Veritas™ Cluster Server Agent for EMC MirrorView Configuration Guide

Windows Server 2008 (x64), Windows Server 2008 R2 (x64)

6.0

October 2011

Symantec™

Veritas Cluster Server Agent for EMC MirrorView Configuration Guide

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

Agent version: 6.0.0

Document version: 6.0.0.0

Legal Notice

Symantec Corporation
350 Ellis Street
Mountain View, CA 94043

http://www.symantec.com

Technical Support

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

Symantec's support offerings include the following:

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

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

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

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

Contacting Technical Support

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

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

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

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

- Product release level

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

Licensing and registration

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

www.symantec.com/business/support/

Customer service

Customer service information is available at the following URL:

www.symantec.com/business/support/

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

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

Support agreement resources

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

Asia-Pacific and Japan	customercare_apac@symantec.com
Europe, Middle-East, and Africa	semea@symantec.com
North America and Latin America	supportsolutions@symantec.com

Documentation

Product guides are available on the media in PDF format. Make sure that you are using the current version of the documentation. The document version appears on page 2 of each guide. The latest product documentation is available on the Symantec Web site.

https://sort.symantec.com/documents

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

doc_feedback@symantec.com

About Symantec Connect

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

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

Contents

Chapter 1

Introducing the Veritas agent for EMC MirrorView

This chapter includes the following topics:

About the agent for EMC MirrorView

The Veritas agent for EMC MirrorView provides support for application failover and recovery. The agent provides this support in environments that use MirrorView to replicate data between CLARiiON arrays.

The agent monitors and manages the state of replicated CLARiiON arrays that are attached to VCS nodes. The agent ensures that the system that has the MirrorView resource online also has safe and exclusive access to the configured arrays.

You can use the agent in replicated data clusters and in global clusters that run VCS.

The agent supports configuring EMC MirrorView in the synchronous or asynchronous modes. In asynchronous mode, you can replicate either individual LUNs or replicate consistency groups. MirrorView can also replicate LUNs or metaLUNs. In synchronous mode, you can replicate individual LUNs but cannot replicate consistency groups.

Supported hardware for EMC MirrorView

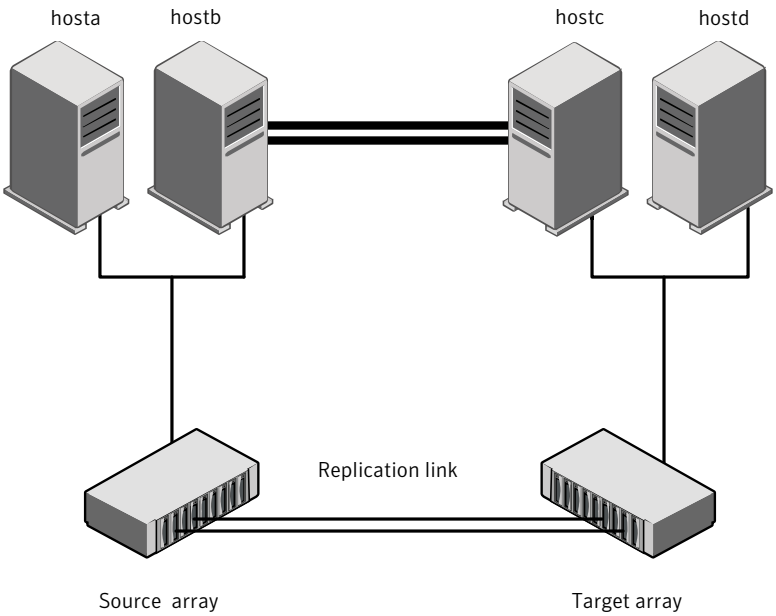
The Veritas agent for EMC Mirrorview supports both Navisphere Secure CLI, naviseccli as well as the Java based CLI, navcli.jar.

To determine the supported versions of NaviCLI and FLARE code that are on CLARiiON arrays, consult the EMC hardware compatibility list.

Typical EMC MirrorView setup in a VCS cluster

Figure 1-1 displays a typical cluster setup in a MirrorView environment.

Figure 1-1 Typical clustering setup for the agent



Clustering in a MirrorView environment typically consists of the following hardware infrastructure:

- The source array (array1) has one or more hosts. The hosts have a direct connection to a CLARiiON array. The array contains the mirror that is the primary image and the direct connection uses either SCSI or Fibre Channel.
- The target array (array2) consists of one or more hosts. These hosts have a direct connection to another CLARiiON array. The array contains the mirror

that is the secondary image and the connection uses either SCSI or Fibre Channel.

