# Volume Replicator 7.3.1 Administrator's Guide - Windows

VERITAS™

Last updated: 2017-11-05

## Legal Notice

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

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

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

http://www.veritas.com

## Technical Support

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

https://www.veritas.com/support

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

https://my.veritas.com

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

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

## Documentation

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

https://sort.veritas.com/documents

## Documentation feedback

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

doc.feedback@veritas.com

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

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

## Veritas Services and Operations Readiness Tools (SORT)

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

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

# Contents

Chapter 2

# Replication concepts

## Chapter 11 Setting up replication in cloud environments .......... 387

## Chapter 12 Troubleshooting Volume Replicator .......... 403

Chapter 1

# Understanding Volume Replicator

This chapter includes the following topics:

## About Volume Replicator

Volume Replicator is an extension of the logical volume management capability of Storage Foundation (SFW). It works as an integrated component of SFW and can use the existing SFW configurations. Any application, even with existing data, can be configured to use Volume Replicator transparently, in a SFW configuration. Volume Replicator benefits from the robustness, ease of use, and high performance of SFW, and at the same time, adds replication capability to SFW.

Volume Replicator replicates data from initially synchronized volumes at a source location, to one or more remote locations across any distance, including the Azure and AWS cloud environment. It provides a consistent and up-to-date copy of application data at the remote locations.

A major trend affecting businesses today is the reliance upon the data that is geographically distributed. When a disaster occurs, quick recovery and availability of data becomes the most important need. One way to maintain a consistent copy of the application data at a remote site is by using a replication service to replicate the data to a remote site. In case of a disaster, the remote site can be used to bring up the application and the user data without much delay.

Volume Replicator is a data replication service that helps you to maintain a consistent copy of the application data at a remote site. It is built to contribute to an effective disaster recovery plan. If the Primary data center is destroyed, the application data is immediately available at the remote site, and the application can be restarted at the remote site.

## Feature highlights of Volume Replicator

Volume Replicator supports volume level replication of application or file system data.

The features of Volume Replicator are as follows:

- Supports the replication of data over any IP network (IPv4 and IPv6), LAN, or WAN.
- Supports data replication in all of the following scenarios:
  - From an on-premise data center to an on-premise data center that is located at a same or different location
  - From an on-premise data center to an on-cloud data center
  - From an on-cloud data center to an on-cloud data center that is located in a same or different region
- Runs on all storage hardware that Storage Foundation supports.
- Supports the replication over Firewall.
- Provides a volume-level replication of application or file system data, including support of commercial database management systems. It also supports replication of raw volumes or volumes that are mounted on file systems.
- Performs the replication of volume groups in asynchronous or synchronous modes, ensuring complete data integrity and consistency in either mode.
- Maintains write-order fidelity so that the updates on the Secondary host are performed in the same order as that on the Primary host.
- Performs intelligent synchronization for the initial synchronization of NTFS and ReFS volumes using the SwiftSync feature.

- Provides an In-band Control (IBC) messaging facility that allows the sequencing of events between the local and remote sites.
- Enables the efficient usage of the available bandwidth by controlling the maximum network bandwidth that Volume Replicator can use for replication.
- Supports both the TCP transport protocol and the UDP transport protocol to exchange data messages.
- Enables taking over the Primary role with fast-failback if the Primary becomes unavailable due to a disaster or some other reason.
- Supports Bunker replication, which enables zero Recovery Point Objective (RPO) or best RPO for a required Recovery Time Objective (RTO).

## About Volume Replicator support for IPv6

Volume Replicator includes support for Internet Protocol Version 6 (IPv6) addresses. To configure replication, you can specify IPv6 addresses.

Note the following:

- You must set the IP preference, whether Volume Replicator should use IPv4 or IPv6 addresses, before you configure replication.

  When you specify host names while configuring replication, Volume Replicator resolves the host names with the IP addresses associated with them. This setting determines which IP protocol Volume Replicator uses to resolve the host names.

  Use Veritas Enterprise Administrator (VEA) (Control Panel > VVR Configuration > IP Settings tab) to set the IP preference.

  See "Changing the IPv6 preference through the Control Panel" on page 109.
- Using the Replicated Data Set (RDS) wizard, you can specify IPv6 addresses that are associated with the Primary and Secondary host names.
- Using the VVR Security Service Configuration Wizard, you can specify IPv6 addresses for hosts on which you want to configure the VxSAS service.
- Volume Replicator commands that use an IP address, either as an input parameter or as an output, now support IPv6 addresses.

  For example, the vxrds `changeip` command that is used to change the host name or IP address of the Primary or Secondary RLINKs now accepts IPv6 addresses as input.
- Volume Replicator does not support replication in cases where the Primary and Secondary systems in an RDS use different IP addresses. For example, if the Primary host uses an IPv4 address and the Secondary host uses an IPv6 address, this configuration is not supported.

In cases where the Primary host uses only an IPv4 address, and the Secondary host uses both IPv4 and IPv6 addresses, Volume Replicator automatically selects an IPv4 address for the Secondary.

- Volume Replicator does not support replication for a IPv6-only system. An IPv6-only system is a system that implements only IPv6. It only has an IPv6 address in the name service database.

# Basic Volume Replicator terms

It is helpful to know certain Volume Replicator specific terms to know and understand its functioning. The terms 'node' and 'host' have been used interchangeably throughout this document and mean the same.

A list of some of the common Volume Replicator terms are described in the following section:

- See “Primary and Secondary host” on page 19.
- See “Write-order fidelity” on page 19.
- See “Consistent data versus up-to-date data” on page 20.
- See “Heartbeat protocol” on page 20.

## Primary and Secondary host

Data is replicated from a source host to a remote target host. The source is referred to as the Primary and the target host is referred to as the Secondary. Any single host in the configuration can simultaneously perform the role of the Primary or Secondary, always replicating an exclusive set of volumes. This enables you to have very flexible replication configurations.

## Write-order fidelity

To use the Secondary in a disaster recovery scenario, write-order fidelity must be maintained. The term write-order fidelity means that Volume Replicator tracks writes on the Primary in the order in which they are received and applies them on the Secondary in the same order. It is important to maintain write-order fidelity to ensure that the data on the Secondary is consistent with the data on the Primary. While the data at the Secondary can be behind in time, it must be a consistent image of the Primary at a known point in the past.

Without write-order fidelity, there is no guarantee that a Secondary has consistent, recoverable data. Volume Replicator maintains write-order fidelity across all the data volumes that are covered under replication, regardless of the modes of replication that are used. For example, in a database environment, the log and data are typically on different volumes. On the Primary, Volume Replicator tracks the order of the writes that are made to the log and data volumes and maintains this order when applying the writes on the Secondary. If the write-order fidelity is not maintained, the database application may not recover successfully when failed over to the Secondary.

## Consistent data versus up-to-date data

Data is considered to be consistent if the system or application using it can be successfully restarted using this data. For example, if the data belongs to a file system, the data is consistent if the `chkdsk` command can be run successfully on it. If the data contains a database, the data is consistent if the database recovery program can be run on it and the database can be restarted.

The data on the Secondary is consistent if it correctly reflects the data on the Primary at some time in the past. Volume Replicator tries to maintain the data at the Secondary in a consistent state at all times.

Data is considered consistent only if it contains all the updates up to some point-in-time and none of the updates that come after that point. For example, in the case of a file system, the most recently created files may be missing when it is abruptly stopped. In the case of a database, one or more of the most recently committed transactions may be missing.

Data that is up-to-date contains all the latest changes. For example, if you replicate a database, all the committed transactions will be available on the Secondary host.

You can choose whether you want the data on the Secondary to always be up-to-date by using either the asynchronous or synchronous mode of replication.

See “Modes of replication” on page 28.

The synchronous mode of replication ensures that the data on the Secondary is always up-to-date. However, in the asynchronous mode Volume Replicator cannot guarantee that the data is up-to-date. Another mode of replication that Volume Replicator supports is synchronous override. In this mode, Volume Replicator replicates synchronously as long as the required network bandwidth is continuously available. However, if the network becomes unavailable, then Volume Replicator replicates asynchronously. Note that Volume Replicator maintains write-order fidelity irrespective of the mode of replication used.

## Heartbeat protocol

To ensure that the Secondary host can always detect communication loss regardless of update activity, the Primary host periodically sends a heartbeat message to the

Secondary. If the Secondary misses a fixed number of heartbeat messages, it detects a communication loss and tries to reconnect. The reconnecting process triggers the heartbeat protocol. Likewise, if the Primary is unable to send a heartbeat message or if its heartbeat messages go unacknowledged, the Primary also detects a communication loss and enters its recovery procedure. Heartbeat messages use the UDP protocol for communication.

On successful completion of the heartbeat protocol, update activity resumes automatically unless some interim administrative command or error prevents it.

# Building blocks of Volume Replicator

Volume Replicator uses replication objects to set up replication.

The following topics describe the replication objects:

- See "Replicated Volume Group" on page 21.
- See "Replicator Log volume" on page 22.
- See "Replication Link—RLINK" on page 23.
- See "Replicated Data Set" on page 23.
- See "Data Change Map " on page 24.

## Replicated Volume Group

The Volume Replicator replicates the data that may be present on one or more Storage Foundation (SFW) volumes. This set of volumes on a Volume Replicator-managed host is called a Replicated Volume Group (RVG).

An RVG is always associated with a SFW disk group. The disk group can consist of volumes. All related volumes must always be a part of the same RVG. Unrelated volumes must not be grouped together in an RVG. Multiple RVGs can be configured inside one disk group.

The RVG is the unit of replication. Set of volumes on a host that need to be replicated are grouped under an RVG and are referred to as the Primary RVG. The destination host to which the volume data needs to be replicated, also has a similar setup as the Primary RVG to maintain consistency. This volume group on the destination host is referred to as the Secondary RVG.

The updates to the volumes in an RVG on the Primary host are also sent to its Secondary hosts. Access to the data volumes on the Secondary hosts is not allowed when replication is active.

Volumes that are associated with an RVG and contain application data are called data volumes. Data volumes are replicated Storage Foundation volumes and are distinct from the Replicator Log volume. The data volumes in an RVG may be under the control of an application such as a Database Management System that expects write-order fidelity to be maintained for the updates to the volumes during replication to ensure that each remote volume is always consistent, both internally and with all other volumes of the RVG.

