the PassCInfo key is 1, the agent function receives the container information that is defined in the service group's ContainerInfo attribute. An example use of this value is to pass the name of the container to the agent.

Project-aware resources

At present Process, Application and Oracle resources are project aware. If a service group, which is configured for Solaris Project contains resources other than Process, Application, or Oracle, Symantec recommends you set `RunInContainer` to 0.

About the Project agent

The Project agent monitors Solaris Project, brings them online, and takes them offline.

For more information about the agent, see the *Veritas Cluster Server Bundled Agents Reference Guide*.

Configuring VCS in Solaris projects

Configuring VCS in projects involves the following tasks:

- | | |
|--------|--|
| First | Review the prerequisites.
See "Prerequisites for configuring VCS in projects" on page 56. |
| Second | Decide on the location of the project root, which is either on local storage or shared storage. |
| Third | Install the application in the project. |
| Fourth | Create the application service group and configure its resources. |

Prerequisites for configuring VCS in projects

Review the following prerequisites for configuring VCS in projects: VCS support only Process, Application and Oracle agent.

Using custom agents in projects

If you use custom agents, review the following information for their use in projects:

- If you use custom agents to monitor the applications that run in project, make sure that the agents use script-based entry points. VCS does not support running C++ entry points inside a project.
- If you want the custom agent to monitor an application in the project, for the custom agent type, set the following values for the `ContainerOpts` attribute: `RunInContainer = 1` and the `PassCInfo = 0`.

- If you don't want the custom agent to monitor an application in the project, for the custom agent type, set the following values for the ContainerOpts attribute: RunInContainer = 0 and the PassCInfo= 0.

For an example, refer to the *Veritas Cluster Server Bundled Agents Reference Guide*.

Storage Foundation and High Availability Solutions support for Oracle VM Server for SPARC

This chapter includes the following topics:

- [Terminology for Oracle VM Server for SPARC](#)
- [Oracle VM Server for SPARC deployment models](#)
- [Benefits of deploying Storage Foundation High Availability solutions in Oracle VM server for SPARC](#)
- [Features](#)
- [Split Storage Foundation stack model](#)
- [Guest-based Storage Foundation stack model](#)
- [System requirements](#)
- [Veritas product release notes](#)
- [Product licensing](#)
- [Installing Storage Foundation in a Oracle VM Server for SPARC environment](#)
- [Exporting a Veritas volume to a guest domain from the control domain](#)
- [Provisioning storage for a guest domain](#)

- [Using Veritas Volume Manager snapshots for cloning logical domain boot disks](#)
- [Configuring Oracle VM Server for SPARC guest domains for disaster recovery](#)
- [Software limitations](#)
- [Known issues](#)

Terminology for Oracle VM Server for SPARC

The following terminology is used in configuring the Veritas software in Oracle VM Server for SPARC.

Table 4-1 Lists the terminology for Oracle VM Server for SPARC

Term	Definition
Logical domain	Logical domain or virtual machine with its own operating system, resources, and identity within the same physical host.
Hypervisor	A firmware layer that provides a set of hardware-specific support functions to the operating systems running inside logical domains through a stable interface, known as the sun4v architecture. The hypervisor is interposed between the operating system and the hardware layer.
Logical Domains Manager	Software that communicates with the Hypervisor and logical domains to sequence changes, such as the addition and removal of resources or creation of a logical domain. The Logical Domains Manager provides an administrative interface and keeps track of the mapping between the physical and virtual devices in a system.
Control domain	The primary domain which provides a configuration platform to the system for the setup and teardown of logical domains. Executes Logical Domains Manager software to govern logical domain creation and assignment of physical resources.

Table 4-1 Lists the terminology for Oracle VM Server for SPARC (*continued*)

Term	Definition
I/O domain	Controls direct, physical access to input/output devices, such as PCI Express cards, storage units, and network devices. The default I/O domain is the control domain.
Guest domain	Utilizes virtual devices offered by control and I/O domains and operates under the management of the control domain.
Virtual devices	Physical system hardware, including CPU, memory, and I/O devices that are abstracted by the Hypervisor and presented to logical domains within the platform.
Logical Domains Channel (LDC)	A logical domain channel is a point-to-point, full-duplex link created by the Hypervisor. LDCs provide a data path between virtual devices and guest domains and establish virtual networks between logical domains.
Virtual Disk Client	A Solaris kernel module in the guest domain which controls the virtual disks visible to that guest, providing standard device interfaces to applications.
Virtual Disk Server	A Solaris kernel module in the control domain which is responsible for exporting various backend devices as virtual disks to guest domains.

Oracle VM Server for SPARC deployment models

Oracle VM Server for SPARC is a virtualization technology on the Solaris SPARC platform that enables the creation of independent virtual machine environments on the same physical system. This allows you to consolidate and centrally manage your workloads on one system.

Veritas Storage Foundation supports logical domains in the following two deployments models:

- [Split Storage Foundation stack](#)
- [Guest-based Storage Foundation stack](#)

Veritas Storage Foundation Cluster File System (SFCFS) is supported only in the guest-based Storage Foundation stack.

See [“About Veritas Cluster Server configuration models in an Oracle VM Server for SPARC environment”](#) on page 115.

Split Storage Foundation stack

The support for this model was introduced in 5.0 MP1 release and this model continues to be supported in this release.

See [“Split Storage Foundation stack model”](#) on page 66.

See [“Veritas Cluster Server setup to fail over a logical domain on a failure”](#) on page 115.

Guest-based Storage Foundation stack

The support for this model was introduced in 5.0 MP3 release.

See [“Guest-based Storage Foundation stack model”](#) on page 69.

See [“Features”](#) on page 62.

See [“Known issues”](#) on page 100.

Note: The SFCFS stack can be installed across multiple I/O domains within or across physical servers.

See [“Veritas Cluster Server limitations”](#) on page 112.

Benefits of deploying Storage Foundation High Availability solutions in Oracle VM server for SPARC

There are several benefits to a Oracle VM Server for SPARC environment.

Standardization of tools

Independent of how an operating system is hosted, consistent storage management tools save an administrator time and reduce the complexity of the environment.

Storage Foundation in the control domain provides the same command set, storage namespace, and environment as in a non-virtual environment.

Array migration

Data migration for Storage Foundation can be executed in a central location, migrating all storage from an array utilized by Storage Foundation managed hosts.

This powerful, centralized data migration functionality is available with Storage Foundation Manager 1.1 and later.

Moving storage between physical and virtual environments

Storage Foundation can make painful migrations of data from physical to virtual environments easier and safer to execute.

With Storage Foundation, there is no need to copy any data from source to destination, but rather the administrator reassigns the same storage or a copy of the storage for a test migration, to the virtual environment.

Boot Image Management

Using Storage Foundation in this environment the user can utilize features such as instant snapshots to contain boot images and manage them from a central location in the control domain.

Features

This section describes some of the features in Oracle VM Server for SPARC using the products in the Veritas Storage Foundation and High Availability Solutions:

- [Storage Foundation features](#)
- [Oracle VM Server for SPARC features](#)

Storage Foundation features

The following features apply for Storage Foundation:

- [The vxloadm utility enables access to a file system contained in a VxVM volume from the Control Domain](#)
 - [Examples of using the vxloadm utility](#)
- [The vxformat utility automatically relabels the virtual disk backed by a VxVM volume in the guest domain](#)
 - [Example of using the vxformat utility](#)

The vxloadm utility enables access to a file system contained in a VxVM volume from the Control Domain

The `vxloadm` utility lets you access a file system contained inside a VxVM volume from outside the guest domain, that is from the Control Domain. This is done by mapping all the partitions contained within that volume using the `vxlo` driver. The partitions can then be mounted if they contain valid file systems.

To use this vxloadm utility

- 1 Load the `vxlo` driver in memory:

```
# cd /kernel/drv/sparcv9
# add_drv -m '* 0640 root sys' vxlo
# modload vxlo
```

- 2 Check if the driver is loaded in memory:

```
# modinfo | grep vxlo
226 7b3ec000 3870 306 1 vxlo (Veritas Loopback Driver 0.1)
```

- 3 Run the `vxloadm` utility:

```
# /etc/vx/bin/vxloadm
```

- 4 You can now use the utility.

See [“Examples of using the vxloadm utility”](#) on page 63.

- 5 Symantec recommends once you are done using the `vxloadm` utility to unload the `vxlo` driver:

```
# rem_drv vxlo
# modinfo | grep vxlo
226 7b3ec000 3870 306 1 vxlo (Veritas Loopback Driver 0.1)
# modunload -i 226
```

where `226` is the module ID from the `modinfo | grep vxlo` command.

Examples of using the vxloadm utility

Use the `vxloadm addall` command to create device(s) mapping the various partition(s) contained in a VxVM volume. For example:

```
# /etc/vx/bin/vxloadm addall vol1 /dev/vx/dsk/testdg/vol1
```

This creates a device node entry for every slice or partition contained within the volume in the `/dev/vxlo/dsk/` and `/dev/vxlo/rdisk/` directories.

```
# ls -l /dev/vxlo/dsk/
lrwxrwxrwx 1 root root 46 Sep 25 14:04 volls0
-> ../../../../devices/pseudo/vxlo@0:volls0,1,blk
lrwxrwxrwx 1 root root 46 Sep 25 14:04 volls3
-> ../../../../devices/pseudo/vxlo@0:volls3,2,blk

# ls -l /dev/vxlo/rdsk/
lrwxrwxrwx 1 root root 46 Sep 25 14:04 volls0
-> ../../../../devices/pseudo/vxlo@0:volls0,1,raw
lrwxrwxrwx 1 root root 46 Sep 25 14:04 volls3
-> ../../../../devices/pseudo/vxlo@0:volls3,2,raw
```

Use the `vxloadm get` command to display the list of all currently mapped partition(s) created using the `vxloadm` utility. For example:

```
# /etc/vx/bin/vxloadm get
VxVM  INFO V-5-1-0      NAME      FILENAME
MOUNT  OFFSET  C/H/S
VxVM  INFO V-5-1-15260  volls0   /dev/vx/dsk/testdg/voll
6180      6787/1/618
VxVM  INFO V-5-1-15260  volls3   /dev/vx/dsk/testdg/voll
4326000  50902/1/618
```

Use the appropriate file system commands to access the file system(s). For example:

```
# fstyp /dev/vxlo/rdsk/volls0
ufs
# mount -F ufs /dev/vxlo/dsk/volls0 /mnt
```

Use the `vxloadm delete` to remove the partition mappings of a volume. For example:

```
# /etc/vx/bin/vxloadm delete volls0
# /etc/vx/bin/vxloadm delete volls3
```

Note: This `vxloadm` utility should only be used on volumes that are currently not in use or held open by a guest domain.

The `vxformat` utility automatically relabels the virtual disk backed by a VxVM volume in the guest domain

The `vxformat` utility provides the user the ability to automatically relabel a virtual disk backed by a VxVM volume. This utility is meant to be executed from inside the guest domain only.

The `vxformat` utility is particularly useful when a VxVM volume with existing partitions is grown in size and you need to access the new size from the guest domain.

Requirements for relabeling to succeed

- The relabel succeeds only- if it can find a new cylinder size that is aligned with the start and size of each of the existing partitions.
In case the `vxformat` command cannot find such cylinder size, it displays the following descriptive message and then exits:

```
Cannot re-label device /dev/rdsk/c0t1d2s2 since we failed to
find new cylinder size that's aligned with all the existing partitions
```

- The relabel succeeds only - if the available blocks is greater than the last sector of each and every non-s2 partition.
Otherwise, the `vxformat` command displays the following message and then exits:

```
Cannot re-label device /dev/rdsk/c0d2s2 since the last sector of a
non-s2 partition is greater than the available blocks
```

Example of using the vxformat utility

Use the `vxformat` command to relabel the virtual disk. For example:

```
# /etc/vx/bin/vxformat c0d1s2
rawpath: /dev/rdsk/c0d1s2
Old disk capacity: 2097000 blocks
New disk capacity: 4194000 blocks
Device /dev/rdsk/c0d1s2 has been successfully re-labeled.
Please use prtvtoc(1) to obtain the latest partition table information
```

If the underlying device size has not changed, the `vxformat` command displays the following message without changing the label. For example:

```
# /etc/vx/bin/vxformat c0d1s2
Old disk capacity: 2343678 blocks
New disk capacity: 2343678 blocks
size of device /dev/rdsk/c0d2s2 is unchanged
```

Oracle VM Server for SPARC features

SFHA solutions support the following features for Oracle VM server for SPARC:

- [Guest domain migration](#)

- [Virtual I/O dynamic reconfiguration](#)

Guest domain migration

The guest domain migration feature is supported for cold, warm, and live migrations by Storage Foundation with both deployment models:

- Split Storage Foundation stack
- Guest-based Storage Foundation stack

Virtual I/O dynamic reconfiguration

The virtual I/O dynamic reconfiguration feature is supported with both deployment models:

- Split Storage Foundation stack
- Guest-based Storage Foundation stack

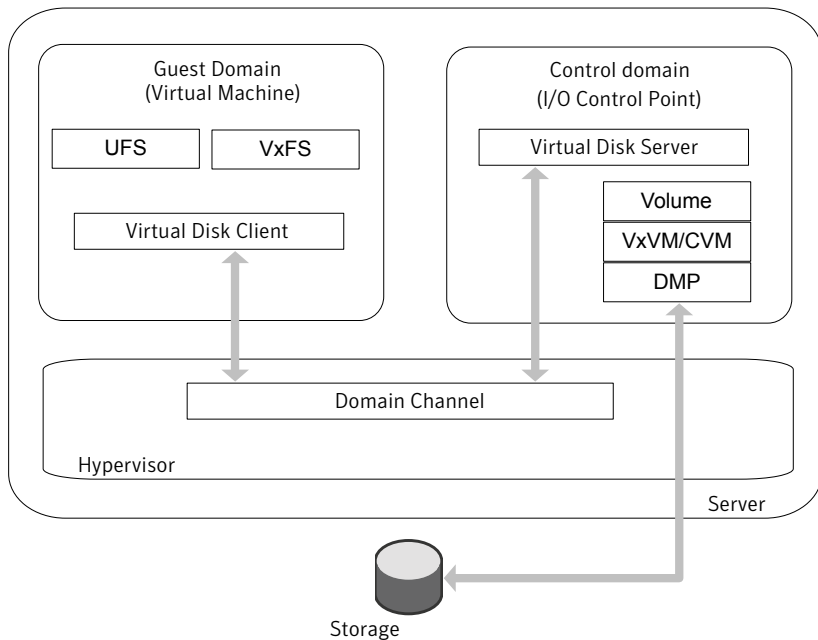
Note: For resizing a volume exported as a single slice: The new size should be visible dynamically in the guest immediately.

For resizing a volume exported as a full disk: Even though the new size is visible dynamically in the guest, the new space allocated in the volume cannot be utilized unless the label in the vdisk has been adjusted to reflect the new sectors. This adjustment of the label needs to be done carefully.

Split Storage Foundation stack model

[Figure 4-1](#) illustrates the split Storage Foundation stack model with Oracle VM Server for SPARC logical domains.

Figure 4-1 Split Storage Foundation stack model with logical domains



How Storage Foundation and High Availability Solutions works in the Oracle VM Server for SPARC

Veritas Storage Foundation and High Availability Solutions supports Oracle VM Server for SPARC logical domains in both single-node, multiple-node, and multiple-node high availability configurations.

Figure 4-1 illustrates the recommended placement of Storage Foundation stack component products in this model.

Following indicates the recommended placement of Storage Foundation stack component products:

- For a single node configuration, Veritas Volume Manager (VxVM) including DMP is placed in the control domain, and Veritas File System (VxFS) is placed in the guest domain.
- For clustered nodes, Cluster Volume Manager (CVM) is placed in the control domain, and VxFS is placed in the guest domain.

See [“Clustering using Cluster Volume Manager”](#) on page 104.

See [“Installing Storage Foundation on multiple nodes in a Logical Domain”](#) on page 105.

See [“Cluster Volume Manager in the control domain for providing high availability”](#) on page 107.

- For clustered nodes in a highly available environment, install Veritas Cluster Server (VCS) in the control domain.
See [“About Veritas Cluster Server in a Oracle VM Server for SPARC environment”](#) on page 111.
See [“About Veritas Cluster Server configuration models in an Oracle VM Server for SPARC environment”](#) on page 115.
See [“Configuring Veritas Cluster Server to fail over a logical domain on a failure”](#) on page 120.
See [“Configuring Veritas Cluster Server to fail over an application on a failure”](#) on page 128.
- VxFS drivers in the guest domain cannot currently interact with the VxVM drivers in the control domain. This renders some features, which require direct VxVM-VxFS coordination, unusable in such a configuration.
See [“Veritas Storage Foundation features restrictions”](#) on page 68.

Note: VxFS can also be placed in the control domain, but there will be no coordination between the two VxFS instances in the guest and the control domain.

Veritas Storage Foundation features restrictions

The following Veritas Storage Foundation software features are restricted in the split Storage Foundation stack model:

- VxVM volume snapshots — Due to the inability of VxFS in the guest domain to coordinate with VxVM in the control domain, taking a data consistent snapshot of a VxVM volume containing a VxFS file system requires shutting down the application and unmounting the file system before taking the snapshot.
- Resizing VxVM volumes and any type of file system on top of the volume with `vxresize` — Resizing any type of file system on the guest whose underlying device is backed by a VxVM volume in the control domain, requires resizing the VxVM volume and the file system in the guest individually.

If you are growing a VxFS file system in the guest whose underlying device is backed by a VxVM volume requires you to first grow the volume in the control domain using the `vxassist` command, and then the file system in the guest domain using the `fsadm` command.

Shrinking a VxFS file system, on the other hand, requires you to first shrink the file system in the guest domain using the `fsadm` command, and then the volume

in the control domain using the `vxassist` command. Using the `vxassist` command requires you to use the `-f` option of the command, as in the following example.

```
# vxassist -g [diskgroup] -f shrinkto volume length
```

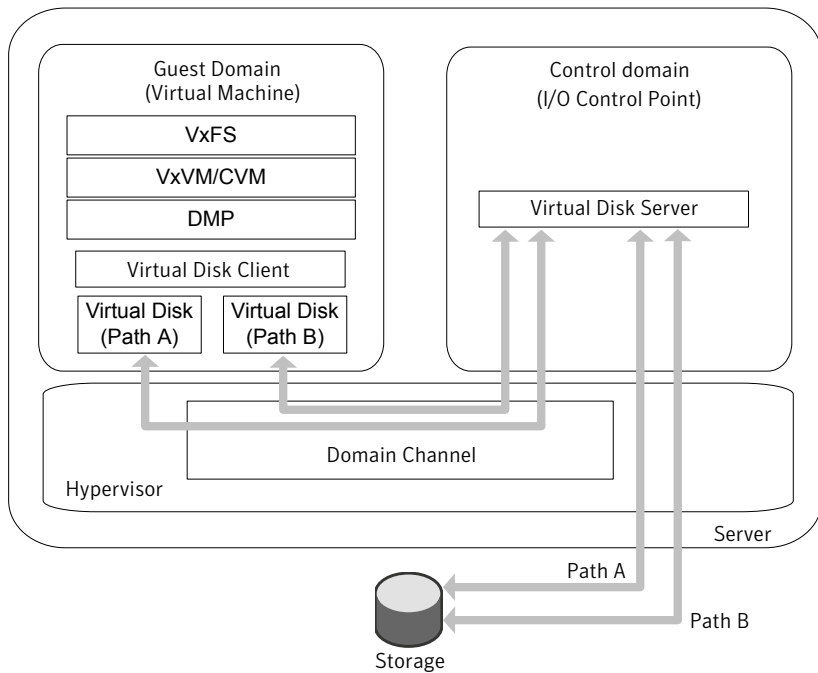
Caution: Do not shrink the underlying volume beyond the size of the VxFS file system in the guest as this can lead to data loss.