The secondary image LUNs pairs with the mirrored LUNs in the source array. The target hosts and the array must be at a significant distance from the source side to survive a source-side disaster.

- Network heartbeating between the two data centers to determine their health; this network heartbeating could be LLT or TCP/IP.
  See "About cluster heartbeats" on page 19.
- In a replicated data cluster environment, all hosts are part of the same cluster. You must connect them with the dual and dedicated networks that support LLT.

  In a global cluster environment, you must attach all hosts in a cluster to the same CLARiiON array.

EMC MirrorView agent functions

The VCS agent for EMC MirrorView monitors and manages the state of replicated CLARiiON LUNs attached to VCS nodes. Agent functions bring resources online, take them offline, and perform different monitoring actions.

The agent performs the following functions:

online	Creates a lock file on the local host. This lock indicates that the resource is online and makes the mirrors available for the application to use. The agent performs specific actions depending on the state of the mirrors. See "About the MirrorView agent's online function" on page 12.
offline	Removes the lock file on the local host.
monitor	Verifies that the lock file exists. If the lock file exists, the monitor entry point reports the status of the resource as online. If the lock file does not exist, the monitor entry point reports the status of the resource as offline.
open	Removes the lock file from the host where the function is called. This action prevents potential concurrency violation if the service group fails over to another node. Note that the agent does not remove the lock file if the agent was started after a `hastop -force` command.
clean	Removes the lock file.

info	The info function gives the information about the mirrors (in case of synchronous mode of replication). It also gives information about the mirrors/groups in case of asynchronous mode of replication. It uses the `-sync listsyncprogress` and `-async -list` or `-async listgroups` commands to get this information.
action	■ resync Performs a resynchronization action. ■ addArrayUser Creates security for the SYSTEM user on the local and remote arrays.
addArrayUser	Action entry point to create user security file for the `naviseccli` command.

About the MirrorView agent's online function

The agent's online operation performs specific actions depending on the state of the mirrors.

Synchronous state

If the state of all local mirrors is MIRRORED, the agent creates a lock file on the local host. This lock indicates that the resource is online and makes the mirrors available for the application to use.

If one or more mirrors are not in the MIRRORED state, the agent runs a NaviCLI command. With this command, the agent brings them into the MIRRORED state, which enables the application to use them.

- For secondary images in the synchronized state, the agent runs the `mirror -sync -promoteimage` command to promote the remote mirror. This command also converts the current primary to secondary.
- For secondary images in the CONSISTENT state, the agent waits to check if the image has transitioned to the SYNCHRONIZED state.
- If the images have transitioned to the SYNCHRONIZED state, the agent then runs the `mirror -sync -promoteimage` command to promote the remote mirror. This command also converts the current primary to secondary.
- If the image has not transitioned to the SYNCHRONIZED state, the agent checks if the remote array is accessible. If the remote array is accessible, then this condition indicates link failure—the image would be in a fractured condition.

In case of fracture:

- If the SplitTakeover attribute is set to 1, the agent forcibly promotes the secondary image.
- If the SplitTakeover attribute is set to 0, the agent does not try to promote the secondary image forcibly, and becomes the administrator's decision.

Asynchronous state

You can configure the online function for either consistency groups or mirrors in asynchronous mode.

Consistency groups

If the state of the group is SYNCHRONIZED, the agent creates a lock file on the local host to indicate that the resource is online. This lock makes the LUNs available for the application to use.

If one or more mirrors are not in the MIRRORED state, the agent checks to see if the remote array is accessible.

- If the remote array is not accessible, then the agent checks the value of the attribute SplitTakeover before proceeding with any further actions.
- If the SplitTakeover attribute is set to 1, the agent forcibly promotes the secondary image.
- If the SplitTakeover attribute is set to 0, the agent does not try to promote the secondary image forcibly, and becomes the administrator's decision.
- If the remote array is accessible, then the agent runs the `mirror -async -promotegroup` command to promote the remote group.
- In case of a successful promotegroup operation, the operation also converts the current primary to secondary.
- If the promotegroup operation is not successful, then the agent initiates a synchronization.
  The agent periodically checks if the group is SYNCHRONIZED. After a successful synchronization, the agent promotes the group using the `mirror -async -promotegroup` command. If the synchronization is not successful, the agent times out.