**Note:** Each RVG can have a maximum of 1023 data volumes.

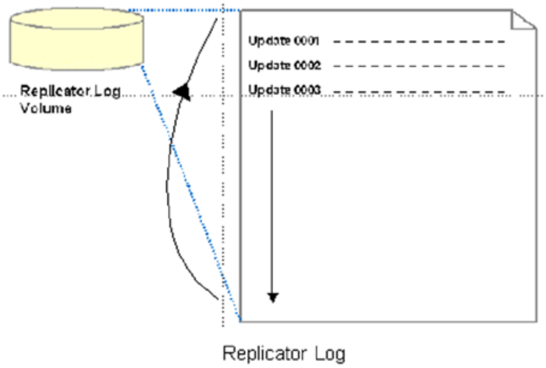
## Replicator Log volume

Volume Replicator uses one of the SFW volumes as a circular log to store updates, and is called the Replicator Log. All updates to the data volumes in the Primary RVG are logged in the Replicator Log volume on the Primary host, before they are sent to the Secondary. Each update to the Primary RVG generates two update write requests; one to the Replicator Log volume and one to a data volume. Each RVG has one Replicator Log volume. Because the Replicator Log plays such an important role in maintaining the consistency of the data between the hosts it is very important to plan the size and layout of the Replicator Log appropriately. The maximum size of the Replicator Log can be derived from various criteria, however, the size of the Replicator Log volume should not be less than 110 MB.

See “Sizing the Replicator Log” on page 43.

**Note:** The terms Replicator Log and Storage Replicator Log (SRL) mean the same. These terms have, therefore, been used interchangeably throughout the document.

**Figure 1-1** Replicator Log volume



The Secondary Replicator Log performs a different function from that of the Primary. Under normal operations, the Secondary Replicator Log volume is not used. It is used to maintain data consistency while Volume Replicator recovers from a temporary failure in communication between the Primary and Secondary, or from a Primary or Secondary host failure.

See “Managing data during failure and recovery” on page 36.

## Replication Link—RLINK

An RLINK is associated with an RVG and establishes the link between the Primary and a Secondary RVG. The RLINK associated to the Primary RVG controls the replication settings such as mode of replication, packet size that is used for replication, latency, or Replicator Log protection and protocol. Each RLINK associated with a Primary RVG represents one Secondary. Each RLINK associated with a Secondary RVG represents a Primary.

**Note:** When using the Graphical User Interface (GUI), these RLINKs are transparent to the user as the Secondary host name is used to indicate a pair of RLINKs between the Primary and the Secondary.

The attributes of an RLINK specify the replication parameters for the corresponding Secondary.

A Primary RVG can have up to 32 associated RLINKs. Although a Secondary RVG can also have 32 associated RLINKs, it can have only one active RLINK; this active RLINK represents the Primary that replicates to this Secondary RVG.

Volume Replicator reads data from the Replicator Log volume and sends it to the Secondary. Each Secondary receives data from the Primary at its own rate. For each Secondary, a write on the Replicator Log volume is marked as done when all the Secondary RVGs have successfully received the writes. If a Secondary does not keep up with the write rate, the Replicator Log volume can overflow for the corresponding RLINK.

## Replicated Data Set

Data is replicated from a Primary host, where the application is running, to one or more Secondary hosts. An RVG on the Primary host, and the corresponding RVGs on the Secondary hosts, make up a Replicated Data Set (RDS).

Most Volume Replicator commands operate on an RDS, that is, the Primary RVG and all the Secondaries in the RDS. You can perform Volume Replicator operations from any host in an RDS, unless otherwise noted. Volume Replicator performs the appropriate task on the required hosts in the RDS.

## Data Change Map

Data Change Map (DCM) is a bitmap representing the data difference between Primary and Secondary volumes.

Volume Replicator uses DCM for the following:

- Performing automatic initial synchronization for the data volumes
- Enabling Replicator Log overflow protection when the log protection mode is set to DCM or AutoDCM
- Resynchronizing the Primary data volumes using the snapshot
- Performing fast-failback

Each data volume in the RVG must have a valid DCM log associated with it before the DCM can be used. Volume Replicator calculates the DCM size based on the size of the volume. The default size of the DCM ranges from 1KB to 256KB depending on the size of the volume. However, you can specify the size of the DCM to a maximum of 2 MB.

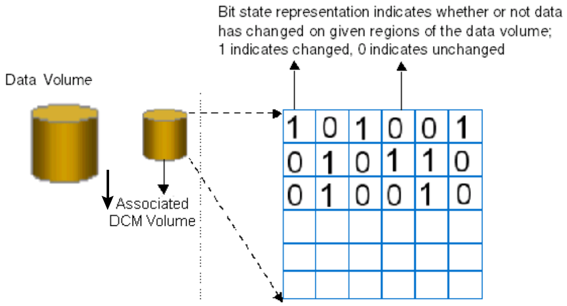
**Note:** If you need to resize the data volumes, then Veritas recommends that you also recreate the DCM proportionate to the new size of the data volume.

When DCM becomes active, the administrator initiates a resynchronization operation and causes Volume Replicator to incrementally synchronize the Secondary with the Primary by looking up the bitmap. Each bit in it represents a region whose contents are different between the Primary and the Secondary. Typically, a region consists of multiples of volume blocks, where each block size is 512 bytes.

**Note:** The Secondary is inconsistent during the period the DCM resynchronization is in progress because the write-order fidelity is not preserved.

After the resynchronization is complete, the Secondary RVG is consistent and replication resumes with write-order fidelity preserved.

**Figure 1-2** DCM layout

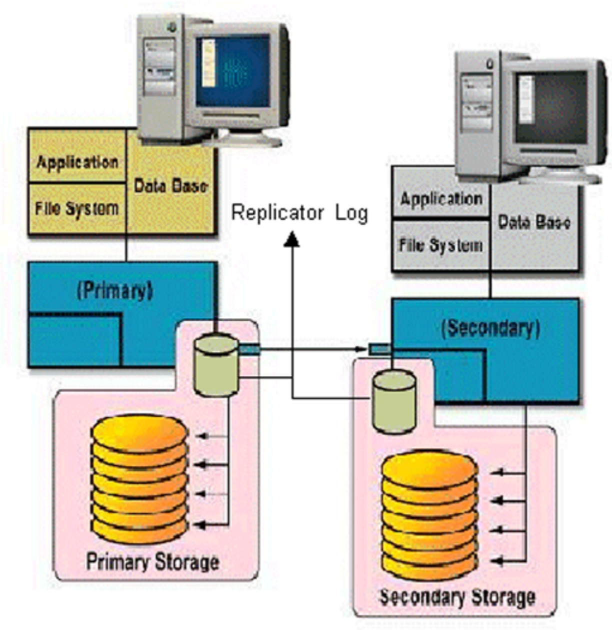

Bit state representation indicates whether or not data has changed on given regions of the data volume; 1 indicates changed, 0 indicates unchanged
Data Volume
Associated DCM Volume

| | | | | | |
|---|---|---|---|---|---|
| 1 | 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 1 | 0 |
| | | | | | |
| | | | | | |
| | | | | | |

# Understanding replication in the Volume Replicator environment

This section describes the Volume Replicator replication process and explains the Volume Replicator setup at the Primary and Secondary host.

**Figure 1-3** Replication process



## Volume Replicator at the Primary

Volume Replicator is configured such that the volumes to be replicated for a specific application are placed in an RVG. Writes to the data volumes are persistently queued in the Replicator Log volume. Volume Replicator uses the Replicator Log volume to track all the writes in the order in which they were received and Volume Replicator transmits the writes to the Secondary using the replication link (RLINK). You can choose to use either the UDP protocol or TCP protocol for network communication between the Primary and Secondary.

The Replicator Log volume is a SFW volume that is configured as part of an RVG. On the Primary, each write to an RVG generates two writes; first to the Replicator Log volume and then to the data volume. Only the write to the Replicator Log volume affects the application. The write to the data volume is written in the background and does not affect application performance.

If the Primary crashes at any point before the write to the data volume is completed, data is fully recoverable from the Replicator Log volume. This is very similar to a database writing to a redo log and later writing to the data files.

Volume Replicator supports several methods to initialize the application data between the Primary location and the remote location which are as follows:

- Automatic synchronization using DCM
- Checkpoints that can be used with block level backups
- Disk group split and join operation, which can be used to move the disks physically to the Secondary site

## Volume Replicator at the Secondary

Volume Replicator sends data to the Secondary RVG as a message, based on the application write size. Each write (update) is divided into one or multiple packets based on the predefined packet size that is specified for a Secondary. These packets are later assembled at the Secondary. When the Secondary receives the message, the Secondary immediately sends an initial acknowledgment of receipt. This is known as the network acknowledgment.

The network acknowledgment allows the Primary to immediately continue processing, as required. The data is not yet written to disk on the Secondary RVG, but it is still safe because it is stored in the Primary Replicator Log volume. After the Secondary writes to the local disk, it sends the second acknowledgment, the data acknowledgment. When the Primary receives the data acknowledgement, this write is discarded from the Replicator Log volume.

The reason for the two-phase acknowledgment is performance. In synchronous mode, the Primary waits for the network acknowledgment from the Secondary before it completes the write for the application. If Volume Replicator were to wait for the write to complete on the Primary and the Secondary, it would increase latency considerably. By using the two-phase acknowledgment, Volume Replicator maintains application performance. Because data is persistently queued in the Primary Replicator Log volume, safety of the data for the Secondary is maintained.

At the Secondary host, Volume Replicator holds the packets until all the previous packets have been received. It then writes to the disks in the correct sequence to maintain consistency at the Secondary. Holding the packets in memory enables Volume Replicator to reassemble out-of-order network traffic before writing, and discover and handle missing packets. To maintain consistency at the Secondary RVG, Volume Replicator never writes an I/O out of order with the Primary RVG. Incoming data from the Primary RVG is serialized and checksummed to support accurate replay to the Secondary volumes.

The Secondary Replicator Log volume is only used in specific conditions which are as follows:

- During recovery, after a Primary or Secondary crash
- To store state of actual underlying volume plexes
- During IBC messaging to a Secondary

## How replication happens in the Volume Replicator environment

The replication process allows data to be replicated across the room or across the world automatically. In general, replication can be used for disaster recovery, providing high availability for the application and data, and load balancing. Volume Replicator is a replication service that provides disaster recovery facility.

When replicating, Volume Replicator sends updates from the Primary host on which the application is running, to the remote host that is the Secondary. Volume Replicator replication is a unidirectional process, whereby the updates on the Primary host are sent to the Secondary host. Volume Replicator setup can have one or more Secondary hosts.

