



# Disaster Recovery using Veritas Storage Foundation Enterprise HA and IBM DS8000 with Metro Mirror

*Solution Installation and Configuration*

• • • • • • • • •

*Document SM101-1*

May 20, 2008



## Table of contents

<b>Abstract</b> .....	<b>4</b>
<b>Introduction</b> .....	<b>4</b>
About High Availability .....	4
About Disaster Recovery .....	4
About IBM .....	4
About Symantec.....	5
About Veritas Storage Foundation / HA.....	5
About IBM System Storage DS8000 Series .....	5
<b>Overview</b> .....	<b>7</b>
<b>Test System Configuration</b> .....	<b>7</b>
<b>Configuring IBM DS8000 and IBM Metro Mirror</b> .....	<b>11</b>
DSCLI Installation on the Application hosts.....	11
IBM DS8000 storage configuration.....	11
<b>Installing VERITAS Storage Foundation</b> .....	<b>15</b>
Symantec product licensing .....	15
Supported AIX operating systems .....	15
Database requirements .....	15
Disk space .....	15
Environment Variables.....	15
Virtual IP Address .....	16
Prerequisites for remote and cluster installation.....	16
Mounting a software disc .....	16
Installing using the VERITAS product installer .....	16
Upgrading VERITAS Storage Foundation HA to 5.0MP1.....	20
<b>Installing and Upgrading VCS agent for DB2</b> .....	<b>21</b>
<b>Installing and Upgrading VCS agent for Oracle</b> .....	<b>21</b>
<b>Installing VCS agent for Metro Mirror</b> .....	<b>21</b>
<b>Installing and Configuring DB2, Oracle</b> .....	<b>22</b>
<b>Configuring Applications for Disaster Recovery</b> .....	<b>22</b>
Quick Setup .....	22
Manual configuration.....	23
Converting DS8000 devices to Metro Mirror devices .....	23
Setting up the Global Cluster.....	23
Linking clusters .....	24
Configuring DB2 / Oracle failover Service groups .....	24
Adding the agents manually in a global cluster.....	24
Configuring and Adding VCS Metro Mirror resource .....	26
Before you configure the agent for Metro Mirror .....	26
About cluster heartbeats .....	26
Configuration concepts.....	26



Metro Mirror resource type definition .....	26
Attribute definitions for the Metro Mirror agent.....	26
Required attributes .....	26
Optional attributes .....	27
Internal attributes.....	27
Adding the agents manually in a global cluster.....	28
<b>Failover Scenarios .....</b>	<b>28</b>
Application host failover .....	28
Disaster Recovery in a Global Cluster Configuration .....	29
<b>Summary .....</b>	<b>30</b>
<b>Appendix A: VCS configuration file (main.cf ) containing DB2, Oracle Application Service group for HA/DR .....</b>	<b>31</b>
<b>Appendix B: Setting up the Database Applications.....</b>	<b>39</b>
Setting up the DB2 Database Application.....	39
Setting up the Oracle Database Application .....	42
<b>Appendix C: Veritas Software filesets listing .....</b>	<b>49</b>
<b>Trademarks and special notices.....</b>	<b>51</b>



## Abstract

---

*This document describes how Symantec and IBM have installed, configured and validated High Availability (HA) and Disaster Recovery (DR) configurations for DB2 and Oracle with IBM® System Storage™ DS8000™. These validations include local HA configurations using Veritas Storage Foundation™ and Veritas Cluster Server™. The configuration was extended to a DR configuration using IBM Metro Mirror for synchronous replication with the VCS agent for Metro Mirror and VCS Global Cluster Option for alternate site failover/failback capability.*

## Introduction

---

Infrastructure for mission critical applications must be able to meet the organization's Recovery Time Objective (RTO) and Recovery Point Objective (RPO) for resuming operation in the event of a site disaster. This solution addresses environments where the RPO is zero and RTO is in the range of minutes to a few hours. While backup is the foundation for any DR plan, a typical RTO for tape only based backup is well beyond these objectives. Also, replication of data alone is not enough as having the application data at a DR site is of limited use without also having the ability to start the correct sequence of database management systems, application servers, and business applications.

Symantec's DR solutions Metro Clustering and Global Clustering are extensions of local HA clustering using Veritas Storage Foundation and Veritas Cluster Server. This validated and documented solution is an example of "Global Clustering", a collection of two or more VCS clusters at separate locations linked together with VCS Global Cluster Option to enable wide area failover and disaster recovery. Each local cluster within the global cluster is connected to its own shared storage. Local clustering provides local failover for each site. IBM Metro Mirror replicates data between IBM System Storage DS8000s at each site to maintain synchronized copies of storage at the two sites. For a disaster that affects an entire site, the customer makes a decision on whether to move operations to the disaster recovery site. When that decision is made, the application is automatically migrated to a system at the DR site.

### About High Availability

The term high availability (HA) refers to a state where data and applications are highly available because software or hardware is in place to maintain the continued functioning in the event of computer failure. High availability can refer to any software or hardware that provides fault tolerance, but generally the term has become associated with clustering. Local clustering provides high availability through database and application failover. Veritas Storage Foundation Enterprise HA (SF/HA) includes Veritas Storage Foundation and Veritas Cluster Server and provides the capability for local clustering.

### About Disaster Recovery

Wide area disaster recovery (DR) provides the ultimate protection for data and applications in the event of a disaster. If a disaster affects a local or metropolitan area, data and critical services are failed over to a site hundreds or thousands of miles away. Veritas Storage Foundation Enterprise HA/DR provides the capability for implementing disaster recovery.

### About IBM

IBM is the world's largest information technology company, with over 80 years of leadership in helping businesses innovate by delivering a wide range of solutions and technologies that enable customers, large and small, to deliver more efficient and effective services. IBM's comprehensive server, storage, software and services portfolio is designed to help you create new business insight by integrating, analyzing and optimizing information on demand. From its foundations of virtualization, openness and innovation through collaboration, IBM can optimize management of information through technology innovations and infrastructure simplification to help achieve maximum business productivity. Visit us at <http://www.ibm.com>.

## About Symantec

Symantec is a global leader in infrastructure software, enabling businesses and consumers to have confidence in a connected world. The company helps customers protect their infrastructure, information and interactions by delivering software and services that address risks to security, availability, compliance and performance. Headquartered in Cupertino, California, Symantec has operations in more than 40 countries.

## About Veritas Storage Foundation / HA

Veritas Storage Foundation HA is a comprehensive solution that delivers data and application availability by bringing together two industry-leading products: Veritas Storage Foundation and Veritas Cluster Server.

Veritas Storage Foundation provides a complete solution for heterogeneous online storage management. Based on the industry-leading Veritas Volume Manager™ and Veritas File System™, it provides a standard set of integrated tools to centrally manage explosive data growth, maximize storage hardware investments, provide data protection and adapt to changing business requirements. Unlike point solutions, Storage Foundation enables IT organizations to manage their storage infrastructure with one tool. With advanced features such as centralized storage management, non-disruptive configuration and administration, dynamic storage tiering, dynamic multi-pathing, data migration and local and remote replication, Storage Foundation enables organizations to reduce operational costs and capital expenditures across the data center.

Veritas Cluster Server is the industry's leading clustering solution for reducing both planned and unplanned downtime. By monitoring the status of applications and automatically moving them to another server in the event of a fault, Veritas Cluster Server can dramatically increase the availability of an application or database. Veritas Cluster Server can detect faults in an application and all its dependent components, including the associated database, operating system, network, and storage resources. When a failure is detected, Cluster Server gracefully shuts down the application, restarts it on an available server, connects it to the appropriate storage device, and resumes normal operations. Veritas Cluster Server can temporarily move applications to a standby server when routine maintenance such as upgrades or patches requires that the primary server be taken offline.

## About IBM System Storage DS8000 Series

The IBM System Storage DS8000 series is designed to support the most demanding business environments. The DS8000 series provides exceptional performance and should be considered for applications that require fast access to data. What's more, for those concerned with maintaining business uptime, the DS8000 provides resiliency features that support continuous availability. With its high-capacity scalability, broad server support and virtualization features, the DS8000 is well suited for simplifying the storage environment by consolidating data from multiple storage systems on a single system



### Common features

- High performance Fibre Channel disk packages containing 16 disks ranging from 73 GB up to 300 GB disk capacities
- 500 GB Fibre Channel ATA disk drive packages for cost-effective second-tier nearline storage
- IBM System Storage FlashCopy®, Global Mirror, Metro Mirror, Metro/Global Mirror and Global Copy provide flexible replication services to address your backup and disaster planning needs



- IBM System Storage Productivity Center centralizes management with an easy-to-use graphical interface

#### **Hardware summary**

- 4-port 4 Gbps or 2 Gbps Fibre Channel/FICON®, 2-port IBM ESCON® host adapter interfaces
- Offers from 2 to 32 host adapters and up to 128 FC/FICON host ports
- FC-AL drive interface
- Minimum of 16 drives
- Dual SMP processor complexes
- Up to 256 GB memory with innovative caching algorithms
- From 1.1 TB up to 512 TB of physical capacity, upgradeable without system disruption

## Overview

This whitepaper illustrates the steps involved in installing and configuring Veritas Storage Foundation HA and IBM Metro Mirror for disaster recovery. Once the setup is complete, it is ready for conducting cluster failover and failback across two cluster sites using DS8000 Metro Mirror and VCS Global Cluster Option.

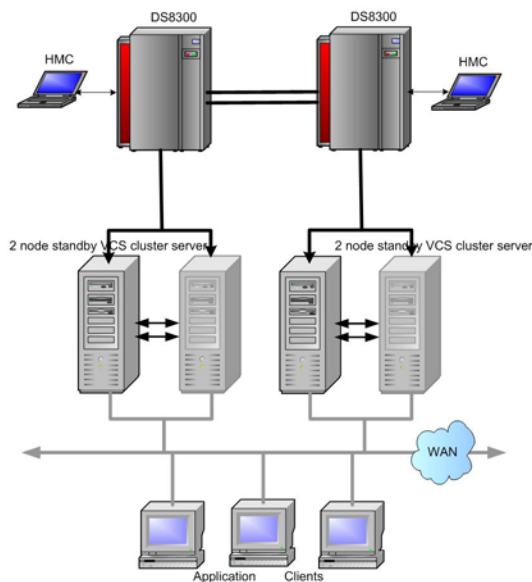
The host failover scenario simulates a fault in one of the cluster nodes and then in the other cluster node in the cluster at Site A causing the cluster to failover to Site B and bringing up the DB2 / Oracle application service group in one of the cluster nodes in Site B.

The disaster recovery robustness is tested by another scenario. A disaster is simulated by introducing a fault in the cluster at Site A and the simulation of a fault of the storage link causing the cluster failover to Site B. This brings up the DB2 / Oracle application service group in one of the cluster nodes in Site B. It also triggers the Metro Mirror failover and failback. In both the scenarios the VCS Metro Mirror agent manages the necessary DS8000 Metro Mirror procedures for failover and failback. The *Failover Scenarios* section provides procedural details.

## Test System Configuration

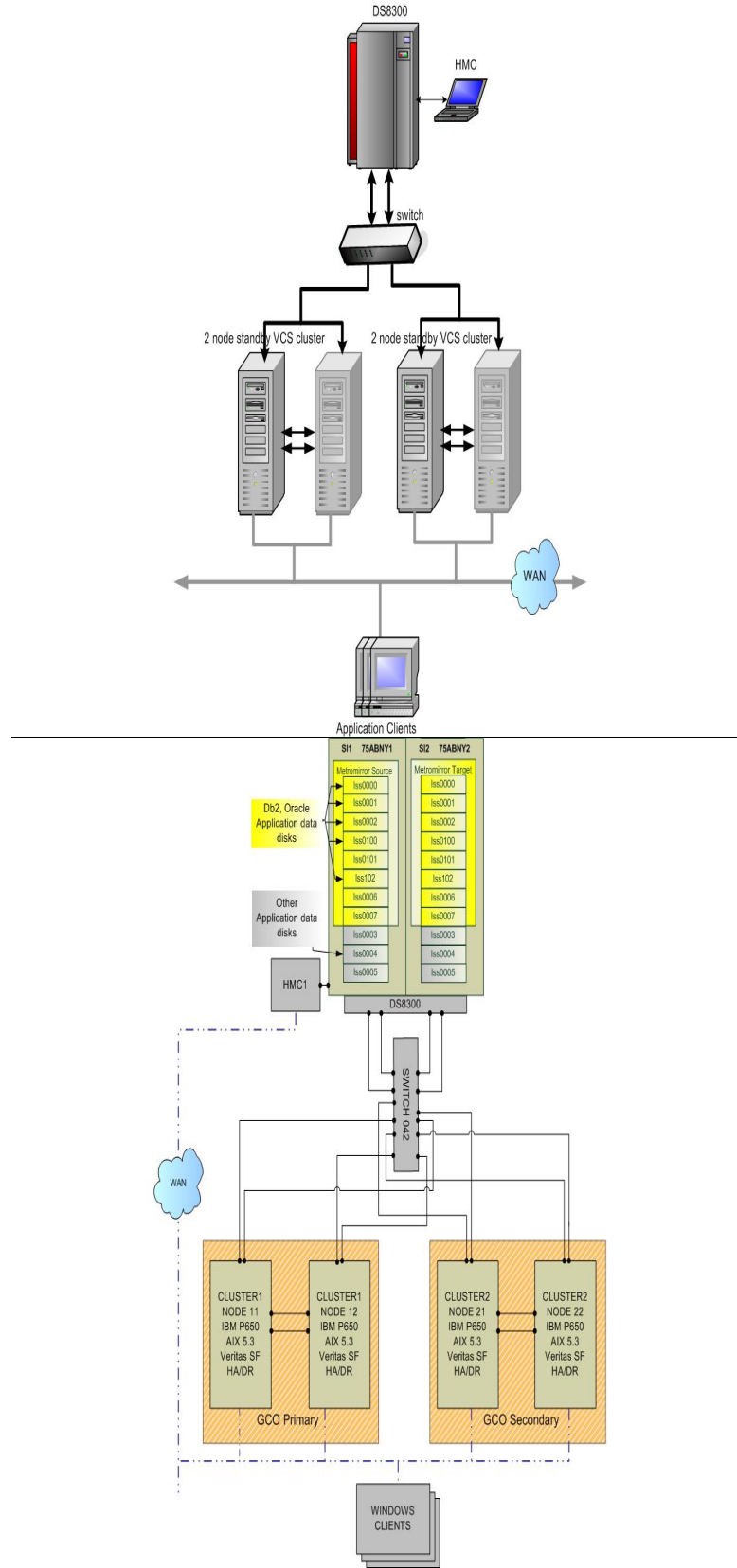
A typical cluster configuration is shown in Figure 1. The configuration includes

- A cluster at Site A consisting of two AIX hosts configured as a two node VCS cluster attached to the primary DS8000 array.
- A cluster at Site B consisting of two AIX hosts configured as a two node VCS cluster attached to the secondary DS8000 array.