Mirrors

If the state of all local mirrors is MIRRORED, the agent creates a lock file on the local host. The lock indicates that the resource is online and makes the mirrors available for the application to use.

If one or more mirrors are not in the MIRRORED state, the agent checks to see if the remote array is accessible.

- If the remote array is not accessible, then the agent checks the value of the attribute SplitTakeover before proceeding with any further actions.
- If the SplitTakeover attribute is set to 1, the agent forcibly promotes the secondary image.
- If the SplitTakeover attribute is set to 0, the agent does not try to promote the secondary image forcibly, and becomes the administrator's decision.
- If the remote array is accessible, then the agent runs the `mirror -async -promoteimage` command to promote the remote mirrors.
- A successful promoteimage operation converts the current primary to secondary.
  If the promoteimage operation is not successful, then the agent initiates a synchronization.
  The agent periodically checks if the group is SYNCHRONIZED. After a successful synchronization, the agent promotes the secondary mirror using the `mirror -async -promoteimage` command. If the synchronization is not successful, the agent times out.

Chapter 2

Configuring the agent for EMC MirrorView

This chapter includes the following topics:

Configuration concepts for the EMC MirrorView agent

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

Resource type definition for the MirrorView agent

The resource type definition defines the agent in VCS.

```
type MirrorView (
    static keylist SupportedActions = { resync }
    static int MonitorInterval = 300
    static int NumThreads = 1
    static int OfflineMonitorInterval = 0
    static int RestartLimit = 1
    static i18nstr ArgList[] = { NaviCliHome, LocalArraySPNames,
    RemoteArraySPNames, Mode, GrpName, MirNames, SplitTakeover }
    str NaviCliHome = "C:\\Program Files\\EMC\\Navisphere CLI"
    str LocalArraySPNames[]
    str RemoteArraySPNames[]
    str Mode
    str GrpName
```

```
str MirNames[]
int SplitTakeover
temp str VCSResLock
)
```

Attribute definitions for the MirrorView agent

Review the description of the agent attributes.

Required attributes

You must assign values to the following required attributes:

NaviCliHome	NaviCLI installation directory `"/opt/Navisphere/bin"` Type and dimension: string-scalar
LocalArraySPNames	The list of storage processors within the array to which the local hosts are connected. Can be names or IP addresses. Type and dimension: string-vector
RemoteArraySPNames	The list of storage processors within the array to which the remote hosts are connected. Can be names or IP addresses. Type and dimension: string-vector
Mode	The replication mode, which is either: `sync` or `async`. Type and dimension: string-scalar
GrpName	The name of the consistency group to which the mirrors belong. This function applies with only one mode: `async`or `sync`. Type and dimension: string-scalar
MirNames	This function specifies the mirrors with only one replication mode: `sync` or `async`. Type and dimension: string-vector

SplitTakeover	This integer indicates whether VCS should forcefully promote a secondary to a primary. In case of a link-failure between the two arrays, the state of the mirror remains consistent or out-of-sync. Under such circumstances, if the application has to failover—due to disaster or user-driven action—mirrors are not in a SYNCHRONIZED state. If the value of the SplitTakeOver attribute is 1: ■ The agent fails over when it discovers link failures ■ The agent determines that mirrors are out of sync If the value of the attribute is 0, agent does not fail over and the administrator must to determine what to do. Type and dimension: integer-scalar

Internal attribute

Do not modify internal attributes. The MirrorView agent currently supports the following internal attribute:

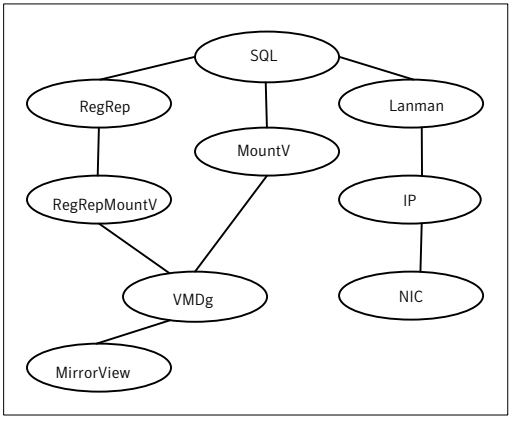
VCSResLock	This agent uses this attribute to guarantee serialized management in case of a parallel application. Do not modify this value. Type and dimension: temporary-string

Sample configuration for the MirrorView agent

Figure 2-1 shows a VCS service group that has a resource of type MirrorView.