**Warning:** You must ensure that no file systems are mounted on the Secondary when replication is active, as this can result in data loss.

If the data at the Primary gets destroyed, one of Secondary hosts can be made the Primary to make the data write-accessible. You can then restart the applications on that Secondary.

See “About using Volume Replicator as a disaster recovery tool” on page 39.

# Modes of replication

Volume Replicator replicates data in three modes.

They are as follows:

- Synchronous
- Asynchronous
- Synchronous override

Each of the modes follows a different method to replicate the data, and behaves differently under different network conditions. You can choose the mode of replication depending on your specific requirements.

The following factors also determine the choice of modes:

- Available bandwidth
- Network round-trip time
- Number of participating hosts
- Amount of data to be replicated
- Geographical distance

Irrespective of the mode that you choose for replication, Volume Replicator maintains complete data integrity. You must, however, ensure that average bandwidth of your network must be adequate for the update rate of the application.

## Synchronous mode of replication

The synchronous mode of replication (also known as hard synchronous mode) ensures that the Secondary host acknowledges, before completing the update at the Primary. In the case of a problem such as a network failure, it ensures that the update fails at the Primary itself.

The synchronous mode of replication is most effective in the following scenarios:

- Application environments that have lower update rates but require all the hosts to always reflect the same data
- Applications where lag in updates between the Primary and Secondary host is not acceptable

### Advantage of synchronous mode of replication

In the event of a disaster at the Primary host, data can be recovered from the surviving Secondary host without any loss, because the Primary and the Secondary host contain the same data.

### Disadvantages of synchronous mode of replication

This section explains disadvantages of synchronous mode of replication.

- The response time that the writing application experiences is affected because the application has to wait for an acknowledgment from the Secondary before it can complete an update.

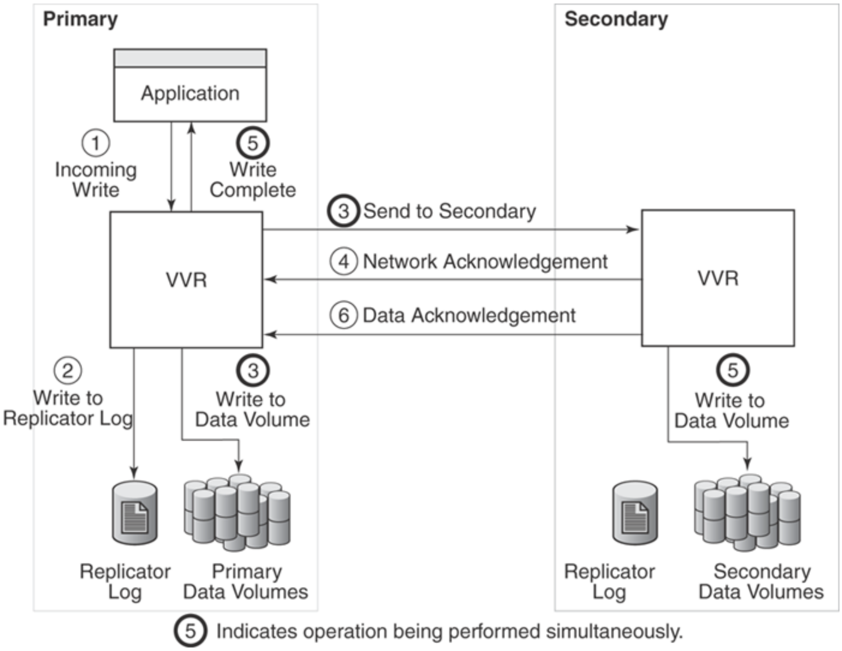
  The following suggestions help to work around the disadvantages to some extent:

  - Add network bandwidth to reduce the degradation in the update response time that the application experiences.
  - Reduce the network round-trip time between each Primary and Secondary pair by using faster network technologies.
- In synchronous mode of replication, if fast failover is set, then the RVG cannot be stopped and started when a disk group fails over to another node. If the RLINK is in hard synchronous mode, it may not be connected when the volume arrives, and the I/Os may fail. In such case, the Event Viewer displays NTFS errors and file system reports the volume as RAW. Therefore, fast failover is not supported if the RLINK is in hard synchronous mode.

## Understanding data flow in Volume Replicator synchronous mode

This section explains how Volume Replicator processes an incoming write when replicating in synchronous mode.

**Figure 1-4** Data flow in synchronous mode of replication



In synchronous mode of replication, Volume Replicator processes an incoming write as follows:

| Task ID | Description |
|---|---|
| 1 | Volume Replicator receives a write on the Primary. |
| 2 | Writes it to the Primary Replicator Log. |
| 3 | Sends the write to the Secondary hosts and waits for the network acknowledgments from the synchronous Secondary hosts. At the same time, Volume Replicator writes to the data volumes on the Primary. |
| 4 | On the Secondary, Volume Replicator receives the write, processes it, and sends a network acknowledgment to the Primary. |

| Task ID | Description |
|---|---|
| 5 | Sends writes to the data volumes on the Secondary; when the Primary receives a network acknowledgment from all the Secondary hosts, Volume Replicator acknowledges to the application that the write is complete.<br>The Secondary RVG sends the network acknowledgment as soon as the write is received. This eliminates the time that is required to write to the Secondary data volumes from the application latency. On the Primary, Volume Replicator does not wait for data to be written to the Secondary data volumes. This improves application performance. However, Volume Replicator tracks all such acknowledged writes that have not been written to the data volumes. Volume Replicator can replay these tracked writes if the Secondary crashes before writing to the data volumes on the Secondary or if the Primary crashes before it receives the data acknowledgment. |
| 6 | When the write is written to the data volumes on the Secondary, Volume Replicator on the Secondary sends a data acknowledgment to the Primary. |

When an RDS containing multiple Secondary RVGs replicates in synchronous mode, the slowest synchronous Secondary determines the application latency. The time to write to the Replicator Log volume, plus the round-trip time that is required to send data to the Secondary RVG and receive the acknowledgment determines the overall performance in synchronous mode.

# Asynchronous mode of replication

In the asynchronous mode of replication, the application updates are immediately reflected at the Primary, but are sent to the Secondary later. The updates are stored in the Replicator Log until they are sent to the Secondary. If the writing application experiences a temporary increase in update rate, this delay may increase.

If a disaster strikes during a period of peak update activity, it is possible that the most recent updates at the Primary host are not reflected in the data at the Secondary host. This is because of the lag between the Primary and Secondary data states called latency. To prevent this, you can configure the latency such that in the event of a disaster the data lag is within acceptable limits. Asynchronous replication ensures that the lag never exceeds this configured maximum.

## Advantages of Asynchronous mode of replication

This section explains certain advantages of replicating in the Asynchronous mode.

Some advantages of the asynchronous mode of replication are as follows:

- The writing application does not suffer from the response time degradation, as there is no network round-trip overhead for each update.
- The rate at which the Replicator Log is drained depends on the maximum available bandwidth or the maximum specified bandwidth. During periods when the update rate is less than the available network bandwidth, the Replicator Log drains faster than it grows. This allows the Secondary data state to catch up with that on the Primary.
- Assures that all completed updates to the Primary volumes are made on the Secondary data volumes, even though it may be with some delay. This is true even in case of failures in communication or system crashes on any of the participating hosts.
- Asynchronous replication can easily handle the temporary network or the Secondary host failure because of its ability to queue updates persistently, and hold them at the Primary for later transmission.

## Disadvantages of Asynchronous mode of replication

This section explains disadvantages of asynchronous mode of replication.

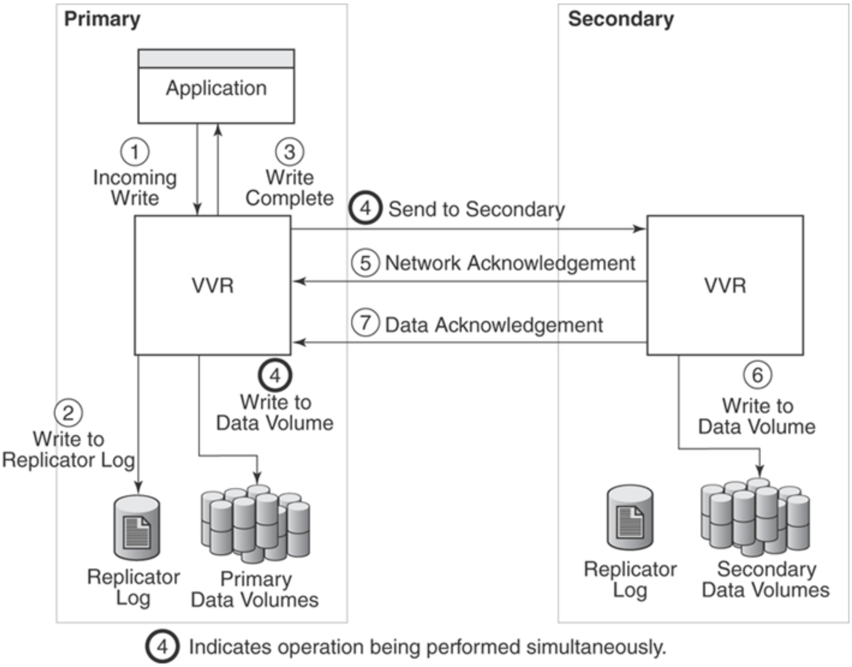
Some disadvantages of the asynchronous mode of replication are as follows:

- The improvement in response time is at the cost of the data at the Secondary host lagging behind the data on the Primary host, during peak update times.
- The volumes at a Secondary host may not have the latest updates when the Primary role is taken over by a Secondary.

# Understanding data flow in Volume Replicator asynchronous mode

This section explains how Volume Replicator processes an incoming write when replicating in asynchronous mode.