**Figure 1. Typical VCS cluster and IBM Metro Mirror for Disaster Recovery**

The setup built for this whitepaper consists of four application hosts. The configuration described below and the hardware and software components are listed in Table 1 and Table 2. The hosts are AIX, IBM P-series servers. The four servers are configured to form two 2-node Storage Foundation cluster servers. Figure 2 shows the test system configuration.



**Figure 2. Test cluster configuration with IBM Metro Mirror**





The two clusters represent two sites, Site A and Site B. The cluster at Site A is the primary cluster and the cluster at Site B is the secondary cluster. Unlike the typical setup shown in Figure 1 above, in this test setup the hosts in both clusters are connected through a switch to a single DS8000 storage unit. The cluster in Site A connects to system image 1 (si1) of the storage unit and the Cluster in Site B connects to system image 2 (si2) of the storage unit. Table 4 shows DS8000 LUN layout.

In this setup both DB2 and Oracle applications are installed on the same cluster. Two separate VCS configuration files (main.cf) are created for each application failover. The failover scenarios are performed for one configuration at a time. Failover application service groups can be included into a single configuration.

From the storage management GUI console ensure that the DS8000 Metro Mirror licenses key is activated.

The application hosts have AIX 5.3 OS installed with the fix requirements for the DS8000 storage microcode level, and the following ODM filesets.

```
devices.fcp.disk.ibm.rte 1.0.0.9 COMMITTED IBM FCP Disk Device
ibm2105.rte 32.6.100.29 COMMITTED IBM 2105 Disk Device
```

Table 1 Hardware Configuration

Application Host Servers				
Cluster Sites	Site A		Site B	
VCS Cluster Names	ClustmmP		ClustmmS	
System / Cluster Node Names	Nd1	Nd2	Nd1	Nd2
System Model	IBM,7029-6C3	IBM,7029-6C3	IBM,7029-6C3	IBM,7029-6C3
Number Of Processors	2	2	2	2
Processor Clock Speed	1452 MHz	1452 MHz	1452 MHz	1452 MHz
CPU Type	64-bit	64-bit	64-bit	64-bit
Kernel Type	64-bit	64-bit	64-bit	64-bit
LPAR Info	1 NULL	1 NULL	1 NULL	1 NULL
Memory Size	4096 MB	4096 MB	4096 MB	4096 MB
Good Memory Size	4096 MB	4096 MB	4096 MB	4096 MB
Platform Firmware level	3F060109	3F060109	3F060109	3F060109
Firmware Version	IBM,RG060109_d79e15_r	IBM,RG060109_d79e15_r	IBM,RG060109_d79e15_r	IBM,RG060109_d79e15_r

Storage	
Array Model	IBM DS8000 (2107 9A2)
Firmware Version	6.1.3.20071026.1
Capacity	18.688TB(128x146GB)
HBA Ports	32



SAN	
SilkWorm 200E	SilkWorm 200E
Firmware version	v5.0.1b
Ports	16

Table 2 Software Configuration

Vendor	Software	Version
IBM	AIX 5.3	5300-06-04-0748
IBM	DB2	9.1
Oracle Corporation	Oracle	10gR2
IBM	IBM DSCLI	5.2.410.401
Symantec	VERITAS Storage Foundation Enterprise	5.0, 5.0MP1
Symantec	VERITAS High Availability 5.0MP1 Agent for DB2 by Symantec	5.0MP1
Symantec	VERITAS High Availability 5.0MP1 Agent for Oracle by Symantec	5.0MP1
Symantec	VERITAS Clustering Support for IBM Metro Mirror	5.0MP1

The following table lists product documentation set required for installation, configuration and troubleshooting the setup.

Table 3 Required Documents

Product	Use this Guide
IBM Metro Mirror	
VERITAS Storage Foundation Enterprise HA	VERITAS Storage Foundation Installation Guide (combined VM, FS, SFUA install)
VERITAS Cluster Server	VERITAS Cluster Server Installation Guide VERITAS Cluster Server User's Guide
VERITAS Volume Manager	VERITAS Volume Manager Administrator's Guide
VERITAS Cluster Server Agents	VERITAS Cluster Server Agent for Oracle Installation and Configuration Guide VERITAS Cluster Server Agent for DB2 Installation and Configuration Guide VERITAS Cluster Server Agent for Metro Mirror Installation and Configuration Guide



## Configuring IBM DS8000 and IBM Metro Mirror

### DSCLI Installation on the Application hosts

Install DSCLI client on each VCS cluster node. VCS Metro Mirror agent connects to DS8000 HMC through the DSCLI client.

1. Mount the CD

```
mount -V cdrfs -o ro /dev/cd0 /mnt
```

2. Execute install command from the mount point directory and follow instructions to complete installation.

```
/mnt/setupaix.bin -console
```

3. Login to the HMC

```
/opt/ibm/dscli/dscli
```

```
Enter the primary management console IP address: 10.140.XX.XX
```

```
Enter the secondary management console IP address:
```

```
Enter your username: abc
```

```
Enter your password: abc123
```

```
Date/Time: Mar 9, 2008 2:22:10 PM PST IBM DSCLI Version: 5.0.4.37
```

```
DS:
```

```
IBM.2107-75ABNY1
```

```
IBM.2107-75ABNY2
```

```
dscli>
```

4. At the dscli prompt issue the command to create a password file. VCS MM agent uses the password file to connect to the HMC.

```
dscli> managepwfile -action add -mc1 10.140.XX.XX -mc2 10.140.XX.XX -name admin  
-pw serv1cece -pwfile /opt/ibm/dscli/profile/ds_pwfile
```

5. You can update the profile file as required. The profile file is located at /opt/ibm/dscli/profile/dscli.profile

### IBM DS8000 storage configuration

This method utilizes IBM's DSCLI utility to configure storage on a DS8000 array. Hierarchical storage elements such as arrays, ranks, and extent pools are created as necessary to provide prerequisites for volume creation. To simplify the Metro Mirror configuration, two separate storage images are configured to provide identical configurations. Metro Mirror provides data replication between storage images and storage images can be physically separate DS8000 arrays, or logical partitions (LPARs) on a single DS8000 array.

Table 4 shows the LUNS configured for Metro Mirror used in this configuration. Table 5 shows the Switch port mappings.



Table 4 DS8000 Metro Mirror LUN Layout

	Storage Image 1 (si1, 75ABNY1) MM SOURCE	Storage Image 2 (si2, 75ABNY2) MM TARGET
Volumes	lss000 (0000)	lss000 (0000)
	lss001 (0001)	lss001 (0001)
	lss002 (0002)	lss002 (0002)
	lss003 (0003)	lss003 (0003)
	lss004 (0004)	lss004 (0004)
	lss100 (0100)	lss100 (0100)
	lss101 (0101)	lss101 (0101)
	lss102 (0102)	lss102 (0102)
	lss006 (0006)	lss006 (0006)
	lss007 (0007)	lss007 (0007)
FC Ports	I0003	I0203

Table 5 Switch Mapping

Switch Port Map	
Switch 001	
2	DS2
3	DS4
6	0-6 ISL
7	4-7 ISL
8	Clust1Nd11
9	Clust1Nd11
10	Clust2Nd21
11	Clust2Nd21
12	Clust1Nd12
13	Clust1Nd12
14	Clust2Nd22
15	Clust2Nd22
Switch 002	
0	0-6 ISL
1	DS1
2	DS3
4	4-7 ISL

Arrays are specific RAID 5 or RAID 10 instances. In this example RAID 5 arrays are created on each storage instance.

```
dscli> mkarray -dev IBM.2107-75ABNY1 -raidtype 5 -arsite S1
dscli> mkarray -dev IBM.2107-75ABNY2 -raidtype 5 -arsite S1
```

Ranks dictate the type of extent such as Fixed Block (FB).

```
dscli> mkrank -dev IBM.2107-75ABNY1 -array A0 -stgtype FB
dscli> mkrank -dev IBM.2107-75ABNY2 -array A0 -stgtype FB
```



Extent pools describe groups of extents.

```
dscli> mkextpool -dev IBM.2107-75ABNY1 -rankgrp 0 -stgtype FB pool00  
dscli> mkextpool -dev IBM.2107-75ABNY2 -rankgrp 0 -stgtype FB pool00
```

Extent pools are assigned to specific ranks.

```
dscli> chrnk -dev IBM.2107-75ABNY1 -extpool P0 R0  
dscli> chrnk -dev IBM.2107-75ABNY2 -extpool P0 R0
```

Fixed Block (FB) volumes are exported to the host or SAN as logical disks. Type, capacity, extent pool, and name are configurable as options. Two volumes are created in each storage instance in this example.

```
dscli> mkfbvol -dev IBM.2107-75ABNY1 -extpool P0 -type ds -cap 300 -name 0001  
dscli> mkfbvol -dev IBM.2107-75ABNY1 -extpool P0 -type ds -cap 300 -name 0002  
dscli> mkfbvol -dev IBM.2107-75ABNY1 -extpool P0 -type ds -cap 300 -name 0100  
dscli> mkfbvol -dev IBM.2107-75ABNY1 -extpool P0 -type ds -cap 300 -name 0102
```

```
dscli> mkfbvol -dev IBM.2107-75ABNY2 -extpool P0 -type ds -cap 300 -name 0001  
dscli> mkfbvol -dev IBM.2107-75ABNY2 -extpool P0 -type ds -cap 300 -name 0002  
dscli> mkfbvol -dev IBM.2107-75ABNY2 -extpool P0 -type ds -cap 300 -name 0100  
dscli> mkfbvol -dev IBM.2107-75ABNY2 -extpool P0 -type ds -cap 300 -name 0102
```

FB volumes are assigned to hosts using volume groups.

```
dscli> mkvolgrp -dev IBM.2107-75ABNY1 -type scsimask -volume 0000,0001,0100,0102  
volumegrp001  
dscli> mkvolgrp -dev IBM.2107-75ABNY2 -type scsimask -volume 0000,0001,0100,0102  
volumegrp001
```

Volume groups are assigned to hosts using mkhostconnect.

```
dscli> mkhostconnect -dev IBM.2107-75ABNY1 -wwname 10000000c9372787 -hosttype pSeries -  
volgrp V0 king2  
dscli> mkhostconnect -dev IBM.2107-75ABNY2 -wwname 10000000c9372787 -hosttype pSeries -  
volgrp V0 king2
```

Available IO ports are identified between the target and source logical subsystems (LSS's).

```
dscli> lsavailpprcport -l -remotedev IBM.2107-75ABNY2 -remotewwnn 5005076303FFCE40 00:01
```

A pair of local and remote ports is selected and assigned to create a path between the source and target LSS's. Multiple paths should be created.

```
dscli> mkpprcpath -dev IBM.2107-75ABNY1 -remotedev IBM.2107-75ABNY2 --remotewwnn  
5005076303FFCE40 -srcLSS 00 -tgtLSS 00 I0003:I0033
```



```
dscli> mkpprcpath -dev IBM.2107-75ABNY1 -remotedev IBM.2107-75ABNY2 -remotewwnn  
5005076303FFCE40 -srcLss 01 -tgtLss 01 I0003:I0033  
dscli> mkpprcpath -dev IBM.2107-75ABNY2 -remotedev IBM.2107-75ABNY1 -remotewwnn  
5005076303FFC640 --srcLss 00 -tgtLss 00 I0203:I0233  
dscli> mkpprcpath -dev IBM.2107-75ABNY2 -remotedev IBM.2107-75ABNY1 -remotewwnn  
5005076303FFC640 --srcLss 01 -tgtLss 01 I0203:I0233
```

Metro Mirror volume pairs can be established once paths have been established.

```
dscli> mkpprc -dev IBM.2107-75ABNY1 -remotedev IBM.2107-75ABNY2 -type mmir 0001:0001  
dscli> mkpprc -dev IBM.2107-75ABNY1 -remotedev IBM.2107-75ABNY2 -type mmir 0002:0002  
dscli> mkpprc -dev IBM.2107-75ABNY1 -remotedev IBM.2107-75ABNY2 -type mmir 0100:0100  
dscli> mkpprc -dev IBM.2107-75ABNY1 -remotedev IBM.2107-75ABNY2 -type mmir 0102:0102
```

Additional help information can be obtained by running the help command under the DSCLI.

```
dscli> help <command>
```



## Installing VERITAS Storage Foundation

### Symantec product licensing

Installation procedure describes how to activate the key. The VRTSvlic package enables product licensing. After VRTSvlic is installed, the following commands and their manual pages are available on the system:

**vxlicinst** Installs a license key for a Symantec product

**vxlicrep** Displays currently installed licenses

**vxlictest** Retrieves features and their descriptions encoded in a license key

Make sure you have activated the VERITAS Storage Foundation Enterprise HA/DR AIX,5.0 license key.

### Supported AIX operating systems

This release of VERITAS Storage Foundation operates on AIX 5.2 and AIX 5.3 operating systems. Product installation scripts verify required update levels. The installation process terminates if the target systems do not meet maintenance level requirements.

For any VERITAS cluster product, all nodes in the cluster must have the same operating system version and update level.

The minimum system requirements for this release are:

- AIX 5.2 ML6 (legacy) or above
- AIX 5.3 TL4 with SP 4

### Database requirements

The following database version are supported by the respective VERITAS High Availability 5.0MP1 Agents.

- DB2 8.1 ESE, DB2 8.2 ESE, DB2 8.2.2 ESE, DB2 9.1
- Oracle9i, Oracle 10g R1, and Oracle 10g R2 (including 64-bit versions)

### Disk space

Use “Perform a Preinstallation Check” (P) option of the product installer to determine whether there is sufficient space.

### Environment Variables

Most of the commands used in the installation are in the /sbin or /usr/sbin directory. However, there are additional variables needed in order to use a VERITAS Storage Foundation product after installation. Add the following directories to your PATH environment variable:

- If you are using Bourne or Korn shell (sh or ksh), use the commands:

```
$ PATH=$PATH:/usr/sbin:/opt/VRTSvxfs/sbin:/opt/VRTSob/bin:/opt/VRTSvcs/bin:/opt/VRTS/bin$  
MANPATH=/usr/share/man:/opt/VRTS/man:$MANPATH  
$ export PATH MANPATH
```



- If you are using a C shell (csh or tcsh), use the commands:  
`% set path = ( $path /usr/sbin /opt/VRTSvxfs/sbin /opt/VRTSvcs/bin /opt/VRTSob/bin /opt/VRTS/bin )% setenv MANPATH /usr/share/man:/opt/VRTS/man:$MANPATH`

Note: The nroff versions of the online manual pages are not readable using the man command if the bos.txt.tfs fileset is not installed; however, the VRTSvxvm and VRTSvxfs packages install ASCII versions in the /opt/VRTS/ man/catman\* directories that are readable without the bos.txt.tfs fileset.