- Exporting a volume set to a guest domain is not supported.
 - Veritas Volume Replicator is not supported in the Split Storage Foundation stack model.
 - Multi-volume filesets/DST
 - File-level Smartsync
 - The following VxFS tunables are not set to their default values based on the underlying volume layout, due to VxFS being in the guest domain and VxVM being installed in the control domain:
 - `read_pref_io`
 - `write_pref_io`
 - `read_nstream`
 - `write_nstream`
- If desired, you can set the values of these tunables based on the underlying volume layout in the `/etc/vx/tunefstab` file.
- See the *Veritas Storage Foundation Administrator's Guide* for more information about tuning I/O.
- Storage Foundation Cluster File System is not recommended in this deployment model.

Guest-based Storage Foundation stack model

Figure 4-2 illustrates the guest-based Storage Foundation stack model with guest logical domains.

Figure 4-2 Guest-based Storage Foundation stack model



How Storage Foundation and High Availability Solutions works in the guest domains

The entire Storage Foundation stack is co-located within the guest in this deployment model.

Symantec recommends that you export all paths to a disk which is being exported to a guest and let Veritas DMP do the multi-pathing of the disk in the guest domain.

Note: Only full SCSI disks can be used under Veritas Volume Manager (VxVM) and DMP in this model. Non-SCSI devices (volume, file, slice, etc) are not supported.

Veritas Storage Foundation and High Availability Solutions and Veritas Storage Foundation Cluster File System supports running in the guest domains in both single-node, multiple-node, and multiple-node high availability configurations.

- For a single node configuration, VxVM (including DMP) and VxFS are co-located in the guest domain.

- For clustered nodes, CVM can also be used inside the guest domain. As with regular physical nodes, forming a CVM cluster of logical domain guests requires shared storage visibility across the guests that are part of the cluster.
See the *Veritas Storage Foundation Cluster File System High Availability Administrator's Guide* for CVM information.
See the *Veritas Storage Foundation Installation Guide* for installation and configuration information.
- For clustered nodes in a highly available environment, install Veritas Cluster Server (VCS) in the guest domains.
See the *Veritas Cluster Server* documentation for more information.

Veritas Volume Replicator (VVR) is supported in the guest-based Storage Foundation stack model in the following configurations:

- A guest domain on one host acting as the VVR primary, and another guest on another host acting as the VVR secondary.
- Two guest domains on the same physical host, but you must export separate LUNs or disks to the data volumes and Storage Replicator Logs of the two guest domains.

Supported configurations with SFCFS and multiple I/O Domains

Figure 4-3 SFCFS cluster across two guest domains

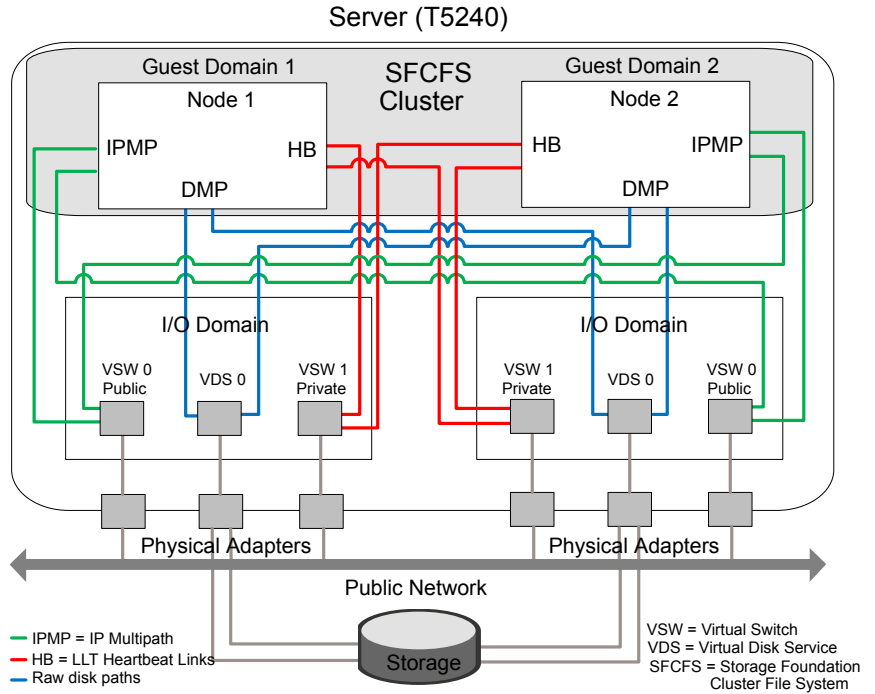


Figure 4-3 illustrates that each guest domain gets network and disk storage redundancy from the two I/O domains.

Figure 4-4 SFCFS cluster across two guest domains

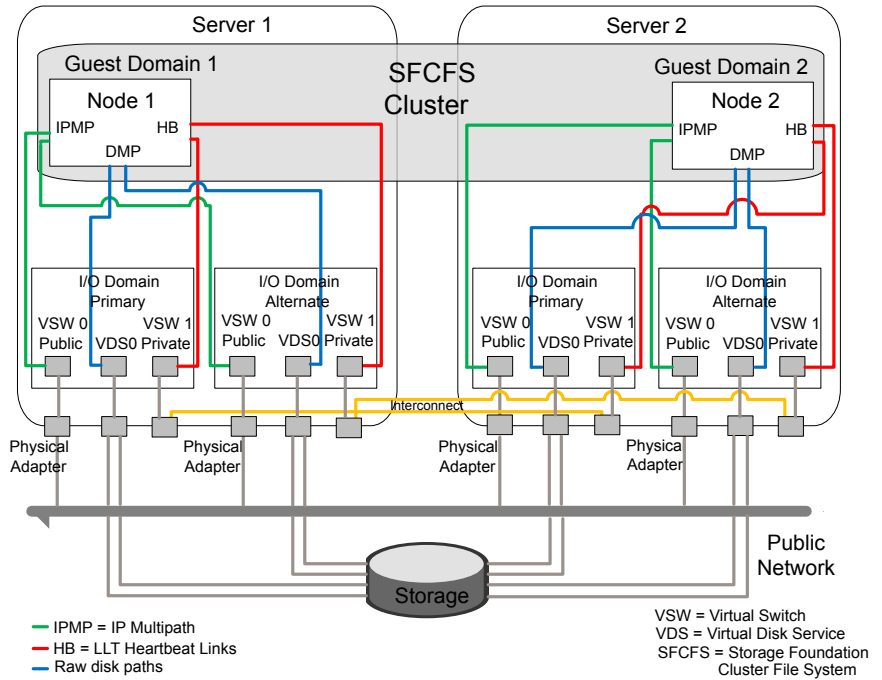


Figure 4-4 illustrates that each guest domain gets network and disk storage redundancy from the two I/O domains on that physical server. The guest cluster spans across two physical servers.

Figure 4-5 SFCFS cluster across four guest domains

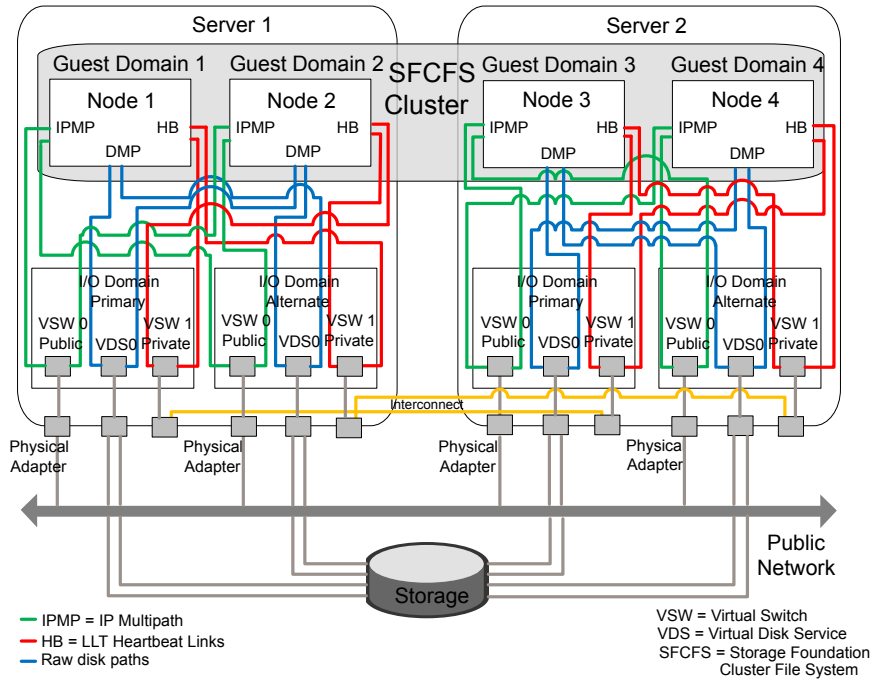


Figure 4-5 illustrates that each guest domain gets network and disk storage redundancy from two I/O domains on that physical server. The guest cluster spans across two physical servers.

Figure 4-6 SFCFS cluster across four guest domains

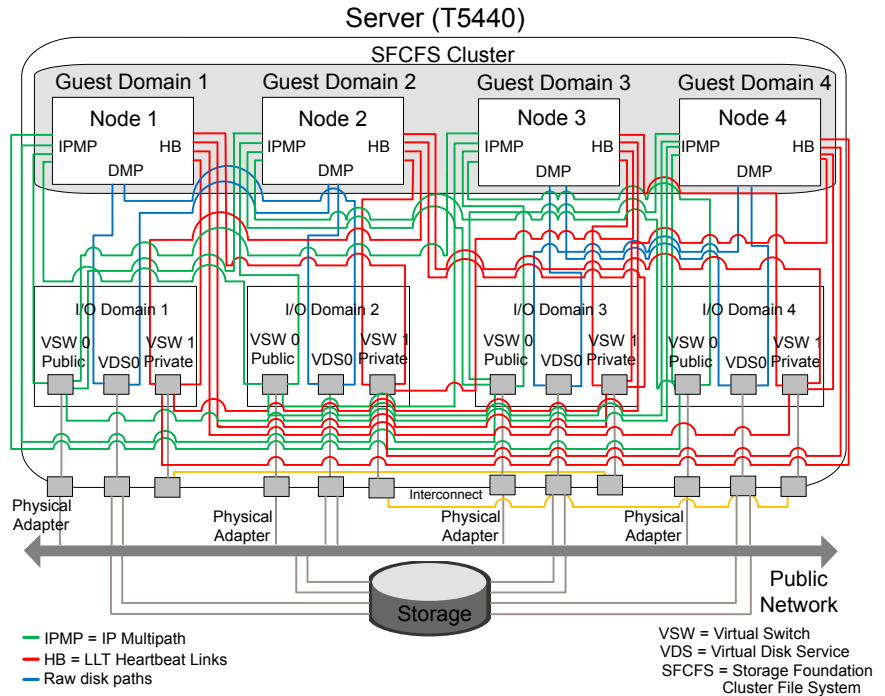


Figure 4-6 illustrates each guest gets its disk storage redundancy from two out of the four I/O domains. Each guest gets its network redundancy from all the four I/O domains.

Veritas Storage Foundation features restrictions

The following Veritas Storage Foundation software features are restricted in the Oracle VM Server for SPARC guest domain environment.

Veritas Volume Replicator bunker replication

Veritas Volume Replicator (VVR) currently does not support configuring a guest domain as a bunker node in the bunker replication mode. This restriction may be removed in a later release.

Mirroring across controllers using vxassist the mirror=ctlr option

Currently, all virtual disks in the guest are under the same virtual controller c0. When `vxassist` tries to look for a second controller to place the mirror on, it fails and therefore the command fails.

All disks fall under the c0 controller, even if they are exported using different physical paths in the backend and coming from different HBAs.

DMP SCSI bypass

The virtual disk client (VDC) driver controls all virtual disks in the guest domain and not the SCSI driver.

Therefore, it is not possible to construct a SCSI packet and send the packet down through DMP. Setting the tunable `dmp_fast_recovery` to on has no effect.

Event Source Daemon (vxesd) fabric monitoring capabilities

One of the features of the `vxesd` daemon is to register with the HBA API to listen to fabric events. Even though the HBA API is loaded in the guest, since there is currently no concept of a directly connected HBA in the guest, this API is not of any use. Therefore, this capability of `vxesd` is not available.

Physical WWN for a path

It is not possible to create a Sub Path Failover Group (SFG) without Physical WWN. Physical World Wide IDs cannot be obtained from inside the guest because they are fetched by DMP using the HBA API which is currently not functional inside the guest.

System requirements

This section describes the system requirements for this release.

Solaris operating system requirements

Veritas Storage Foundation for Oracle VM Server for SPARC is supported on the following Solaris operating systems:

- Solaris 10 (SPARC Platform) Update 7 or later

Note: Symantec recommends installing the latest Solaris 10 Update.

- Logical domain software, version 2.0 and 2.1
Visit the Oracle Web site for information about Solaris operating environments.

Solaris patch requirements

Some required system patches may already be present in your operating system. Check if your system already contains the required Solaris patches needed before installing Oracle VM Server for SPARC. Use the `showrev -p` command to display the patches included on your system.

For more information, see the `showrev(1M)` manual page.

If the patches shown in the required list are not already installed, go to the Oracle support website to download them. You must install the appropriate patches and then reboot your system, if required.

Warning: Patch version and information are determined at the time of product release. Contact your vendor for the most current patch version and information.

Table 4-2 Patches

Oracle Solaris patch number	Notes
141777-01	This patch fixes the following Oracle (Sun) bugs that affects SF functionality: <ul style="list-style-type: none"> ■ Oracle (Sun) bug id: 6795836 — <code>vd_setup_vd()</code> should handle errors from <code>vd_identify_dev()</code> better
139562-02 (obsoleted by 138888-07)	This patch fixes the following Oracle (Sun) bugs that affects SF functionality: <ul style="list-style-type: none"> ■ Oracle (Sun) bug id: 6640564 — <code>vds</code> should not serialize disk image IO ■ Oracle (Sun) bug id: 6699271 — Dynamic virtual disk size management ■ Oracle (Sun) bug id: 6705190 — <code>uscscmd</code> on <code>vdisk</code> can overflow sense buffer ■ Oracle (Sun) bug id: 6716365 — disk images on volumes should be exported using the <code>Idi</code> interface

Table 4-2 Patches (*continued*)

Oracle Solaris patch number	Notes
138042-05	<p>This patch fixes the following Oracle (Sun) bugs that affects SF functionality:</p> <ul style="list-style-type: none"> ■ Oracle (Sun) bug id: 6637560 — disks are not correctly exported with vxdmp ■ Oracle (Sun) bug id: 6558966 — virtual disks created from files do not support EFI labels <p>Oracle recommends installing the latest patches in all domains to ensure that you have the most up-to-date drivers.</p>

Hardware requirements

Visit the Oracle Web site for information about the supported hardware for Oracle VM Server for SPARC

Veritas product release notes

Read the relevant Veritas product release notes before installing any version of Veritas Storage Foundation and High Availability or Veritas Storage Foundation Cluster File System.

Veritas Storage Foundation and High Availability

Release notes for Veritas Storage Foundation are located under the `storage_foundation/docs` directory and the Veritas Cluster Server are located under the `cluster_server/docs` directory of the Veritas Storage Foundation disc.

- *Veritas Storage Foundation Release Notes*
- *Veritas Cluster Server Release Notes*

Veritas Storage Foundation Cluster File System

Release notes for Veritas Storage Foundation Cluster File System are located under the `storage_foundation_cluster_file_system/docs` directory of the Veritas disc.

- *Veritas Storage Foundation Cluster File System Release Notes*

Product licensing

Customers running Veritas Storage Foundation or Veritas Storage Foundation Cluster File System in a Oracle VM Server for SPARC environment are entitled to use an unlimited number of logical domains on each licensed server or CPU.

Installing Storage Foundation in a Oracle VM Server for SPARC environment

This section describes how to install Storage Foundation in several Oracle VM Server for SPARC environments.

To install the Split Storage Foundation stack model environment, you must complete the following sections in order:

- See [“Installing and configuring Oracle VM Server for SPARC and domains”](#) on page 80.
- See [“Installing Storage Foundation in the control domain or guest”](#) on page 80.
- See [“Installing Veritas File System in the guest domain using pkgadd”](#) on page 85.
- See [“Verifying the configuration”](#) on page 86.

To install the Guest based Storage Foundation stack model environment, which includes Veritas Storage Foundation Cluster File System, you must complete the following sections in order:

- See [“Installing and configuring Oracle VM Server for SPARC and domains”](#) on page 80.
- See [“Installing Storage Foundation in the control domain or guest”](#) on page 80.
- See [“Verifying the configuration”](#) on page 86.

To install and configure Veritas Cluster Server in a Oracle VM Server for SPARC environment, see the following sections:

- See [“Configuring Veritas Cluster Server to fail over a logical domain on a failure”](#) on page 120.
- See [“Configuring Veritas Cluster Server to fail over an application on a failure”](#) on page 128.

Installing and configuring Oracle VM Server for SPARC and domains

Refer to the Oracle documentation for instructions about installing and configuring the Oracle VM Server for SPARC software and configuring the control and guest domains.

Installing Storage Foundation in the control domain or guest

This section describes how to install Storage Foundation in the control domain or guest domain.

Installing the split Storage Foundation stack model

If you are installing the split Storage Foundation stack model, the entire stack must be in the control domain and VxFS must be in the guest domain.

Use the procedures in the Veritas installation documentation and Release Notes to install Storage Foundation in the control domain.

To install the split Storage Foundation stack model

- ◆ Install the product.
 - See the *Veritas Storage Foundation Installation Guide for Solaris*.
 - See the *Veritas Storage Foundation Release Notes for Solaris*.

Installing the guest-based Storage Foundation stack model

If you are installing the guest-based Storage Foundation stack model environment, the entire stack must be in the guest domain.

Use the procedures in the SF or SFCFS Installation Guide and Release Notes, as appropriate, to install SF or SFCFS in a guest domain.

To install the guest-based Storage Foundation stack model

- ◆ Install the product.
 - See the *Veritas Storage Foundation Installation Guide for Solaris* for SF.
 - See the *Veritas Storage Foundation Release Notes for Solaris* for SF.
 - See the *Veritas Storage Foundation Cluster File System Installation Guide for Solaris* for SFCFS.
 - See the *Veritas Storage Foundation Cluster File System Release Notes for Solaris* for SFCFS.

DMP as a multi-pathing solution in an Oracle VM Server for SPARC environment

Using Dynamic Multi-Pathing (DMP), you can choose to manage multiple paths to a system's storage in the following domains:

- control domain
- I/O domains
- guest domains

For configurations that have multiple I/O domains, Symantec recommends that you have DMP manage the paths inside of the I/O domains.

Note: Not having a multi-pathing solution can cause accidental data corruption when the same device is accessed.

Using DMP in the control domain lets you use all of the Storage Foundation (SF) features from inside of the guest. You must enable DMP in the control and alternate I/O domains.

See [“Enabling DMP in the control and alternate I/O domains”](#) on page 81.

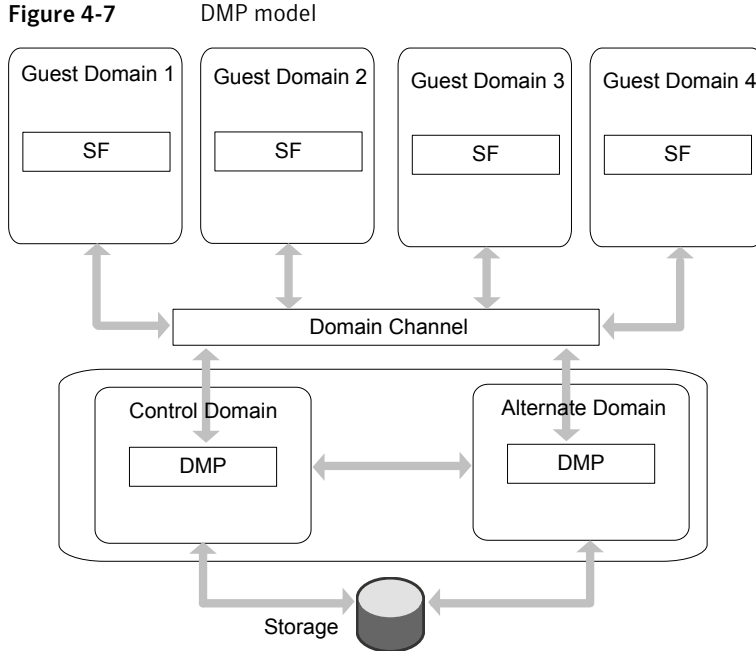
You can enable DMP path failover in the guest domain.