**Figure 1-5** Data flow in asynchronous mode of replication



In asynchronous mode of replication, Volume Replicator processes an incoming write as follows:

| Task ID | Description |
|---|---|
| 1 | Volume Replicator receives a write on the Primary. |
| 2 | Writes it to the Primary Replicator Log. |
| 3 | On the Primary, acknowledges to the application that the write is complete. |
| 4 | Sends the writes to the asynchronous Secondary hosts, in the order in which they were received on the Primary, and at the same time, writes to the Primary data volumes. |
| 5 | When the Primary receives the network acknowledgment, it knows that the write has been received in the Secondary memory buffer. |
| 6 | Volume Replicator sends the writes to the data volumes on the Secondary and then sends a data acknowledgement to the Primary. |

| Task ID | Description |
|---|---|
| 7 | When the Primary receives the data acknowledgment, Volume Replicator marks the write as complete in the Replicator Log. |

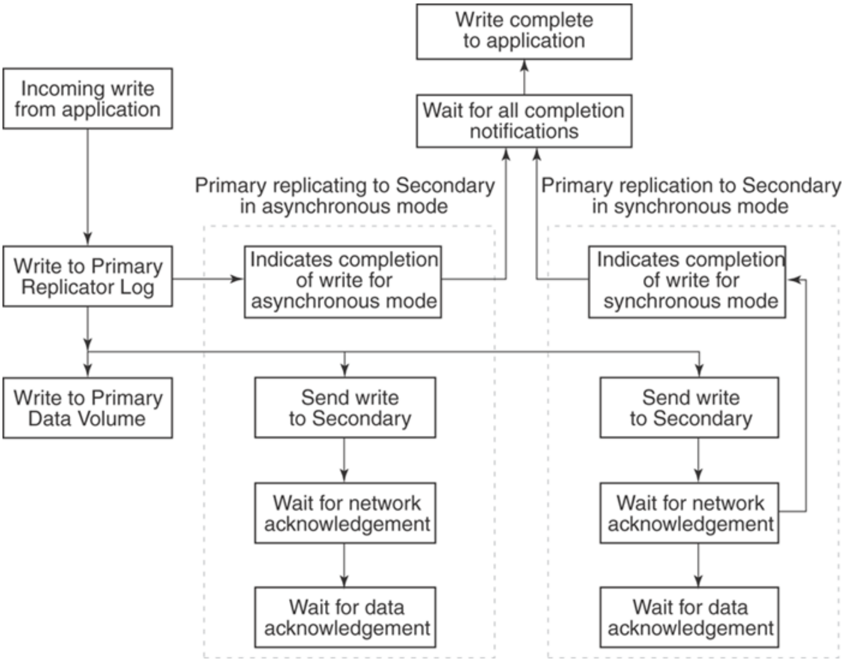
## Synchronous override mode

The synchronous override mode of replication (also known as soft synchronous mode) is a mode where replication is synchronous, as long as the network is available. If the network becomes unavailable, then replication is continued in the asynchronous mode. The pending updates are sent to the Secondary when the network becomes available. When the data becomes completely synchronized then the replication mode reverts back to being synchronous. Depending on specific needs where you want to have synchronous replication, you can use the synchronous override mode of replication for maximum continuity.

# Understanding data flow in an RDS that contains multiple Secondary hosts

An RDS can have multiple Secondary hosts. This section explains how Volume Replicator processes an incoming write for a Replicated Data Set containing multiple Secondary hosts, some replicating in asynchronous mode and some in synchronous mode.

**Figure 1-6** Data flow in case of multiple Secondary hosts in an RDS


Write complete to application
Incoming write from application
Wait for all completion notifications
Primary replicating to Secondary in asynchronous mode
Primary replication to Secondary in synchronous mode
Write to Primary Replicator Log
Indicates completion of write for asynchronous mode
Indicates completion of write for synchronous mode
Write to Primary Data Volume
Send write to Secondary
Send write to Secondary
Wait for network acknowledgement
Wait for network acknowledgement
Wait for data acknowledgement
Wait for data acknowledgement

In asynchronous and synchronous mode of replication, Volume Replicator processes an incoming write as follows, in the presented order:

- Receives a write from the application.
- Writes it to the Replicator Log.
- Volume Replicator first sends the update to all the Secondary hosts replicating in synchronous mode. It then writes to the data volumes under the Primary RVG, and then sends it to the Secondary hosts replicating in asynchronous mode.
- On the Secondary, Volume Replicator receives the write, processes it, and sends a network acknowledgement to the Primary.
- When the Primary receives a network acknowledgment from the Secondary hosts replicating in synchronous mode, Volume Replicator acknowledges to the application that the write is complete. The Secondary RVG sends the network acknowledgment as soon as the write is received. This eliminates the time that is required to write to the Secondary data volumes from the application latency. On the Primary, Volume Replicator waits only for the network acknowledgment from all the synchronous Secondary hosts and not for the data to be written to the Secondary data volumes. This improves application performance. However, Volume Replicator tracks all such acknowledged writes that have not been

written to the data volumes. Volume Replicator can replay these tracked writes if the Secondary crashes before writing to the data volumes on the Secondary or if the Primary crashes before receiving the data acknowledgement.

- When the write is written to the data volumes on the Secondary, Volume Replicator sends a data acknowledgment from the Secondary to the Primary in both synchronous and asynchronous mode.
- When the Primary receives the data acknowledgment from all the Secondary hosts, Volume Replicator marks the write as complete in the Replicator Log.

# Managing data during failure and recovery

This section gives an overview of the methods of preventing data loss and maintaining data consistency even during a failure and subsequent recovery process.

Some concerns that need to be considered during a failure and the subsequent recovery are described in the following sections:

- See “Preventing data loss” on page 36.
- See “Maintaining data consistency” on page 37.
- See “Detecting host and connection failures” on page 37.
- See “Securing Volume Replicator” on page 38.

## Preventing data loss

This section describes the techniques that Volume Replicator uses to prevent data loss.

### Preventing data loss during normal operations

During normal operation, Volume Replicator prevents data loss by logging all the updates to the Primary Replicator Log volume and ensuring that this operation is completed before writing to the Primary and Secondary data volumes. The Primary Replicator Log volume can be used to obtain the correct contents of all the data volumes, except in the case of failure of the Primary Replicator Log volume or the data volume itself.

### Preventing data loss during a Primary host failure

In the case of a Primary host failure, the Primary data volumes may slightly lag behind the Primary Replicator Log volume. During recovery, the first Primary Replicator Log volume entry that has not yet been written to the data volumes is identified, and the Primary Replicator Log volume is replayed from that point. During the recovery period, the RVG is not available for Input/Output operations. The recovery time is short because there are only a few blocks that have not been written to the data volumes.

Volume Replicator also supports fast-failback to the original Primary, once the original Primary becomes available. This is achieved by using the DCM logs.

See “Performing takeover with fast-failback ” on page 202.

## Maintaining data consistency

Volume Replicator uses co-ordinating operations to maintain data consistency by maintaining the same write-order on each Secondary as on the Primary. The Primary Replicator Log volume is time-ordered and contains the data for each individual write. The disk modifications also occur in the same order on the Secondary as on the Primary.

If the Primary recovers after a crash, Volume Replicator locates the last entry in the Primary Replicator Log volume that the Secondary did not acknowledge as successful, before the crash. Updates to this Secondary continue from that point onwards.

When the Primary or the Secondary crashes, the Volume Replicator recovery process ensures that all the pending updates on the Primary are sent to the Secondary in such a way that there is no data loss, and the data is consistent at the end of the recovery. Secondary Replicator Log is used for this purpose.

Volume Replicator is designed to maintain consistency between the Primary RVG and the Secondary RVG even in the event of network failures and the temporary loss of the Primary or Secondary host, or both. When the problem is corrected, and the Primary and Secondary are again both active and able to communicate, the Primary and Secondary automatically resynchronize themselves and continue replication. A Secondary may become temporarily inconsistent during this resynchronization phase. However, because synchronization is achieved in a protected manner, a subsequent network or host failure during this phase cannot cause inconsistency on the Secondary, even if the Primary host is permanently lost.

## Detecting host and connection failures

The Primary and Secondary hosts exchange messages periodically even when there is no replication activity using the heartbeat protocol. This helps to detect host or connection failure between the Primary and Secondary.

See “Replicator Log protection when Primary and Secondary are disconnected” on page 50.

# Securing Volume Replicator

Volume Replicator is capable of replicating over a firewall and also supports Network Address Translation (NAT).

Volume Replicator operations can be performed directly from the VEA or using the CLI. You can perform the operations on the various Volume Replicator objects which include RVG, RDS, replicated volumes, and the RLINKs (Secondaries). Some Volume Replicator operations involve more than one host as a part of their operations. Before executing such an operation, Volume Replicator first validates whether the originator host is allowed to execute the specified operation on the target hosts. If not, the specified operation fails. The Volume Replicator Security Service (VxSAS) wizard manages the validation process, also known as the security check. These measures provide a higher level of security to your application and data.

See “Security considerations for Volume Replicator” on page 77.

Chapter 2

# Replication concepts

This chapter includes the following topics:

## About using Volume Replicator as a disaster recovery tool

This chapter explains the important concepts of Volume Replicator, the most important one being able to transfer the Primary role and failing back. Veritas recommends that you read this chapter before setting up replication. The term RLINK has been used to explain important Volume Replicator concepts.

See “Replication Link—RLINK” on page 23.

For detailed information about configuring DR solutions, refer to the Storage Foundation and High Availability Solutions HA and Disaster Recovery solutions guides.

One of the key advantages of Volume Replicator is its capability to provide a disaster recovery solution. In the case of a Primary host failure or a disaster at the Primary site, it may become necessary to transfer the role of the Primary to the Secondary. At times, it may be necessary to bring down the Primary host for maintenance purposes. This can be achieved by transferring the Primary role to any Secondary having up-to-date data.

Volume Replicator enables you to transfer the Primary role from a healthy or failed Primary using the Graphical User Interface (GUI) or the command line options. It also enables you to failback to the original Primary using a simple set of operations.

The following topics describe the methods that Volume Replicator offers to transfer the Primary role:

- See “Migrating the Primary role ” on page 40.
- See “Taking over the Primary role” on page 40.
- See “Performing a takeover using the fast-failback option” on page 41.

## Migrating the Primary role

Migrating the Primary role involves interchanging the role of a healthy Primary with that of a Secondary, when the application that is involved in replication is inactive. You can also plan to change the role of the Primary if you need to perform some maintenance activities or some other configuration changes to the Primary. To migrate successfully, the data between the Primary and the Secondary must be up-to-date.

Volume Replicator provides options from the GUI as well as the command line to migrate a healthy Primary. The migrate operation involves migrating the Primary role of an RVG to a Secondary, thus converting the Secondary RVG to a Primary RVG.

## Taking over the Primary role

When the original Primary fails or is destroyed because of a disaster, the takeover procedure enables you to convert a consistent Secondary to a Primary.

To determine whether the takeover of the Primary by a Secondary is successful, you must first consider whether the data is consistent and how up-to-date it is.