## Virtual IP Address

This configuration will need several IP addresses depending on the products you are enabling. Have at least six virtual IPs' allocated for the two clusters. The list below shows virtual IPs' required for this configuration.

Purpose	ClustmmP	ClustmmS
DB2 failover	10.140.xx.01	10.140.xx.02
Oracle failover	10.140.xx.03	10.140.xx.04
GCO	10.140.xx.04	10.140.xx.05

## Prerequisites for remote and cluster installation

Establishing communication between nodes is required to install VERITAS software from a remote system, or to install and configure a cluster. The node from which the installation utility is run must have permissions to run rsh (remote shell) or ssh (secure shell) utilities as root on all cluster nodes or remote systems. Make sure that the hosts to be configured as cluster nodes have two or more NIC cards and are connected for heartbeat links. See the *VERITAS Cluster Server installation Guide* for more details.

## Mounting a software disc

You must have superuser (root) privileges to load the VERITAS software.

To mount the VERITAS software disc

1. Log in as superuser.
2. Place the VERITAS software disc into a DVD drive connected to your system.
3. Mount the disc by determining the device access name of the DVD drive.

The format for the device access name is cdX where X is the device number. After inserting the disc, type the following commands:

```
# mkdir -p /cdrom  
# mount -V cdrfs -o ro /dev/cdX /cdrom
```

## Installing using the VERITAS product installer

You can perform an upgrade to Storage Foundation using the VERITAS product installer or product installation script if you already have Storage Foundation installed. Go to the upgrade procedure explained in the next section.





Note: VERITAS products are installed under the /opt directory on the specified host systems. Ensure that the directory /opt exists and has write permissions for root before starting an installation procedure.

The VERITAS product installer is the recommended method to license and install the product. The installer also enables you to configure the product, verify pre-installation requirements, and view the product's description.

You can use the product installer to install VERITAS Storage Foundation and VERITAS Storage Foundation enterprise HA.

At most points during an installation, you can type **b** ("**back**") to return to a previous section of the installation procedure. The **back** feature of the installation scripts is context-sensitive, so it returns to the beginning of a grouped section of questions. If an installation procedure hangs, use Control-c to stop and exit the program. There is a short delay before the script exits.

To install a Storage Foundation product execute the steps from one node in each cluster.

1 Make sure the disc is mounted. See "Mounting a software disc"

2 To invoke the common installer, run the installer command on the disc as shown in this example:

```
# cd /cdrom/disc_name/installer -rsh clustmmPNd1 clustmmPNd2
```

3 Enter **I** to install a product and press Return to begin.

4 When the list of available products is displayed, select the product you want to install and enter the corresponding number and press Return. The product installation begins automatically.

6 Enter the Storage Foundation Enterprise HA/DR product license information.

```
Enter a product_name license key for clustmmPNd1: [?] XXXX-XXXX-XXXX-XXXX-XXXX-XXXX-X
```

```
XXXX-XXXX-XXXX-XXXX-XXXX-XXXX-X successfully registered on clustmmPNd1
```

```
Do you want to enter another license key for clustmmPNd1? [y,n,q,?] (n)
```

```
Enter a product_name license key for clustmmPNd2: [?] XXXX-XXXX-XXXX-XXXX-XXXX-XXXX-X
```

```
XXXX-XXXX-XXXX-XXXX-XXXX-XXXX-X successfully registered on clustmmPNd2
```

```
Do you want to enter another license key for clustmmPNd2? [y,n,q,?] (n)
```

```
Enter n if you have no further license keys to add for a system.
```

You are then prompted to enter the keys for the next system.

Note: Each system requires a product license before installation. License keys for additional product features should also be added at this time.

7 Choose to install all filesets.

SF can be installed without optional filesets to conserve disk space. Additional filesets are typically installed to simplify future upgrades.

1) Required VERITAS Storage Foundation filesets - 928 MB required

2) All VERITAS Storage Foundation filesets - 1063 MB required

```
Select the filesets to be installed on all systems? [1-2,q,?] (2)
```



4. At the installed product list page, enter **y** or press Return to configure the Storage Foundation and VCS products.

It is possible to install SF filesets without performing configuration.

It is optional to configure SF now. If you choose to configure SF later, you can do so manually or run the `installsf -configure` command.

Are you ready to configure SF? [y,n,q] (**y**)

Do you want to configure VCS on these systems at this time? [y,n,q] (**y**) .

5. The installer will ask for details for configuring the VCS cluster for SF. Enter the unique cluster name and Cluster ID number.

```
Enter the unique cluster name: [?] ClustmmP
```

```
Enter the unique Cluster ID number between 0-65535: [b,?] 8
```

6. The installer discovers the NICs available on the first system and reports them:

```
Discovering NICs on clustmmPNd1 ...discovered en0 en1 en2
```

7. Enter private heartbeat NIC information for each host.

```
Enter the NIC for the first private heartbeat link on clustmmPNd1: [b,?] en1
```

```
Would you like to configure a second private heartbeat link?[y,n,q,b,?] (y) y
```

```
Enter the NIC for the second private heartbeat link on clustmmPNd1: [b,?] en2
```

```
en2
```

```
Would you like to configure a third private heartbeat link?[y,n,q,b,?] (n) n
```

```
Do you want to configure an additional low priority heartbeat link? [y,n,q,b,?] (n) n
```

```
(n) n
```

```
Are you using the same NICs for private heartbeat links on all systems?
```

```
[y,n,q,b,?] (y) y
```

Note: When answering **y**, be sure that the same NICs are available on each system; the installer does not verify this.

Notice that in this example, en0 is not selected for use as a private heartbeat NIC because it is already in use as the public network interface.

8. A summary of the information you entered is given. When prompted, confirm that the information is correct.

```
Is this information correct? [y,n,q] (y)
```

If the information is correct, press Return. If the information is not correct, enter **n**. The installer prompts you to enter the information again.

9. When prompted to configure the product to use VERITAS Security Services, enter **y** or **n** to configure.

Note: Before configuring a cluster to operate using VERITAS Security Services, another system must already have VERITAS Security Services installed and be operating as a Root Broker. Refer to the VERITAS Cluster Server Installation Guide for more information on configuring a VxSS Root Broker.

```
Would you like to configure product_name to use VERITAS Security Services? [y,n,q]
```

```
(n) n
```



10. A message displays notifying you of the information required to add users. When prompted, set the user name and /or password for the Administrator.

```
Do you want to set the username and/or password for the Admin user (default
username = 'admin', password='password')? [y,n,q] (n)
```

11. Enter **n** if you want to decline. If you enter **y**, you are prompted to change the password. You are prompted to add another user to the cluster.

```
Do you want to add another user to the cluster? [y,n,q] (n)
```

Enter **n** if you want to decline, enter **y** if you want to add another user. You are prompted to verify the user.

```
Is this information correct? [y,n,q] (y)
```

Enter **y** or **n** to verify if this information is correct.

12. You are prompted to configure the cluster management console. Enter **y** or **n** to configure the cluster management console.

```
Do you want to configure the Cluster Management Console [y,n,q] (n) n
```

13. You are prompted to configure the cluster connector. Enter **y** or **n** to configure the cluster connector.

```
Do you want to configure the cluster connector [y,n,q] (n)
```

14. When prompted to configure SMTP notification, enter **y** to not configure SMTP.

```
Do you want to configure SMTP notification? [y,n,q] (n)
```

15. When prompted to configure SNMP notification, enter **y** to not configure SMTP notification.

```
Do you want to configure SNMP notification? [y,n,q] (n)
```

16. When prompted to set up the default disk group for each system, enter **n** to set up the disk group for each system.

```
Do you want to set up a default disk group for each system?
[y,n,q,?] (n)
```

17. You are prompted to enter the fully qualified hostname of system clustmmP. Enter **y** for the clustmmP.domain\_name.

```
Is the fully qualified hostname of system "clustmmPNd1"
="clustmmPNd1.domain_name"? [y,n,q] (y)
```

18. 23 You are prompted to enter the fully qualified hostname of system clustmmP. Enter **y** for the clustmmS.domain\_name.

```
Is the fully qualified hostname of system "clustmmNd2"
="clustmmPNd2.domain_name"? [y,n,q] (y)
```

19. You are prompted to enable Storage Foundation Management Server Management.

```
Enable Storage Foundation Management Server Management? [y,n,q] (n)
```

20. Enter **n** to enable Storage Foundation Management Server Management. You are prompted to start Storage Foundation.

```
Do you want to start VERITAS Storage Foundation processes now?
[y,n,q] (y)...Startup completed successfully on all systems
```

```
You declined to set up the name of the default disk group for
clustmmPNd1.
```



You declined to set up the name of the default disk group for clustmmPNd2.

Installation log files, summary file, and response file are saved at:

```
/opt/VRTS/install/logs/installsf-7ai12i
```

When installsf installs software, some software may be applied rather than committed. It is the responsibility of the system administrator to commit the software, which can be performed later with the `-c` option of the `installp` command.

21. Proceed to the next section to upgrade VERITAS Storage Foundation 5.0 to 5.0MP1

## Upgrading VERITAS Storage Foundation HA to 5.0MP1

This release of VERITAS products operates on AIX 5.2 and AIX 5.3 operating systems. The minimum system requirements for this release are:

- AIX 5.2 TL8

- AIX 5.3 TL5 with SP 1

Product installation scripts verify the required update levels. The installation process terminates if the target systems do not meet the maintenance level requirements.

You install upgrade packages by running the `installmp` installation script from the command line. The `installmp` script is at the top level of the Maintenance Pack disc. The `installmp` script checks for any installed VERITAS package and replaces it with the updated package.

All VERITAS product processes are stopped after running `installmp`. To ensure that processes restart correctly, a manual restart or system reboot may be required. If you want to configure a product, run the product installation scripts from the `/opt/VRTS/install` directory using the `-configure` option.

See the individual product release notes for information on how to restart processes, or how to uninstall a maintenance pack and return to the previous product level.

Execute the steps from one node in each cluster.

1. Make sure the disc is mounted. See "Mounting a software disc".
2. To install the Storage Foundation software, you must invoke the `installmp` command from one of your cluster nodes using the option that corresponds to your configuration:
3. To install on more than one system using remote shell (RSH) utilities, enter the following command:

```
# ./installmp clustmmP clustmmS -rsh
```

4. After the initial system checks have completed successfully, press Enter to start the requirements checks for the installation.
5. After the requirement checks have completed successfully, press Enter to begin installing the packages.
6. Disable io-fencing on each node in the two clusters by following the steps

```
#cd /etc/VRTSvcS/conf/config
```

```
#echo "vxfen_mode=disabled" > vxfenmode
```



7. After the installation of the packages is complete, use the following command to shut down the system:

```
# shutdown -r now
```

## Installing and Upgrading VCS agent for DB2

---

For complete details refer to the *VERITAS Cluster Server Agent for DB2 Installation and Configuration Guide*. You must install the DB2 agent on each node in the cluster. In global cluster environments, install the agent on each node in each cluster. These instructions assume that you have already installed Cluster Server. Follow steps below to install the agent.

1. Make sure the disc is mounted. See “Mounting a software disc”.
2. Navigate to the location of the agent packages:  
**# cd /cdrom/cluster\_server\_agents/db2\_agent/pkg**
3. Add the filesets for the software  
**# installp -ac -d VRTSvcsdb.rte.bff VRTSvcsdb**
4. Navigate to the location of the 5.0MP1 agent packages:  
**# cd /cdrom/cluster\_server\_agents/db2\_agent/patches**
5. Add the filesets for the software  
**# installp -ac -d VRTSvcsdb.rte.bff VRTSvcsdb**

## Installing and Upgrading VCS agent for Oracle

---

For complete details refer to the *VERITAS Cluster Server Agent for Oracle Installation and Configuration Guide*. You must install the Oracle agent on each node in the cluster. In global cluster environments, install the agent on each node in each cluster. These instructions assume that you have already installed Cluster Server. Follow steps below to install the agent.

1. Make sure the disc is mounted. See “Mounting a software disc”.
2. Navigate to the location of the agent packages:  
**# cd /cdrom/cluster\_server\_agents/oracle\_agent/pkg**
3. Add the filesets for the software  
**# installp -ac -d VRTSvcsor.rte.bff VRTSvcsor**
4. Navigate to the location of the 5.0MP1 agent packages:  
**# cd /cdrom/cluster\_server\_agents/oracle\_agent/patches**
5. Add the filesets for the software  
**# installp -ac -d VRTSvcsor.rte.bff VRTSvcsor**

## Installing VCS agent for Metro Mirror

---

For complete details refer to the *VERITAS Cluster Server Agent for Metro Mirror Installation and Configuration Guide*. You must install the IBM Metro Mirror agent on each node in the cluster. In global cluster environments, install the agent on each node in each cluster. These instructions assume that the Cluster Server is already installed. Follow the steps below to install the agent.

1. Make sure the disc is mounted. See “Mounting a software disc”.

2. Navigate to the location of the agent packages:  
`# cd /cdrom/aix/replication/metro_mirror_agent/version/pkg`  
The variable *version* represents the version of the agent. We have installed version 5.0.2.0
3. Add the filesets for the software  
`# installp -ac -d VRTSvcsi.rte.bff VRTSvcsi`

All of the required Software components have now been installed. You should be able to list out the filesets in Appendix C on each application host.

## Installing and Configuring DB2, Oracle

---

This step involves

- Installation of DB2 and Oracle software
- Creation of a DB2 instance
- Creation of Database

Install DB2 and Oracle on all the nodes of the clusters in Site A and Site B. Make sure that the installation setups are identical, especially the login ids passwords, owner and group permissions and listener port ids.

Refer to the appropriate sections in Appendix B for instructions to setup the database. In this configuration a database representing TPC-C schema is built. A database workload utility is used to populate and simulate the TPC-C workload. You will need workload kits to exercise the database application.

While configuring VxVM make sure to enable the mode for scsi3 disk discovery. The instructions to enable it are included in Appendix B.

## Configuring Applications for Disaster Recovery

---

Most clustered applications can be adapted to a disaster recovery environment by:

- Converting DS8000 devices to Metro Mirror devices and synchronizing the devices
- Setting up the Global cluster
- Setting up DB2 / Oracle failover service group
- Configuring Metro Mirror to the service group

To quickly setup the applications follow the quick setup section. You can follow the manual configuration with the help of reference guides mentioned in the manual configuration section.