See [“Enabling DMP path failover in the guest domain”](#) on page 83.

Enabling DMP in the control and alternate I/O domains

This section describes how to enable Dynamic Multi-Pathing (DMP) in the control and alternate I/O domains.

[Figure 4-7](#) illustrates the architecture of DMP.



To enable DMP in the control and alternate I/O domains

- 1 Install VRTSvxvm and VRTSaslapm packages on both the control and alternate I/O domains.

For information on manual installation of DMP:

See the *Veritas Dynamic Multi-Pathing Installation Guide*.

- 2 Turn on the enclosure-based naming default setting in both the control and alternate domains.

```
# vxddladm set namingscheme=ebn
```

- 3 Create the Virtual Disk Service (VDS) devices on DMP metanodes that can be provisioned to the guest domains.

For example:

```
# ldm add-vdsdev /dev/vx/dmp/xyz vol10015-001-p1@primary-vds0
# ldm add-vdsdev /dev/vx/dmp/xyz vol10015-001-p2@alternate-vds0
```

- 4 While provisioning, export the DMP metanodes from both the control and alternate I/O domain:

```
# ldm add-vdsdev /dev/vx/dmp/xyz vol10015-001-p1@primary-vds0
# ldm add-vdsdev /dev/vx/dmp/xyz vol10015-001-p2@alternate-vds0
# ldm add-vdisk timeout=30 vdisk0015-001-p1 \
vol10015-001-p1@primary-vds0 hxxd0015
# ldm add-vdisk timeout=30 vdisk0015-001-p2 \
vol10015-001-p2@alternate-vds0 hxxd0015
```

This allows DMP in the guest domain to see two access paths, one through the control domain and the other through the alternate domain to the storage.

DMP in the guest domain can take care of the control and alternate I/O domain failures.

Enabling DMP path failover in the guest domain

In Oracle VM Server configurations the Virtual Disk Client (VDC) driver timeout is set to zero by default which signifies infinity. This can cause the failed I/O not to return to the guest domain, if either the control or alternate I/O domain crashes unexpectedly. As a result the guest domain cannot get back the failed I/Os and cannot route them through the alternate domain. If this issue occurs or to avoid this issue, you must set the VDC driver timeout.

There are two ways to set the VDC driver timeout:

Modify globally all of the LUNs that are exported to the current guest domain. This requires a reboot to all the guest domains.

See [“To change the VDC driver timeout globally”](#) on page 84.

Manually export every LUN directly to the guest domain and set the timeout parameter to 30 seconds. No reboot is required.

See [“To change the VDC driver timeout for each LUN”](#) on page 84.

To change the VDC driver timeout globally

- 1 On each guest domain, edit the `/etc/system` file and add the following line to set the VDC driver timeout to 30 seconds:

```
set vdc:vdc_timeout=30
```

- 2 Reboot the guest domains.

To change the VDC driver timeout for each LUN

- 1 Create the primary domain using four internal disks and get all required SAN LUNs for the guest domains allocated to the primary domain.
- 2 Turn on enclosure-based naming in the primary domain.

```
# vxddladm set namingscheme=ebn
```

- 3 Remove half of the system's I/O from the primary domain:

```
# ldm remove-io pci_X primary_domain_name
```

where `pci_x` is the name of the PCI bus for your system.

where `primary_domain_name` is the name of the primary domain.

For example:

```
# ldm remove-io pci_@400 primary
```

- 4 Create the alternate I/O domain on the other four internal disks and add the I/O that was removed from the primary domain:

```
# ldm add-io pci_X primary_domain_name
```

where `pci_x` is the name of the PCI bus for your system.

where `primary_domain_name` is the name of the primary domain.

For example:

```
# ldm add-io pci_@400 primary
```

- 5 Turn on enclosure-based naming in the alternate I/O domain.

```
# vxddladm set namingscheme=ebn
```

At this point you have one of the two paths to each SAN LUN in the primary domain and the other path is in the alternate I/O domain. Both will have the same name: `/dev/vx/dmp/enclousre_based_name`.

- 6 On the primary domain, create the guest domains. In the sample the enclosure-based name of one of the LUNs is `xyz` and the guest domain is `hsxd0015`:

```
# ldm add-vdsdev /dev/vx/dmp/xyz vol0015-001-p1@primary-vds0
# ldm add-vdsdev /dev/vx/dmp/xyz vol0015-001-p2@alternate-vds0
# ldm add-vdisk timeout=30 vdisk0015-001-p1 \
vol0015-001-p1@primary-vds0 hsxd0015
# ldm add-vdisk timeout=30 vdisk0015-001-p2 \
vol0015-001-p2@alternate-vds0 hsxd0015
```

The same set of four commands for each SAN LUN that gets placed in a guest domain. Use three SAN LUNs for SAN boot in the guest domain and the rest for application data. Each LUN in the guest domain has one path backup through the primary domain and one backup through the alternate domain. That means each LUN only uses one LDC in each domain. Also, since you are using DMP you still only use 1 LDC in each domain, even if the LUN has more than two paths from the array.

Installing Veritas File System in the guest domain using pkgadd

This section describes how to install VxFS 6.0 in the guest domain.

To install Veritas File System in the guest domain

- 1 Copy the VxFS packages from the `/pkgs` directory on the disc to a location in the guest domain where you have write permissions.
- 2 Install the packages:

```
# pkgadd -d VRTSvlic.pkg
# pkgadd -d VRTSvxfs.pkg
# pkgadd -d VRTSfssdk.pkg
```

- 3 Reboot the guest domain.

Verifying the configuration

Verify the configuration of Oracle VM server for SPARC in the control domain and the guest domain. Refer to the Oracle documentation for details.

Verify the Storage Foundation installation in both the control domain and the guest domain. Refer to the following guides for more information:

- See the *Veritas Storage Foundation Installation Guide for Solaris*.
- See the *Veritas Storage Foundation Release Notes for Solaris*.
- See the *Veritas Storage Foundation and High Availability Installation Guide for Solaris*.
- See the *Veritas Storage Foundation and High Availability Release Notes for Solaris*.
- See the *Veritas Storage Foundation Cluster File System and High Availability Installation Guide for Solaris*.
- See the *Veritas Storage Foundation Cluster File System and High Availability Release Notes for Solaris*.

Exporting a Veritas volume to a guest domain from the control domain

Use the following procedure to migrate a VxVM disk group from a non-logical domain environment to a Oracle VM Server for SPARC environment.

Note: This section applies to only the Split Storage Foundation model.

In the following example control domain is named `primary` and the guest domain is named `ldom1`. The prompts in each step show in which domain to run the command.

To create virtual disks on top of the Veritas Volume Manager data volumes using the ldm command

- 1 The VxVM diskgroup on the target host is imported in the control domain, after which volumes are visible from inside the control domain.

See the *Veritas Storage Foundation Administrator's Guide* to move disk groups between systems.

- 2 In the control domain (primary), configure a service exporting the VxVM volume containing a VxFS or UFS filesystem as a slice using the `options=slice` option:

```
primary# ldm add-vdiskserverdevice options=slice \  
/dev/vx/dsk/dg-name/volume_name \  
volume_name volume_name@primary-vds0
```

Caution: A volume by default shows up as a full disk in the guest. The Virtual Disk Client driver writes a VTOC on block 0 of the virtual disk, which will end up as a WRITE on block 0 of the VxVM volume. This can potentially cause data corruption, because block 0 of the VxVM volume contains user data. Using `options=slice` exports a volume as a slice to the guest and does not cause any writes to block 0, therefore preserving user data.

- 3 Add the exported disk to a guest domain:

```
primary# ldm add-vdisk vdisk1 volume_name  
volume_name@primary-vds0 ldom1
```

- 4 Start the guest domain, and ensure that the new virtual disk is visible.

```
primary# ldm bind ldom1  
  
primary# ldm start ldom1
```

- 5 If the new virtual disk device nodes do not show up in the `/dev/[r]dsk` directories, then run the `devfsadm` command in the guest domain:

```
ldom1# devfsadm -C
```

In this example, the new disk appears as `/dev/[r]dsk/c0d1s0`.

```
ldom1# ls -l /dev/dsk/c0d1s0
```

```
lrwxrwxrwx 1 root root 62 Sep 11 13:30 /dev/dsk/c0d1s0 ->
../../devices/virtual-devices@100/channel-devices@200/disk@1:a
```

- 6 Mount the file system on the disk to access the application data:

```
ldom1# mount -F vxfs /dev/dsk/c0d1s0 /mnt
```

```
ldom1# mount -F ufs /dev/dsk/c0d1s0 /mnt
```

Provisioning storage for a guest domain

Use the following procedure to provision storage for a guest domain. You can provision both boot disks and data disks.

Note: This section applies to the Split Storage Foundation stack model only.

For the guest-based Storage Foundation model:

See [“How Storage Foundation and High Availability Solutions works in the guest domains”](#) on page 70.

Provisioning Veritas Volume Manager volumes as data disks for guest domains

The following procedure uses VxVM volumes as data disks (virtual disks) for guest domains.

VxFS can be used as the file system on top of the data disks.

The example control domain is named `primary` and the guest domain is named `ldom1`. The prompts in each step show in which domain to run the command.

To provision Veritas Volume Manager volumes as data disks

- 1 Create a VxVM disk group (`mydatadg` in this example) with some disks allocated to it:

```
primary# vxdg init mydatadg TagmaStore-USP0_29 TagmaStore-USP0_30
```

- 2 Create a VxVM volume of the desired layout (in this example, creating a simple volume):

```
primary# vxassist -g mydatadg make datavol1 500m
```

- 3 Configure a service exporting the volume `datavol1` as a virtual disk:

```
primary# ldm add-vdiskserverdevice /dev/vx/dsk/mydatadg/datavol1 \  
datadisk1@primary-vds0
```

- 4 Add the exported disk to a guest domain.

```
primary# ldm add-vdisk vdisk1 datadisk1@primary-vds0 ldom1
```

- 5 Start the guest domain, and ensure that the new virtual disk is visible:

```
primary# ldm bind ldom1
```

```
primary# ldm start ldom1
```

- 6 If the new virtual disk device node entries do not show up in the `/dev/[r]dsk` directories, then run the `devfsadm` command in the guest domain:

```
ldom1# devfsadm -C
```

- 7 Label the disk using the `format` command to create a valid label before trying to access it.

See the `format(1M)` manual page.

- 8 Create the file system where `c0d1s2` is the disk.

```
ldom1# mkfs -F vxfs /dev/rdsk/c0d1s2
```

9 Mount the file system.

```
ldom1# mount -F vxfs /dev/dsk/c0d1s2 /mnt
```

10 Verify that the file system has been created:

```
ldom1# df -hl -F vxfs
```

```
Filesystem size used avail capacity Mounted on  
/dev/dsk/c0d1s2 500M 2.2M 467M 1% /mnt
```

Provisioning Veritas Volume Manager volumes as boot disks for guest domains

The following procedure provisions boot disks for a guest domain.

A VxVM volume appears as a full disk by default and can be used as a boot disk for a guest domain.

The following process gives the outline of how a VxVM volume can be used as a boot disk.

The example control domain is named `primary` the guest domain is named `ldom1`. The prompts in each step show in which domain to run the command.

To provision Veritas Volume Manager volumes as boot disks for guest domains

- 1 On the control domain, create a VxVM volume of a size that is recommended for Solaris 10 installation. In this example, a 7GB volume is created:

```
primary# vxassist -g boot_dg make bootdisk-vol 7g
```

- 2 Configure a service by exporting the `/dev/vx/dsk/boot_dg/bootdisk1-vol` volume as a virtual disk:

```
primary# ldm add-vdiskserverdevice \  
/dev/vx/dsk/boot_dg/bootdisk1-vol bootdisk1-vol@primary-vds0
```

- 3 Add the exported disk to `ldom1`:

```
primary# ldm add-vdisk vdisk1 bootdisk1-vol@primary-vds0 ldom1
```

- 4 Follow Oracle's recommended steps to install and boot a guest domain, and use the virtual disk `vdisk1` as the boot disk during the net install.

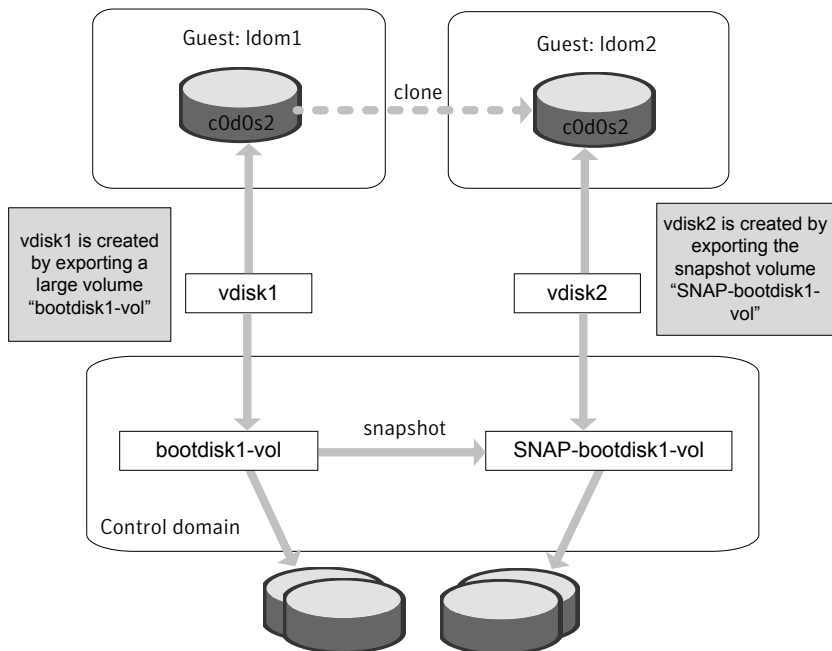
Using Veritas Volume Manager snapshots for cloning logical domain boot disks

The following highlights the steps to clone the boot disk from an existing logical domain using VxVM snapshots, and makes use of the third-mirror breakoff snapshots.

See [“Provisioning Veritas Volume Manager volumes as boot disks for guest domains”](#) on page 90.

[Figure 4-8](#) illustrates an example of using Veritas Volume Manager snapshots for cloning Logical Domain boot disks.

Figure 4-8 Example of using Veritas Volume Manager snapshots for cloning Logical Domain boot disks



Before this procedure, Idom1 has its boot disk contained in a large volume, `/dev/vx/dsk/boot_dg/bootdisk1-vol`.

This procedure involves the following steps:

- Cloning the logical domain configuration to form a new logical domain configuration.

This step is a Solaris logical domain procedure, and can be achieved using the following commands:

```
# ldm list-constraints -x
```

```
# ldm add-domain -i
```

Refer to the Oracle documentation for more information about cloning the logical domain configuration to form a new logical domain configuration.

See the *Logical Domains Administration Guide*.

- After cloning the configuration, clone the boot disk and provision it to the new logical domain.
To create a new logical domain with a different configuration than that of `ldom1`, skip this step of cloning the configuration and create the desired logical domain configuration separately.

To clone the boot disk using Veritas Volume Manager snapshots

- 1 Create a snapshot of the source volume bootdisk1-vol. To create the snapshot, you can either take some of the existing ACTIVE plexes in the volume, or you can use the following command to add new snapshot mirrors to the volume:

```
primary# vxsnap [-b] [-g diskgroup] addmir volume \  
[nmirror=N] [alloc=storage_attributes]
```

By default, the `vxsnap addmir` command adds one snapshot mirror to a volume unless you use the `nmirror` attribute to specify a different number of mirrors. The mirrors remain in the SNAPATT state until they are fully synchronized. The `-b` option can be used to perform the synchronization in the background. Once synchronized, the mirrors are placed in the SNAPDONE state.

For example, the following command adds two mirrors to the volume, bootdisk1-vol, on disks mydg10 and mydg11:

```
primary# vxsnap -g boot_dg addmir bootdisk1-vol \  
nmirror=2 alloc=mydg10,mydg11
```

If you specify the `-b` option to the `vxsnap addmir` command, you can use the `vxsnap snapwait` command to wait for synchronization of the snapshot plexes to complete, as shown in the following example:

```
primary# vxsnap -g boot_dg snapwait bootdisk1-vol nmirror=2
```

- 2 To create a third-mirror break-off snapshot, use the following form of the `vxsnap` `make` command.

Caution: Shut down the guest domain before executing the `vxsnap` command to take the snapshot.

```
primary# vxsnap [-g diskgroup] make \  
source=volume[/newvol=snapvol] \  
{/plex=plex1[,plex2,...]|/nmirror=number}
```

Either of the following attributes may be specified to create the new snapshot volume, `snapvol`, by breaking off one or more existing plexes in the original volume:

- `plex` Specifies the plexes in the existing volume that are to be broken off. This attribute can only be used with plexes that are in the ACTIVE state.
- `nmirror` Specifies how many plexes are to be broken off. This attribute can only be used with plexes that are in the SNAPDONE state. Such plexes could have been added to the volume by using the `vxsnap addmir` command.

Snapshots that are created from one or more ACTIVE or SNAPDONE plexes in the volume are already synchronized by definition.

For backup purposes, a snapshot volume with one plex should be sufficient.

For example,

```
primary# vxsnap -g boot_dg make \  
source=bootdisk1-vol/newvol=SNAP-bootdisk1-vol/nmirror=1
```

Here `bootdisk1-vol` makes source; `SNAP-bootdisk1-vol` is the new volume and `1` is the `nmirror` value.

The block device for the snapshot volume will be
`/dev/vx/dsk/boot_dg/SNAP-bootdisk1-vol`.

- 3 Configure a service by exporting
the `/dev/vx/dsk/boot_dg/SNAP-bootdisk1-vol` file as a virtual disk.

```
primary# ldm add-vdiskserverdevice \  
/dev/vx/dsk/boot_dg/SNAP-bootdisk1-vol vdisk2@primary-vds0
```

- 4 Add the exported disk to ldom1 first.

```
primary# ldm add-vdisk vdisk2 \  
SNAP-bootdisk1-vol@primary-vds0 ldom1  
  
primary# ldm bind ldom1  
  
primary# ldm start ldom1
```

- 5 Start ldom1 and boot ldom1 from its primary boot disk vdisk1.

```
primary# ldm bind ldom1  
  
primary# ldm start ldom1
```

- 6 If the new virtual disk device node entries do not show up in the `/dev/[r]disk` directories, then run the `devfsadm` command in the guest domain:

```
ldom1# devfsadm -C
```

where vdisk2 is the c0d2s# device.

```
ldom1# ls /dev/dsk/c0d2s*  
  
/dev/dsk/c0d2s0 /dev/dsk/c0d2s2 /dev/dsk/c0d2s4 /dev/dsk/c0d2s6  
/dev/dsk/c0d2s1 /dev/dsk/c0d2s3 /dev/dsk/c0d2s5 /dev/dsk/c0d2s7
```

- 7 Mount the root file system of c0d2s0 and modify the `/etc/vfstab` entries such that all `c#d#s#` entries are changed to `c0d0s#`. You must do this because ldom2 is a new logical domain and the first disk in the operating system device tree is always named as `c0d0s#`.