Volume Replicator provides the takeover operation to transfer the Primary role both from the graphical user interface as well as the command line. Upon successful completion of the takeover, the Secondary becomes the Primary.

**Note:** The takeover operation can be performed only on the Secondary host, when the Primary becomes unavailable, or the Secondary cannot communicate with the Primary.

## Performing a takeover using the fast-failback option

In the case of a Primary failure or if the Primary needs to be brought down for some maintenance tasks, the role of the Primary needs to be taken over by the Secondary. When the old (original) Primary comes up you can failback from the new Primary to the original Primary. The fast-failback feature enables you to do this quickly and efficiently as it performs incremental synchronization, for only the changed data. This feature uses the DCMs of the data volumes of the new Primary, to keep track of the changed content and the new content. This process of logging on the DCM is called failback logging.

You can perform the takeover operation with fast-failback by selecting the failback logging option on one of the Secondaries. After the takeover operation is complete the applications are started on the new Primary. All the subsequent writes from the applications running on the new Primary are then tracked on the DCM of the new Primary. When the original Primary recovers, it discovers that one of its Secondaries has taken over as the new Primary and it starts acting as a Secondary. The synchronization to the original Primary can be started manually or automatically depending on the options that are specified during takeover. The RVG volumes on the original Primary disallow access permissions to the applications and need to be synchronized with the new Primary by playing back the DCM. You need to perform the resynchronization operation to start the DCM replay. At the start of the DCM replay, the original Primary becomes a Secondary and starts receiving the missing updates.

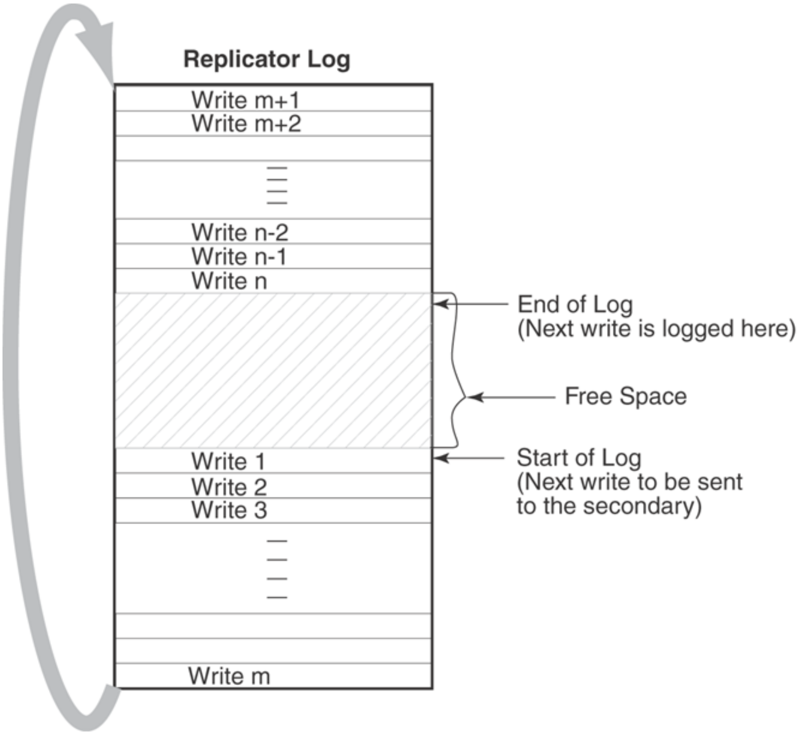
You can then continue to use the current setup after takeover, as is, or, you can complete the failback process by using the migrate operation to change the Primary role back to the original Primary. If you want to migrate the role of Primary back to the original Primary then you do not need to perform the operation to add the other Secondaries back to the original Primary. The RLINKs from the other Secondaries to the original Primary are still retained, and once the Primary role is migrated back to the original Primary (current Secondary) these Secondaries automatically become Secondary hosts to the original Primary.

# Understanding how Volume Replicator logs writes to the Replicator Log

Volume Replicator receives writes from the application and queues them in the Primary Replicator Log for transmission to the Secondary hosts. If a Primary RVG is connected to multiple Secondary RVGs, the Replicator Log on the Primary is used to manage the writes for these Secondary hosts. The Replicator Log header contains a specific set of pointers for each Secondary which indicates the writes that have not been sent to the corresponding Secondary.

This section explains the working of the Replicator Log as a circular buffer.

**Figure 2-1** Illustrates the working of the Replicator Log as a circular buffer

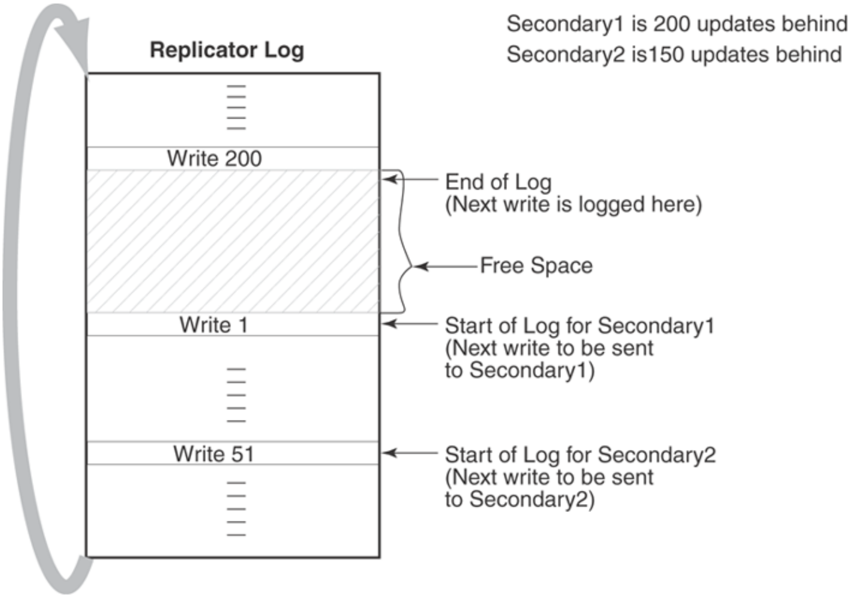


The first write that comes in is Write 1, which also represents the start of log for the Secondary. Volume Replicator logs Write 2, Write 3, Write m one after the other until it reaches the end of the Replicator Log. Because the Replicator Log is a circular log the next write, Write m+1 wraps around and logging continues. When the Primary receives the data acknowledgment from this Secondary host for Write

1, Volume Replicator marks the Write 1 as complete in the Replicator Log. Volume Replicator then processes Write 2, Write 3, and so on.

Secondary1 is 200 writes or updates behind, whereas Secondary2 is 150 writes behind. If the end of log pointer reaches the start of log pointer of the Secondary, the Replicator Log overflows for this Secondary.

**Figure 2-2** The working of the Replicator Log when the Secondary is behind


Secondary1 is 200 updates behind
Secondary2 is150 updates behind
Replicator Log
Write 200
End of Log
(Next write is logged here)
Free Space
Write 1
Start of Log for Secondary1
(Next write to be sent
to Secondary1)
Write 51
Start of Log for Secondary2
(Next write to be sent
to Secondary2)

The Secondary hosts for which the replication is configured in synchronous mode are usually up-to-date. Typically, the number of simultaneous I/O operationsthat the application performs separates the start and the end of log pointers of synchronous RLINKs (Secondaries). For asynchronous RLINKs, the difference between the start of log pointer and end of log pointers reflect how many outstanding writes have yet to be processed, that is, how behind is the RLINK. Different RLINKs usually have start of log pointers indicating different places in the Replicator Log; this reflects the difference in the rate at which data is sent to the Secondary. After the Primary receives the data acknowledgment from all the Secondary hosts, Volume Replicator marks the write as complete in the Replicator Log volume.

## Sizing the Replicator Log

The size of the Replicator Log is critical to the performance of replication. In the asynchronous mode of replication, due to network latency, the writes may be pending on the Primary Replicator Log. In this case, the Primary Replicator Log may overflow if the number of pending writes exceed the number of updates it can store.

When the Replicator Log overflows for a particular Secondary, the RLINK corresponding to that Secondary is marked STALE and becomes out of date until a complete resynchronization with the Primary is performed. Because resynchronization is a time-consuming process and during this time the data on the Secondary cannot be used, it is important to avoid Replicator Log overflows.

Thus, the Replicator Log size needs to be large enough to satisfy the following constraints:

- It must not overflow for asynchronous RLINKs during periods of peak usage when replication over the RLINK may fall far behind the application.
- It must not overflow while a Secondary RVG is synchronized.
- It must not overflow while a Secondary RVG is restored.
- It must not overflow during extended outages (network or Secondary node).

## Determining the size of the Replicator Log

To determine the size of the Replicator Log, you must evaluate each of the following constraints individually. Then, choose a value at least equal to the maximum so that all constraints are satisfied.

**Note:** If the size of the Replicator Log specified is not enough to meet new business requirements, then you can resize the Replicator Log.

For more information, See “Expanding the Replicator Log” on page 165.

To determine the size of the Replicator Log, you need the following information:

- The maximum expected downtime for Secondary nodes.
- The maximum expected downtime for the network connection.
- The method for synchronizing Secondary data volumes with data from Primary data volumes.

  If the application is shut down to perform the synchronization, the Replicator Log is not used and the method is not important. Otherwise, this information can include, the time that is required to copy the data over a network, or the time that is required to copy it to a tape or disk, to send the copy to the Secondary site, and to load the data onto the Secondary data volumes.

  Note that if you use the **Synchronize Automatically** option from VEA to synchronize the Secondary the previous paragraph is not a concern.

In the case of Secondary data volume failure if you perform Secondary backup to avoid complete synchronization, the information needed includes the following:

- The frequency of Secondary backups.
- The maximum expected delay to detect and repair a failed Secondary data volume.
- The expected time to reload backups onto the repaired Secondary data volume.

# Understanding replication settings for a Secondary

The Volume Replicator replication settings determine the replication behavior between the Primary RVG and the corresponding Secondary RVG.

Volume Replicator behaves differently based on the option that has been specified for the following:

- Mode of replication
- Replicator Log overflow protection
- Latency protection

To use these replication settings effectively in your environment, it is important to understand how each setting affects replication when the Primary and Secondary are connected and disconnected. A Secondary is said to be disconnected from the Primary if there is communication loss between Primary and Secondary RVG because of a network outage or administrative action.