### Quick Setup

1. Make sure you have all of the resource names ready
2. Halt the cluster server from any node in the clusters in Site A and Site B  
`##opt/VRTSvcs/hastop -all`
3. cut and paste the appropriate main.cf in Appendix A, one for the DB2 configuration and the other for the Oracle configuration to files in /etc/VRTSvcs/conf/config directory as shown below.  
**On cluster nodes clustmmPNd1, clustmmPNd2 in Site A as :**  
main.cf.db2.siteA and main.cf.ora.siteA

**On cluster nodes clustmmSNd1, clustmmSNd2 in Site B as :**

main.cf.db2.siteB and main.cf.ora.siteB

4. Modify the values of hostnames, IP addresses, mount points etc. to match your configuration.
5. Copy the VCS agent resource files  

```
#cp /etc/VRTSagents/ha/conf/Oracle/OracleTypes.cf /etc/VRTSvcs/conf/config/
#cp /etc/VRTSagents/ha/conf/Db2udb/Db2udbTypes.cf /etc/VRTSvcs/conf/config/
#cp /etc/VRTSvcs/conf/MetroMirrorTypes.cf /etc/VRTSvcs/conf/config/
```
6. Depending on which application you want to test first copy the files as shown below  
 Example for DB2 : On Site A cluster node 1 ( ClustmmPNd1)  

```
#cd /etc/VRTSvcs/con/config
#cp main.cf.db2.siteA main.cf
#rcp main.cf ClustmmPNd2:/etc/VRTSvcs/conf/config/main.cf
```

 On Site B cluster node 1 ( ClustmmPNd1)  

```
#cd /etc/VRTSvcs/con/config
#cp main.cf.db2.siteB main.cf
#rcp main.cf ClustmmSNd2:/etc/VRTSvcs/conf/config/main.cf
```
7. Verify that the main.cf does not have any errors and fix it if there are any issues.  

```
#/opt/VRTSvcs/bin/hacf -verify
```

 If there are no errors the command exits with a zero, and returns to the prompt.
8. start the cluster on each node in the clusters in Site A and B.  

```
#/opt/VRTSvcs/hastop -all
#/opt/VRTSvcs/hastop -all
```
9. Start the cluster Manager from any node in the cluster SiteA. Login to one of the nodes as admin/password  

```
#export DISPLAY=xhost:0
#/opt/VRTSvcs/hagui &
```
10. Now you are ready to manage the clusters

## Manual configuration

You can use this set of procedures if you want to manually configure VCS. Follow the guides mentioned below for more detailed configuration steps.

## Converting DS8000 devices to Metro Mirror devices

DS8000 volumes are configured as resources of type "Metro Mirror". See the "Configuring DS8000 and IBM Metro Mirror" section for converting and synchronizing the DS8000 devices.

## Setting up the Global Cluster

From any node in the clusters in Site A and Site B run the GCO Configuration wizard to create or update the ClusterService group. The wizard verifies your configuration and validates it for a global cluster setup.

```
#/opt/VRTSvcs/bin/gcoconfig
```

The wizard discovers the NIC devices on the local system and prompts you to enter the device to be used for the global cluster.

Specify the name of the device and press Enter. If you do not have NIC resources in your configuration, the wizard asks you whether the specified NIC will be the public NIC used by all systems. Enter **y** if it is the public NIC; otherwise enter **n**. If you entered **n**, the wizard prompts you to enter the names of NICs on all systems.

Enter the virtual IP to be used for the global cluster which you already have identified. If you do not have IP resources in your configuration, the wizard prompts you for the netmask associated with the virtual IP. The wizard detects the netmask; you can accept the suggested value or enter another value. The wizard starts running commands to create or update the ClusterService group. Various messages indicate the status of these commands. After running these commands, the wizard brings the ClusterService group online.

## Linking clusters

Now link the two clusters. The Remote Cluster Configuration wizard provides an easy interface to link clusters. Before linking clusters, verify the virtual IP address for the ClusterAddress attribute for each cluster is set. Use the same IP address as the one assigned to the IP resource in the ClusterService group.

Run the wizard from any cluster. From Cluster Manager, click Edit>Add/Delete Remote Cluster. Follow the instructions in the GUI.

## Configuring DB2 / Oracle failover Service groups

The DB2 and Oracle VCS service groups can be setup once the VCS agents have been installed. In this setup both Applications (DB2 and Oracle) are installed on the same cluster. Two separate VCS configuration files (main.cf) are created for each application. The failover scenarios are performed for one configuration at a time. As an alternative configuration you can include both applications into a single configuration.

Figure 3 shows the dependency graph of the VCS resources within each service group. There are two service groups. The DB2, Oracle service groups and the Cluster service group. The DB2 and Oracle are Global service groups. The DiskGroup resource depends on the Metro Mirror resource. The complete listing of the configuration files (main.cf) are provided in Appendix A.

Refer to the *VERITAS Cluster Server Agent for Oracle Installation and Configuration Guide* and the *VERITAS Cluster Server Agent for Oracle Installation and Configuration Guide* for adding and configuring the DB2 and Oracle resource types.

### Adding the agents manually in a global cluster

Configuring the agent manually in a global cluster involves the following tasks.  
To configure the agent in a global cluster

1. Start Cluster Manager and log on to the cluster.  
**#opt/VRTS/vcs/bin/hagui &**
2. If the agent resource type (Oracle / DB2 / Metro Mirror) is not added to your configuration, add it. From the Cluster Manager **File** menu, choose **Import Types** and select **/etc/VRTSagents/ha/conf/Db2udb/Db2udbTypes.cf** OR **Oracle/OracleTypes.cf**
3. Click **Import**.
4. Save the configuration.
5. Create the Service groups for DB2 / Oracle
6. Add the resources mentioned in the main.cf to the DB2 / Oracle service group.
7. Configure the resources as mentioned in the main.cf and changing the values for your installation
8. Configure the service group as a global group using the Global Group Configuration Wizard. See the *VERITAS Cluster Server User's Guide* for more information.



- Change the ClusterFailOverPolicy from the default, if necessary. Symantec recommends keeping the default, which is Manual, to minimize the chance of failing over on a split-brain.

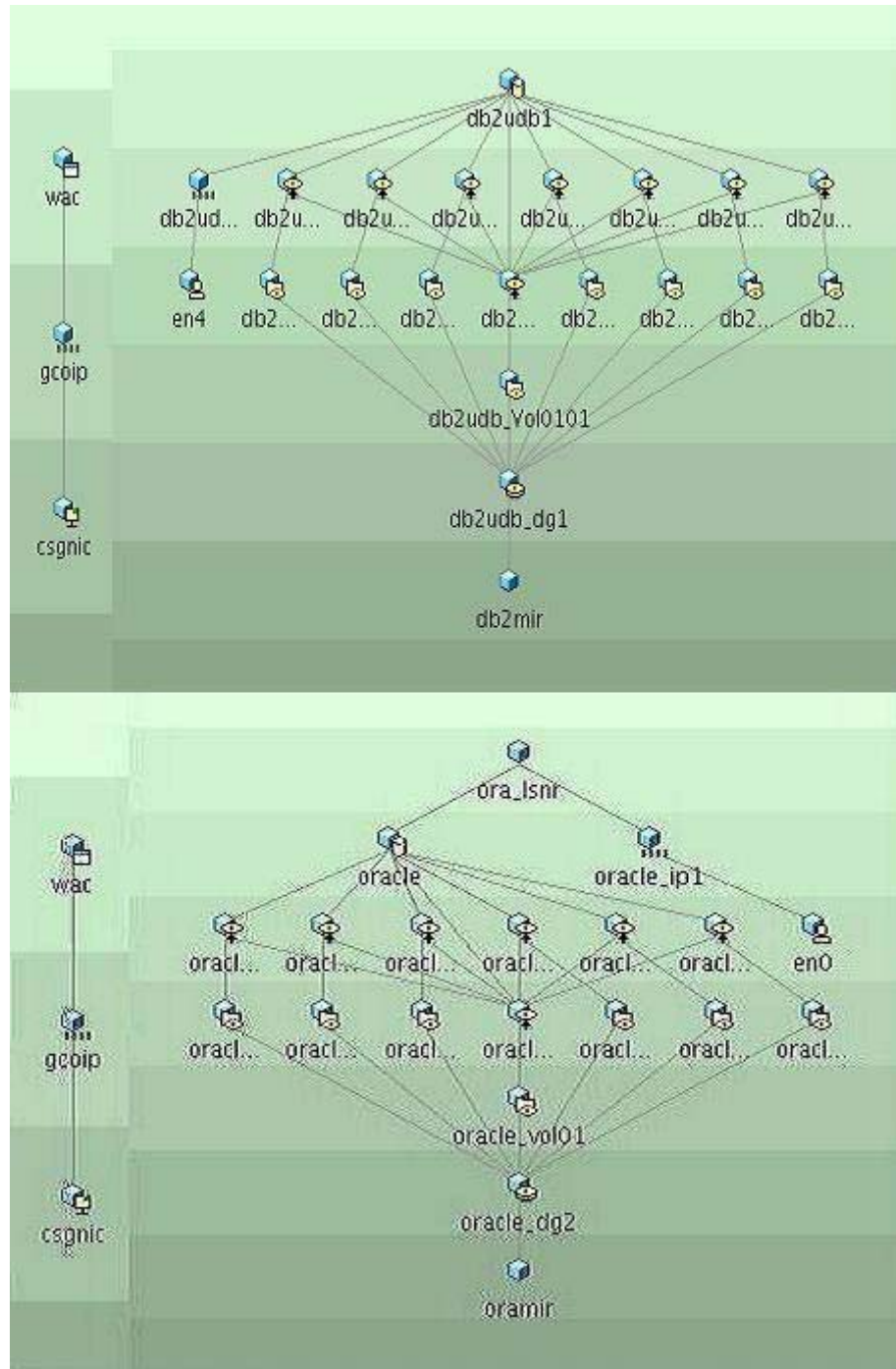


Figure 3. VCS resources dependency tree with Metro Mirror resource

## Configuring and Adding VCS Metro Mirror resource

### Before you configure the agent for Metro Mirror

Before configuring the agent, review the following information:

- Review the configuration concepts, which describe the agent's type definition and attributes. See "Configuration concepts".
- Verify that the agent is installed on all systems in the cluster.
- Ensure that Metro Mirror paths are configured in both directions between the source and the target LSS. Metro Mirror role reversal fails if paths are not configured from the current target LSS to the current source LSS.
- Make sure the cluster has an effective heartbeat mechanism in place. See "About cluster heartbeats".
- Generate the DSCLI password file. Check "DSCLI Installation on the Application hosts".

### About cluster heartbeats

In a VCS cluster, robust heartbeating is accomplished through dual, dedicated networks over which the Low Latency Transport (LLT) runs. Additionally, you can configure a low-priority heartbeat across public networks. In a global cluster, Cluster Server sends ICMP pings over the public network between the two sites for network heartbeating. To minimize the risk of split-brain, VCS sends ICMP pings to highly available IP addresses. VCS global clusters also notify the administrators when the sites cannot communicate.

### Configuration concepts

Review the resource type definition and the attribute definitions for the agent.

### Metro Mirror resource type definition

The IBM Metro Mirror agent is represented by the Metro Mirror resource type in VCS.

```

type Metro Mirror (
  static keylist SupportedActions = {failback}
  static int MonitorInterval = 300
  static int NumThreads = 1
  static str ArgList[] = { DSCLiHome, HMC1, HMC2, User,
  PasswdFile, LocalStorageImageID,
  RemoteStorageImageID, VolIds }
  str DSCLiHome = "/opt/ibm/dscli"
  str HMC1
  str HMC2
  str User
  str PasswdFile
  str LocalStorageImageID
  str RemoteStorageImageID
  str VolIds{}
  temp str VCSResLock

```

### Attribute definitions for the Metro Mirror agent

Review the description of the agent attributes.

### Required attributes

You must assign values to required attributes.

HMC1	IP address or host name of the primary management console.
------	--



Type-dimension:	string-scalar
User	User name for issuing DS CLI commands from the command line.
Type-dimension:	string-scalar
PasswdFile	Specifies the password file that contains your password. See the managepwfile DS CLI command for information on how to generate a password file.
Type-dimension:	string-scalar
LocalStorageImageID	The image ID of the local storage, which consists of manufacturer, type, and serial number. For example, IBM.2107-75FA120
Type-dimension:	string-scalar
RemoteStorageImageID	The image ID of the local storage, which consists of manufacturer, type, and serial number. For example, IBM.3108-75GB248
Type-dimension:	string-scalar
VolIds	IDs of local DS8000 Metro Mirror volumes that the agent manages.
Type-dimension:	string-keylist

### Optional attributes

Configuring these attributes is optional.

DScliHome	Path to the DS8000 command line interface. Default is /opt/ibm/dscli
Type-dimension:	string-scalar
HMC2	IP address or host name of the secondary management console.
Type-dimension:	string-scalar

### Internal attributes

These attributes are for internal use only. Do not modify their values.

VCSResLock	The agent uses the VCSResLock attribute to guarantee serialized management in case of a parallel application.
Type-dimension:	string-scalar

A resource of type Metro Mirror may be configured as follows in main.cf:

```
Metro Mirror db2mir (  
    HMC1 = "XX.XXX.XX.XX"  
    User = admin  
    PasswdFile = "/opt/ibm/dscli/profile/ds_pwfile"  
    LocalStorageImageID = "IBM.2107-75ABNY2"  
    RemoteStorageImageID = "IBM.2107-75ABNY1"  
    VolIds = { 0001, 0002, 0100, 0102 }  
)
```

This resource manages the following objects:

- A group of two Metro Mirror volumes: 001,002, 0100 and 0102 on the local array with the storage image ID IBM.2107-75ABNY1.



- The local array is managed by the HMC dsmc1.abc.symantec.com.
- The Metro Mirror target volumes are on the remote array with the storage image ID IBM.2107-75ABNY2.
- The password file, created using the managepwfile DSCLI command, is located at the path /opt/ibm/dscli/ds\_pwfile.

### Adding the agents manually in a global cluster

Configuring the agent manually in a global cluster involves the following tasks.  
To configure the agent in a global cluster

1. Start Cluster Manager and log on to the cluster.  
**#opt/VRTS/vcs/bin/hagui &**
2. If the agent resource type is not added to your configuration, add it. From the Cluster Manager **File** menu, choose **Import Types** and select /etc/VRTSvcs/conf/Metro MirrorTypes.cf.
3. Click **Import**.
4. Save the configuration.
5. Create the Service groups for Metro Mirror
6. Add a resource of type Metro Mirror at the bottom of the service group.
7. Configure the attributes of the Metro Mirror resource.

## Failover Scenarios

The section describes the procedures to perform the failover scenarios. The setup contains two database applications, a DB2 and an Oracle application. For the failover scenarios it is recommended that you setup two different VCS configuration files one for DB2 and one for Oracle. The scenarios explained below are executed once with VCS configuration for DB2 service group enabled and then with the Oracle service group enabled.