The DiskGroup resource depends on the MirrorView resource.

Figure 2-1 Dependency tree



You can configure a resource of type MirrorView in the `main.cf` file.

In this example, the resource is configured for asynchronous mode and consistency groups.

```
MirrorView mir (
    NaviCliHome = "C:\Program Files\EMC\Navisphere CLI"
    LocalArraySPNames @sys1= = { "Local_SP1_Name", "Local_SP1_IP" }
    LocalArraySPNames @sys2 = { "Local_SP2_Name", "Local_SP2_IP" }
    RemoteArraySPNames @sys1 = { "Local_SP2_IP", "Remote_SP2_Name" }
    RemoteArraySPNames @sys2= { "Local_SP1_IP", "Remote_SP1_Name" }

    Mode = async
    GrpName = consistency_grp1

    SplitTakeover = 0
    )

MirrorView mir (
    NaviCliHome = "/opt/Navisphere/bin"
    LocalArraySPNames = { "Local_SP1_IP", "Local_SP2_IP" }
    RemoteArraySPNames = { "Remote_SP1_IP", "Remote_SP2_IP" }
    Mode = sync
    GrpName = sync_consistency_grp1
    SplitTakeover = 0
    )
```

If you want to configure the resource in synchronous mode and specify the individual mirror names, configure the MirNames attribute, instead of the GrpNames attribute, as follows:

```
Mode = sync
MirNames = { "sync_mir1", "sync_mir2" }
GrpName = ""
```

If you want to configure the resource in asynchronous mode and specify the individual mirror names, configure the MirNames attribute, instead of the GrpNames attribute, as follows:

```
Mode = async
MirNames = { "async_mir1", "async_mir2" }
GrpName = ""
```

Before you configure the agent for EMC MirrorView

Before you configure the agent, review the following information:

- Review the configuration concepts, which describe the agent's type definition and attributes.
  See "Configuration concepts for the EMC MirrorView agent" on page 15.
- Verify that you have installed the agent on all systems in the cluster.
- Verify the hardware setup for the agent.
  See "Typical EMC MirrorView setup in a VCS cluster" on page 10.
- Make sure that the cluster has an effective heartbeat mechanism in place.
  See "About cluster heartbeats" on page 19.
  See "About preventing split-brain" on page 21.
- Set up system zones in replicated data clusters.
  See "About configuring system zones in replicated data clusters" on page 20.

About cluster heartbeats

In a replicated data cluster, ensure robust heartbeating by using 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, VCS 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.

Heartbeat loss may occur due to the failure of all hosts in the primary cluster. In such a scenario, a failover may be required even if the array is alive. In any case, a host-only crash and a complete site failure must be distinguished. In a host-only crash, only the ICMP heartbeat signals a failure by an SNMP trap. No cluster failure notification occurs because a surviving heartbeat exists. This trap is the only notification to fail over an application.

About configuring system zones in replicated data clusters

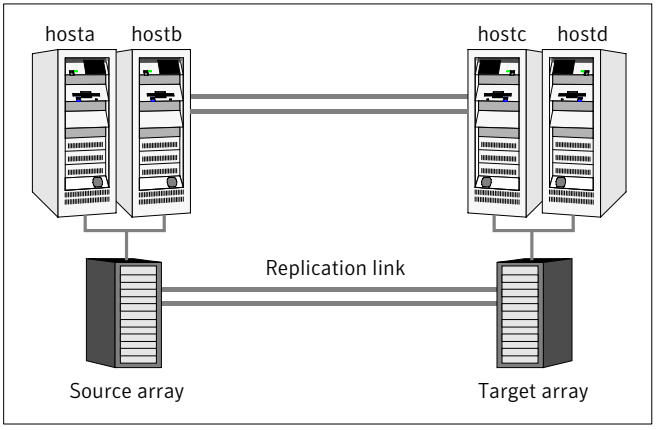
In a replicated data cluster, you can prevent unnecessary MirrorView failover or failback by creating system zones. VCS attempts to fail over applications within the same system zone before failing them over across system zones.

Configure the hosts that are attached to an array as part of the same system zone to avoid unnecessary failover.

Figure 2-2 depicts a sample configuration where hosta and hostb are in one system zone, and hostc and hostd are in another system zone.