Volume Replicator enables you to set the replication mode, latency protection, and Replicator Log protection using both the GUI and the CLI. The following sections explain the concepts that are associated with these settings, with the help of the command line attributes `synchronous`, `latencyprot`, and `srlprot` respectively. These attributes are of the form `attribute=value`. Each attribute setting can affect replication and must be set up with care. These settings can also be changed from the GUI using the Change Replication Settings dialog box.

## Mode of replication—synchronous attribute

Volume Replicator replicates in two modes: synchronous and asynchronous. The decision to use synchronous or asynchronous mode must be made with an understanding of the effects of this choice on the replication process and the application performance. You can set up Volume Replicator to replicate to a Secondary in synchronous override or asynchronous mode by setting the `synchronous` attribute of the Secondary to `override`, or `off` respectively.

**Note:** While you set the mode of replication from the GUI, the synchronous override is the default mode of replication.

The following table summarizes the effect of RLINK on modes of replication.

**Table 2-1** Effect of RLINK state on modes of replication

| Value of `synchronous` Attribute | When RLINK (Secondary) is connected | When RLINK (Secondary) is disconnected |
|---|---|---|
| `synchronous=off` | Asynchronous | Asynchronous |
| `synchronous=override` | Synchronous | Asynchronous |
| `synchronous=fail` | Synchronous | I/O error to application |

These terms have been explained as follows:

- `synchronous=off`

  Specifying the attribute value as `off` sets the replication mode to asynchronous.

- `synchronous=override`

  Specifying the attribute value as `override` sets the replication mode to synchronous override. During normal operation, Volume Replicator replicates in synchronous mode, but if the RLINK is disconnected, Volume Replicator switches temporarily to asynchronous mode and continues to receive writes from the application and logs them in the Replicator Log. After the connection is restored and the RLINK is up-to-date, the RLINK automatically switches back to synchronous mode. Most system administrators set `synchronous=override`.

- `synchronous=fail`

  Specifying the attribute value as `fail` sets the replication mode to synchronous. During normal operation, Volume Replicator replicates in synchronous mode, but if the RLINK is disconnected, Volume Replicator fails incoming writes to the Primary.

## Using the available bandwidth effectively

Volume Replicator uses the network to replicate data from the Primary to the Secondary. The Bandwidth Throttling feature enables you to control the maximum network bandwidth that Volume Replicator uses for replication. Bandwidth Throttling controls the rate at which data is sent from the Primary to the Secondary; it does not limit the rate at which the network acknowledgments are sent from the Secondary to the Primary.

By default, Volume Replicator uses the entire available network. However, you might want to control the bandwidth that Volume Replicator uses depending on factors such as, whether other applications can use the available network connection or is it exclusively for Volume Replicator, the network costs, and network usage over time. For example, if the network is used for purposes other than replication, you might have to control the network bandwidth used by Volume Replicator. Volume Replicator enables you to change the network bandwidth that is used for replication to the Secondary, even when replication is in progress.

If you want Volume Replicator to use the entire available network bandwidth then do not set any value for the bandwidth parameter either using the GUI or command line.

### Bandwidth of the available network connection

The type of connection determines the maximum bandwidth available between the two locations. However, the important factor to consider is whether other applications can use the available connection or is it exclusively reserved for replicating to a single Secondary. If other applications use the same line, it is important to be aware of the bandwidth requirements of these applications and subtract them from the total network bandwidth. If any applications sharing the line have variations in their usage pattern, it is also necessary to consider whether their times of peak usage are likely to coincide with peak network usage by Volume Replicator. Additionally, overhead added by Volume Replicator and the various underlying network protocols reduces effective bandwidth by a small amount, typically 3% to 5%.

## Choosing the network protocol

Volume Replicator exchanges two types of messages between the Primary and the Secondary: heartbeat messages and data messages. The heartbeat messages are transmitted using the UDP transport protocol. Volume Replicator can use the TCP transport protocol or the UDP transport protocol to exchange data messages. If the setup includes a Bunker node and the storage is shared between the Primary and the Bunker node, then the storage is visible on the Primary. In this case, you can import the Bunker disk group on the Primary and then use the storage protocol for transmitting information to the Bunker Secondary.

The choice of protocol to use for the data messages is based on the network characteristics. Volume Replicator uses the UDP transport protocol by default and in most networks, Volume Replicator with UDP may perform better. However, you must experiment with TCP and UDP protocols to determine the one that performs better in your network environment.

**Note:** You must specify the same protocol for the Primary and Secondary; otherwise, the nodes cannot communicate and the RLINKs do not connect. This also applies to all nodes in a cluster environment.

# Measures to protect log overflow and replication latency

This section describes some key parameters that you can set to protect replication from being stopped. Setting the `srlprot` attribute appropriately prevents the Replicator Log from overflowing. Similarly, you can set the `latencyprot` attribute to make sure that the Secondary does not lag too far behind the Primary.

See "Replicator Log overflow protection—`srlprot` attribute" on page 48.

See "Latency protection—`latencyprot` attribute" on page 52.

## Replicator Log overflow protection—`srlprot` attribute

Volume Replicator provides the following modes of overflow protection: Override, Fail, DCM, and AutoDCM. You can also turn off the Replicator Log overflow protection feature by setting the attribute to `off`.

If the network is down or the Secondary is unavailable, the number of writes in the Replicator Log waiting to be sent to the Secondary could increase until the Replicator Log fills up. When the Replicator Log cannot accommodate a new write without overwriting an existing one, the condition is called Replicator Log overflow. The new writes are held up, DCM is activated, or the Replicator Log overflows depending on the srlprot setting.

Circumstances that can cause the Replicator Log to overflow when you replicate in the asynchronous mode are as follows:

- A temporary burst of writes, or a temporary congestion in the network, causing the current update rate to exceed the currently available bandwidth between the Primary and the Secondary.
- A temporary failure of the Secondary node or the network connection between the Secondary and the Primary.
- An administrator pauses the RLINK from the VEA GUI or by executing a `vxrlink pause` command.
- Inability of the network bandwidth to keep up with the update rate at the Primary on a sustained basis. This is not a temporary condition and can be corrected

only by increasing the network bandwidth or reducing the application update rate, if possible.

If the Replicator Log overflows, the Secondary becomes out-of-date and must be completely synchronized to bring it up-to-date with the Primary. The Replicator Log protection feature of Volume Replicator enables you to either prevent the Replicator Log from overflowing or tracks the writes using Data Change Map (DCM) in the case of Replicator Log overflow. You must weigh the trade-off between allowing the overflow or affecting the application. You can prevent Replicator Log overflow by using the srlprot setting.

Volume Replicator provides the following modes of Replicator Log overflow protection: AutoDCM, DCM, Override, and Fail. These modes are activated only when the Replicator Log is about to overflow. You can set up Replicator Log protection by setting the `srlprot` attribute of the corresponding RLINKs to AutoDCM, DCM, Override, or Fail. You can turn off the Replicator Log protection by setting the `srlprot` attribute to `off`.

The following table describes the effect of the RLINK state on the Replicator Log protection.

**Table 2-2** Effect of RLINK state on the Replicator Log Protection

| Value of the `srlprot` Attribute | When RLINK is Connected | When RLINK is Disconnected |
|---|---|---|
| `autodcm` | Convert to DCM logging | Convert to DCM logging |
| `dcm` | Protect<br>**Note:** SRL protects writes by stalling application writes until Replicator Log drains 5% to become 95% full or drains 20 MB, whichever is smaller. | Convert to DCM logging |
| `override` | Protect | Overflow |
| `fail` | Protect | I/O error to application |

**Note:** When `srlprot=off`, the Replicator Log overflows irrespective of whether the RLINK is connected or disconnected.

If the Replicator Log overflow protection is enabled and if a write is about to cause the log to overflow, then the Replicator Log protection is turned on.

## Replicator Log protection when Primary and Secondary are connected

This section explains how Volume Replicator works when the Replicator Log is about to overflow while the Primary and Secondary are connected, for different settings of the srlprot attribute.

Different settings of srlprot attribute when Primary and Secondary are connected are as follows:

- `srlprot=override`, `fail`, or `dcm`

  New writes are throttled in the operating system of the Primary host until a predetermined amount of space, that is, 5% or 20 MB (whichever is lesser) becomes available in the Replicator Log.

- `srlprot=autodcm`

  Volume Replicator activates the DCM, instead of throttling writes. Each data volume in the RVG must have a DCM. If every data volume has a DCM attached to it then by default, the Replicator Log protection is set to the `AutoDCM` mode.

- `srlprot=off`

  Disables Replicator Log protection and allows the Replicator Log to overflow.

## Replicator Log protection when Primary and Secondary are disconnected

This section explains in detail how Volume Replicator works when the Replicator Log is about to overflow while the Primary and Secondary are disconnected for different settings of the srlprot attribute.

Different settings of srlprot attribute when Primary and Secondary are disconnected are as follows:

- `srlprot=override`

  Writes by the application to the Primary are allowed to complete even if it overflows the Replicator Log.

- `srlprot=off`

  Disables Replicator Log protection and lets the Replicator Log overflow.

- `srlprot=fail`

  Writes by the application to the Primary are failed to make sure that the Replicator Log does not overflow.

- `srlprot=dcm, autodcm`

  DCM protection is activated and writes are written to the DCM. Each data volume in the RVG must have a DCM.

## Changing between the states of Replicator Log protection

To enable Replicator Log protection you can set the srlprot attribute to any one of the modes: fail, override, DCM, or AutoDCM. Volume Replicator allows transition between the srlprot values, but there are some situations when the transitions between the states do not succeed.

**Note:** When the DCM logging is enabled as part of Replicator Log protection mechanism, changing to the Fail or Override mode is disallowed.

The following table highlights the valid state transitions when the Secondary (RLINK) is connected.

**Table 2-3** Valid state transitions with Secondary RLINK connected

| **Changing Replicator Log Protection (`srlprot`) > From** | **Changing Replicator Log Protection (`srlprot`) > To** | **Outcome of the original state** | **Result after state transition** |
|---|---|---|---|
| `Fail` | `AutoDCM` | The writes are correctly throttled, until the Replicator Log is freed of some space. | Changing the mode from `Fail` to `AutoDCM` cannot guarantee that the DCM logging is enabled. |
| `Override` | `AutoDCM` | The writes are correctly throttled, until the Replicator Log is freed of some space. | Changing the mode from `override` to `AutoDCM` cannot guarantee that the DCM logging is enabled. |
| `DCM` | `AutoDCM` | The writes are correctly throttled, until the Replicator Log is freed of some space. | Changing the mode from `DCM` to `AutoDCM` cannot guarantee that the DCM logging is enabled. |
| `AutoDCM` | `DCM` | The DCM logging is enabled. | Since the DCM logging is already enabled, the RLINK (Secondary) is not disconnected. |

The following table highlights the valid state transitions when the RLINK (Secondary) is disconnected.