- 8 Stop and unbind ldom1 from its primary boot disk vdisk1.

```
primary# ldm stop ldom1  
primary# ldm unbind ldom1
```

- 9 After you change the `vfstab` file, unmount the file system and unbind vdisk2 from ldom1:

```
primary# ldm remove-vdisk vdisk2 ldom1
```

10 Bind vdisk2 to ldom2 and then start and boot ldom2.

```
primary# ldm add-vdisk vdisk2 vdisk2@primary-vds0 ldom2
```

```
primary# ldm bind ldom2
```

```
primary# ldm start ldom2
```

After booting ldom2, appears as ldom1 on the console because the other host-specific parameters like hostname and IP address are still that of ldom1.

```
ldom1 console login:
```

11 To change the parameters bring ldom2 to single-user mode and run the `sys-unconfig` command.

12 Reboot ldom2.

During the reboot, the operating system prompts you to configure the host-specific parameters such as hostname and IP address, which you must enter corresponding to ldom2.

13 After you have specified all these parameters, ldom2 boots successfully.

Configuring Oracle VM Server for SPARC guest domains for disaster recovery

The Oracle VMs can be configured for disaster recovery by replicating the boot disk using replication methods like Hitachi TrueCopy, EMC SRDF, Veritas Volume Replicator, and so on. The network configuration for the Oracle VM in the primary site may not be effective in the secondary site if the two sites are in different IP subnets. You will need to make these additional configuration changes to the LDom resource.

To configure the guest domains for disaster recovery, you need to configure VCS on both the sites in the Control Domains with GCO option.

Refer to the *Veritas Cluster Server Administrator's Guide* for more information about global clusters.

To set up the guest domain for disaster recovery

- 1 On the primary site, create the guest domain using `ldm` commands and configure the network-related parameters.
- 2 On the primary site after you boot the guest domain, copy and install the package `VRTSvcsnr` from the VCS installation media in the guest domain. This package installs the `vcs-network-reconfig` service in the guest domain. This service makes sure that the site-specific network parameters are applied when the guest domain boots.
- 3 On the primary site, shut down the guest domain.
- 4 Use replication specific commands to failover the replication to the secondary site from the primary site.
- 5 Repeat step 1 on the secondary site.
- 6 Perform step 7, step 8, step 9, and step 10 on both the primary cluster and the secondary clusters.
- 7 Create a VCS service group and add a VCS LDom resource for the guest domain.

Configure the following disaster recovery-related attributes on the LDom resource with site-specific values for each: `IPAddress`, `Netmask`, `Gateway`, `DNS` (DNS Server).

Set the value of the `ConfigureNetwork` attribute to 1 to make the changes effective. The LDom agent does not apply the disaster recovery-related attributes to the guest domain if the value of the `ConfigureNetwork` attribute is 0.

- 8 Add the appropriate `Mount` and `DiskGroup` resources in the service group for the file system and the disk group on which the boot image of the guest domain resides.

Add a resource dependency from the LDom resource to the `Mount` resource and another dependency from the `Mount` resource to the `Diskgroup` resource.

- 9 Add the appropriate VCS replication resource in the service group. Examples of hardware replication agents are `SRDF` for EMC SRDF, `HTC` for Hitachi TrueCopy, `MirrorView` for EMC MirrorView, etc.

Refer to the appropriate VCS Replication agent guide for configuring the replication resource.

For VVR-based replication, add the appropriate `RVGPrimary` resource to the service group.

Refer to the following manuals for more information:

- For information about configuring VVR-related resources, see the *Veritas Storage Foundation and High Availability Solutions Replication Administrator's Guide*.
- For information about the VVR-related agents, see the *Veritas Cluster Server Bundled Agents Reference Guide*.

10 Add a dependency from the DiskGroup resource to the replication resource.

Figure 4-9 Sample resource dependency diagram for hardware replication based guest domains

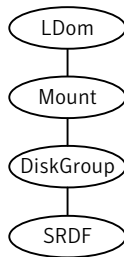
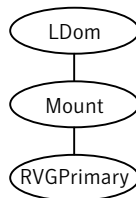


Figure 4-10 Sample resource dependency diagram for VVR replication-based guest domains



The replication resource makes sure that when the resource is online in a site, the underlying replicated devices are in primary mode and the remote devices are in secondary mode. Thus, when the LDom resource goes online, the underlying storage will always be in read-write mode. When the LDom resource goes online, it sets the DR related parameters in the EEPROM parameter network-boot-arguments for the guest domain before starting the guest domain. When the guest domain boots, the vcs-network-reconfig service starts inside the guest domain. This service reads the EEPROM parameter and applies the disaster recovery related parameters by modifying the appropriate files inside the guest domain.

Software limitations

The following section describes some of the limitations of the Oracle VM server for SPARC software and how those software limitations affect the functionality of the Veritas Storage Foundation products.

Memory Corruption in the guest domain during SCSI commands

When running VxVM and DMP the system panics in the guest domain. The following error messages display on the console:

```
vxdmp: NOTICE: VxVM vxdmp V-5-3-289 DMP: Memory Overrun!  
Caller dmpscsi_sys.c(170) Ptr 0x3000528e580 Size 0x100
```

```
vxdmp: NOTICE: VxVM vxdmp V-5-3-289 DMP: Memory Overrun!  
Caller dmpgrio.c(824) Ptr 0x3002d7f1a80 Size 0xe8
```

These error messages are due to a kernel memory corruption occurring in the Solaris kernel driver stacks (virtual disk drivers). This issue occurs when issuing USCSICMD with the sense request enable (USCSI_RQENABLE) on a virtual disk from the guest.

Symantec has an open escalation with Oracle and an associated Oracle (SUN) bug id for this issue:

Oracle (SUN) Escalation number: 1-23915211

Oracle (SUN) bug id: 6705190 (ABS: uscsicmd on vdisk can overflow the sense buffer)

This Oracle (SUN) bug has been fixed in Sun patch 139562-02.

See [“Solaris patch requirements”](#) on page 77.

Exporting the raw volume device node fails

Following messages can be observed in the `/var/adm/messages` file:

```
vds: [ID 998409 kern.info] vd_setup_vd():  
/dev/vx/rdisk/testdg/testvol identification failed  
vds: [ID 790452 kern.info] vd_setup_vd():  
/dev/vx/rdisk/testdg/testvol can not be exported as a virtual disk  
(error 5)  
vds: [ID 556514 kern.info] vds_add_vd(): Failed to add vdisk ID 21  
vds: [ID 970787 kern.info] vds_add_vd(): No vDisk entry found for  
vdisk ID 21
```

Workaround: Export VxVM volumes using their block device nodes instead. Oracle is investigating this issue.

Oracle (SUN) bug id: 6716365 (disk images on volumes should be exported using the ldi interface)

This Oracle (Sun) bug is fixed in Oracle (Sun) patch 139562-02.

See [“Solaris patch requirements”](#) on page 77.

Resizing a Veritas Volume Manager volume (exported as a slice or full disk) does not dynamically reflect the new size of the volume in the guest

On resizing a VxVM volume exported to a guest, the virtual disk still shows the old size of the volume. The virtual disk drivers do not update the size of the backend volume after the volume is resized.

Oracle has an RFE for this issue (CR 6699271 Dynamic virtual disk size management).

Workaround: The guest must be stopped and rebound for the new size to be reflected.

This Oracle (Sun) bug is fixed in Oracle (Sun) patch 139562-02.

See [“Solaris patch requirements”](#) on page 77.

Known issues

The following section describes some of the known issues of the Oracle VM Server for SPARC software and how those known issues affect the functionality of the Veritas Storage Foundation products.

Guest-based known issues

The following are new known issues in this release of Veritas Storage Foundation and High Availability Solutions Support for Oracle VM Server for SPARC.

Encapsulating a non-scsi disk may fail

Trying to encapsulate a non-scsi disk which is a slice of a disk or a disk exported as a slice may fail with the following error:

```
VxVM vxslicer ERROR V-5-1-599 Disk layout does not support swap shrinking  
VxVM vxslicer ERROR V-5-1-5964 Unsupported disk layout.
```

Encapsulation requires at least 0 sectors of unused space either at the beginning or end of the disk drive.

This is because while installing the OS on such a disk, it is required to specify the entire size of the backend device as the size of slice "s0", leaving no free space on the disk.

Boot disk encapsulation requires free space at the end or the beginning of the disk for it to proceed ahead.

Guest domain node shows only one PGR key instead of two after rejecting the other node in the cluster

For configuration information concerning the guest domain node shows only 1 PGR key instead of 2 after rejecting the other node in the cluster:

See [Figure 4-4](#) on page 73.

This was observed while performing a series of reboots of the primary and alternate I/O domains on both the physical hosts housing the two guests. At some point one key is reported missing on the coordinator disk.

This issue is under investigation. The `vxfen` driver can still function as long as there is 1 PGR key. This is a low severity issue as it will not cause any immediate interruption. Symantec will update this issue when the root cause is found for the missing key.

Disk paths intermittently go offline while performing I/O on a mirrored volume

This was observed while testing the SFCFS stack inside a 4-node guest cluster where each node gets its network and virtual disk resources from multiple I/O domains within the same host.

See [“Supported configurations with SFCFS and multiple I/O Domains”](#) on page 72.

While performing I/O on a mirrored volume inside a guest, it was observed that a vdisk would go offline intermittently even when at least one I/O domain which provided a path to that disk was still up and running.

Symantec recommends that you install Solaris 10 Update 7 that contains the fix for Oracle (Sun) bug id 6742587 (vds can ACK a request twice).

Deadlock between DMP kernel and VDC driver during volume creation

This was observed by Oracle during their interoperability testing of 5.0 MP3. The deadlock happens when creating a mirrored VxVM volume, while there is ongoing

I/O to a UFS file system on another disk which is not under VxVM. This issue has been observed typically in a large configuration such as 14 virtual CPUs and around 10Gig of memory configured for the guest domain running VxVM.

Relevant Oracle (Sun) bug id: 6744348

This known issue has been fixed and verified in the 5.0 MP3 RP1 release.

For the latest information on updates, patches, documentation, and known issues regarding this 5.0 MP3 RP1 release, see the following TechNote on the Symantec Technical Support website:

For Solaris SPARC:

<http://entsupport.symantec.com/docs/281987>

For Solaris x64:

<http://entsupport.symantec.com/docs/286955>

Split Storage Foundation stack known issues

The following are new known issues in this release of Veritas Storage Foundation and High Availability Solutions Support for Oracle VM Server for SPARC.

Caching of data writes on the backend volume in the service domain

This was observed by a customer during their evaluation of Oracle VM Server for SPARC with Storage Foundation. This issue occurs because data written to the virtual disk might get cached into the service domain before it is effectively written to the virtual disk backend. This can cause potential data loss if the service domain crashes while the data is still cached.

Oracle (Sun) bug id is: 6684721 (file backed virtual I/O should be synchronous)

This Oracle (Sun) bug is fixed in Oracle (Sun) patch 139562-02 that has been obsoleted by 138888-07.

See “[Solaris patch requirements](#)” on page 77.

A volume can become inaccessible from the guest in the event of control domain reboot

All access to such a volume hangs if the primary domain is rebooted. This is due to the vdisk corresponding to the volume does not come back online after the control domain reboots.

This issue has been identified and fixed under Oracle (Sun) bug id: 6795836 (vd_setup_vd() should handle errors from vd_identify_dev() better)

This Oracle (Sun) bug is fixed in Oracle (Sun) patch 141777-01.

Veritas Cluster Server support for using CVM with multiple nodes in a Oracle VM Server for SPARC environment

This chapter includes the following topics:

- [Clustering using Cluster Volume Manager](#)
- [Installing Storage Foundation on multiple nodes in a Logical Domain](#)
- [Cluster Volume Manager in the control domain for providing high availability](#)

Clustering using Cluster Volume Manager

The Veritas Volume Manager cluster functionality (CVM) makes logical volumes and raw device applications accessible throughout a cluster.

In the split Storage Foundation model, CVM is placed in the control domain and VxFS is placed in the guest domain. In this model, CVM provides high availability and shared storage visibility at the control domain level across multiple physical nodes in the cluster.

See [“Cluster Volume Manager in the control domain for providing high availability”](#) on page 107.

In the guest-based Storage Foundation stack model, CVM is placed in the guest domain, providing high availability and shared storage visibility at the guest domain level across multiple guest domains that act as the nodes in the cluster.

Installing Storage Foundation on multiple nodes in a Logical Domain

To install Storage Foundation on multiple nodes in a Solaris Logical Domains environment, you must complete the following operations, which are the same as on a single node:

- See [“Installing and configuring Oracle VM Server for SPARC and domains”](#) on page 80.
- See [“Installing Storage Foundation in the control domain or guest”](#) on page 80.
- See [“Installing Veritas File System in the guest domain using pkgadd”](#) on page 85.
- See [“Verifying the configuration”](#) on page 86.

Reconfiguring the clustering agents for Cluster Volume Manager

This section applies to only the Split Storage Foundation model.

For a Storage Foundation CVM, the following additional configuration steps are necessary:

- See [“Removing the vxfsckd resource”](#) on page 105.
- See [“Creating CVMVolDg in a group”](#) on page 106.

Removing the vxfsckd resource

After configuring Storage Foundation and CVM, remove the vxfsckd resource.

To remove the vxfsckd resource

- 1 Make the configuration writeable:

```
# haconf -makerw
```

- 2 Delete the resource:

```
# hares -delete vxfsckd
```

3 Make the configuration read-only:

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

4 Stop the resources:

```
# hstop -all
```

5 Restart the resources.

```
# hstart
```

Run the `hstart` command on all nodes in the cluster.

Creating CVMVoIDg in a group

The following procedure creates CVMVoIDg in a given group.

To create CVMVoIDg

1 Make the configuration writeable:

```
# haconf -makerw
```

2 Add the CVMVoIDg resource:

```
# hares -add name_of_resource CVMVoIDg name_of_group
```

3 Add a diskgroup name to the resource:

```
# hares -modify name_of_resource CVMDiskGroup diskgroup_name
```

4 Make the attribute local to the system:

```
# hares -local name_of_resource CVMActivation
```

5 Add the attribute to the resource.

```
# hares -modify name_of_resource CVMActivation \  
activation_value -sys nodename
```

Repeated this step on each of the nodes.

6 If you want to monitor volumes, enter the following command:

```
# # hares -modify name_of_resource CVMVolume -add \  
name_of_volume
```

In a database environment, Symantec recommends you monitor volumes.

7 Modify the resource so that a failure of this resource does not bring down the entire group:

```
# hares -modify name_of_resource Critical 0
```

8 Enable the resources:

```
# hares -modify name_of_resource Enabled 1
```

9 Make the configuration read-only:

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

10 Verify the configuration:

```
# hacf -verify
```

This should put the resource in the `main.cf` file.

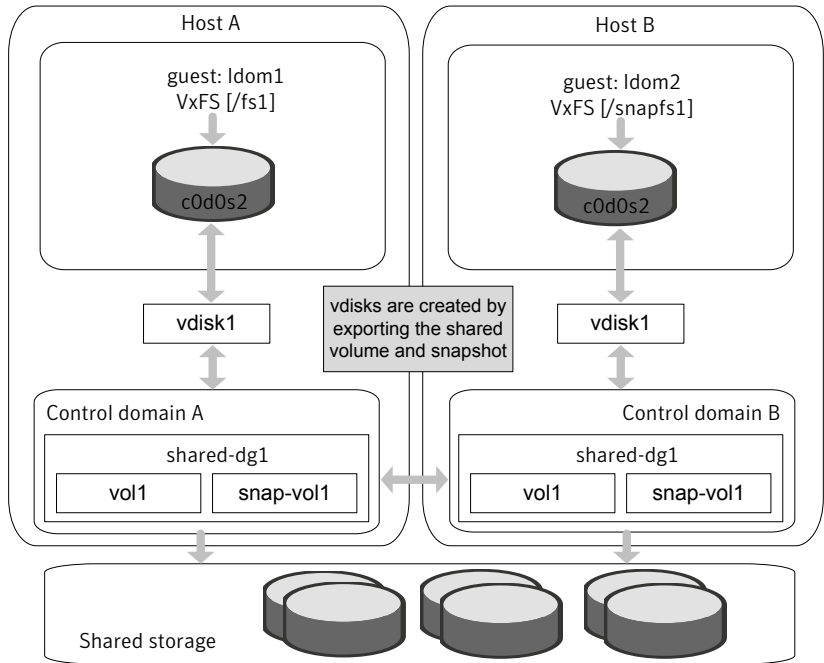
Cluster Volume Manager in the control domain for providing high availability

The main advantage of clusters is protection against hardware failure. Should the primary node fail or otherwise become unavailable, applications can continue to run by transferring their execution to standby nodes in the cluster.

CVM can be deployed in the control domains of multiple physical hosts running Oracle VM Server for SPARC, providing high availability of the control domain.

[Figure 5-1](#) illustrates a CVM configuration.

Figure 5-1 CVM configuration in an Oracle VM Server for SPARC environment



If a control domain encounters a hardware or software failure causing the domain to shut down, all applications running in the guest domains on that host are also affected. These applications can be failed over and restarted inside guests running on another active node of the cluster.

Caution: As such applications running in the guests may resume or time out based on the individual application settings. The user must decide if the application must be restarted on another guest on the failed-over control domain. There is a potential data corruption scenario if the underlying shared volumes get accessed from both of the guests simultaneously.

Shared volumes and their snapshots can be used as a backing store for guest domains.

Note: The ability to take online snapshots is currently inhibited because the file system in the guest cannot coordinate with the VxVM drivers in the control domain.

Make sure that the volume whose snapshot is being taken is closed before the snapshot is taken.

The following example procedure shows how snapshots of shared volumes are administered in such an environment. In the example, `datavol1` is a shared volume being used by guest domain `ldom1` and `c0d1s2` is the front end for this volume visible from `ldom1`.

To take a snapshot of `datavol1`

- 1 Unmount any VxFS file systems that exist on `c0d1s0`.
- 2 Stop and unbind `ldom1`:

```
primary# ldm stop ldom1
```

```
primary# ldm unbind ldom1
```

This ensures that all the file system metadata is flushed down to the backend volume, `datavol1`.

- 3 Create a snapshot of `datavol1`.

See the *Veritas Storage Foundation Administrator's Guide* for information on creating and managing third-mirror break-off snapshots.

- 4 Once the snapshot operation is complete, rebind and restart `ldom1`.

```
primary# ldm bind ldom1
```

```
primary# ldm start ldom1
```

- 5 Once `ldom1` boots, remount the VxFS file system on `c0d1s0`.

Veritas Cluster Server: Configuring Oracle VM Server for SPARC for high availability

This chapter includes the following topics:

- [About Veritas Cluster Server in a Oracle VM Server for SPARC environment](#)
- [About Veritas Storage Foundation Cluster File System in an Oracle VM Server for SPARC environment](#)
- [About Veritas Cluster Server configuration models in an Oracle VM Server for SPARC environment](#)
- [About configuring Veritas Cluster Server for Oracle VM Server for SPARC with multiple I/O domains](#)
- [Configuring Veritas Cluster Server to fail over a logical domain on a failure](#)
- [Configuring Veritas Cluster Server to fail over an application on a failure](#)
- [Oracle VM Server for SPARC guest domain migration in VCS environment](#)
- [Configuring VCS to manage a Logical Domain using services from multiple I/O domains](#)
- [Configuring VCS on logical domains to manage applications using services from multiple I/O domains](#)

About Veritas Cluster Server in a Oracle VM Server for SPARC environment

Veritas Cluster Server (VCS) provides high availability (HA) for Oracle VM Server for SPARC. You can configure VCS to monitor the Logical Domain, services to the Logical Domain, and the applications that run in logical domain, or to monitor only the applications that run in the logical domain.

See [“About Veritas Cluster Server configuration models in an Oracle VM Server for SPARC environment”](#) on page 115.

[Table 6-1](#) lists the failure scenarios and the VCS failover options based on which you can plan your VCS configuration in an Oracle VM Server for SPARC environment.