Before you start the scenarios make sure that the both clusters in Site A and Site B are up and running. Login in to any node on each cluster as super user and issue the following command

```
#opr/VRTSvcs/bin/hastatus
```

Start the Cluster management gui

```
#export DISPLAY=xhost:xx.0  
#opt/VRTSvcs/bin/hagui &
```

Make sure that, the Clusterservice groups are online on one node in each cluster. The DB2 / Oracle service group is online on a node of the cluster in Site A.

### Application host failover

In this scenario, a node in the cluster at Site A where the application is online is lost. The application fails over to the second node in the cluster. Next, that node is also lost and since all nodes in the cluster are down the application fails over to a node in the cluster at Site B.

To perform the host failure test



1. Disable the host port on the switch of cluster node say clustmmPNd1 belonging to site A on which the application is online. This action introduces a fault. The service group fails over to the second node i.e. clustmmPNd2 in the cluster in Site A. While the Application is running, ensure all volumes belonging to the Application on the source side are in FULL DUPLEX state and the target volumes are in TARGET FULL DUPLEX state. No copy or synchronization is in progress.
2. Disable the host port on the switch of cluster node say clustmmPNd2
3. A cluster down alert appears and gives the admin opportunity to fail over the service group manually to one of the cluster nodes at Site B, choose say clustmmSNd1. The Metro Mirror devices attached to Site B transition from the TARGET FULL DUPLEX to the FULL DUPLEX state and start on clustmmSNd1.
4. Enable the switch ports of the two cluster nodes at Site A.
5. Switch the service group to its original host i.e. clustmmPNd1 when VCS starts. In the Service Groups tab of the Cluster Manager configuration tree, right-click the service group. Click **Switch To**, and click the clustmmPNd1 on which the service group was initially online.
7. The service group comes online on clustmmPNd1 and Metro Mirror devices swap roles again.

## Disaster Recovery in a Global Cluster Configuration

Test how robust your cluster is in case of a disaster. In this case simulate a disaster by introducing fault to all hosts and the storage at Site A simultaneously.

To perform a disaster test

1. Disable the scsi3 mode on all of the cluster nodes.

***#vxddladm disable SCSI3***

2. Make sure the Application is up and running on one of the cluster node, for example clustmmPNd1 in Site A. While the Application is running, ensure all volumes belonging to the Application on the source side are in FULL DUPLEX state and the target volumes are in TARGET FULL DUPLEX state. No copy or synchronization is in progress at the time of disaster
3. Disable the host ports on the switch of both cluster nodes belonging to site A. This action mimics a disaster scenario from the point of view of Site B. Then after the failover, the Metro Mirror devices attached to Site B go to SUSPENDED state (Reason = "Host Source").
4. The administrator is notified of the failure. Select the Node in Site B on which you the Application to come up. The Application will come online on the selected cluster node in Site B, say clustmmSNd1
5. Reboot the cluster nodes in SiteA.
6. Take the service group offline. Run the command from any node in the cluster clustmmS

***#hagrp -offline db2udb\_grp1 -any***

7. Manually resynchronize the volumes using dscli. After the resynchronization completes, the state of the original target volumes changes to FULL DUPLEX (Reason = "-"). The state of the original source volumes changes to TARGET FULL DUPLEX (Reason = "-"). Establish the pprc path and failback using the `-resetreserve` parameter

***dscli>failbackpprc command: /opt/ibm/dscli/dscli -hmc1 10.140.89.99 -user admin -pwfile /opt/ibm/dscli/profile/ds\_pwfile failbackpprc -dev IBM.2107-75ABNY2 -remotedev IBM.2107-75ABNY1 -type mmir -resetreserve -tgtread 0004:0004 0003:0003***

8. Do a remote online of the Application service group to Site A



## Summary

---

Clustering software, like Veritas Cluster Server, has for many years been the standard approach to protect against failures of individual hardware or software components. As more and more organizations look to add robust disaster recovery capabilities to their mission critical systems, merely shipping backup tapes to an offsite location is not adequate. This whitepaper has shown how a local HA cluster can be extended with DR capabilities. The IBM MetroMirror feature of the System Storage DS8000 is utilized to add data replication capabilities to the solution. VCS Wizards are used to convert two independent clusters (at two different locations) into a global cluster with automated failover capability between locations in the event of a site disaster. The result is a robust DR environment capable of meeting stringent Recovery Time Objectives.



## Appendix A: VCS configuration file (main.cf ) containing DB2, Oracle Application Service group for HA/DR

Site A : Primary Cluster Nodes : clustmmPNd1 clustmmPNd2	Site B : Secondary Cluster Nodes : clustmmSNd1 , clustmmSNd2
VCS configuration file for DB2	
main.cf.db2.siteA	main.cf.db2.siteB
<pre>include "types.cf" include "Db2udbTypes.cf" include "MetroMirrorTypes.cf"  cluster clustmmP (   UserNames = { admin = GLMeLGIIhMMkUMgLJ     = "D&amp;m6aqCd" }   ClusterAddress = "10.140.xx.05"   Administrators = { admin } )  remotecluster clustmmS (   ClusterAddress = "10.140.xx.06" )  heartbeat lcmp (   ClusterList = { clustmmS }   AYATimeout = 30   Arguments @clustmmS = { "10.140.xx.06" } )  system clustmmPNd1 ( )  system clustmmPNd2 ( )  group ClusterService (   SystemList = { clustmmPNd1 = 0, clustmmPNd2 = 0 }   AutoStartList = { clustmmPNd1, clustmmPNd2 }   OnlineRetryLimit = 3   OnlineRetryInterval = 120 )  Application wac (   StartProgram = "/opt/VRTSvcs/bin/wacstart"   StopProgram = "/opt/VRTSvcs/bin/wacstop"   MonitorProcesses = { "/opt/VRTSvcs/bin/wac" }   RestartLimit = 3 )  IP gcoip (   Device = en0   Address = "10.140.xx.05"   NetMask = "255.255.248.0" )  NIC csgnic (   Enabled = 0   Device = en0 )  gcoip requires csgnic</pre>	<pre>include "types.cf" include "Db2udbTypes.cf" include "MetroMirrorTypes.cf"  cluster clustmmS (   UserNames = { admin = eHlAhChEIdllgQlChF     = "tp&amp;NGtWb" }   ClusterAddress = "10.140.xx.06"   Administrators = { admin } )  remotecluster clustmmP (   ClusterAddress = "10.140.xx.05" )  heartbeat lcmp (   ClusterList = { clustmmP }   AYATimeout = 30   Arguments @clustmmP = { "10.140.xx.05" } )  system clustmmSNd1 ( )  system clustmmSNd2 ( )  group ClusterService (   SystemList = { clustmmSNd1 = 0, clustmmSNd2 = 0 }   AutoStartList = { clustmmSNd1, clustmmSNd2 }   OnlineRetryLimit = 3   OnlineRetryInterval = 120 )  Application wac (   StartProgram = "/opt/VRTSvcs/bin/wacstart"   StopProgram = "/opt/VRTSvcs/bin/wacstop"   MonitorProcesses = { "/opt/VRTSvcs/bin/wac" }   RestartLimit = 3 )  IP gcoip (   Device = en4   Address = "10.140.xx.06"   NetMask = "255.255.248.0" )  NIC csgnic (   Enabled = 0   Device = en4 )  gcoip requires csgnic</pre>

<pre>wac requires gcoip  // resource dependency tree // // group ClusterService // { //   Application wac //   { //     IP gcoip //     { //       NIC csgnic //     } //   } // }  group db2udb_grp1 (   SystemList = { clustmmPNd1 = 0, clustmmPNd2 = 1 }   ClusterList = { clustmmP = 0, clustmmS = 1 }   AutoStartList = { clustmmPNd1, clustmmPNd2 }   ClusterFailOverPolicy = Auto )  Db2udb db2udb1 (   DB2InstOwner = db2inst   DB2InstHome = "/db2inst/instance"   IndepthMonitor = 1   DatabaseName = TPCC1 )  DiskGroup db2udb_dg1 (   DiskGroup = db2dg   StartVolumes = 0 )  IP db2udb_ip1 (   Device = en0   Address = "10.140.xx.01"   NetMask = "255.255.248.0" )  MetroMirror db2mir (   HMC1 = "10.140.xx.xxx"   User = admin   PasswdFile = "/opt/ibm/dscli/profile/ds_pwfile"   LocalStorageImageID = "IBM.2107-75ABNY1"   RemoteStorageImageID = "IBM.2107-75ABNY2"   VolIds = { 0001, 0002, 0100, 0102 } )  Mount db2udb_mnt1 (   MountPoint = "/db2inst"   BlockDevice = "/dev/vx/dsk/db2dg/base"   FSType = vxfs   FscOpt = "-y" )  Mount db2udb_mnt2 (   MountPoint = "/db2inst/db/temp"   BlockDevice = "/dev/vx/dsk/db2dg/dbtemp"   FSType = vxfs   FscOpt = "-y" )  Mount db2udb_mnt3 (   MountPoint = "/db2inst/db/mnt1"   BlockDevice = "/dev/vx/dsk/db2dg/mnt1"</pre>	<pre>wac requires gcoip  // resource dependency tree // // group ClusterService // { //   Application wac //   { //     IP gcoip //     { //       NIC csgnic //     } //   } // }  group db2udb_grp1 (   SystemList = { clustmmSNd1 = 0, clustmmSNd2 = 1 }   ClusterList = { clustmmP = 0, clustmmS = 1 }   Authority = 1   AutoStartList = { clustmmSNd2, clustmmSNd1 }   ClusterFailOverPolicy = Auto )  Db2udb db2udb1 (   DB2InstOwner = db2inst   DB2InstHome = "/db2inst/instance"   IndepthMonitor = 1   DatabaseName = TPCC1 )  DiskGroup db2udb_dg1 (   DiskGroup = db2dg   StartVolumes = 0 )  IP db2udb_ip1 (   Device = en4   Address = "10.140.xx.02"   NetMask = "255.255.248.0" )  MetroMirror db2mir (   HMC1 = "10.140.xx.xxx"   User = admin   PasswdFile = "/opt/ibm/dscli/profile/ds_pwfile"   LocalStorageImageID = "IBM.2107-75ABNY2"   RemoteStorageImageID = "IBM.2107-75ABNY1"   VolIds = { 0001, 0002, 0100, 0102 } )  Mount db2udb_mnt1 (   MountPoint = "/db2inst"   BlockDevice = "/dev/vx/dsk/db2dg/base"   FSType = vxfs   FscOpt = "-y" )  Mount db2udb_mnt2 (   MountPoint = "/db2inst/db/temp"   BlockDevice = "/dev/vx/dsk/db2dg/dbtemp"   FSType = vxfs   FscOpt = "-y" )  Mount db2udb_mnt3 (   MountPoint = "/db2inst/db/mnt1"</pre>
--	---



<pre> FSType = vxfs FsckOpt = "-y" )  Mount db2udb_mnt4 (   MountPoint = "/db2inst/db/mnt2"   BlockDevice = "/dev/vx/dsk/db2dg/mnt2"   FSType = vxfs   FsckOpt = "-y" )  Mount db2udb_mnt5 (   MountPoint = "/db2inst/db/mnt3"   BlockDevice = "/dev/vx/dsk/db2dg/mnt3"   FSType = vxfs   FsckOpt = "-y" )  Mount db2udb_mnt6 (   MountPoint = "/db2inst/db/mnt4"   BlockDevice = "/dev/vx/dsk/db2dg/mnt4"   FSType = vxfs   FsckOpt = "-y" )  Mount db2udb_mnt7 (   MountPoint = "/db2inst/db/log"   BlockDevice = "/dev/vx/dsk/db2dg/log"   FSType = vxfs   FsckOpt = "-y" )  Mount db2udb_mnt8 (   MountPoint = "/db2inst/backup"   BlockDevice = "/dev/vx/dsk/db2dg/backup"   FSType = vxfs   FsckOpt = "-y" )  Proxy en0 (   Enabled = 0   TargetResName = csgnic )  Volume db2udb_Vol0101 (   Volume = base   DiskGroup = db2dg )  Volume db2udb_Vol0102 (   Volume = dbtemp   DiskGroup = db2dg )  Volume db2udb_Vol0103 (   Volume = mnt1   DiskGroup = db2dg )  Volume db2udb_Vol0104 (   Volume = mnt2   DiskGroup = db2dg )  Volume db2udb_Vol0105 (   Volume = mnt3   DiskGroup = db2dg ) </pre>	<pre> BlockDevice = "/dev/vx/dsk/db2dg/mnt1" FSType = vxfs FsckOpt = "-y" )  Mount db2udb_mnt4 (   MountPoint = "/db2inst/db/mnt2"   BlockDevice = "/dev/vx/dsk/db2dg/mnt2"   FSType = vxfs   FsckOpt = "-y" )  Mount db2udb_mnt5 (   MountPoint = "/db2inst/db/mnt3"   BlockDevice = "/dev/vx/dsk/db2dg/mnt3"   FSType = vxfs   FsckOpt = "-y" )  Mount db2udb_mnt6 (   MountPoint = "/db2inst/db/mnt4"   BlockDevice = "/dev/vx/dsk/db2dg/mnt4"   FSType = vxfs   FsckOpt = "-y" )  Mount db2udb_mnt7 (   MountPoint = "/db2inst/db/log"   BlockDevice = "/dev/vx/dsk/db2dg/log"   FSType = vxfs   FsckOpt = "-y" )  Mount db2udb_mnt8 (   MountPoint = "/db2inst/backup"   BlockDevice = "/dev/vx/dsk/db2dg/backup"   FSType = vxfs   FsckOpt = "-y" )  Proxy en4 (   Enabled = 0   TargetResName = csgnic )  Volume db2udb_Vol0101 (   Volume = base   DiskGroup = db2dg )  Volume db2udb_Vol0102 (   Volume = dbtemp   DiskGroup = db2dg )  Volume db2udb_Vol0103 (   Volume = mnt1   DiskGroup = db2dg )  Volume db2udb_Vol0104 (   Volume = mnt2   DiskGroup = db2dg )  Volume db2udb_Vol0105 (   Volume = mnt3   DiskGroup = db2dg ) </pre>
--	---