**Table 2-4** Replication State Transitions when Secondary RLINK is disconnected

| **Changing Replicator Log Protection (`srlprot`) > From** | **Changing Replicator Log Protection (`srlprot`) > To** | **Outcome of the original state** | **Result after state transition** |
|---|---|---|---|
| `Fail` | `AutoDCM` | Results in an error to the application for the current write. | The DCM logging is enabled on the next Input/Output operation by the application. |
| `Override` | `AutoDCM` | The Replicator Log Overflows. | The DCM logging is not used since the Replicator Log has already overflowed and replication is stopped. The replication must be started to the Secondary using the Automatic Synchronization operation. |
| `DCM` | `AutoDCM` | The DCM logging is enabled. | The DCM logging is already enabled.<br>Secondary must be made up-to-date by using the resynchronization operation. |
| `AutoDCM` | `DCM` | The DCM logging is enabled. | The DCM logging is already enabled.<br>Secondary must be made up-to-date by using the resynchronization operation. |

## Latency protection—`latencyprot` attribute

Volume Replicator provides the following modes of latency protection: Override and Fail. You can also turn off the latency protection feature by setting the `latencyprot` attribute to `off`. This section describes how you can use the latency protection feature to prevent the Secondary from falling far too behind.

## Understanding latency protection

When you replicate in asynchronous mode, it is normal for the Replicator Log to have writes waiting to be sent to the Secondary. If your network has been sized based on the average update rate of the application on the Primary node, the number of writes waiting in the Primary Replicator Log is likely to be within an acceptable range.

The number of writes in the Replicator Log that would grow under the following circumstances are as follows:

- A temporary burst of writes or a temporary congestion in the network, which causes the current update rate to exceed the currently available bandwidth between the Primary and the Secondary.
- A temporary failure of the Secondary node or the network connection between the Secondary and the Primary.
- Performing the pause operation.
- Inability of the network bandwidth to keep up with the update rate at the Primary, on a sustained basis. This is not a temporary condition and can be corrected only by increasing the network bandwidth or reducing the application update rate, if possible.

If the Primary Replicator Log has a large number of writes waiting to be transferred to the Secondary, the Secondary data is considerably behind the Primary. If a disaster strikes the Primary and the Secondary takes over, the Secondary does not contain all the data in the Primary Replicator Log. In this case, the data on the Secondary is consistent but out of date when the Secondary takes over. In such a scenario, to prevent the Secondary from being too far behind the Primary, you can limit the number of writes waiting in the Primary Replicator Log for transmission to the Secondary, by setting up latency protection.

Latency protection has two components, its mode, and the `latency_high_mark` and `latency_low_mark` values which specify when the protection is active or inactive. The `latency_high_mark` specifies the maximum number of pending updates by which the Secondary can be behind the Primary. If the number of such updates in the Replicator Log reaches the specified `latency_high_mark` value, then, further writes to the Primary are stalled or failed, depending on the mode of latency protection. In this situation, the writes continue to be stalled or failed until the number of pending updates in the Replicator Log falls to the specified `latency_low_mark` value. Hence, the `latency_low_mark` value must be a number lower than the `latency_high_mark` value.

You can enable latency protection by setting the `latencyprot` attribute to either `override` or `fail`. Setting the attribute to `latencyprot=off`, which is the default, disables latency protection.

The following table summarizes how the state of the RLINK affects the latency protection.

**Table 2-5** The state of RLINK and latency protection

| Value of `latencyprot` Attribute | When RLINK is Connected | When RLINK is Disconnected |
|---|---|---|
| `override` | Protect | Drop protection |
| `off` | No protection | No protection |
| `fail` | Protect | I/O error to application |

The following sections explain how Volume Replicator controls replication depending on the setting of the `latencyprot` attribute of the RLINK when the Primary and Secondary are either connected or disconnected.

## Latency protection when Primary and Secondary are connected

Under normal operation, if the number of waiting writes increase and reach the `latency_high_mark`, the consecutive writes are stalled in the operating system of the Primary until the Replicator Log drains to bring the number of waiting writes below the `latency_low_mark`.

`latencyprot=fail`

## Latency protection when Primary and Secondary are disconnected

Primary and Secondary are said to be disconnected when they are in the PAUSED state or are disconnected because of a network outage, or an outage of the Secondary node.

The attributes are as follows:

- `latencyprot=override`

  Volume Replicator allows the number of writes in the Replicator Log to exceed the `latency_high_mark`. In this case, Volume Replicator causes latency protection to be overridden and allows incoming writes from the application whose data is replicated. Volume Replicator does not stall incoming writes because the Replicator Log is not draining, and incoming writes may be stalled indefinitely. Stalling of incoming writes is undesirable for the writing application. Most system administrators set `latencyprot=override`.

- `latencyprot=fail`

If the number of writes in the Replicator Log reaches the `latency_high_mark` while the Primary and the Secondary are disconnected, Volume Replicator causes new writes at the Primary to fail. This prevents the Secondary from falling further behind than specified by the `latency_high_mark`.

# Pausing the replication

Using the Pausing the replication feature, you can temporarily stop sending the updates to the Secondary hosts.

## Pausing the replication from the Primary host

It is a good practice to backup the Secondary data volumes at frequent intervals. During this period you can pause updates to any Secondary from the Primary host. During a pause, the Primary continues to keep a history of volume updates but does not send the updates to the Secondary. The network session between the Primary and paused Secondary (on behalf of the Secondary) is broken.

Sometimes, pausing the replication from the Primary host may be required to perform some maintenance tasks on the Secondary or to make configuration changes such as changes to the network connecting the two hosts. This can be done effectively by pausing the Secondary from the Primary.

You can use the resume feature to reestablish the network session between the Primary and Secondary host and allow updates to continue from the point of the pause. If there are any updates to the volume during the pause, a synchronous Secondary is forced to become asynchronous, until it catches up.

## Pausing the replication from the Secondary host

You can also pause updates to the Secondary from the Secondary host. Unlike the pause that is initiated from the Primary, the network session between the Primary and Secondary is maintained. Maintaining the connection allows the Secondary to notify the Primary when it wants updates to the RVG to continue.

**Note:** If the Secondary host has lost contact with the Primary host, then you cannot take backups of the Secondary RVG volumes using checkpoints.

## Applications of the pause feature

You can use the pause feature of Volume Replicator to perform maintenance tasks, to backup Secondary data, and to change mode of replication.

Using the pause feature, you can do the following tasks:

- To perform network maintenance tasks such as changing IP addresses on the Primary and Secondary host.
- To stop using the network for some time to allow some other application to use it.
- To backup the Secondary data which can be restored later, if required.
- To change the mode of replication for a Secondary from the synchronous override mode to asynchronous mode when the network is slow, to avoid the writes from being stalled.

# Understanding checkpoints

Volume Replicator checkpoints are user-defined markers in the Primary Replicator Log. The RVG and RLINK (Secondary) checkpoints are two types of checkpoints. The RVG checkpoint has a start (checkstart) and an end (checkend) and can be used for initial synchronization. The RLINK (Secondary) checkpoint is used to restore Secondary volumes in case of failure.

Checkpoints are used to perform the tasks which are as follows:

- Synchronizing the Secondary while the Primary application is active
- Restoring the Secondary data volumes

The Secondary data volumes must be synchronized with the Primary data volumes before replication can start, that is, after you add a Secondary to the RDS, after a Secondary data volume error, or after Replicator Log overflow. Volume Replicator enables you to synchronize the Secondary data volumes while the application is active on the Primary. If you use the Automatic Synchronization feature of Volume Replicator to synchronize the Secondary data volumes over the network, Volume Replicator ensures that the Secondary data volumes are consistent and up-to-date when the synchronization process completes.

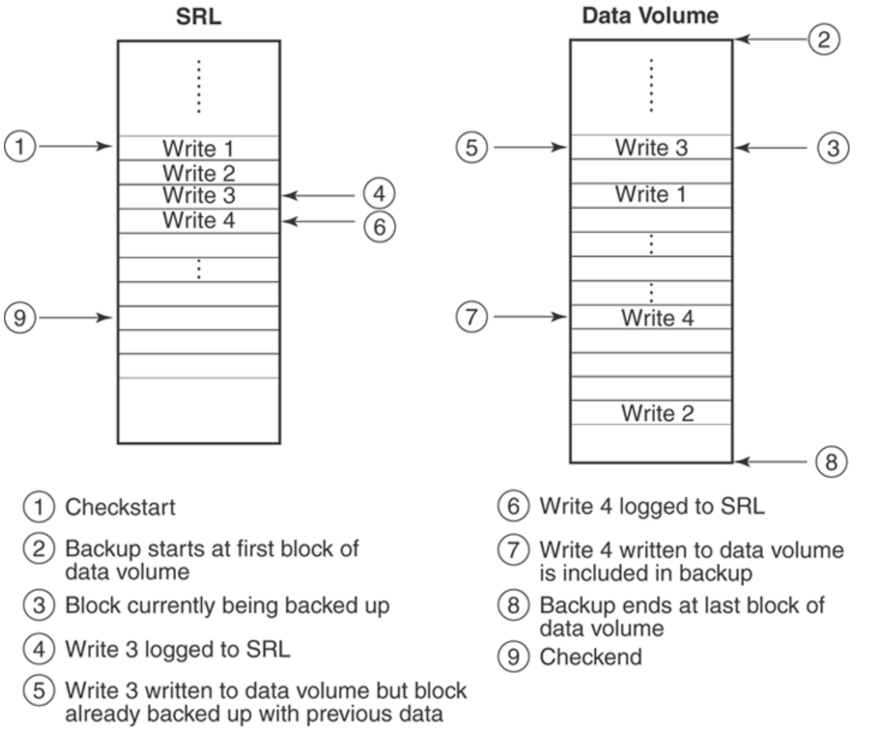
If you use the backup and checkpoint method for synchronizing the Secondary and if the Primary application is active during the backup process, then, after you restore the backup on the Secondary, the Secondary data volumes are inconsistent and not up-to-date.