Use the SystemZones attribute to create these zones.

Figure 2-2 Example system zone configuration



Modify the SystemZones attribute using the following command:

```
hagrp -modify grpname SystemZones hosta 0 hostb 0 hostc 1 hostd 1
```

The variable grpname represents the service group in the cluster.

Global clusters do not require system zones because failover occurs on a remote cluster if all local targets have been exhausted.

As long as a secondary image is available, MirrorView sends the writes to the secondary image immediately in synchronous mode. It does so periodically in asynchronous mode.

If the period is too long, you can perform synchronization using the resync action. The supported resync action is defined in the MirrorView resource type.

About preventing split-brain

Split-brain occurs when all heartbeat links between the primary and secondary hosts are cut. In this situation, each side mistakenly assumes that the other side is down. You can minimize the effects of split-brain by ensuring that the cluster heartbeat links pass through a similar physical infrastructure as the replication links. When you ensure that both pass through the same infrastructure, if one breaks, so does the other.

Sometimes you cannot place the heartbeats alongside the replication links. In this situation, a possibility exists that the cluster heartbeats are disabled, but the replication link is not. A failover transitions the original source to target and vice-versa. In this case, the application faults because its underlying volumes become write-disabled, causing the service group to fault. VCS tries to fail it over to another host, causing the same consequence in the reverse direction. This phenomenon continues until the group comes online on the final node. You can avoid this situation by setting up your infrastructure such that loss of heartbeat links also mean the loss of replication links.

Configuring the agent for EMC MirrorView

You can adapt most clustered applications to a disaster recovery environment by:

- Converting their LUNs to CLARiiON LUNs
- Synchronizing the mirrors
- Adding the EMC MirrorView agent to the service group

After configuration, the application service group must follow the dependency diagram.

See “Sample configuration for the MirrorView agent” on page 17.

Note: You must not change the replication state of devices primary to secondary and viceversa, outside of a VCS setup. The agent for EMC MirrorView fails to detect a change in the replication state if the role reversal is done externally.

Configuring the agent 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.
2. If the agent resource type (mirrorview) is not added to your configuration, add it. From the Cluster Explorer **File** menu, choose **Import Types** and select:

   systemdrive\Program Files\Veritas\cluster server\conf\Sample_mirrorview\mirrorviewTypes.cf

3. Click **Import**.
4. Save the configuration.
5. Add a resource of type mirrorview at the bottom of the service group.
6. Configure the attributes of the mirrorview resource.
7. If the service group is not configured as a global service group, configure the service group using the Global Group Configuration Wizard.

   See the *Veritas Cluster Server Administrator's Guide* for more information.
8. Change the ClusterFailOverPolicy attribute from the default, if necessary. Symantec recommends keeping the default, which is Manual, to minimize the chance of failing over on a split-brain.
9. Repeat step 5 through step 8 for each service group in each cluster that uses replicated data.
10. The configuration must be identical on all cluster nodes, both primary and disaster recovery.

Configuring the agent manually in a replicated data cluster

Configuring the agent manually in a replicated data cluster involves the following tasks:

To configure the agent in a replicated data cluster

1. Start Cluster Manager and log on to the cluster.
2. If the agent resource type (mirrorview) is not added to your configuration, add it. From the Cluster Explorer **File** menu, choose **Import Types** and select:

   systemdrive\Program Files\Veritas\Cluster Server\conf\config\mirrorviewTypes.cf.

3 Click **Import**.

4 Save the configuration.

5 In each service group that uses replicated data, add a resource of type mirrorview at the bottom of the service group.

6 Configure the attributes of the mirrorview resource. Note that some attributes must be localized to reflect values for the hosts that are attached to different arrays.

7 Set the SystemZones attribute for the service group to reflect which hosts are attached to the same array.

Executing the addArrayUser action

Executing the addArrayUser action is a critical step in configuring the EMC MirrorView agent in both global clusters and replicated data clusters.

The MirrorView agent runs under the system user context. You must execute the addArrayUser action to create the required security files, thus allowing you as a system user to run commands on both local and remote arrays.

To execute the addArrayUser action

1 Right click the mirrorview resource and select **Actions**.

2 From the list of actions, select **addArrayUser**.

3 Select the system on which to execute the action.

4 From the list of action arguments, click the plus button to add a new argument. Type in a non-blank password for the system user to use.