Table 6-1 Veritas Cluster Server failover options for logical domain failure

Failure scenario	VCS failover	Typical VCS configuration
Logical domains, their storage, or switches fail	VCS fails over the logical domain from one node to another node	VCS is installed in the control domain of each node. See “Veritas Cluster Server setup to fail over a logical domain on a failure” on page 115.
Logical domains, their storage, or switches fail Or Applications that run in logical domains fail	VCS fails over the logical domain from one node to another node. The application starts on the same logical domain after the logical domain failover.	VCS is installed in the control domain of each node, and single node VCS is installed on each guest domain. See “Veritas Cluster Server setup to fail over a logical domain on a failure” on page 115.
Applications that run in logical domains fail Or logical domain where the application is running fails	VCS fails over the application from one logical domain to another.	VCS is installed in the guest domain of each node. See “Veritas Cluster Server setup to fail over an application on a failure” on page 118.

Dynamic reconfiguration of memory and CPU of a guest domain

VCS supports dynamic reconfiguration of memory and CPU assigned to a guest domain. Modify the values of the Memory and NumCPU attributes of an LDom resource to dynamically reconfigure the memory and CPU of a guest domain.

Veritas Cluster Server prerequisites

This document assumes a working knowledge of VCS.

Review the prerequisites in the following documents to help ensure a smooth VCS installation:

- *Veritas Cluster Server Release Notes*
Find this in the `cluster_server/docs` directory of the product disc.
- *Veritas Cluster Server Installation Guide*
Find this in the `cluster_server/docs` directory of the product disc.

Unless otherwise noted, all references to other documents refer to the Veritas Cluster Server documents version 6.0 for Solaris.

Veritas Cluster Server requirements

For installation requirements:

See [“System requirements”](#) on page 76.

For the configuration model where VCS is installed on the control domain:

- VCS requires shared storage that is visible across all the nodes in the cluster.
- Configure Logical Domain on each cluster node.
- The logical domain’s boot device and application data must reside on shared storage.

For the configuration model where VCS is installed on the guest domain:

- VCS requires the application data to reside on the shared storage.
- Each node can have more than one logical domain.
- Each logical domain can have its own boot device.

Veritas Cluster Server limitations

Depending on the configuration model, the following limitations apply to VCS in an Oracle VM Server for SPARC environment.

Limitations for VCS configuration in the control domain:

- Each logical domain configured under VCS must have at least two VCPUs. With one VCPU, the control domain always registers 100% CPU utilization for the logical domain. This is an Oracle VM Server for SPARC software issue.

Limitation for VCS configuration in the guest domain when disk-based fencing is configured:

- Do not export a physical device to more than one guest domains on the same physical node. For example, if you configure I/O fencing in a guest domain, and if one of the guest domains dies, then I/O fencing fences off the other guest domains as well.
- For co-ordinator disks and data disks, disks must be given from different HBAs if more than one guest domains forming the cluster are residing on the same physical host.

Note: If you plan to use disk-based I/O fencing inside a guest domain along with OVM live migration feature, ensure that the DMP version inside the guest domain is at least 6.0.5.

- The disks exported for co-ordinator disks and data disks must be full disks and SCSI compliant.

Veritas Cluster Server known issues

The following section describes known issues with VCS in a Oracle VM Server for SPARC environment.

Shutting down the control domain may cause the guest domain to crash (1631762)

Set up

Two Oracle SPARC Enterprise T5240 server physical boxes, each with a control domain and a guest domain. The guest domains in each of the physical boxes form a two node cluster. The nodes are named node 0 and node 1 in the following text.

Symptom

Gracefully shutting down the control domain of node 0 causes the guest domain of node 0 to crash.

Analysis

Even though the guest domain can continue to function when the control domain is shut down, the heartbeats between node 0 and node 1 are lost as the control domain shuts down. As a result, the cluster forms two separate sub-clusters without the sub-clusters being able to see each others' heartbeats.

I/O fencing resolves the split brain situation and determines that only one sub-cluster will continue to function while the other sub-cluster should panic. Therefore, the panic of node 0 is expected behavior.

Resolution:

None; this is expected behavior. However, Symantec recommends keeping the control domain highly available for the proper function of the SFCFS and SFRAC stack in the guest domains.

If you have set up a virtual private LLT heartbeats between the two guests (node 0 and node1), the guest will not crash.

About Veritas Storage Foundation Cluster File System in an Oracle VM Server for SPARC environment

Veritas Storage Foundation Cluster File System (SFCFS) allows clustered servers to mount and use a file system simultaneously as if all applications using the file system were running on the same server for a Oracle VM Server for SPARC.

Veritas Storage Foundation Cluster File System limitations

Depending on the configuration model, the following limitations apply to using SFCFS in an Oracle VM Server for SPARC environment.

Limitations for SFCFS configuration in the guest domain:

- There is no support for replicating a shared disk group using VVR, when one or more guest domains share the disk group.
- If you want to configure I/O fencing in guest domain, then do not export physical devices to more than one guest domain on the same physical node. Otherwise, I/O fencing fences off the device whenever one of the guest domain dies. This situation causes the other guest domains also to lose access to the device.

Symantec recommends you to disable I/O fencing if you exported the same physical device to multiple guest domains.

About Veritas Cluster Server configuration models in an Oracle VM Server for SPARC environment

When you configure VCS in an Oracle VM Server for SPARC environment, you need some specific information about the logical domain, network, and the storage devices that the logical domain requires to run.

You need to know the following information about your logical domain:

- The logical domain's name
- The names of the primary network interfaces for each node
- The virtual switch that the logical domain uses
- The name and type of storage that the logical domain uses

VCS configuration depends on whether you want VCS to fail over the logical domain or the application on a failure:

- [Veritas Cluster Server setup to fail over a logical domain on a failure](#)
- [Veritas Cluster Server setup to fail over an application on a failure](#)

See [“About Veritas Cluster Server in a Oracle VM Server for SPARC environment”](#) on page 111.

Veritas Cluster Server setup to fail over a logical domain on a failure

You can configure VCS to monitor the logical domain, or to monitor both the logical domain and the application in the guest domain.

- See [“Veritas Cluster Server installed in the control domain to monitor logical domains”](#) on page 116.
- See [“Veritas Cluster Server installed in the control domain to monitor the applications in the guest domain”](#) on page 117.

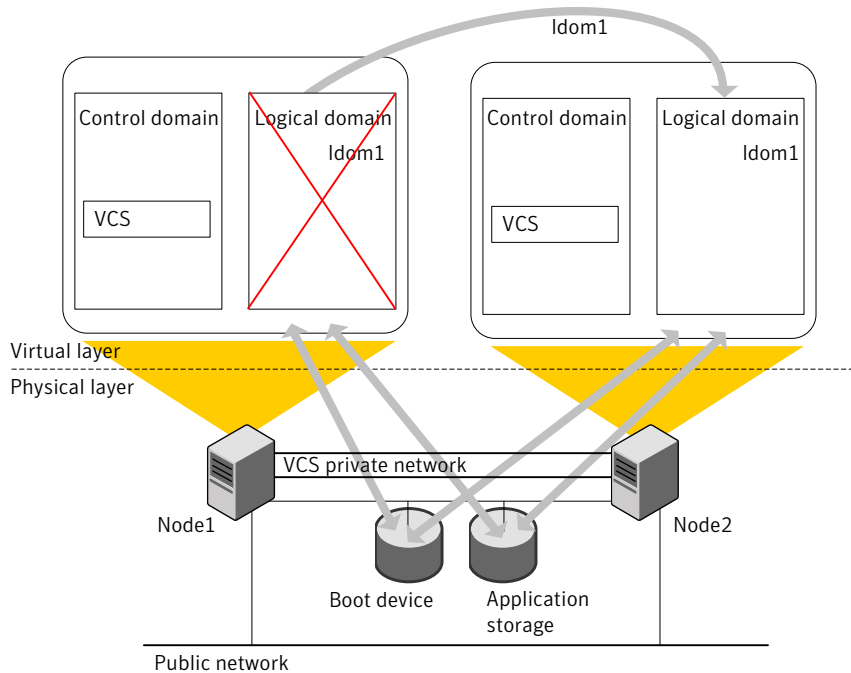
When you configure VCS in the Oracle VM Server for SPARC environment, VCS monitors the health of the logical domain, and its supporting components. VCS uses the LDom agent to monitor the logical domain and storage and network agents to monitor storage and network services. If the agent detects that the logical domain resource has failed, VCS moves the service group that contains the logical domain resource to another node in the cluster.

See the *Veritas Cluster Server Bundled Agents Reference Guide* for information on the logical domain agent and other bundled agents.

Veritas Cluster Server installed in the control domain to monitor logical domains

Figure 6-1 illustrates a typical setup where VCS installed in the control domain provides high availability to the logical domain and its resources.

Figure 6-1 Typical setup for logical domain high availability with VCS installed in control domains



A typical two-node VCS configuration for logical domain high availability has the following software and hardware infrastructure:

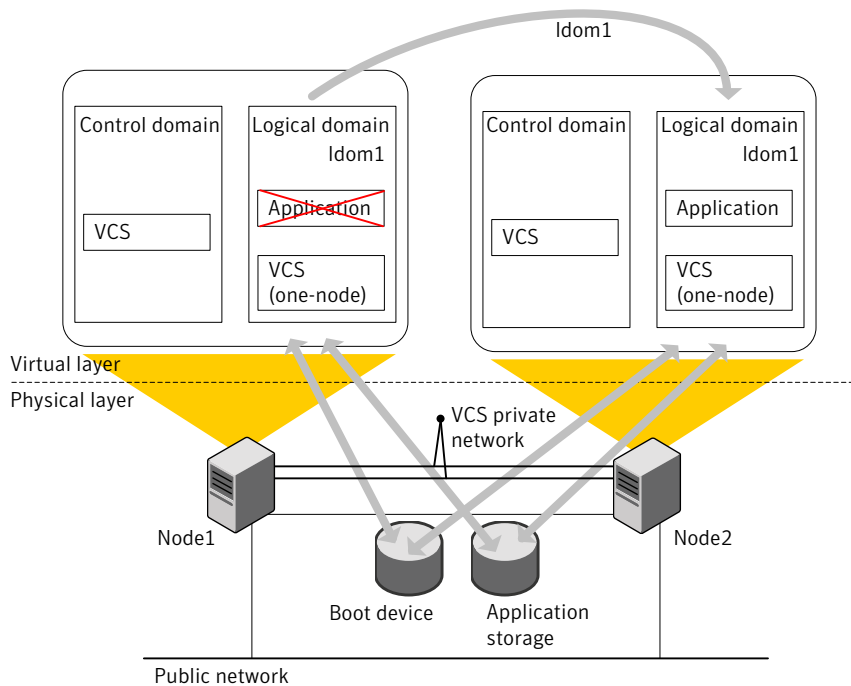
- Oracle VM Server for SPARC software is installed on each system Node1 and Node2.
- Shared storage is attached to each system.
- A logical domain (ldom1) exists on both the nodes with a shared boot device.
- Each LDom has an operating system installed.

- VCS is installed in the control domains of each node.

Veritas Cluster Server installed in the control domain to monitor the applications in the guest domain

Figure 6-2 illustrates a typical setup where VCS installed in the control domain provides high availability to applications that run in the guest domains.

Figure 6-2 Typical setup for application high availability with VCS installed in control domains



A typical two-node VCS configuration that fails over the logical domains to keep the applications that run in them highly available has the following infrastructure:

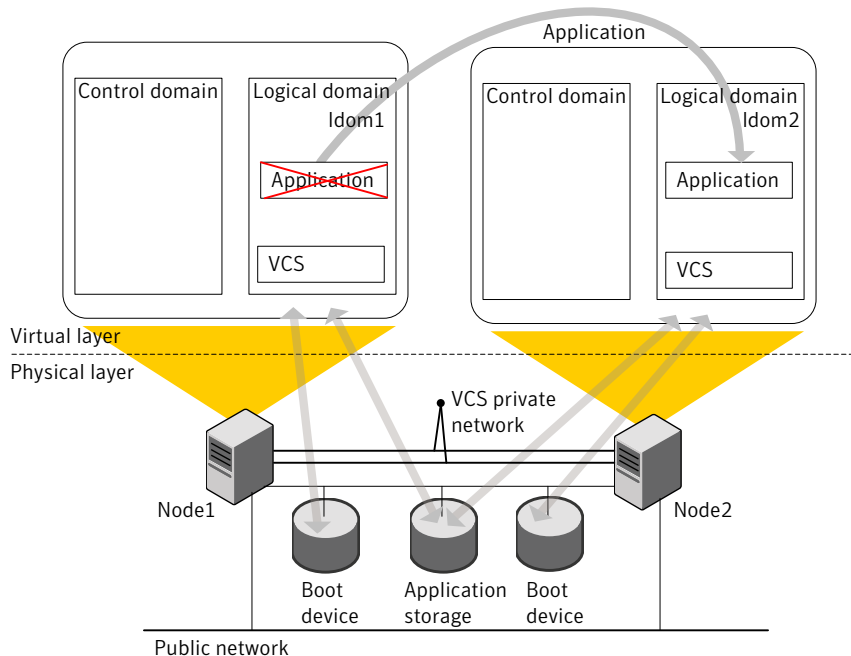
- Oracle VM Server for SPARC software is installed on each system Node1 and Node2.
- Shared storage is attached to each system.
- A logical domain (logical domain1) with same configuration details exists on both the nodes with a shared boot device.
- Each logical domain has an operating system installed.

- VCS is installed in the control domains of each node.
- Each guest domain has single-node VCS installed. VCS kernel components are not required.
- VCS service group exists inside the guest domain for the application running inside the guest domain. VCS in the control domain uses this service group to manage the applications using RemoteGroup resource.
- Service group with RemoteGroup resource exists in the control domain to monitor the application service group in the guest domain.
- Service group with LDom resource exists in the control domain.
- An online global firm dependency exists from RemoteGroup service group to the LDom service group.

Veritas Cluster Server setup to fail over an application on a failure

Figure 6-3 illustrates a typical VCS setup to provide high availability for applications that run in guest domains.

Figure 6-3 Typical setup for applications high availability with Veritas Cluster Server installed in guest domains



A typical two-node VCS configuration that fails over the applications to keep the applications that run in logical domains highly available has the following infrastructure:

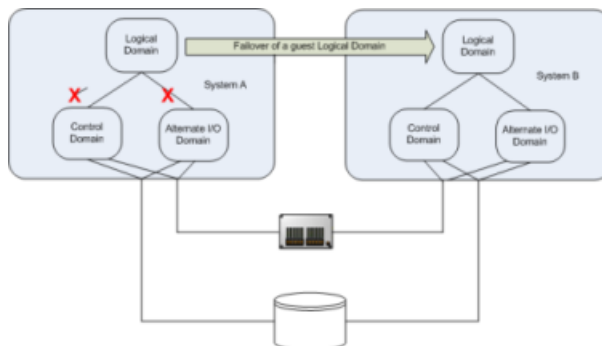
- Oracle VM Server for SPARC software is installed on each system Node1 and Node2.
- Shared storage is attached to each system.
- Logical domains are created on both the nodes that may have local boot devices.
- Each LDom has an operating system installed.
- VCS is installed in the guest domains of each node.

Note: When the guest domains are forming a cluster and I/O fencing is enabled, live migration of the guest domain is not supported.

About configuring Veritas Cluster Server for Oracle VM Server for SPARC with multiple I/O domains

With Oracle VM Server for SPARC virtualization technology, you can create multiple I/O domains (control domain and alternate I/O domain) to provide redundant storage and network services to a guest Logical Domain. A typical cluster-setup configured to use multiple I/O domains has two physical systems. On each physical system, the control domain and alternate I/O domain provide I/O services from back-end storage and network devices to a Logical Domain.

Figure 6-4 with guest Logical Domain on System A and System B using storage and network services from the control domain and alternate I/O domain



If there is a failure of storage or network services from one of the domains, the guest Logical Domain continues to function on the same physical system because it gets I/O services from the other I/O domain. However, when there is failure of services from both the I/O domains on a physical system, the Logical Domain on the physical system fails.

Configure VCS on multiple I/O domains to manage a Logical Domain. VCS fails over the Logical Domain from one system to a Logical Domain on another system when services from both the domains fail.

Note: Failover happens when I/O services from all I/O domains fail or the control domain goes down.

About Alternate I/O domain

Alternate I/O domain, is an Oracle technology available on Oracle VM server for SPARC, which provides highly available storage and network services to guest domains on a physical system.

Setting up the Alternate I/O domain

While setting up a system to support Logical Domain, the control domain (primary domain) owns all the I/O devices on the physical system. To create alternate I/O domain, you need to relinquish ownership of one of the PCI Express bus from control domain and assign it to a Logical Domain.

For more information on creating alternate I/O domain, refer to the Oracle Sun documentation.

Configuring VCS to manage a Logical Domain with multiple I/O domains

Proceed to configure VCS. See [“Configuring VCS to manage a Logical Domain using services from multiple I/O domains”](#) on page 137.

Configuring Veritas Cluster Server to fail over a logical domain on a failure

You can configure VCS to keep the logical domains highly available. In addition to monitoring the logical domain, you can also configure VCS to monitor the applications that run in them.

You need to perform additional steps for VCS in the control domain to manage applications in the guest domains. After you install VCS in the control domain, you must create separate service groups for the RemoteGroup resource and logical domain resource with online global firm dependency.

Note: If you create the RemoteGroup resource as part of the logical domain service group, then the RemoteGroup resource state remains as UNKNOWN if the logical domain is down. VCS does not probe the service group and cannot bring the logical domain online. The online global firm dependency between the service groups allows VCS to fail over a faulted child logical domain service group independent of the state of the parent RemoteGroup service group.

Perform the following tasks to configure VCS to fail over a logical domain due to its failure:

- Review the configuration scenarios
See [“Configuration scenarios”](#) on page 122.
- Configure logical domains
See [“Configuring logical domain”](#) on page 124.
- Install VCS on control domain
See [“Installing Veritas Cluster Server inside the control domain”](#) on page 124.
- Create VCS service group for the logical domain
See [“Creating the Veritas Cluster Server service groups for logical domains”](#) on page 125.

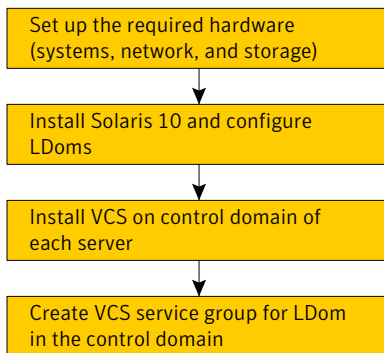
Perform the following additional tasks to configure VCS to fail over a logical domain on an application failure:

- Install single-node VCS on guest domain
See [“Installing single-node Veritas Cluster Server inside the guest domain”](#) on page 125.
- Configure VCS in control domain to monitor the application in guest domain
See [“Configuring Veritas Cluster Server to monitor the application in the guest domain”](#) on page 125.

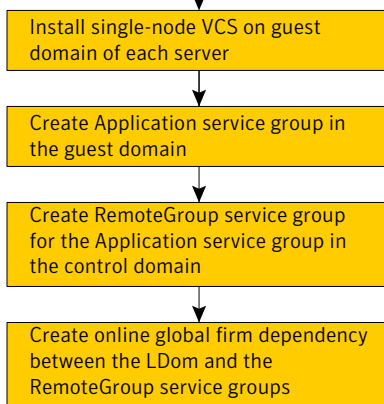
Figure 6-5 depicts the workflow to configure VCS to manage the failure of a logical domain or the failure of an application that runs in an logical domain.

Figure 6-5 Workflow to configure VCS to fail over a logical domain on a failure

For VCS to monitor the LDom:



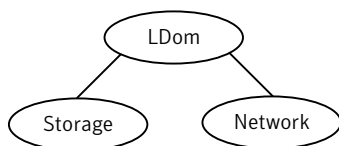
For VCS to monitor the application in guest domain:



Configuration scenarios

Figure 6-6 shows the basic dependencies for an logical domain resource.