<pre> Volume db2udb_Vol0106 (   Volume = mnt4   DiskGroup = db2dg )  Volume db2udb_Vol0107 (   Volume = log   DiskGroup = db2dg )  Volume db2udb_Vol0108 (   Volume = backup   DiskGroup = db2dg )  db2udb1 requires db2udb_ip1 db2udb1 requires db2udb_mnt1 db2udb1 requires db2udb_mnt2 db2udb1 requires db2udb_mnt3 db2udb1 requires db2udb_mnt4 db2udb1 requires db2udb_mnt5 db2udb1 requires db2udb_mnt6 db2udb1 requires db2udb_mnt7 db2udb1 requires db2udb_mnt8 db2udb_Vol0101 requires db2udb_dg1 db2udb_Vol0102 requires db2udb_dg1 db2udb_Vol0103 requires db2udb_dg1 db2udb_Vol0104 requires db2udb_dg1 db2udb_Vol0105 requires db2udb_dg1 db2udb_Vol0106 requires db2udb_dg1 db2udb_Vol0107 requires db2udb_dg1 db2udb_Vol0108 requires db2udb_dg1 db2udb_dg1 requires db2mir db2udb_ip1 requires en0 db2udb_mnt1 requires db2udb_Vol0101 db2udb_mnt2 requires db2udb_Vol0102 db2udb_mnt2 requires db2udb_mnt1 db2udb_mnt3 requires db2udb_Vol0103 db2udb_mnt3 requires db2udb_mnt1 db2udb_mnt4 requires db2udb_Vol0104 db2udb_mnt4 requires db2udb_mnt1 db2udb_mnt5 requires db2udb_Vol0105 db2udb_mnt5 requires db2udb_mnt1 db2udb_mnt6 requires db2udb_Vol0106 db2udb_mnt6 requires db2udb_mnt1 db2udb_mnt7 requires db2udb_Vol0107 db2udb_mnt7 requires db2udb_mnt1 db2udb_mnt8 requires db2udb_Vol0108 db2udb_mnt8 requires db2udb_mnt1 </pre>	<pre> )  Volume db2udb_Vol0106 (   Volume = mnt4   DiskGroup = db2dg )  Volume db2udb_Vol0107 (   Volume = log   DiskGroup = db2dg )  Volume db2udb_Vol0108 (   Volume = backup   DiskGroup = db2dg )  db2udb1 requires db2udb_ip1 db2udb1 requires db2udb_mnt1 db2udb1 requires db2udb_mnt2 db2udb1 requires db2udb_mnt3 db2udb1 requires db2udb_mnt4 db2udb1 requires db2udb_mnt5 db2udb1 requires db2udb_mnt6 db2udb1 requires db2udb_mnt7 db2udb1 requires db2udb_mnt8 db2udb_Vol0101 requires db2udb_dg1 db2udb_Vol0102 requires db2udb_dg1 db2udb_Vol0103 requires db2udb_dg1 db2udb_Vol0104 requires db2udb_dg1 db2udb_Vol0105 requires db2udb_dg1 db2udb_Vol0106 requires db2udb_dg1 db2udb_Vol0107 requires db2udb_dg1 db2udb_Vol0108 requires db2udb_dg1 db2udb_dg1 requires db2mir db2udb_ip1 requires en4 db2udb_mnt1 requires db2udb_Vol0101 db2udb_mnt2 requires db2udb_Vol0102 db2udb_mnt2 requires db2udb_mnt1 db2udb_mnt3 requires db2udb_Vol0103 db2udb_mnt3 requires db2udb_mnt1 db2udb_mnt4 requires db2udb_Vol0104 db2udb_mnt4 requires db2udb_mnt1 db2udb_mnt5 requires db2udb_Vol0105 db2udb_mnt5 requires db2udb_mnt1 db2udb_mnt6 requires db2udb_Vol0106 db2udb_mnt6 requires db2udb_mnt1 db2udb_mnt7 requires db2udb_Vol0107 db2udb_mnt7 requires db2udb_mnt1 db2udb_mnt8 requires db2udb_Vol0108 db2udb_mnt8 requires db2udb_mnt1 </pre>
VCS configuration file for Oracle	
main.cf.ora.siteA	main.cf.ora.siteB
<pre> include "types.cf" include "ClusterConnectorConfigType.cf" include "MetroMirrorTypes.cf" include "OracleTypes.cf"  cluster clustmmP (   UserNames = { admin = GLMeLGIIhMMkUMgLJ                 = "D&amp;m6aqCd" }   ClusterAddress = "10.140.xx.05"   Administrators = { admin } )  remotecluster clustmmS (   ClusterAddress = "10.140.xx.06" ) </pre>	<pre> include "types.cf" include "ClusterConnectorConfigType.cf" include "MetroMirrorTypes.cf" include "OracleTypes.cf"  cluster clustmmS (   UserNames = { admin = eHlaHChEldIlgQIcHF                 = "tp&amp;NGtWb" }   ClusterAddress = "10.140.xx.06"   Administrators = { admin } )  remotecluster clustmmP (   ClusterAddress = "10.140.xx.05" ) </pre>

<pre> heartbeat lcmp (   ClusterList = { clustmmS }   AYATimeout = 30   Arguments @clustmmS = { "10.140.xx.06" } )  system clustmmPNd1 ( )  system clustmmPNd2 ( )  group ClusterService (   SystemList = { clustmmPNd1 = 0, clustmmPNd2 = 0 }   AutoStartList = { clustmmPNd1, clustmmPNd2 }   OnlineRetryLimit = 3   OnlineRetryInterval = 120 )  Application wac (   StartProgram = "/opt/VRTSvcs/bin/wacstart"   StopProgram = "/opt/VRTSvcs/bin/wacstop"   MonitorProcesses = { "/opt/VRTSvcs/bin/wac" }   RestartLimit = 3 )  IP gcoip (   Device = en0   Address = "10.140.xx.05"   NetMask = "255.255.248.0" )  NIC gconic (   Device = en0 )  gcoip requires gconic wac requires gcoip  // resource dependency tree // // group ClusterService // { //   Application wac //   { //     IP gcoip //     { //       NIC gconic //     } //   } // }  group oracle_grp1 (   SystemList = { clustmmPNd1 = 0, clustmmPNd2 = 1 }   ClusterList = { clustmmP = 0, clustmmS = 1 }   Authority = 1   AutoStartList = { clustmmPNd1, clustmmPNd2 }   ClusterFailOverPolicy = Auto )  DiskGroup oracle_dg2 (   DiskGroup = oradata )  IP oracle_ip1 ( </pre>	<pre> heartbeat lcmp (   ClusterList = { clustmmP }   AYATimeout = 30   Arguments @clustmmP = { "10.140.xx.05" } )  system clustmmSNd1 ( )  system clustmmSNd2 ( )  group ClusterService (   SystemList = { clustmmSNd1 = 0, clustmmSNd2 = 0 }   AutoStartList = { clustmmSNd1, clustmmSNd2 }   OnlineRetryLimit = 3   OnlineRetryInterval = 120 )  Application wac (   StartProgram = "/opt/VRTSvcs/bin/wacstart"   StopProgram = "/opt/VRTSvcs/bin/wacstop"   MonitorProcesses = { "/opt/VRTSvcs/bin/wac" }   RestartLimit = 3 )  IP gcoip (   Device = en4   Address = "10.140.xx.06"   NetMask = "255.255.248.0" )  NIC csgnic (   Device = en4 )  gcoip requires csgnic wac requires gcoip  // resource dependency tree // // group ClusterService // { //   Application wac //   { //     IP gcoip //     { //       NIC csgnic //     } //   } // }  group oracle_grp1 (   SystemList = { clustmmSNd1 = 0, clustmmSNd2 = 1 }   ClusterList = { clustmmP = 0, clustmmS = 1 }   AutoStartList = { clustmmSNd1, clustmmSNd2 }   ClusterFailOverPolicy = Auto )  DiskGroup oracle_dg2 (   DiskGroup = oradata )  IP oracle_ip1 ( </pre>
--	--



<pre>Device = en0 Address = "10.140.xx.03" NetMask = "255.255.248.0" )  MetroMirror oramir ( HMC1 = "10.140.xx.xxx" User = admin PasswdFile = "/opt/ibm/dscli/profile/ds_pwfile" LocalStorageImageID = "IBM.2107-75ABNY1" RemoteStorageImageID = "IBM.2107-75ABNY2" VolIds = { 0003, 0004 } )  Mount oracle_mnt1 ( MountPoint = "/oradata" BlockDevice = "/dev/vx/dsk/oradata/mnt1" FSType = vxfs FsckOpt = "-y -o full" )  Mount oracle_mnt2 ( MountPoint = "/oradata/tpcc/mnt2" BlockDevice = "/dev/vx/dsk/oradata/mnt2" FSType = vxfs FsckOpt = "-y -o full" )  Mount oracle_mnt3 ( MountPoint = "/oradata/tpcc/mnt3" BlockDevice = "/dev/vx/dsk/oradata/mnt3" FSType = vxfs FsckOpt = "-y -o full" )  Mount oracle_mnt4 ( MountPoint = "/oradata/tpcc/mnt4" BlockDevice = "/dev/vx/dsk/oradata/mnt4" FSType = vxfs FsckOpt = "-y -o full" )  Mount oracle_mnt5 ( MountPoint = "/oradata/tpcc/temp" BlockDevice = "/dev/vx/dsk/oradata/dbtemp" FSType = vxfs FsckOpt = "-y -o full" )  Mount oracle_mnt6 ( MountPoint = "/oradata/tpcc/log" BlockDevice = "/dev/vx/dsk/oradata/log" FSType = vxfs FsckOpt = "-y -o full" )  Mount oracle_mnt7 ( MountPoint = "/oradata/backup" BlockDevice = "/dev/vx/dsk/oradata/backup" FSType = vxfs FsckOpt = "-y -o full" )  Netlsnr ora_lsnr ( Owner = oracle Home = "/oracle/orahome/" TnsAdmin = "/oracle/orahome/network/admin" Listener @gl-ax04 = LISTENER_clustmmPNd1</pre>	<pre>Device = en4 Address = "10.140.xx.04" NetMask = "255.255.248.0" )  MetroMirror oramir ( HMC1 = "10.140.xx.xxx" User = admin PasswdFile = "/opt/ibm/dscli/profile/ds_pwfile" LocalStorageImageID = "IBM.2107-75ABNY2" RemoteStorageImageID = "IBM.2107-75ABNY1" VolIds = { 0003, 0004 } )  Mount oracle_mnt1 ( MountPoint = "/oradata" BlockDevice = "/dev/vx/dsk/oradata/mnt1" FSType = vxfs FsckOpt = "-y -o full" )  Mount oracle_mnt2 ( MountPoint = "/oradata/tpcc/mnt2" BlockDevice = "/dev/vx/dsk/oradata/mnt2" FSType = vxfs FsckOpt = "-y -o full" )  Mount oracle_mnt3 ( MountPoint = "/oradata/tpcc/mnt3" BlockDevice = "/dev/vx/dsk/oradata/mnt3" FSType = vxfs FsckOpt = "-y -o full" )  Mount oracle_mnt4 ( MountPoint = "/oradata/tpcc/mnt4" BlockDevice = "/dev/vx/dsk/oradata/mnt4" FSType = vxfs FsckOpt = "-y -o full" )  Mount oracle_mnt5 ( MountPoint = "/oradata/tpcc/temp" BlockDevice = "/dev/vx/dsk/oradata/dbtemp" FSType = vxfs FsckOpt = "-y -o full" )  Mount oracle_mnt6 ( MountPoint = "/oradata/tpcc/log" BlockDevice = "/dev/vx/dsk/oradata/log" FSType = vxfs FsckOpt = "-y -o full" )  Mount oracle_mnt7 ( MountPoint = "/oradata/backup" BlockDevice = "/dev/vx/dsk/oradata/backup" FSType = vxfs FsckOpt = "-y -o full" )  Netlsnr ora_lsnr ( Owner = oracle Home = "/oracle/orahome/" TnsAdmin = "/oracle/orahome/network/admin" Listener @gl-ax04 = LISTENER_clustmmSND1</pre>
--	--



<pre>Listener @gl-ax05 = LISTENER_ clustmmPNd2 LsnrPwd = iwmWiuLul )  Oracle oracle (   Sid = tpcc   Owner = oracle   Home = "/oracle/orahome"   Pfile = "/oracle/orahome/dbs/inittpcc.ora"   StartUpOpt = STARTUP   EnvFile = "/oracle/.profile"   DetailMonitor = 1   User = tpcc   Pword = GUKuGSjSJ   Table = test )  Proxy en0 (   TargetResName = gconic )  Volume oracle_vol01 (   Volume = mnt1   DiskGroup = oradata )  Volume oracle_vol02 (   Volume = mnt2   DiskGroup = oradata )  Volume oracle_vol03 (   Volume = mnt3   DiskGroup = oradata )  Volume oracle_vol04 (   Volume = mnt4   DiskGroup = oradata )  Volume oracle_vol05 (   Volume = dbtemp   DiskGroup = oradata )  Volume oracle_vol06 (   Volume = log   DiskGroup = oradata )  Volume oracle_vol07 (   Volume = backup   DiskGroup = oradata )  ora_lsnr requires oracle ora_lsnr requires oracle_ip1 oracle requires oracle_mnt1 oracle requires oracle_mnt2 oracle requires oracle_mnt3 oracle requires oracle_mnt4 oracle requires oracle_mnt5 oracle requires oracle_mnt6 oracle requires oracle_mnt7 oracle_dg2 requires oramir oracle_ip1 requires en0 oracle_mnt1 requires oracle_vol01</pre>	<pre>Listener @gl-ax05 = LISTENER_ clustmmSnd2 LsnrPwd = iwmWiuLul )  Oracle oracle (   Sid = tpcc   Owner = oracle   Home = "/oracle/orahome"   Pfile = "/oracle/orahome/dbs/inittpcc.ora"   StartUpOpt = STARTUP   EnvFile = "/oracle/.profile"   DetailMonitor = 1   User = tpcc   Pword = GUKuGSjSJ   Table = test )  Proxy en0 (   TargetResName = gconic )  Volume oracle_vol01 (   Volume = mnt1   DiskGroup = oradata )  Volume oracle_vol02 (   Volume = mnt2   DiskGroup = oradata )  Volume oracle_vol03 (   Volume = mnt3   DiskGroup = oradata )  Volume oracle_vol04 (   Volume = mnt4   DiskGroup = oradata )  Volume oracle_vol05 (   Volume = dbtemp   DiskGroup = oradata )  Volume oracle_vol06 (   Volume = log   DiskGroup = oradata )  Volume oracle_vol07 (   Volume = backup   DiskGroup = oradata )  ora_lsnr requires oracle ora_lsnr requires oracle_ip1 oracle requires oracle_mnt1 oracle requires oracle_mnt2 oracle requires oracle_mnt3 oracle requires oracle_mnt4 oracle requires oracle_mnt5 oracle requires oracle_mnt6 oracle requires oracle_mnt7 oracle_dg2 requires oramir oracle_ip1 requires en0 oracle_mnt1 requires oracle_vol01</pre>
--	--