5 Click **OK** to execute the action.

6 Review the results of the action and click **OK**.

7 Repeat these steps for all nodes that need to access the arrays.

Chapter 3

Managing and testing clustering support for EMC MirrorView

This chapter includes the following topics:

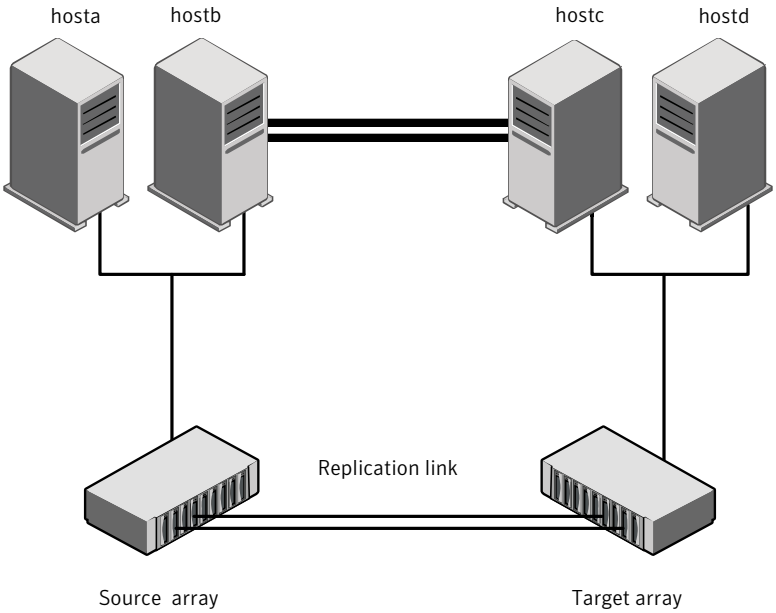
Typical test setup for the EMC MirrorView agent

A typical test environment includes the following characteristics:

- Two hosts (hosta and hostb) are attached to the source CLARiiONarray.
- Two hosts (hostc and hostd) are attached to the target CLARiiON array.
- The application runs on hosta and devices in the local array are read-write enabled in the SYNCHRONIZED state.
- A replicated data cluster has two dedicated heartbeat links.
  A global cluster has one network heartbeat.

Figure 3-1 depicts a typical test environment.

Figure 3-1 Typical test setup



Testing service group migration

Verify the service group can migrate to different hosts in the cluster and across clusters.

To perform the service group migration test

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

   Migrate the service group to a host that is attached to the same array.

2. Click **Switch To**, and click the system that is attached to the same array (hostb) from the menu.

   The service group comes online on hostb and local image remains in the MIRRORED state.

3. In the Service Groups tab of the Cluster Explorer configuration tree, right-click the service group.

   Migrate the service group to a host that is attached to a different array.

4 Click **Switch To**, and click the system that is attached to another array (hostc) from the menu.

The service group comes online on hostc and the role of the images there transition to primary.

Accumulate dirty tracks on the new source-side and update them back on the target:

```
hares -action mirrorview_res_name resync -sys hostc
```

The variable *mirrorview_res_name* represents the name of the MirrorView resource.

5 In the Service Groups tab of the Cluster Explorer configuration tree, right-click the service group.

After the devices transition to a source SYNCHRONIZED state, migrate the service group back to its original host.

6 Click **Switch To**, and click the system on which the group was initially online (hosta).

The group comes online on hosta. The devices return to the RW/SYNCINPROG state at the array that is attached to hosta and hostb, and then eventually transition to the SYNCHRONIZED state.

Testing host failure

In this scenario, the host where the application runs is lost. Eventually all the hosts in the system zone or cluster are lost.

To perform the host failure test

1 Halt or shut down the host where the application runs (hosta).

The service group fails over to hostb and devices are in the SYNCHRONIZING state.

2 Halt or shut down hostb.

In a replicated data cluster, the group fails over to hostc or hostd depending on the FailOverPolicy attribute in the cluster.

In a global cluster, a cluster down alert appears and gives you the opportunity to fail over the service group manually.

In both environments, the role of the devices changes from secondary to primary and starts on the target host.

3 Power on the two hosts that were shut down.

4 Switch the service group to its original host when VCS starts.

Do the following:

- In the **Service Groups** tab of the Cluster Explorer configuration tree, right-click the service group.
- Click **Switch To**, and click the system on which the service group was initially online (hosta).
  The service group comes online on hosta and devices transition to the SYNCHRONIZING state and then to the SYNCHRONIZED state.