Figure 6-6 A logical domain resource depends on storage and network resources



Network configuration

Use the NIC agent to monitor the primary network interface, whether it is virtual or physical. Use the interface that appears using the `ifconfig` command.

Recommended network device to monitor.

[Figure 6-7](#) is an example of an logical domain service group. The logical domain resource requires both network (NIC) and storage (Volume and DiskGroup) resources.

See the *Veritas Cluster Server Bundled Agents Reference Guide* for more information about the NIC agent.

Storage configurations

Depending on your storage configuration, use a combination of the Volume, DiskGroup, and Mount agents to monitor storage for logical domains.

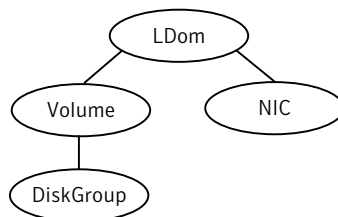
Note: Symantec recommends volumes or flat files in volumes that are managed by VxVM for LDom storage for configurations where VCS is in control domain.

Veritas Volume Manager exposed volumes

Veritas Volume Manager (VxVM) exposed volumes is the recommended storage solution for VCS in a control domain configuration. Use the Volume and DiskGroup agents to monitor a VxVM volume. VCS with VxVM provides superior protection for your highly available applications.

[Figure 6-7](#) shows an logical domain resource that depends on a Volume and DiskGroup resource.

Figure 6-7 The logical domain resource can depend on resources such as NIC, Volume, and DiskGroup depending on the environment



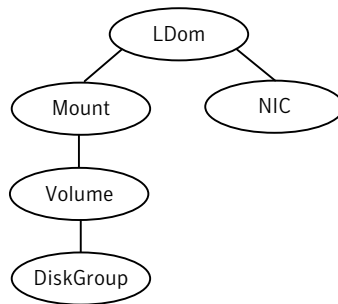
For more information about the Volume and DiskGroup agents, refer to the *Veritas Cluster Server Bundled Agents Reference Guide*.

Image files

Use the Mount, Volume, and DiskGroup agents to monitor an image file.

Figure 6-8 shows how the LDom resource depends on the resources with image file.

Figure 6-8 The LDom resource in conjunction with different storage resources



See the *Veritas Cluster Server Bundled Agents Reference Guide* for more information about the Mount agent.

Configuring logical domain

You must perform the following steps to configure logical domain.

To configure logical domain

- 1 Make sure that the network and storage setup meet the VCS requirements.
See “[Veritas Cluster Server requirements](#)” on page 112.
- 2 Make sure that you have installed the required Solaris operating system in the logical domain.
- 3 Create logical domain (ldom1) on each system with an identical configuration and boot device.

Installing Veritas Cluster Server inside the control domain

You must install VCS in the control domain of each system.

To install Veritas Cluster Server inside the control domain

- ◆ Install and configure VCS in the control domain of each system.

The process for installing VCS in the control domain is very similar to the regular installation of VCS. However, you must specify the host name of the control domain for the name of the host where you want to install VCS.

See the *Veritas Cluster Server Installation Guide* for VCS installation and configuration instructions.

Creating the Veritas Cluster Server service groups for logical domains

You can also create and manage service groups using the Veritas Operations Manager (VOM), or through the command line.

See the *Veritas Cluster Server Administrator's Guide* for complete information about using and managing service groups.

Configuring Veritas Cluster Server for application monitoring

You must perform the following procedures to configure VCS to monitor the application in the guest domain:

- See [“Installing single-node Veritas Cluster Server inside the guest domain”](#) on page 125.
- See [“Configuring Veritas Cluster Server to monitor the application in the guest domain”](#) on page 125.

Installing single-node Veritas Cluster Server inside the guest domain

Perform the following steps to install one-node Veritas Cluster Server (VCS) inside the guest domain:

To install and configure one-node Veritas Cluster Server inside the logical domain

- 1 Install one-node VCS (no kernel components required) in the guest domain.
See the *Veritas Cluster Server Installation Guide* to perform a single-node VCS installation in the logical domains.
- 2 Start the VCS engine.

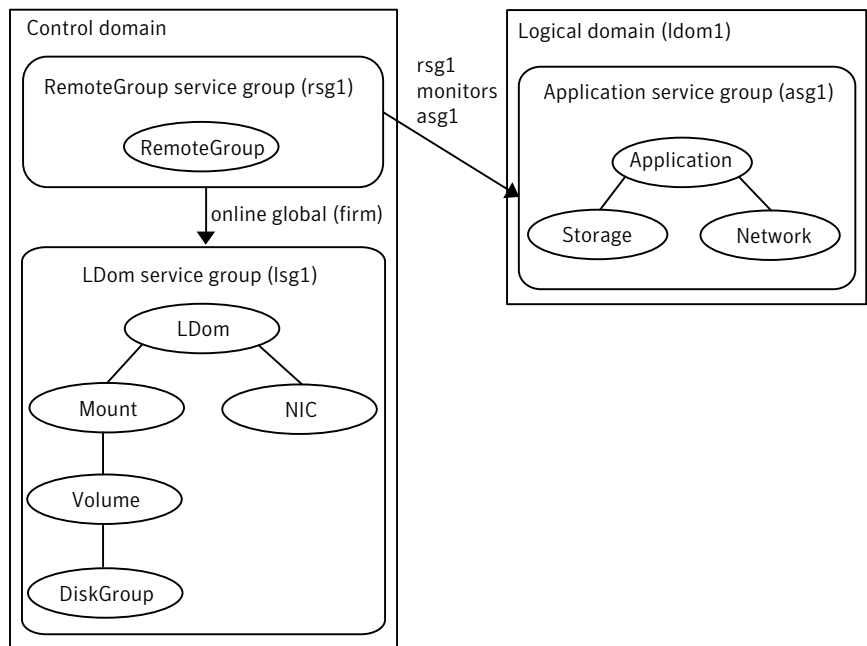
Configuring Veritas Cluster Server to monitor the application in the guest domain

Perform the following steps to configure Veritas Cluster Server (VCS) in the control domain to monitor the application in the guest domain.

To configure Veritas Cluster Server to monitor the application in the guest domain

- 1 Configure a VCS service group (lsg1) for the application.
 The ManualOps attribute of the service group must remain set to true, the default value.
- 2 Add a VCS user (lsg1 -admin) with the minimum privilege as the group operator of the VCS service group (lsg1).
- 3 Configure a RemoteGroup service group (rsg1) in the control domain to monitor the VCS service group (lsg1) that was configured in 1.
- 4 Set the value of the following RemoteGroup resource attributes as follows:
 - RestartLimit attribute of the resource or type to 1 or higher
 - OfflineWaitLimit attribute of the resource or type to 1
 - ToleranceLimit attribute of the resource or type to 1
- 5 Create the dependencies between the groups and resources as shown in [Figure 6-9](#).

Figure 6-9 Group and resource dependency diagram



Note: RemoteGroup and Application service groups are required only when you want to configure VCS to monitor the application in the guest domain.

RemoteGroup resource definition

The resource definition for the RemoteGroup resource is as follows:

```
RemoteGroup rsg1 (  
    GroupName = lsg1  
    IpAddress = <IP address of ldom1>  
    ControlMode = OnOff  
    Username = lsg1-admin  
    Password = <lsg1-admin's password>  
)
```

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

Verifying a logical domain service group failover

Verify the configuration in different situations.

Using hagrp - switch command

Switch the logical domain service group to another node in the cluster to make sure the service group fails over. If all the resources are properly configured, the service group shuts down on the first node and comes up on the second node.

Other verification scenarios

In all of these verification scenarios, you are stopping or moving an logical domain, or stopping a resource for that logical domain. VCS should detect the failure, or the movement, and either failover the effected logical domain or take no action.

The following list presents some quick testing scenarios;

- From outside of VCS control, stop the logical domain. VCS should fail the logical domain over to the other node.
- Boot the logical domain through VCS by entering a `hagrp -online` command. move the logical domain to another node by shutting it down through VCS on the node where the logical domain is running. Boot the logical domain outside of VCS control on the other node- the service group comes online on that node.

Configuring Veritas Cluster Server to fail over an application on a failure

You must install and configure Veritas Cluster Server (VCS) in the guest domains of each system to enable VCS to manage applications in the guest domains.

To configure Veritas Cluster Server to manage applications in the guest domains

- 1 Install and configure VCS in the guest domains of each system.
See the *Veritas Cluster Server Installation Guide* for installation and configuration instructions.
- 2 Create two virtual NICs using private virtual switches for private interconnects.
You can configure virtual switches with no physical network interfaces if you want the failover across logical domains in the same control domain.
- 3 Configure VCS service group for the application that you want to monitor.
 - Configure Mount and Disk resources to monitor the storage.
 - Configure NIC resources to monitor the network.
 - Configure application resources using the application-specific agent.

See the *Veritas Cluster Server Administrator's Guide* for more details on configuring applications and resources in VCS.

See the *Veritas Cluster Server Bundled Agents Reference Guide* for details on the storage and networking bundled agents.

Oracle VM Server for SPARC guest domain migration in VCS environment

VCS supports cold, warm, and live migration, also known as domain migration, for Oracle VM Server for SPARC guest domains.

Domain migration enables you to migrate a guest domain from one host system to another host system. The system on which the migration is initiated is the source system. The system to which the domain is migrated is the target system.

While a migration operation is in progress, the domain that you want to migrate transfers from the source system to the migrated domain on the target system.

The domain migration until Oracle VM Server for SPARC 2.0 release is a warm migration. A warm migration is where the domain that you want to migrate enters a suspended state before the migration.

The Oracle VM Server for SPARC 2.1 introduces live migration, which provides performance improvements that enable an active domain to migrate while it continues to run.

In addition to live migration, you can migrate bound or inactive domains. This migration is a cold migration.

You can use domain migration to perform tasks such as the following:

- Balancing the load between systems
- Performing hardware maintenance while a guest domain continues to run

Overview of a warm migration

The Logical Domains Manager on the source system accepts the request to migrate a domain and establishes a secure network connection with the Logical Domains Manager running on the target system. Once this connection has been established, the migration occurs.

The migration operation occurs in the following phases:

- Phase 1** After connecting with the Logical Domains Manager running in the target host, information about the source system and domain are transferred to the target host. The Logical Domains Manager on the target host uses this information to perform a series of checks to determine whether a migration is possible. The checks differ depending on the state of the source domain. For example, if the source domain is active the Logical Domains Manager performs a different set of checks than if the domain is bound or inactive.
- Phase 2** When all checks in Phase 1 have passed, the source and target systems prepare for the migration. The Logical Domains Manager on the source suspends the source domain. On the target system, the Logical Domains Manager creates a domain to receive the source domain.
- Phase 3** For an active domain, the next phase is to transfer all the runtime state information for the domain to the target. The Logical Domains Manager retrieves this information from the hypervisor. On the target, the Logical Domains Manager installs the state information in the hypervisor.
- Phase 4** Handoff—after all state information is transferred, the handoff occurs when the target domain resumes execution (if the source was active). The Logical Domain Manager on the source destroys the source domain. From this point on, the target domain is the sole version of the domain running.

Overview of a live migration

The Logical Domains Manager on the source system accepts the request to migrate a domain and establishes a secure network connection with the Logical Domains Manager that runs on the target system. The migration occurs after this connection has been established.

The migration operation occurs in the following phases:

- Phase 1** After the source system connects with the Logical Domains Manager that runs in the target system, the Logical Domains Manager transfers information about the source system and the domain to be migrated to the target system. The Logical Domains Manager uses this information to perform a series of checks to determine whether a migration is possible. The Logical Domains Manager performs state-sensitive checks on the domain that you plan to migrate. The checks it performs is different for an active domain than for bound or inactive ones.
- Phase 2** When all checks in Phase 1 have passed, the source and target systems prepare for the migration. On the target system, the Logical Domains Manager creates a domain to receive the domain. If the domain that you plan to migrate is inactive or bound, the migration operation proceeds to Phase 5.
- Phase 3** If the domain that you want to migrate is active, its runtime state information is transferred to the target system. The domain continues to run, and the Logical Domains Manager simultaneously tracks the modifications that the operating system makes to this domain. The Logical Domains Manager on the source retrieves this information on the source from the source hypervisor and sends the information to the Logical Domains Manager on the target. The Logical Domains Manager on the target installs this information in the hypervisor for the target.
- Phase 4** The Logical Domains Manager suspends the domain that you want to migrate. At this time, all of the remaining modified state information is re-copied to the target system. In this way, there should be little or no perceivable interruption to the domain. The amount of interruption depends on the workload.
- Phase 5** A handoff occurs from the Logical Domains Manager on the source system to the Logical Domains Manager on the target system. The handoff occurs when the migrated domain resumes execution (if the domain to be migrated was active), and the domain on the source system is destroyed. From this point forward, the migrated domain is the sole version of the domain running.

With Oracle VM Server for SPARC 2.1, the default domain migration attempted is Live Migration. If the installed version of Oracle VM Server for SPARC is 2.0, the domain migration defaults to warm migration. For more details on supported configurations, read Chapter 9, Migrating Domains in the *Oracle® VM Server for SPARC 2.1 Administration Guide*.

VCS supports the following three varieties of Oracle VM Server for SPARC domain migration:

- Guest migration from one VCS node to other VCS node in the cluster, for example:

```
sys1# ldm migrate ldom1 sys2
```

- Guest migration from a VCS node to a non-VCS node, for example:

```
sys1# ldm migrate ldom1 sys3
```

- Renaming the logical domain during the migration to the target system, for example:

```
sys1# ldm migrate ldom1 sys2:ldom2
```

Prerequisites before you perform domain migration

Perform the following prerequisites before you perform a domain migration:

- Verify that the value of IntentionalOffline attribute for LDom type is 1. Note that 1 is the default value.
- Make sure that the LDom resource for the LDom that you plan to migrate is in an ONLINE or OFFLINE steady state.
- To rename the logical domain when you migrate it, make sure that the LDomName attribute for the LDom resource in VCS is localized with the target LDom name for the target node. When you rename it, VCS can continue to monitor the LDom after migration.
- Make sure that CfgFile attribute for LDom is configured before migration.
- Make sure that RemoveLDomConfigForMigration attribute is set before migration. If this attribute is set, the LDom Agent removes the LDom configuration from the system on which an offline or clean is called. This arrangement helps in the scenario when a LDom is failed-over to a target node and is migrated back to the source node. In the presence of LDom configuration on the source node, migration would not be possible. Refer to the LDom Agent attribute description in the *Bundled Agents Reference Guide* for Solaris for more information.

Supported deployment models for Oracle VM Server for SPARC domain migration with VCS

The following are the supported deployment models for Oracle VM Server for SPARC domain migration with VCS:

- See “[Migrating Oracle VM guest when VCS is installed in the control domain that manages the guest domain](#)” on page 132.
- See “[Migrating Oracle VM guest when VCS is installed in the control domain and single-node VCS is installed inside the guest domain to monitor applications inside the guest domain](#)” on page 133.
- See “[Migrating Oracle VM guest when VCS cluster is installed in the guest domains to manage applications for Oracle VM Server for SPARC version 2.1 and above](#)” on page 135.
- See “[Migrating Oracle VM guest when VCS cluster is installed in the guest domains to manage applications for Oracle VM Server for SPARC version 2.0](#)” on page 135.

Migrating Oracle VM guest when VCS is installed in the control domain that manages the guest domain

Use the following information to migrate when you have VCS installed in the control domain that manages the guest domain.

To perform a migration of an LDom when VCS is installed in the control domain

- ◆ Use the `ldm` command for migration.

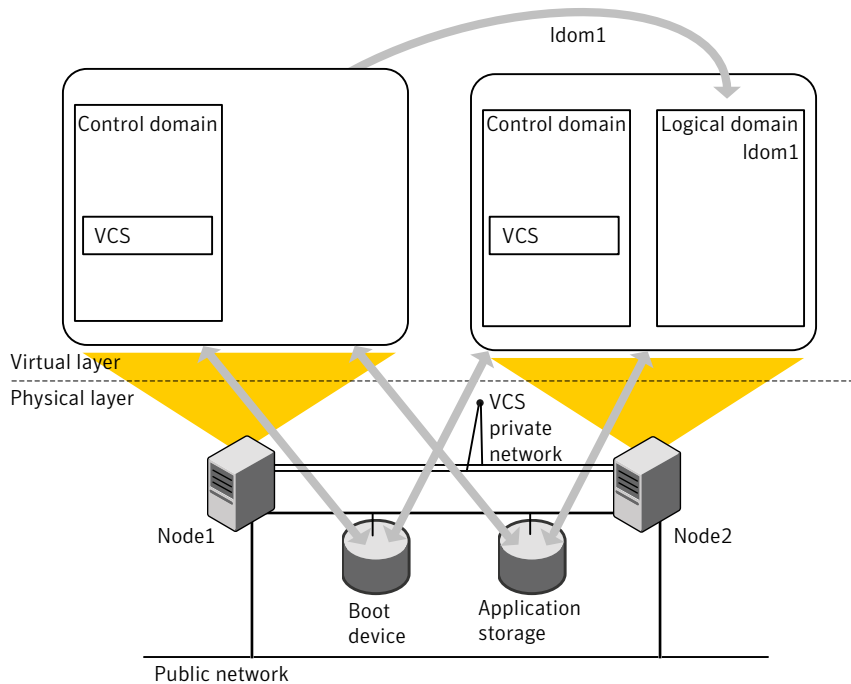
```
ldm migrate [-f] [-n] [-p password_file] source_ldom \  
[user@target_host[:target_ldom]
```

For example:

```
Sys1# ldm migrate ldom1 Sys2
```

[Figure 6-10](#) illustrates a logical domain migration when VCS is clustered between control domains.

Figure 6-10 Logical domain migration when VCS is clustered between control domains



Migrating Oracle VM guest when VCS is installed in the control domain and single-node VCS is installed inside the guest domain to monitor applications inside the guest domain

Use the following information to migrate when you have:

- VCS installed in the control domain
- The VCS in control domain manages the application in the guest domain, and
- Single-node VCS installed in the guest domain monitors the application in guest domain.

To perform a migration of the LDom when VCS is installed in the control domain that manages the applications in the guest domains

- ◆ Use the `ldm` command for migration.

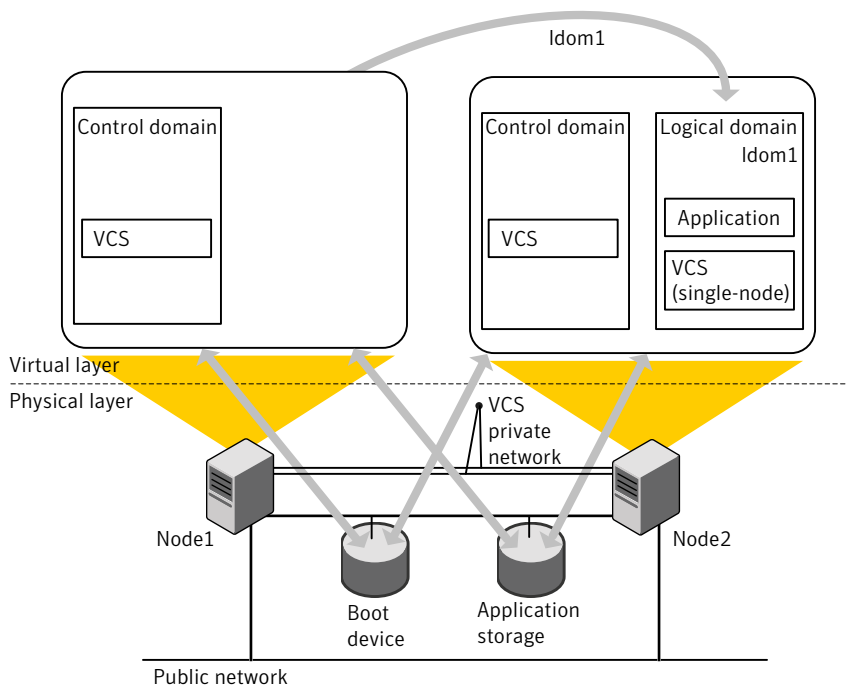
```
ldm migrate [-f] [-n] [-p password_file] source_ldom \  
[user@target_host[:target_ldom]
```

For example:

```
Sys1# ldm migrate ldom1 Sys2
```

Figure 6-10 illustrates a logical domain migration when VCS is clustered between control domains and single-node VCS in the guest domain monitors applications.