<p>oracle_mnt2 requires oracle_vol02 oracle_mnt2 requires oracle_mnt1 oracle_mnt3 requires oracle_vol03 oracle_mnt3 requires oracle_mnt1 oracle_mnt4 requires oracle_vol04 oracle_mnt4 requires oracle_mnt1 oracle_mnt5 requires oracle_vol05 oracle_mnt5 requires oracle_mnt1 oracle_mnt6 requires oracle_vol06 oracle_mnt6 requires oracle_mnt1 oracle_mnt7 requires oracle_vol07 oracle_mnt7 requires oracle_mnt1 oracle_vol01 requires oracle_dg2 oracle_vol02 requires oracle_dg2 oracle_vol03 requires oracle_dg2 oracle_vol04 requires oracle_dg2 oracle_vol05 requires oracle_dg2 oracle_vol06 requires oracle_dg2 oracle_vol07 requires oracle_dg2</p>	<p>oracle_mnt2 requires oracle_vol02 oracle_mnt2 requires oracle_mnt1 oracle_mnt3 requires oracle_vol03 oracle_mnt3 requires oracle_mnt1 oracle_mnt4 requires oracle_vol04 oracle_mnt4 requires oracle_mnt1 oracle_mnt5 requires oracle_vol05 oracle_mnt5 requires oracle_mnt1 oracle_mnt6 requires oracle_vol06 oracle_mnt6 requires oracle_mnt1 oracle_mnt7 requires oracle_vol07 oracle_mnt7 requires oracle_mnt1 oracle_vol01 requires oracle_dg2 oracle_vol02 requires oracle_dg2 oracle_vol03 requires oracle_dg2 oracle_vol04 requires oracle_dg2 oracle_vol05 requires oracle_dg2 oracle_vol06 requires oracle_dg2 oracle_vol07 requires oracle_dg2</p>
---	---

## Appendix B: Setting up the Database Applications

---

### Setting up the DB2 Database Application

#### Create Groups

```
mkgroup -A id=310 db2adm
mkgroup -A id=311 db2fadm
```

#### Create Users

```
DB2INST=db2inst
mkdir /$DB2INST
mkuser id=320 pgrp=db2adm groups=db2adm home=/$DB2INST/instance db2inst
mkuser id=321 pgrp=db2fadm groups=db2fadm home=/$DB2INST/db2fenc db2fenc
passwd db2inst
passwd db2fenc
```

su to another user and then to the user to confirm the password

```
su guest
su db2inst
su db2fenc
```

#### Edit /etc/services

Update /etc/services on all the systems that will be running the database, add the following:

```
DB2_tpcc1      60000/tcp
DB2_tpcc1_1    60001/tcp
DB2_tpcc1_2    60002/tcp
DB2_tpcc1_END  60003/tcp
DB2_db2inst    60004/tcp
DB2_db2inst_1  60005/tcp
DB2_db2inst_2  60006/tcp
DB2_db2inst_END 60007/tcp
```

NOTE: "tpcc1" needs to match what the "SVCENAME" is set to for the database. (db2 get dbm cfg | grep SVCENAME)

NOTE: "db2inst" needs to match the instance name. This is for the DAS instance.

#### Configure VxVM Disks

```
vxddladm enablescsi3 policy=aa
```

```
DG=db2dg
DB2INST=db2inst
vxdg init $DG <disk-name> cds=off
```

```
# If you want simple volumes
```



```
vxassist -g $DG make base 2g layout=mirror &
vxassist -g $DG make dbtemp 2g layout=mirror &
vxassist -g $DG make mnt1 2g layout=mirror &
vxassist -g $DG make mnt2 2g layout=mirror &
vxassist -g $DG make mnt3 2g layout=mirror&
vxassist -g $DG make mnt4 2g layout=mirror&
vxassist -g $DG make log 20g layout=concat &
vxassist -g $DG make backup 6g layout=concat &
wait
vxvol -g $DG startall
V=V
mkfs -$V vxfs /dev/vx/rdisk/$DG/base
mkfs -$V vxfs /dev/vx/rdisk/$DG/dbtemp
mkfs -$V vxfs /dev/vx/rdisk/$DG/mnt1
mkfs -$V vxfs /dev/vx/rdisk/$DG/mnt2
mkfs -$V vxfs /dev/vx/rdisk/$DG/mnt3
mkfs -$V vxfs /dev/vx/rdisk/$DG/mnt4
mkfs -$V vxfs /dev/vx/rdisk/$DG/backup
mkfs -$V vxfs /dev/vx/rdisk/$DG/log
mkdir -p /$DB2INST
mount -$V vxfs /dev/vx/dsk/$DG/base /$DB2INST
mkdir -p /$DB2INST/db/temp
mount -$V vxfs /dev/vx/dsk/$DG/dbtemp /$DB2INST/db/temp
mkdir -p /$DB2INST/db/mnt1
mkdir -p /$DB2INST/db/mnt2
mkdir -p /$DB2INST/db/mnt3
mkdir -p /$DB2INST/db/mnt4
mount -$V vxfs /dev/vx/dsk/$DG/mnt1 /$DB2INST/db/mnt1
mount -$V vxfs /dev/vx/dsk/$DG/mnt2 /$DB2INST/db/mnt2
mount -$V vxfs /dev/vx/dsk/$DG/mnt3 /$DB2INST/db/mnt3
mount -$V vxfs /dev/vx/dsk/$DG/mnt4 /$DB2INST/db/mnt4
mkdir -p /$DB2INST/db/log
mount -$V vxfs /dev/vx/dsk/$DG/log /$DB2INST/db/log
mkdir -p /$DB2INST/backup
mount -$V vxfs /dev/vx/dsk/$DG/backup /$DB2INST/backup
chown db2inst /$DB2INST /$DB2INST/db /$DB2INST/db/* /$DB2INST/backup
chgrp db2adm /$DB2INST /$DB2INST/db /$DB2INST/db/* /$DB2INST/backup
mkdir /$DB2INST/instance /$DB2INST/db2fenc
touch /$DB2INST/instance/.profile
echo "+ +" >/$DB2INST/dbinst/.rhosts
chown db2inst /$DB2INST/instance /$DB2INST/instance/.profile
chown db2fenc /$DB2INST/db2fenc
chgrp db2adm /$DB2INST/instance /$DB2INST/instance/.profile
chgrp db2fadm /$DB2INST/db2fenc
chown db2inst /opt/VRTSdb2ed
chgrp db2adm /opt/VRTSdb2ed
```

## Database installation

On each system install the db2 bits. Mount the Db2 Software disk OR have access to the DB2 software store.

```
cd ese.dbs
./db2_install -p ESE
```





## Database creation

Create instance (as root)

```
USER=db2inst
DB2INST=db2inst
[ -d /opt/IBM/db2/V9.1/instance ]
cd /opt/IBM/db2/V9.1/instance
[ ! -f /db2inst/instance/.profile ]
./db2idrop db2inst
./db2icrt -a server -p 70000 -s ese -u db2inst db2inst
```

Set up the database (db2inst)

NOTE: DO NOT RUN THIS WHILE THE DATABASE IS UNDER VCS CONTROL  
NOTE: DO NOT RUN THIS WHILE THE DATABASE IS UNDER VCS CONTROL

```
su - db2inst
echo "PATH=$PATH:/opt/VRTSdb2ed/bin" >> .profile
exit
su - db2inst
```

Create database

```
DB2INST=db2inst
DBNAME=tpcc1
SVNAME=DB2_tpcc1
LOGDIR=/$DB2INST/db/log
BACKUP=/$DB2INST/backup
db2 update dbm cfg using diaglevel 4
db2 force applications all
db2stop
db2start
db2 drop database $DBNAME
(rm -f $LOGDIR/* $LOGDIR/NODE*/*.LOG $BACKUP/*)
db2stop
db2start
db2 create database $DBNAME on /$DB2INST/db/\
temporary tablespace managed by system using (\
'/$DB2INST/db/temp/tmp00',\
'/$DB2INST/db/temp/tmp01',\
'/$DB2INST/db/temp/tmp02') extentsize 16 prefetchsize 64"

db2 "update db cfg for $DBNAME using NEWLOGPATH $LOGDIR"
db2 "update db cfg for $DBNAME using SORTHEAP 2048"
db2 "update db cfg for $DBNAME using logretain on"
db2 "update db cfg for $DBNAME using LOGFILSIZ 2000"
db2 "update db cfg for $DBNAME using LOGPRIMARY 100"
db2 "update db cfg for $DBNAME using LOGSECOND 150"
db2set DB2COMM=tcPIP
db2 "update dbm cfg using svcname $SVNAME"
# "tpcc1" needs to match up to the SVCENAME for the database
db2stop
db2start
```



```
exit $?
```

Backup the database

```
DB2INST=db2inst
DBNAME=tpcc1
BACKUP=/$DB2INST/backup
db2 backup database $DBNAME to $BACKUP
exit $?
```

Create table spaces and buffer pools

```
DB2INST=db2inst
DBNAME=tpcc1
db2 connect to tpcc1
db2 CREATE Bufferpool TPCC1_TS1 IMMEDIATE SIZE 65536 PAGESIZE 4 K
db2 CREATE REGULAR TABLESPACE TS1 PAGESIZE 4 K MANAGED BY DATABASE USING "(
FILE '/db2inst/db/mnt1/dfa1' 1900M ) EXTENTSIZE 16 PREFETCHSIZE 16 NO FILE SYSTEM
CACHING BUFFERPOOL TPCC1_TS1"
db2 CREATE Bufferpool TPCC1_TS2 IMMEDIATE SIZE 65536 PAGESIZE 4 K
db2 CREATE REGULAR TABLESPACE TS2 PAGESIZE 4 K MANAGED BY SYSTEM USING "(
'/db2inst/db/mnt2/dfb1') EXTENTSIZE 16 PREFETCHSIZE 16 NO FILE SYSTEM CACHING
BUFFERPOOL TPCC1_TS2"
db2 CREATE Bufferpool TPCC1_TS3 IMMEDIATE SIZE 65536 PAGESIZE 4 K
db2 CREATE REGULAR TABLESPACE TS3 PAGESIZE 4 K MANAGED BY DATABASE USING "(
FILE '/db2inst/db/mnt3/dfc1' 1900M ) EXTENTSIZE 16 PREFETCHSIZE 16 FILE SYSTEM
CACHING BUFFERPOOL TPCC1_TS3"
db2 CREATE Bufferpool TPCC1_TS4 IMMEDIATE SIZE 65536 PAGESIZE 4 K
db2 CREATE REGULAR TABLESPACE TS4 PAGESIZE 4 K MANAGED BY SYSTEM USING "(
'/db2inst/db/mnt4/dfd1') EXTENTSIZE 16 PREFETCHSIZE 16 FILE SYSTEM CACHING
BUFFERPOOL TPCC1_TS4"
db2 CONNECT RESET
db2 force application all
db2stop
db2start
exit $?
```

## Setting up the Oracle Database Application

-->

### Create Groups

```
mkgroup -A id=410 oracle
mkgroup -A id=411 dba
```

### Create Users

```
ORAINST=oracle
mkdir /$ORAINST
mkuser id=320 pgrp=dba groups=db2adm home=/$ORAINST/orahome oracle
passwd oracle
```

su to another user and then to the user to confirm the password



```
su guest
su oracle
```

### Configure VxVM Disks

```
vxddladm enablescsi3 policy=aa
DG=oracle
  vxdg init $DG <disk-name>_0 cds=off
  vxassist -g $DG make base 15g &
```

V=V

```
mkfs -$V vxfs /dev/vx/rdisk/$DG/base
mkdir -p /$DG
mount -$V vxfs /dev/vx/dsk/$DG/base /$ORAINST
mkdir /$ORAINST/orahome
touch /$ORAINST/orahome/.profile
echo "+ +" >/$ORAINST/orahome/.rhosts
chown -R oracle:dba /$ORAINST
```

```
DG1=oradata
mkdir -p /$DG1
  vxdg init $DG1 SAN_VC0_1 cds=off
```

#for simple volumes

```
vxassist -g $DG1 make mnt1 4g &
vxassist -g $DG1 make mnt2 4g &
vxassist -g $DG1 make mnt3 4g &
vxassist -g $DG1 make mnt4 6g &
vxassist -g $DG1 make log 20g &
vxassist -g $DG1 make dbtemp 4g &
vxassist -g $DG1 make backup 24g &
```

```
wait
mkfs -$V vxfs /dev/vx/rdisk/$DG1/mnt1
mkfs -$V vxfs /dev/vx/rdisk/$DG1/mnt2
mkfs -$V vxfs /dev/vx/rdisk/$DG1/mnt3
mkfs -$V vxfs /dev/vx/rdisk/$DG1/mnt4
mkfs -$V vxfs /dev/vx/rdisk/$DG1/log
mkfs -$V vxfs /dev/vx/rdisk/$DG1/temp
mkfs -$V vxfs /dev/vx/rdisk/$DG1/backup
```

```
mount -$V vxfs /dev/vx/dsk/$DG1/mnt1 /$DG1
```

```
mkdir -p /$DG1/tpcc
mkdir -p /$DG1/tpcc/mnt2
mkdir -p /$DG1/tpcc/mnt3
mkdir -p /$DG1/tpcc/mnt4
mkdir -p /$DG1/tpcc/log
```

```

mkdir -p /$DG1/tpcc/dbtemp
mkdir -p /$DG1/backup

mount -$V vxfs /dev/vx/dsk/$DG1/mnt2 /$DG1/tpcc/mnt2
mount -$V vxfs /dev/vx/dsk/$DG1/mnt3 /$DG1/tpcc/mnt3
mount -$V vxfs /dev/vx/dsk/$DG1/mnt4 /$DG1/tpcc/mnt4
mount -$V vxfs /dev/vx/dsk/$DG1/log /$DG1/tpcc/log
mount -$V vxfs /dev/vx/dsk/$DG1/dbtemp /$DG1/tpcc/temp
mount -$V vxfs /dev/vx/dsk/$DG1/backup /$DG1/backup

chown -R oracle:dba /$DG1

```

## Database installation

On each system install the Oracle 10gR2. Mount the Oracle Software disk OR have access to the Oracle software store. Install Oracle as oracle user. Edit the .profile file and set the following environment variables. Follow instructions in the installer gui and complete installation

```

#su - oracle
export ORACLE_BASE=/oracle
export ORACLE_HOME=$ORACLE_BASE/orahome
export ORACLE_SID=tpcc
export LIBPATH=/usr/dt/lib:$ORACLE_HOME/lib32:$ORACLE_HOME/lib
export NLSPATH=/usr/lib/nls/msg/%L/%N:/usr/lib/nls/msg/%L/%N.cat
export
PATH=$PATH:/oradata/tpcc/scripts:$ORACLE_BASE:$ORACLE_HOME:$ORACLE_HOME/bin
export DISPLAY=xhost:XX.0

$ ./profile
$ cd <oracle software disk path>/Disk1
./Disk1/runInstaller -ignoreSysPrereqs

```

## Database creation

Copy the scripts listed below to a directory under /oracle. Login as oracle and run the script as shown. Ensure that the oracle users env has the ORACLE\_SID variable set .

```

$./build.sh sys manager /oracle/scripts