Performing a disaster test

Test how robust your cluster is in case of a disaster.

To perform a disaster test

1 Shut down all hosts on the source side and shut down the source array.

If you cannot shut down the source array, change the value of the RemoteArraySPNames in the target side to non-existent names and IP addresses. This action mimics a disaster scenario from the target's point of view.

2 In a replicated data cluster, the service group fails over to hostc or hostd in the following conditions:

- All devices were originally in the SYNCHRONIZED state.
- No synchronization was in progress at the time of disaster.

3 In a global cluster, the administrator is notified of the failure. The administrator can then initiate the failover.

Performing the failback test

You can set up your cluster for a failback test.

The failback test verifies the application can fail back to its original host after a failover to a remote site.

To perform the failback test for asynchronous mode with Consistency groups

1 Remove all the mirrors form the consistency group on the old primary.

2 Destroy the consistency group on the old primary.

3 Forcefully destroy the remote mirrors on the old primary.

4 Remove the LUNs from the storage group on the old primary.

5 Remove the mirrors from the consistency group on the new primary.

6 Add secondary images to each of the remote mirrors on the new primary.

7 Add the mirrors into the consistency group on the new primary.

Between step 5 and step 7, the LUNs become vulnerable to data corruption. For example, if one of the LUNs has sustained hardware damage and failed.

During this window, the mirrors are not a part of the consistency group. The writes to other mirrors that were a part of the consistency group are not stopped. This situation could result in data corruption.

8 Add the LUNs, where the secondary image resides, into the appropriate storage group on the old primary.

To perform the failback test for synchronous and asynchronous mode with Individual mirrors

1 Forcefully destroy the remote mirrors on the old primary.

2 Remove the LUNs from the storage group on the old primary.

3 Add secondary images to each of the remote mirrors on the new primary.

4 Add the LUNs, where the secondary image resides, into the appropriate storage group on the old primary.

In either of the modes, the original contents of the old primary are lost.

Failure scenarios for EMC MirrorView

Review the failure scenarios and agent behavior in response to failure.

Site disaster

In a total site failure, all hosts and the array are completely disabled, either temporarily or permanently.

In a replicated data cluster, site failure is detected the same way as a total host failure, that is, the loss of all LLT heartbeats.

In a global cluster, VCS detects site failure by the loss of all configured heartbeats.

A total disaster renders the devices on the surviving array in the FRACTURED state. If the SplitTakeover attribute is set to its default value of 1, the online entry point runs the ‘promote’ operation. If the attribute is set to 0, no takeover occurs and the online entry point times out and faults.

The online entry point detects whether any synchronization was in progress when the source array was lost. Since the target devices are inconsistent until the synchronization completes, the agent does not write-enable the devices, but it times out and faults. You must restore consistent data from a snapshot or tape backup.

All host or all application failure

Even if both arrays are operational, the service group fails over in the following conditions:

- All hosts on the source CLARiiON side are disabled.
- The application cannot start successfully on any source host.

In replicated data cluster environments, the failover can be automatic, whereas in global cluster environments failover requires user confirmation by default.

In replicated data cluster environments, site failure is detected the same way as a total host failure, that is, the loss of all LLT heartbeats. This type of failure is communicated by the VCS engine to the other site.

In both replicated data cluster and global cluster environments, multiple service groups can fail over in parallel.

Replication link failure

Before the MirrorView takes any action, it waits for the synchronization to complete in the following situations:

- The two arrays are healthy and the link that failed is restored.
- A failover is initiated while synchronization is in progress.

After the synchronization completes, the MirrorView runs the promote operation.

If the agent times out before the synchronization completes, the resource faults.

If the SplitTakeover attribute is set to 0, the agent does not attempt a promote operation, but it times out and faults. If you write-enable the devices manually, the agent can come online after it is cleared.

Split-brain in a MirrorView environment

You must resynchronize the volumes manually using the `symrdf merge` or `symrdf restore` commands.

In a global cluster, you can confirm the failure before failing over the service groups. You can check with the site administrator to identify the cause of the

failure. If a fail over mistakenly occurs, the situation is similar to the replicated data cluster case. However, when the heartbeat is restored, VCS does not stop HAD at either site. VCS forces you to choose which group to take offline. You must resynchronize the data manually.

Index

T

V