Figure 6-11 Logical domain migration when VCS is clustered between control domains and single-node VCS in the guest domain monitors applications



Migrating Oracle VM guest when VCS cluster is installed in the guest domains to manage applications for Oracle VM Server for SPARC version 2.1 and above

Perform one of the following procedures when you want a migration of guest domain when VCS cluster is configured between guest domains.

To perform a migration of the LDom when VCS is installed in the guest domain that manages the applications in the guest domains for Oracle VM Server for SPARC version 2.1 and above

- ◆ Use the `ldm` command for migration.

```
ldm migrate [-f] [-n] [-p password_file] source_ldom \  
[user@target_host[:target_ldom]
```

For example:

```
Sys1# ldm migrate ldom1 Sys2
```

Migrating Oracle VM guest when VCS cluster is installed in the guest domains to manage applications for Oracle VM Server for SPARC version 2.0

The domain migration is a warm migration.

Note: You do not have to start and stop LLT and GAB. In a warm migration, LLT and GAB restart themselves gracefully.

To perform a domain migration for an LDom when VCS is installed in guest domains

- 1 Stop VCS engine. Use the `hastop -local -force` command on the system that has the logical domain that you plan to migrate. Perform this step so that GAB does not have to kill the VCS engine process when the migration is complete. GAB wants all clients to reconfigure and restart when the configuration is not in sync with other members in the cluster.
- 2 If CVM is configured inside the logical domain, perform this step. Set the value of the LLT `peerinact` parameter to 0 on all nodes in the cluster. You set the value to zero so that while the logical domain is in migration, the system is not thrown out of the cluster by the other members in the cluster. If the CVM stack is unconfigured, the applications can stop.

See the *Veritas Cluster Server Administrator's Guide* for LLT tunable parameter configuration instructions.

- 3 If fencing is configured in single instance mode inside the logical domain, perform this step. Unconfigure and unload the vxfen module in the logical domain. Perform this step so that GAB does not panic the node when the logical domain migration is complete.
- 4 Migrate the logical domain from the control domain using the `ldm` interface. Wait for migration to complete.

```
ldm migrate [-f] [-n] [-p password_file] source_ldom \  
[user@target_host[:target_ldom]
```

For example:

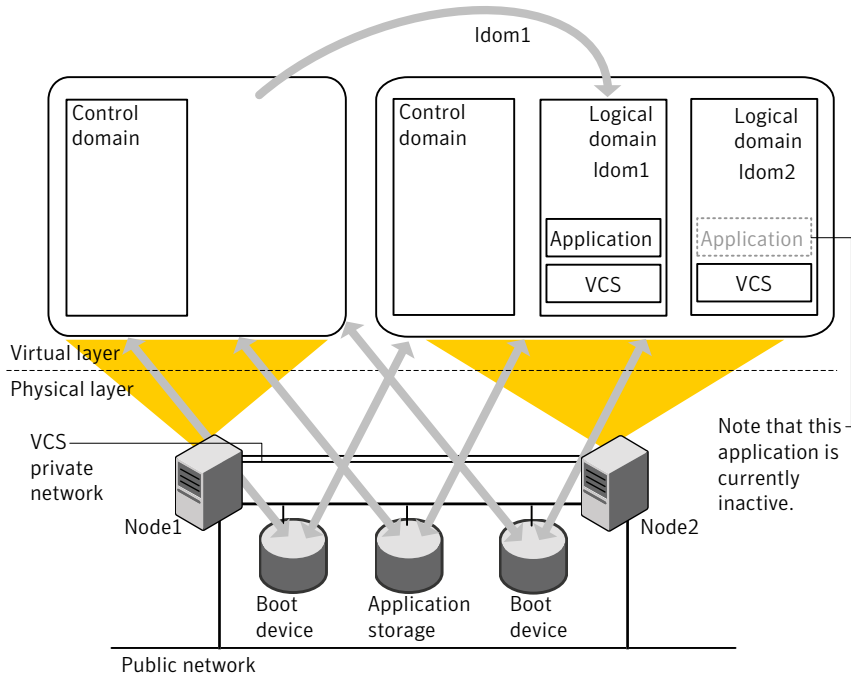
```
Sys1# ldm migrate ldom1 Sys2
```

- 5 Perform this step if you performed step 3. Load and configure vxfen module in the logical domain. See the Veritas Cluster Server Administrator's Guide for information about I/O fencing and its administration.
- 6 Perform this step if you performed step 2. Reset the value of the LLT `peerinact` parameter to its original value on all nodes in the cluster.

See the *Veritas Cluster Server Administrator's Guide* for LLT tunable parameter configuration instructions.
- 7 Use the `hastart` command to start VCS engine inside the logical domain.

[Figure 6-12](#) illustrates a logical domain migration when VCS is clustered between control domains and single-node VCS in the guest domain monitors applications.

Figure 6-12 The logical domain migration when VCS is clustered between guest domains



Configuring VCS to manage a Logical Domain using services from multiple I/O domains

VCS provides high availability to Logical Domains using I/O services from multiple I/O domains. When I/O services from the control domain and alternate I/O domain fail, VCS fails over the LDom from one system to another system. LDom continues to be functional on the same system and does not need a fail over if one of the I/O domain continues to provide service.

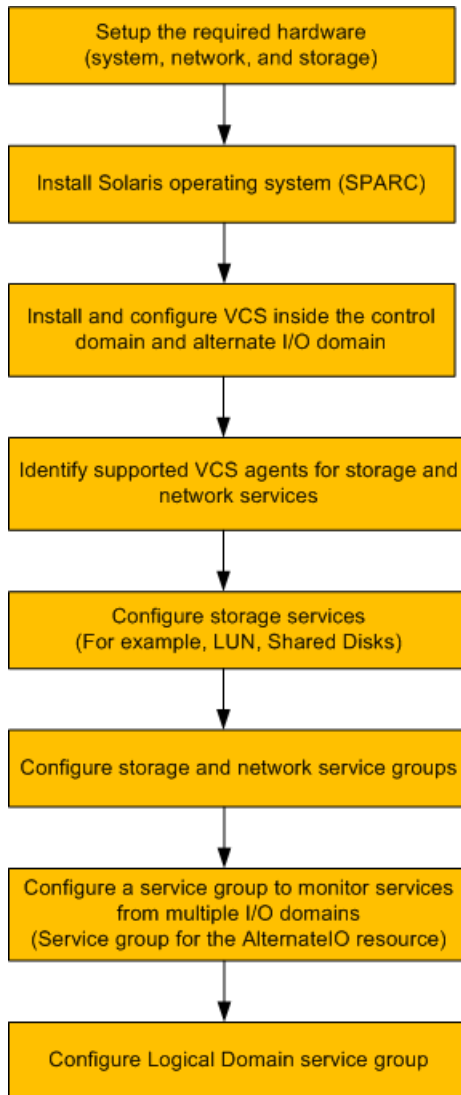
VCS uses service groups and resources to manage the storage and network services that are provided to a Logical Domain. These service groups are monitored by the AlternateIO resource. The AlternateIO service group provides the information about the state of the storage and network services to the LDom agent. VCS fails over the Logical Domain when services from both the I/O domains fail.

Perform the following tasks to configure VCS to manage a Logical Domain:

- Identify supported storage and network services
 See [Identify supported storage and network services](#)

- Determine the number of nodes to form VCS cluster
See [Determine the number of nodes to form VCS cluster](#)
- Install and configure VCS inside the control and alternate I/O domains
See [Install and configure VCS inside the control domain and alternate I/O domain](#)
- Configure storage services
See [Configuring storage services](#)
- Configure storage service groups
See [Configure storage service groups](#)
- Configure network service groups
See [Configure network service groups](#)
- Configure service group to monitor services from multiple I/O domains
See [Configure a service group to monitor services from multiple I/O domains](#)
- Configure AlternatelIO resource
See [Configure the AlternatelIO resource](#)
- Configure Logical Domain service group
See [Configure the service group for a Logical Domain](#)

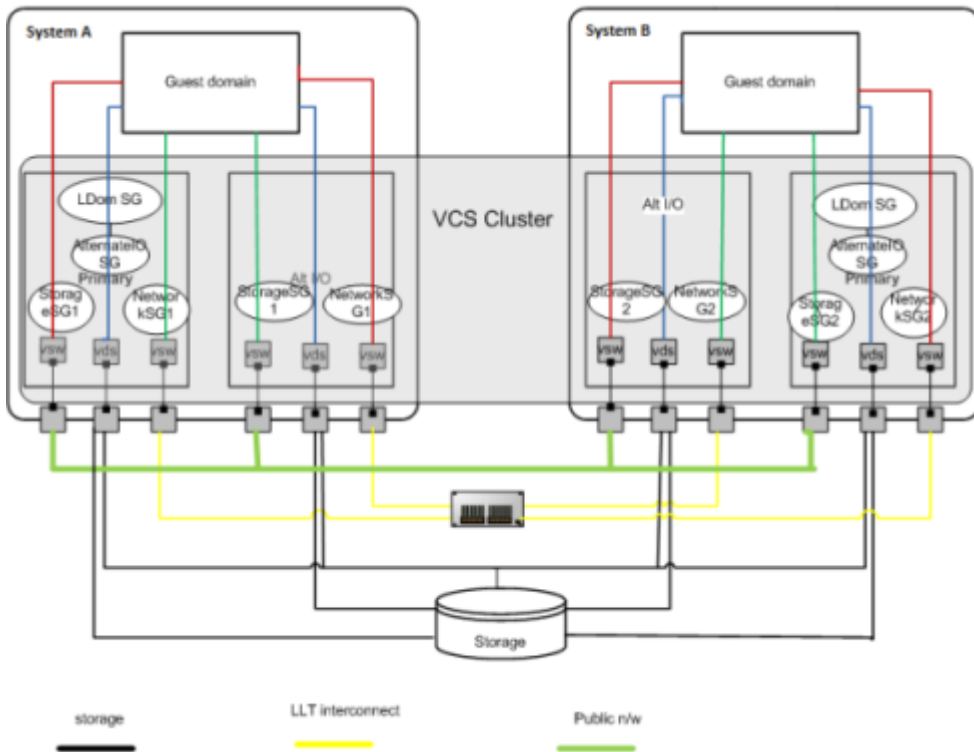
Figure 6-13 Workflow to configure VCS on a physical system to manage a Logical Domain



A typical setup for a Logical Domain with multiple I/O services

Guest logical domain using I/O services from multiple I/O domains.

Figure 6-14 shows VCS configuration to monitor a logical domain using I/O services from multiple I/O domains



System A, System B - T5440 servers

LDom SG - Logical Domain service group

AlternateIO SG - AlternateIO service group

Storage SG - Storage service group

Network SG - Network service group

Identify supported storage and network services

The following back-end storage and network services can be exported to Logical Domains.

I/O services	Back-end device	VCS agents to be used
Storage	LUN, shared disk	Disk
	Flat file	Mount
	zpool	Zpool
	Veritas CVM volume	CVMVoIDG
Network	NIC	NIC

Determine the number of nodes to form VCS cluster

The total number of nodes that form a VCS cluster depends on the number of physical systems times the the control domain and alternate I/O domain on each physical system.

For example, if you have two physical systems, each having a control domain and an alternate I/O domain, you need to configure VCS as a four node cluster.

Install and configure VCS inside the control domain and alternate I/O domain

Install and configure VCS inside the control domain and alternate I/O domain

For more details, refer to the *Veritas™ Cluster Server Installation Guide*.

Configuring storage services

Depending upon the storage service, the configuration procedure differs.

- LUN, Shared Disk, or CVM volume
See [“About virtual disk multipathing for storage services”](#) on page 142.
See [“Configuring virtual disk multipathing for LUN, Shared Disk, or CVM Volume”](#) on page 142.
- ZFS volume
See [“Configuring storage services when back-end device is a ZFS volume”](#) on page 144.
- Flat file
For more information, refer to the LDom Administrator’s guide.
- Zpool
For more information, refer to the LDom Administrator’s guide.

About virtual disk multipathing for storage services

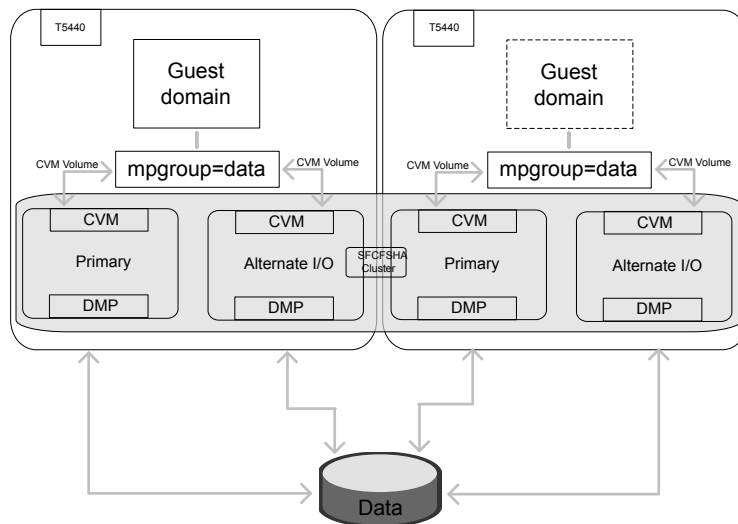
Virtual disk multipathing (`mpgroup`) enables you to configure virtual disk on a guest logical domain to access its back-end storage path through more than one I/O domain. This feature ensures that the virtual disk accessed by the guest logical domain remains accessible as long as services from one of the I/O domain are available.

For example, if you set up a virtual disk multipathing (`mpgroup`) to access a physical disk from a shared storage (SAN) which is connected to more than one I/O domains, when the guest domain accesses that virtual disk, the virtual disk driver goes through one of the I/O domains to access the back-end storage. If the virtual disk driver cannot connect to an I/O domain, then the virtual disk attempts to reach the back-end storage through a different I/O domain.

You must configure virtual disk multipathing (`mpgroup`) only if the storage back-end device exported to the guest domain is a LUN, Shared Disk, or CVM volume.

See [“Configuring virtual disk multipathing for LUN, Shared Disk, or CVM Volume”](#) on page 142.

Figure 6-15 shows a sample diagram for `mpgroup` with CVM Volume



Configuring virtual disk multipathing for LUN, Shared Disk, or CVM Volume

To enable virtual disk multipathing (`mpgroup`), you must export a virtual disk back-end path from each I/O domain and add the virtual disks to a multipathing group (also

known as `mpgroup`). The `mpgroup` is identified by a name and is configured when you export the virtual disk back-end path.

To configure virtual disk multipathing

- 1 Add the physical disk back-end path of the disk service to the primary domain.

```
# ldm add-vdsdev mpgroup=data backend_path1 volume@primary-vds0
```

where `backend_path1` is the path to the virtual disk back-end from the primary domain.

- 2 Add the physical disk back-end path of the disk service to the alternate I/O domain for disk added in step 1.

```
# ldm add-vdsdev mpgroup=data backend_path2 volume@alternate-vds0
```

where `backend_path2` is the path to the virtual disk back-end from the alternate I/O domain.

- 3 Export the virtual disk to the guest domain.

```
# ldm add-vdisk disk_name volume@primary-vds0 ldom_name
```

where `disk_name` is the name of the virtual storage device.

where `ldom_name` is the name of the Logical Domain.

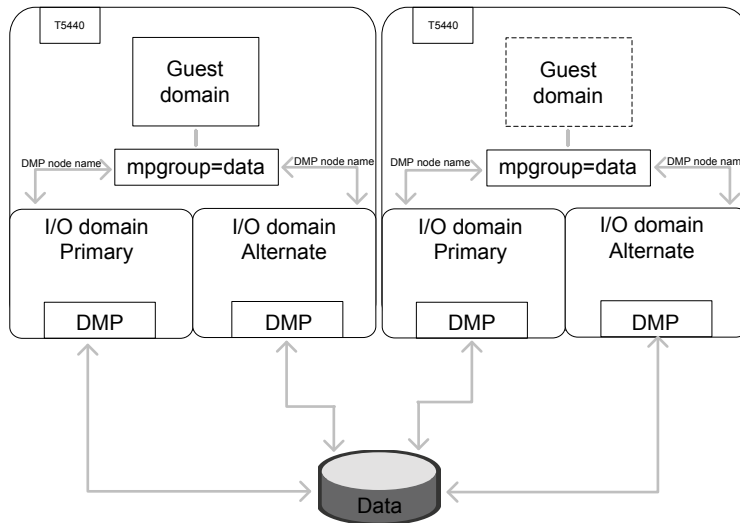
Note: Do not set the value of the **Options** attribute to `exclusive`; **excl**. If set to `exclusive`, Logical Domain cannot use the multipathing functionality.

For more details on configuring virtual disk multipathing, refer to the *Oracle VM server for SPARC Administration Guide*.

Virtual disk multi-pathing (mpgroup) configurations with DMP

If the disk exported to guest domain has more than one I/O path from each I/O domain, install Veritas Dynamic Multi-pathing (DMP) in each I/O domain. Use the DMP node name to configure the virtual disk multi-pathing (`mpgroup`) to the disk exported to the guest domain.

Figure 6-16 shows how mpgroup and DMP works together



Configuring storage services when back-end device is a ZFS volume

If you export ZFS volume as a back-end storage to Logical Domain, you need to

- 1 Export ZFS volume created in the control domain.
You do not need to configure mpgroup.
- 2 Export ZFS volume created in the alternate I/O domain.
You do not need to configure mpgroup.

Note: Ensure the ZFS volume size created in both domains matches.

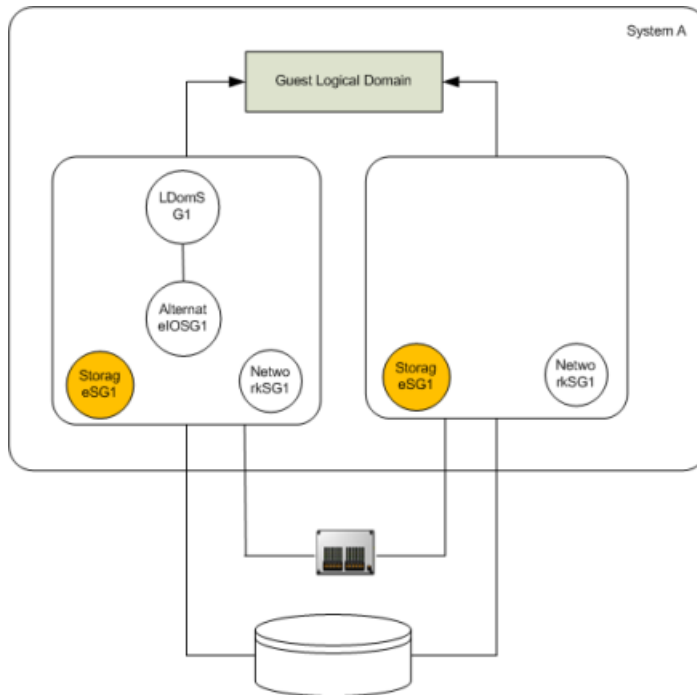
- 3 Create ZFS root pool mirror inside the Logical Domain from the volumes exported from control domain and alternate I/O domain.

Configure storage service groups

VCS agents manage storage services that are made available to a guest Logical Domain. Depending on the back-end storage device, use the appropriate VCS agent. For more information on supported VCS agents, see [Identify supported storage and network services](#).

Note: You must configure a storage service group on each physical system in the cluster.