```

```

build.sh

```

---

```

#!/usr/bin/ksh
Usage ()
{
print -u2 "
    $0 syspassword systempassword scriptpath
    Example : $0 sys manager /tmp/scripts
"
exit 2
}
if [ $# -ne 3 ]; then
    Usage
fi

```



```
export DBNAME=$ORACLE_SID
export sysPassword=$1
export systemPassword=$2
export SCRPT_PATH=$3

if [ -z $DBNAME ]
then
    print -u2 " Oracle SID (ORACLE_SID) is not set. Set it in the oracle users .profile and execute it.
Rerun command again.
"
    exit 2
fi

if [ ! -x $SCRPT_PATH/build.sh ]
then
    print -u2 " Script directory path : $SCRPT_PATH is not valid.
Rerun command with correct path.
"
    exit 2
fi

mkdir -p /oradata/$DBNAME
mkdir -p /oradata/$DBNAME/bdump
mkdir -p /oradata/$DBNAME/cdump
mkdir -p /oradata/$DBNAME/create
mkdir -p /oradata/$DBNAME/hdump
mkdir -p /oradata/$DBNAME/pfile
mkdir -p /oradata/$DBNAME/udump
mkdir -p /oradata/backup
echo Add this entry in the oratab: $DBNAME:/oracle/orahome:Y
$ORACLE_HOME/bin/sqlplus /nolog @$SCRPT_PATH/build.sql $DBNAME $sysPassword
$systemPassword $SCRPT_PATH
```

---

build.sql

```
set verify off
DEFINE DBNAME = &1
DEFINE sysPassword = &2
DEFINE systemPassword = &2
DEFINE SCRPT_PATH = &4

host /oracle/orahome/bin/orapwd file=/oracle/orahome/dbs/orapw.&&DBNAME
password=&&sysPassword force=y
@&&SCRPT_PATH/CreateDB.sql
@&&SCRPT_PATH/CreateDBFiles.sql
@&&SCRPT_PATH/CreateDBCatalog.sql
host echo "SPFILE='/${SCRPT_PATH}/spfile${DBNAME}.ora" >
/oracle/orahome/dbs/init${DBNAME}.ora
@&&SCRPT_PATH/postDBCcreation.sql
```

---

CreateDB.sql



```
connect SYS/&&sysPassword as SYSDBA
set echo on
spool /oracle/orahome/assistants/dbca/logs/CreateDB.log
shutdown immediate ;
startup nomount pfile="&& SCRPT_PATH /init.ora";
DEFINE DBNAME = &1
CREATE DATABASE &&DBNAME
CONTROLFILE REUSE
MAXINSTANCES 32
MAXLOGHISTORY 1
MAXLOGFILES 192
MAXLOGMEMBERS 3
MAXDATAFILES 1024
DATAFILE '/oradata/&&DBNAME/system01.dbf' SIZE 300M REUSE AUTOEXTEND ON NEXT
10240K MAXSIZE UNLIMITED EXTENT MANAGEMENT LOCAL
SYSAUX DATAFILE '/oradata/&&DBNAME/sysaux01.dbf' SIZE 120M REUSE AUTOEXTEND ON
NEXT 10240K MAXSIZE UNLIMITED
DEFAULT TEMPORARY TABLESPACE TEMP TEMPFILE
'/oradata/&&DBNAME/temp/temp01.dbf' SIZE 20M REUSE AUTOEXTEND ON NEXT 640K
MAXSIZE UNLIMITED
UNDO TABLESPACE "UNDOTBS1" DATAFILE '/oradata/&&DBNAME/undotbs01.dbf' SIZE 200M
REUSE AUTOEXTEND ON NEXT 5120K MAXSIZE UNLIMITED
CHARACTER SET WE8ISO8859P1
NATIONAL CHARACTER SET AL16UTF16
LOGFILE GROUP 1 ('/oradata/&&DBNAME/log/redo01.log') SIZE 10240K REUSE,
GROUP 2 ('/oradata/&&DBNAME/log/redo02.log') SIZE 10240K REUSE
;
spool off
```

---

#### CreateDBFiles.sql

```
connect SYS/&&sysPassword as SYSDBA
set echo on
spool /oracle/orahome/assistants/dbca/logs/CreateDBFiles.log
CREATE TABLESPACE "USERS1" LOGGING DATAFILE
'/oradata/&&DBNAME/mnt2/users01.dbf' SIZE 5M REUSE AUTOEXTEND ON NEXT 1280K
MAXSIZE UNLIMITED EXTENT MANAGEMENT LOCAL SEGMENT SPACE MANAGEMENT
AUTO ;
CREATE TABLESPACE "USERS2" LOGGING DATAFILE
'/oradata/&&DBNAME/mnt3/users02.dbf' SIZE 5M REUSE AUTOEXTEND ON NEXT 1280K
MAXSIZE UNLIMITED EXTENT MANAGEMENT LOCAL SEGMENT SPACE MANAGEMENT
AUTO ;
CREATE TABLESPACE "USERS3" LOGGING DATAFILE
'/oradata/&&DBNAME/mnt4/users03.dbf' SIZE 5M REUSE AUTOEXTEND ON NEXT 1280K
MAXSIZE UNLIMITED EXTENT MANAGEMENT LOCAL SEGMENT SPACE MANAGEMENT
AUTO ;
spool off
```

---

#### CreateDBCatalog.sql

```
connect SYS/&&sysPassword as SYSDBA
set echo on
```



```
spool /oracle/orahome/assistants/dbca/logs/CreateDBCatalog.log
@/oracle/orahome/rdbms/admin/catalog.sql;
@/oracle/orahome/rdbms/admin/catblock.sql;
@/oracle/orahome/rdbms/admin/catproc.sql;
@/oracle/orahome/rdbms/admin/catoctk.sql;
@/oracle/orahome/rdbms/admin/owminst.plb;
connect SYSTEM/manager
@/oracle/orahome/sqlplus/admin/pupbld.sql;
connect SYSTEM/manager
set echo on
spool /oracle/orahome/assistants/dbca/logs/sqlPlusHelp.log
@/oracle/orahome/sqlplus/admin/help/hlpbld.sql helpus.sql;
spool off
```

---

#### postDBCcreation.sql

```
connect SYS/ &&sysPassword as SYSDBA
set echo on
spool /oracle/orahome/assistants/dbca/logs/postDBCcreation.log
create spfile='/oradata/tpcc/pfile/spfiletpcc.ora' FROM pfile='&&SCRPT_PATH/init.ora';
shutdown immediate;
host cp $SCRPT_PATH/init.ora /oracle/orahome/dbs/inittpcc.ora
connect SYS/ &&sysPassword as SYSDBA
startup ;
select 'utl_recomp_begin: ' || to_char(sysdate, 'HH:MI:SS') from dual;
execute utl_recomp.recomp_serial();
select 'utl_recomp_end: ' || to_char(sysdate, 'HH:MI:SS') from dual;
alter user scott account unlock ;
alter user scott identified by tiger replace tiger ;
create user tpcc identified by tpcc;
grant dba, connect, resource to tpcc;
disconnect
connect tpcc/tpcc
create table tpcc.test ( tstamp date );
insert into tpcc.test (tstamp) values (SYSDATE);
disconnect
connect tpcc/tpcc
update tpcc.test set ( tstamp ) = SYSDATE;
select TO_CHAR(tstamp, 'MON DD, YYYY HH:MI:SS AM') from tpcc.test;
exit;
```

---

#### Listener.ora Files

Login as oracle and execute netca to configure the listener

The listener files used in this configuration is listed below.

```
# listener.ora Network Configuration File: /oracle/orahome/network/admin/listener.ora
# Generated by Oracle configuration tools.
```

```
LISTENER_clustmmPNd1=
(DESCRIPTION_LIST =
(DESCRIPTION =
(ADDRESS = (PROTOCOL = TCP)(HOST = clustmmPNd1.pdx.symantec.com)(PORT = 1521))
(ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC0))
)
)
```



```
)  
  
LISTENER_clustmmPNd2=  
  (DESCRIPTION_LIST =  
    (DESCRIPTION =  
      (ADDRESS = (PROTOCOL = TCP)(HOST = clustmmPNd2.pdx.symantec.com)(PORT = 1521))  
      (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC0))  
    )  
  )  
)  
  
LISTENER_clustmmSNd1=  
  (DESCRIPTION_LIST =  
    (DESCRIPTION =  
      (ADDRESS = (PROTOCOL = TCP)(HOST = clustmmSNd1.pdx.symantec.com)(PORT = 1521))  
      (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC0))  
    )  
  )  
)  
  
LISTENER_clustmmSNd2=  
  (DESCRIPTION_LIST =  
    (DESCRIPTION =  
      (ADDRESS = (PROTOCOL = TCP)(HOST = clustmmSNd2.pdx.symantec.com)(PORT = 1521))  
      (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC0))  
    )  
  )  
)  
  
SID_LIST_TPCC =  
  (SID_LIST =  
    (SID_DESC =  
      (SID_NAME = PLSExtProc)  
      (ORACLE_HOME = /oracle/orahome)  
      (PROGRAM = extproc)  
    )  
  )  
)
```





## Appendix C: Veritas Software filesets listing

List of Veritas Storage Foundation and VCS Agent Packages installed with the steps followed in the procedure mentioned in this whitepaper.

Package	Contents
VRTSaa	Veritas Enterprise Administrator Action Agent by Symantec
VRTSacclib.rte	Veritas Cluster Server ACC Library 5.0 by Symantec
VRTSalloc	Veritas Storage Foundation Intelligent Storage Provisioning by Symantec
VRTSat	Authentication client Fileset
VRTSat	Authentication server Fileset
VRTSccg	Veritas Enterprise Administrator Central Control Grid by Symantec
VRTScmccc.rte	Veritas Cluster Management Console Cluster Connector 5.0MP1 by Symantec
VRTScmcs.rte	Veritas Cluster Management Console for single cluster environments 5.0MP1 by Symantec
VRTScscm.rte	Veritas Cluster Manager - Java Console 5.0MP1 by Symantec
VRTScscw.rte	Veritas Cluster Server Configuration Wizard by Symantec
VRTScssim.rte	Veritas Cluster Server 5.0MP1 Simulator by Symantec
VRTScutil.rte	Symantec Veritas Cluster Utility
VRTSdcli	Veritas Distributed Command Line by Symantec
VRTSddlpr	Veritas Device Discovery Services Provider Library by Symantec
VRTSdsa	Veritas Datacenter Storage Agent by Symantec
VRTSfsdoc	Veritas File System Documentation by Symantec
VRTSfsman	Veritas File System Manual Pages by Symantec
VRTSfsmnd	Veritas File System SDK Manual Pages by Symantec
VRTSfspro	Veritas File System Services Provider by Symantec 5.0MP1t
VRTSfssdk	Veritas Libraries and Header Files for VxFS by Symantec
VRTSgab.rte	Veritas Group Membership and Atomic Broadcast 5.0MP1 by Symantec 11/29/2006-23
VRTSgapms	Veritas Generic Array Plugin
VRTSicsco	Symantec Infrastructure Core Services Common Fileset
VRTSjre15.rte	Symantec JRE Redistribution
VRTSilt.rte	Veritas Low Latency Transport 5.0MP1 by Symantec 11/29/2006-23
VRTSmapro	Veritas Storage Foundation Mapping Provider from Symantec, PSTAMP=05.22.2006.17.57, BUILD=GA01
VRTSmh	Veritas Enterprise Administrator Managed Host by Symantec
VRTSob	Veritas Enterprise Administrator Service
VRTSobc33	Veritas Enterprise Administrator Service
VRTSobgui	Veritas Enterprise Administrator Service
VRTSpbx	Symantec Private Branch Exchange Fileset
VRTSperl.rte	Perl 5.8.8 for Veritas
VRTSsmf	Symantec Service Management Framework Fileset
VRTSspt	Veritas Support Tools by Symantec



VRTSvail	Veritas Array Providers
VRTSvcs.doc	User Documentation for Veritas Cluster Server by Symantec
VRTSvcs.man	Manual Pages for Veritas Cluster Server by Symantec
VRTSvcs.msg.en_US	Veritas Cluster Server English Message Catalogs by Symantec
VRTSvcs.rte	Veritas Cluster Server 5.0MP1 by Symantec 11/29/2006-23
VRTSvcsag.rte	Veritas Cluster Server 5.0MP1 Bundled Agents by Symantec
VRTSvcsdb.rte	Veritas High Availability 5.0MP1 Agent for DB2 by Symantec
VRTSvcsi.rte	VERITAS Clustering Support for IBM Metro Mirror(tm)
VRTSvcsor.rte	Veritas High Availability 5.0MP1 Agent for Oracle by Symantec.
VRTSvcsvr	Veritas Cluster Server Volume Replicator Agents by Symantec
VRTSvdid	Veritas Device Identifier (VDID)
VRTSveki	Veritas Kernel Interface by Symantec
VRTSvlic	VRTSvlic Symantec License Utilities
VRTSvm.doc	User Documentation for Veritas Volume Manager by Symantec
VRTSvm.man	manual pages for Veritas Volume Manager by Symantec
VRTSvm.pro	Veritas Volume Manager Servs Provider by Symantec
VRTSvr.doc	User Documentation for Veritas Volume Replicator by Symantec
VRTSvr.pro	VERITAS Volume Replicator Management Services Provider
VRTSvrw	Veritas Volume Replicator Web Console by Symantec
VRTSvsvc	Veritas Volume Server and Client Providers by Symantec
VRTSvxfen.rte	Veritas I/O Fencing 5.0MP1 by Symantec 11/29/2006-23
VRTSvxf.s	Veritas File System by Symantec
VRTSvxmsa	VERITAS - VxMS Mapping Service, Application Libraries
VRTSvxvm	Veritas Volume Manager by Symantec
VRTSweb.rte	Symantec Web Server



## Trademarks and special notices

---

© International Business Machines 1994-2008. IBM, the IBM logo, System Storage, and other referenced IBM products and services are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. All rights reserved.

References in this document to IBM products or services do not imply that IBM intends to make them available in every country.

Storage Foundation Enterprise HA, Symantec, the Symantec logo, Veritas, and Veritas Storage Foundation are trademarks or registered trademarks of Symantec Corporation or its affiliates in the U.S. and other countries.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

Information is provided "AS IS" without warranty of any kind.

Information concerning non-IBM products was obtained from a supplier of these products, published announcement material, or other publicly available sources and does not constitute an endorsement of such products by IBM. Sources for non-IBM list prices and performance numbers are taken from publicly available information, including vendor announcements and vendor worldwide homepages. IBM has not tested these products and cannot confirm the accuracy of performance, capability, or any other claims related to non-IBM products. Questions on the capability of non-IBM products should be addressed to the supplier of those products.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.