Figure 6-17 shows storage service groups in the control domain and alternate I/O domain on a physical system



Configuration parameter

Localize resource attribute value

Description

You may need to localize VCS resource attributes depending on the type of back-end storage device.

For example, for Disk resource, if the back-end storage paths from the control and alternate I/O domains are different, you need to localize the partition attribute.

```
Disk disk1
(
  Partition @primary = "/dev/rdisk/c3t50060E8000C46C50d2s2"
  Partition @alternate = "/dev/rdisk/c1t50060E8000C46C50d2s2"
)
```

Configuration parameter	Description
Service group type	Service groups that manage storage services in the control domain and alternate I/O domain must be configured as a parallel service group.
Configure the SystemList attribute	Modify the SystemList attribute of the service group to add hostnames of the control domain and alternate I/O domain configured on the physical system.
Configure Phantom resource	<p>If all the resources are of the type Disk, configure a Phantom resource.</p> <p>The Disk resource is of the type OnOnly and does not contribute to determine the state of the service group. The Phantom resource enables VCS to determine the state of parallel service groups that do not include OnOff resources.</p> <p>For more information on the Phantom agent, refer to the Veritas™ Cluster Server Bundled Agents Reference Guide.</p>

An example of storage service group configuration from main.cf configuration (for a setup that has two physical systems)

Control domain host names – primary1, primary2

Alternate domain host names – alternate1, alternate2

```
group primary1-strsg (  
    SystemList = { primary1 = 0, alternate1 = 1 }  
    AutoStartList = { primary1, alternate1 }  
    Parallel = 1  
)  
  
Disk disk1  
(  
    Partition @primary1 = "/dev/rdisk/c3t50060E8000C46C50d2s2"  
    Partition @alternate1 = "/dev/rdisk/c1t50060E8000C46C50d2s2"  
)  
  
Phantom ph1 (  
)  
)  
  
group primary2-strsg (  
    SystemList = { primary2 = 0, alternate2 = 1 }  
    AutoStartList = { primary2, alternate2 }  
    Parallel = 1  
)
```

```
Disk disk2
(
  Partition @primary2 = "/dev/rdisk/c3t50060E8000C46C50d2s2"
  Partition @alternate2 = "/dev/rdisk/clt50060E8000C46C50d2s2"
)

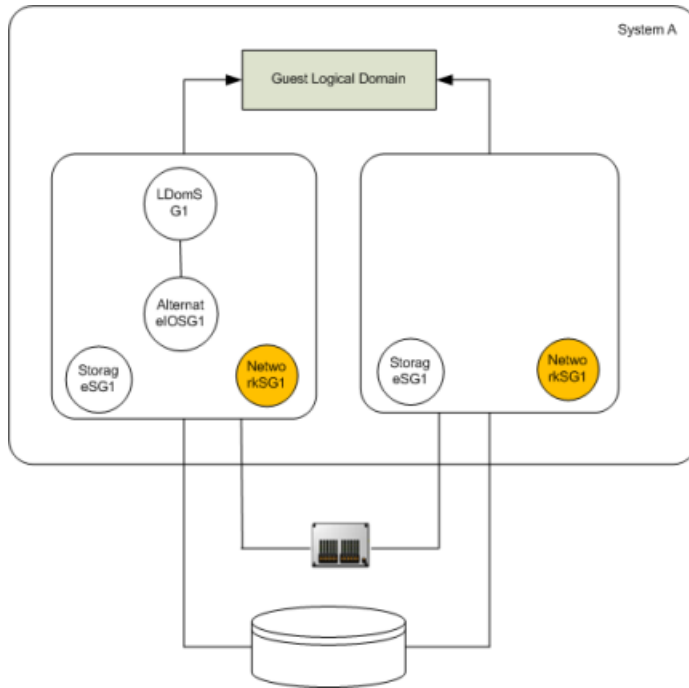
Phantom ph2 (
)
```

Configure network service groups

VCS agents manage network resources that are made available to a guest Logical Domain. Depending on the back-end storage device, use appropriate VCS agent. For more information, see [Identify supported storage and network services](#).

Note: You must configure a network service group on each physical system in the cluster.

Figure 6-18 shows network service groups in the control domain and alternate I/O domain



Perform the configuration steps for the network service group on each physical system.

Configuration parameter	Description
Localize network resource attribute	<p>You may need to localize VCS resources depending on the back-end network device.</p> <p>For example, for disk agent, if the network device exported from control and alternate I/O domain are different, you need to localize the Device attribute.</p> <pre>NIC primary1-network (Device @primary = nxge3 Device @alternate = nxge4)</pre>

Configuration parameter	Description
Service group type	Service groups that manage network services in the control domain and alternate I/O domain must be configured as a parallel service group.
Configure the SystemList attribute	Modify the SystemList attribute in the service group to add host names of the control domain and alternate I/O domain configured on the physical system.
Configure Phantom resource	<p>If all the resources are of the type NIC, configure a Phantom resource.</p> <p>The NIC resource is of the type OnOnly and does not contribute to determine the state of the service group. The Phantom resource enables VCS to determine the state of parallel service groups that do not include OnOff resources.</p> <p>For more information on the Phantom agent, refer to the Veritas™ Cluster Server Bundled Agents Reference Guide.</p>

An example of network service group configuration from main.cf (for a setup that has two physical systems)

Control domain host names – primary1, primary2

Alternate domain host names – alternate1, alternate2

```
group primary1-nwsg (
    SystemList = { primary1 = 0, alternate1 = 1 }
    AutoStartList = { primary1, alternate1 }
    Parallel = 1
)

NIC nicres1 (
    Device @primary1 = nxge3
    Device @alternate1 = nxge1
)

Phantom ph3 (
)

group primary2-nwsg (
    SystemList = { primary2= 0, alternate2 = 1 }
    AutoStartList = { primary2, alternate2 }
    Parallel = 1
)

NIC nicres2(
```

```
Device @primary2= nxge3
Device @alternate2 = nxge1
)

Phantom ph4 (
)
```

Configure a service group to monitor services from multiple I/O domains

Configure a service group for the AlternatelO resource to monitor storage and network services that are exported from back-end devices to a Logical Domain.

Configuration notes for the service group:

- Configure the service group as a parallel or a failover service group. See, Type of service group configuration for the AlternatelO resource.
- If multiple storage services are exported to a Logical Domain, you can configure separate service groups to monitor each of the storage services. For example, you can configure separate service groups to monitor LUN and ZFS volume storage services.
- The SystemList attribute of the service group must contain only host names of the control domains present on each physical system in the cluster.
- Localize the StorageSG attribute of the service group with the list of the storage service groups configured for each node.
- Enable preonline trigger for a fail over type service group

```
# hagr -modify aiosg TriggerPath bin/AlternatelO
```

where aiosg is the name of the service group

```
# hagr -modify aiosg TriggersEnabled PREONLINE
```

Type of service group configuration for the AlternatelO resource

Service group type	Condition
Parallel	If storage services are simultaneously accessible to all nodes in the cluster, the service group of the AlternatelO resource must be configured as a parallel service group. For example, shared LUNs, shared Disks, CVM Volume.

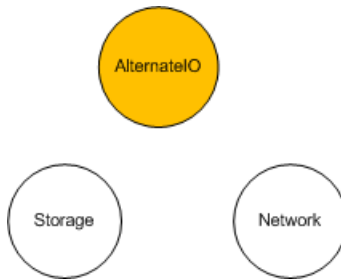
Fail over If storage services must be accessible only on one physical system (control domain and alternate I/O domain) in the cluster, configure the service group of the AlternateIO resource as a failover service group.

For example, zpool.

Configure the AlternateIO resource

The AlternateIO resource monitors storage and network services that are exported to a guest Logical Domain. The AlternateIO resource is not dependent on storage or network resources. However, its state depends upon the state of storage or network service groups.

Figure 6-19 shows that the AlternateIO resource does not have any dependency on storage or network resources.



Configuration parameter

Description

StorageSG attribute

This attribute is a key value pair. A storage service group is the key and the value of the key can either be 0 or 1.

Set the value of the key to 1 to bring the service group online when the Alternateio resource comes online and to take the service group offline when the Alternateio resource goes offline.

Localize the StorageSG attribute of the AlternateIO resource with the list of storage service groups that are configured on each node.

```
AlternateIO altiores1
(
  StorageSG @primary1 = { primary1-strsg1 = 1 }
  StorageSG @primary2 = { primary2-strsg1 = 1 }
)
```

Configuration parameter	Description
NetworkSG attribute	<p>This attribute is a key value pair. A network service group is the key and the value of the key can either be 0 or 1.</p> <p>Set the value of the key to 1 to bring the service group online when the AlternateIO resource comes online and to take the service group offline when the AlternateIO resource goes offline.</p> <p>Localize the NetworkSG attribute of the AlternateIO resource with the list of network service groups that are configured on each node.</p> <pre>AlternateIO altiores1 (NetworkSG @primary1 = { primary1-nwsg = 0 } NetworkSG @primary2 = { primary2-nwsg = 0 })</pre>
Preonline trigger	<p>For any of the service groups configured in the StorageSG or NetworkSG attributes, if you set the value to 1, configure the preonline trigger at the service group level.</p> <p>Configuring the preonline trigger ensures that service groups listed in the StorageSG attributes are offline on all systems except onto the system where failover or a manual switch over is initiated.</p> <p>For information on enabling the preonline trigger, see Configure service group to monitor services from multiple I/O domains.</p>

Sample service group configuration for the AlternateIO resource

Assumption – Storage and network service groups are of the type parallel.

```
group aiosg (
  SystemList = { primary1 = 0, primary2 = 1 }
  AutoStartList = { primary1, primary2 }
  Parallel = 1
)

AlternateIO aiores1 (
  StorageSG @primary1 = { primary1-strsg = 0 }
  StorageSG @primary2 = { primary2-strsg = 0 }
  NetworkSG @primary1 = { primary1-nwsg = 0 }
```



```
NetworkSG @primary2 = { primary2-nwsg = 0 }  
)
```

Configure the service group for a Logical Domain

VCS uses the LDom agent to manage a guest logical domain. The Logical Domain resource has an online local hard dependency on the AlternatelIO resource.

Figure 6-20 shows the dependency of LDom service group on the AlternatelIO service group.



Configuration notes:

- Configure the service group as a fail over type service group.
- The SystemList attribute in the LDom service group must contain only host names of the control domains from each physical system in the cluster.
- The LDom service group must have online local hard dependency with the AlternatelIO service group.

Sample configuration for LDom service group

The LDom service group must have online local hard dependency with the AlternatelIO service group.

```
group ldmsg (  
    SystemList = { primary1 = 0, primary2 = 1 }  
    AutoStartList = { primary1, primary2 }  
)  
  
LDom ldmres (  
    LDomName = ldg1  
)
```

Failover scenarios

Scenario	Control domain	Alternate I/O domain	VCS behavior
State of each storage service group	Online	Online	No fail over
	Offline/FAULT	Online	No fail over
	Online	Offline/FAULT	No fail over
	Offline/FAULT	Offline/FAULT	Fail over
State of each network service group	Online	Online	No fail over
	Offline/FAULT	Online	No fail over
	Online	Offline/FAULT	No fail over
	Offline/FAULT	Offline/FAULT	Fail over
Domain state	Up	Up	No fail over
	Up	down	No fail over
	Down	Up	Fail over
	Down	down	Fail over

Recommendations while configuring VCS and Oracle VM Server for SPARC with multiple I/O domains

- Online and offline operations for service groups in the StorageSG attribute
 To manually bring online or take offline service groups that are configured in the StorageSG attribute do not use the AlternatelIO resource or its service group. Instead, use service groups configured in the StorageSG attribute.
- Freeze the service group for the AlternatelIO resource
 Freeze the AlternatelIO service group before you bring online or take offline service groups configured in the StorageSG attribute of the AlternatelIO resource. If you do not freeze the service group, the behavior of the Logical Domain is unknown as it is dependent on the AlternatelIO service group.
- Configuring preonline trigger for storage service groups
 You must configure preonline trigger in the following scenario:
 When the service groups configured in the StorageSG attribute of the AlternatelIO resource are of fail over type, and if you accidentally bring storage service groups online on another physical system in the cluster.

It is possible to bring the storage service groups online on another physical system because resources configured to monitor back-end storage services are present in different service groups on each physical system. Thus, VCS cannot prevent resources coming online on multiple systems. This may cause data corruption.

To configure preonline trigger for each service group listed in the StorageSG attribute, run the following commands:

- # hagrps -modify *stg-sg* TriggerPath bin/AlternateIO/StorageSG
where *stg-sg* is the name of the storage service group
- # hagrps -modify *stg-sg* TriggersEnabled PREONLINE

Note: Perform this procedure for storage service groups on each node.

- Set connection time out period for virtual disks
When a disk device is not available, I/O services from the guest domain to the virtual disks are blocked.
Symantec recommends to set a connection time out period for each virtual disk so that applications times out after the set period instead of waiting indefinitely.
ldm add-vdisk timeout=seconds disk_name volume_name@service_name
ldom

Sample VCS configuration for AlternateIO resource configured as a fail over type

```
include "types.cf"
cluster altio-cluster (
  UserNames = { admin = XXXXXXXXXXXX }
  Administrators = { admin }
  HacliUserLevel = COMMANDROOT
)

system primary1 (
)

system alternatel (
)

system primary2 (
)
```

```
system alternate2 (
)

group aiosg (
  SystemList = { primary1 = 0, primary2 = 1 }
  AutoStartList = { primary1 }
  TriggerPath = "bin/AlternateIO"
  TriggersEnabled @primary1 = { PREONLINE }
  TriggersEnabled @primary2 = { PREONLINE }
)

AlternateIO altiores (
  StorageSG @primary1 = { primary1-strsg = 1 }
  StorageSG @primary2 = { primary2-strsg = 1 }
  NetworkSG @primary1 = { primary1-nwsg = 0 }
  NetworkSG @primary2 = { primary2-nwsg = 0 }
)

// resource dependency tree
//
// group aiosg
// {
// AlternateIO altiores
// }

group ldomsg (
  SystemList = { primary1 = 0, primary2 = 1 }
  AutoStartList = { primary1 }
)

LDom ldmguest (
  LDomName = ldg1
)

requires group aiosg online local hard

// resource dependency tree
//
// group ldomsg
```

```
// {  
// LDom ldg1  
// }  
  
group primary1-strsg (  
    SystemList = { primary1 = 0, alternatel = 1 }  
    AutoStart = 0  
    Parallel = 1  
    TriggerPath = "bin/AlternateIO/StorageSG"  
    TriggersEnabled @primary1 = { PREONLINE }  
    TriggersEnabled @alternatel = { PREONLINE }  
    AutoStartList = { primary1, alternatel }  
)  
  
    Zpool zpres1 (  
        PoolName @primary1= zfsprim  
        PoolName @alternatel = zfsmirr  
        ForceOpt = 0  
    )  
  
    // resource dependency tree  
    //  
    //     group primary1-strsg  
    //     {  
    //     Zpool zpres1  
    //     }  
  
group primary1-nwsg (  
    SystemList = { primary1 = 0, alternatel = 1 }  
    Parallel = 1  
)  
  
Phantom ph1 (  
)  
  
NIC nicres1 (  
    Device @primary1 = nxge3  
    Device @alternatel = nxge4
```

```
)

// resource dependency tree
//
// group primary1-nwsg
// {
// Phantom ph1
// Proxy nicres1
// }

group primary2-strsg (
  SystemList = { primary2 = 0, alternate2 = 1 }
  Parallel = 1
  TriggerPath = "bin/AlternateIO/StorageSG"
  TriggersEnabled @primary2 = { PREONLINE }
  TriggersEnabled @alternate2 = { PREONLINE }
)

  Zpool zpres2 (
    PoolName @ primary2 = zfsprim
    PoolName @ alternate2 = zfsmirr
    ForceOpt = 0
  )

// resource dependency tree
//
//      group primary2-strsg
//      {
//      Zpool zpres2
//      }

group primary2-nwsg (
  SystemList = { primary2 = 0, alternate2 = 1 }
  Parallel = 1
)

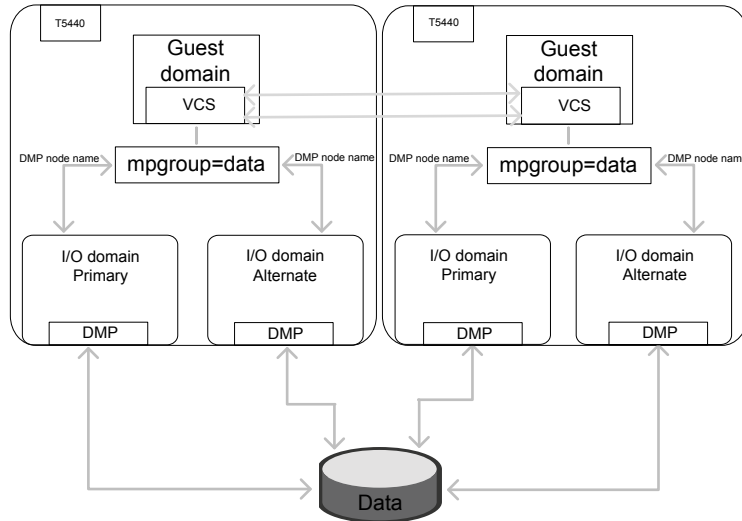
Phantom ph2 (
```

```
)  
  
NIC nicres2 (  
    Device @primary2 = nxge3  
    Device @alternate2 = nxge4  
)  
  
// resource dependency tree  
//  
// group primary2-nwsg  
// {  
//   Phantom ph2  
//   Proxy nicres2  
// }
```

Configuring VCS on logical domains to manage applications using services from multiple I/O domains

This section describes how to configure Veritas Cluster Server (VCS) on logical domains to manage applications using services from multiple I/O domains.

Figure 6-21 Typical two node VCS cluster setup is configured on logical domains where each guest domain receives I/O services from multiple I/O domains



The typical two node VCS cluster at guest domain fails over the applications between guest domains to keep the applications highly available.

Configuring VCS on guest domains that uses services from multiple I/O domains

- 1 Set up the required hardware, system, storage, and network.
- 2 Install the operating system and the Oracle VM Server for SPARC on the physical system.
 Refer to the *Oracle* documentation for details.
- 3 Set up the alternate I/O domain.
 Refer to the *Oracle VM Server for SPARC Administration Guide* for details.
- 4 Set up the guest domain and provide the I/O services from all the I/O domains to the guest domain.
 Refer to the *Oracle VM Server for SPARC Administration Guide* for details.

During the setup of the guest domain, you can choose one of the following:

- Local disk or SAN disk as boot disk to guest domain
- SAN disk as data disk for application

Use SAN disk for boot disk and data disk if you plan to migrate the guest domain.

- If the storage device contains multiple paths from each I/O domain, you have the following options:
 - Install Veritas Dynamic Multipathing (DMP) on the I/O domains
Export DMP node name from each I/O domain to guest domain.
Configure virtual disk multipathing (`mpgroup`) for the DMP node name between the I/O domains.
I/O Fencing cannot be configured with VCS on the guest domain when `mpgroup` is set for storage devices at I/O domain level.
 - Install Veritas Dynamic Multipathing (DMP) on the guest domain
Export all the paths to storage device from each I/O domain and do not configure the virtual disk multipathing (`mpgroup`). DMP at the guest domain takes care of path management between I/O domains.
 - Install Veritas Dynamic Multipathing (DMP) on I/O domains and the guest domain
Export DMP node name from each I/O domain to the guest domain.
DMP at the guest domain takes care of path management between I/O domains.
I/O Fencing cannot be configured with VCS on the guest domain when DMP is configured on both I/O domains and guest domain.
- 5 Install VCS on the guest domain.
See the *Veritas Cluster Server Installation Guide* for details.
- 6 Configure the application service group.
See the *Veritas Cluster Server Administrator's Guide* and *Veritas Cluster Server Bundled Agents Reference Guide* for details.