To make the Secondary consistent and up-to-date, Volume Replicator must transfer all the blocks that changed during the backup process, in the order that they changed. In a Volume Replicator environment, all writes to the Primary data volumes are logged to the Replicator Log; therefore, Volume Replicator can transfer the writes that occurred during the backup to the Secondary. To do this, Volume Replicator must know the start and end of the backup process. RVG checkpoints are used to indicate this start position (checkstart) and end position (checkend) in the Replicator Log.

Because the checkpoint information is stored in the Replicator Log, checkpoints become invalid when the Replicator Log wraps around. The same checkpoint and tape backups can be used to synchronize the data volumes on multiple Secondary hosts if the checkpoint remains valid.

**Note:** If a checkpoint becomes invalid, performing the synchronize operation using that checkpoint fails.

**Figure 2-3** Figure illustrates how Volume Replicator handles checkpoints


SRL
Data Volume
1
2
Write 1
Write 2
Write 3
Write 4
4
6
9
5
Write 3
3
Write 1
7
Write 4
Write 2
8
1 Checkstart
2 Backup starts at first block of data volume
3 Block currently being backed up
4 Write 3 logged to SRL
5 Write 3 written to data volume but block already backed up with previous data
6 Write 4 logged to SRL
7 Write 4 written to data volume is included in backup
8 Backup ends at last block of data volume
9 Checkend

A backup utility may copy previous contents of the blocks corresponding to Write 3 (event 5) but copy updated contents of the blocks corresponding to Write 4 (event 7).

However, Volume Replicator logs all the writes to the Replicator Log (events 4 and 6). Note that a checkstart was performed (event 1) before the backup was started (event 2) and a checkend was performed (event 9) after the backup was completed (event 8). When you start replication with this checkpoint after the backup is restored on Secondary, Volume Replicator can transfer all the writes between checkstart and checkend and make the Secondary data volumes up-to-date and consistent.

# Synchronizing the Secondary

The Secondary must be synchronized with the Primary to have consistent data at all times. Before a Primary can replicate data to a Secondary, or after Replicator Log volume overflows, you must make a block-for-block copy of the data on the Primary to the Secondary, to synchronize the data in the RVGs.

Choose an appropriate method, depending on your environment, bandwidth available on your network, the rate at which the application updates, and the size of the data to be replicated.

Volume Replicator provides features to synchronize the data on the Secondary which are as follows:

- Automatic Synchronization
- Block-level backup and Primary checkpoint
- DCM to incrementally synchronize the Secondary

## Using Automatic Synchronization

You can use Automatic Synchronization to transfer the data on the Primary to the Secondary over the network. You can synchronize the Secondary using Automatic Synchronization either when the application is active or inactive. Volume Replicator uses the Data Change Map (DCM) and the network to synchronize the data. This method requires sufficient network bandwidth for Volume Replicator to transfer the data. The Secondary remains inconsistent until the synchronization is complete.

The Automatic Synchronization procedure transfers data in the Primary data volumes to the Secondary by reading the Primary data volumes from start to finish and sending the data to the Secondary. If there are multiple updates to the same block, only the last is sent, reducing the load on the network. To use Automatic Synchronization successfully, the network must be sized appropriately.

**Note:** Note that the synchronization is complete only if the Primary receives writes at a lesser rate than they can be sent to the Secondary.

If the Primary receives writes at a faster rate than they can be sent to the Secondary, the synchronization might never complete, especially if the writes are dispersed widely in the volume. Depending on the number of volumes and the amount of data that exists, the Automatic Synchronization can take a long time to complete.

### Performing intelligent synchronization

Although large volume sizes may be one of the important criteria in determining the time that is taken for Automatic Synchronization to complete, in many cases only about 50 percent of the volumes are used.

This results in the synchronization process sending unused blocks to the Secondary, therefore taking a longer time to complete and causing an overhead on the network bandwidth.

The SwiftSync feature enables Volume Replicator to perform intelligent synchronization by replicating only those blocks that the application uses. In some cases these blocks may just have the file system on them. Because only the used blocks are transferred, the synchronization is much faster and allows for more efficient usage of the available network bandwidth.

**Note:** The SwiftSync feature can be used only for volumes with the NTFS or ReFS file systems.

By default, Volume Replicator performs intelligent synchronization for volumes with NTFS or ReFS file systems, however if required you can choose to disable this feature.

See “Disabling the SwiftSync feature” on page 171.

**Note:** Automatic Synchronization does not maintain the order of writes; therefore, the Secondary is inconsistent until the process is complete. The Secondary becomes consistent after the Automatic Synchronization completes.

## Using incremental synchronization after log overflow

You can incrementally synchronize the Secondary using the Replicator Log overflow protection feature. To enable Replicator Log overflow protection for a Secondary, you can set the log overflow protection for the corresponding Secondary to DCM or AutoDCM. Each data volume in the RVG must have a DCM log associated with it.

If the Replicator Log volume overflows and log overflow protection is set to DCM or AutoDCM, the Secondary does not need to be synchronized completely before it starts replicating again, because the Secondary can be synchronized incrementally. In this case, the DCM log is used and only the updates that are marked on the DCM after the Replicator Log volume overflows are copied to the Secondary. The Secondary is inconsistent during the period when it is updated from the DCM log.

## Using backup and checkpoint

Using the checkpoint feature, you can synchronize the Secondary using a block-level backup and restore method without interrupting the Primary. The block-level backup can be used to recover the Secondary data in case of data volume failure.

See “Understanding checkpoints” on page 56.

This method is useful for low-bandwidth networks or very large data sets. When you use checkpoints, you take backup of the data on the Primary and physically ship the backup media to the Secondary location, and restore the backup on the Secondary. When you start the backup, mark the starting point, by using the checkstart operation on the Primary. When you end the backup, mark the ending point by using the checkend operation on the Primary. While the backup and restore is going on, updates are written to the Replicator Log volume.

To bring the Secondary data up-to-date, restore the block-level backup. After the restore is complete, start replication to the Secondary with checkpoint using the same checkpoint name that you had specified for the checkstart operation on the Primary.

The advantage of this method is that data on the Secondary is inconsistent for a shorter period although there is a risk that the Replicator Log volume may overflow.

**Note:** The Secondary can be brought up-to-date only if the updates are still present in the Replicator Log volume. Using checkpoints is a multi-step process and therefore, needs to be done very carefully.

# Understanding Volume Replicator support for FlashSnap

The FlashSnap feature available with Storage Foundation enables you to perform off-host operations on volumes by creating independent mirrors of volumes on the server.

FlashSnap comprises of a multi-step process that can include the following operations:

Prepare

Creates a snapshot mirror of the volumes.

The Prepare command replaces the Snap Start command in the GUI. Both `prepare` and `snapstart` keywords are available in the CLI, however `prepare` is the recommended keyword.

Snapshot

Create snapshot volumes by breaking off the mirrors.

Disk group split

Forms a new disk group using these snapshot volumes which can be used for off-host processing.

For detailed steps on creating the snapshots for off-host processing, refer to the steps that are described in the section "Off-Host FlashSnap Procedure (Two Servers)" in the *Storage Foundation Administrator’s Guide*.

**Note:** For creating a snapshot, you must use the `vxrvg snapshot` without the `-f` option to create disk group split friendly snapshots.

See “Conditions for creating disk group split friendly snapshots” on page 63.

Disk group join

Joins the new disk group back to the original disk group once the off-host processing is done.

Snapback

Reattaches the snapshot volumes back to the original volume.

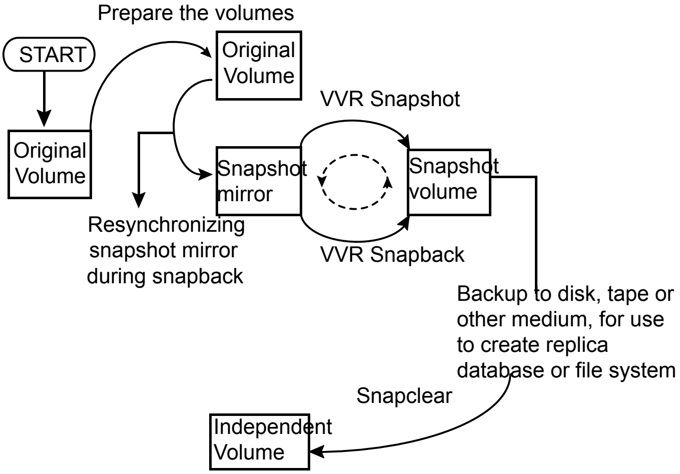
**Note:** A valid license for Storage Foundation FlashSnap feature must be present on all the systems on which you want to use the snapshot operations.

For more information about the FlashSnap feature refer to the *Storage Foundation Administrator’s Guide*.

The need for Volume Replicator to support FlashSnap arises from the fact that if the snapshot volume is created on a disk that is a part of an RVG, then, splitting the disk group with this snapshot volume is not allowed as it breaks the Volume Replicator configuration.

Now as a part of the FlashSnap support, Volume Replicator supports RVG-wide snapshot and snapback operations. This can be performed on the Primary as well as the Secondary RVGs in an RDS. Volume Replicator ensures that only disk group split-friendly snapshots are created.

**Figure 2-4** Working of the snapshot and snapback operations



The data in the original volume may change, however, the snapshot can still be used as a stable and independent copy for various purposes. The snapshots can be used as backup copies to restore the data that may have been lost due to disk failure, software, or human errors. You can perform system backup, upgrade, and other maintenance tasks on point-in-time copies, while providing continuous availability of your critical data. A volume snapshot is also used to execute offline backups without affecting the application performance. They can also be used for restoring data both on the Primary and Secondary, if the original data gets corrupted due to logical or media errors. The snapshot volumes can be replicated and can also be included as a part of the RVG.

**Note:** While the snapshot volume is a part of the RVG it cannot be used for recovery as a consistent point-in-time copy of the data.

Another important advantage of the Volume Replicator snapshot operation is that it supports an RVG friendly disk group split operation. It ensures that the resultant snapshot volume lie on the disks that are not under an RVG, that is, the disks that do not contain any plex of a replicated volume. Thus, a disk group split operation on the snapshot volume(s) keeps the existing Volume Replicator configuration intact and does not fail because the Volume Replicator configuration was disturbed.

**Note:** If the snapshot volumes lie on disks within an RVG, the Volume Replicator snapshot operation fails, provided the force option is not used.

For example, consider the following scenario:

A disk group `dg1` has two disks `disk1` and `disk2`. An RVG with two data volumes and a Replicator Log is created in this disk group. Both the data volumes reside on `disk1` while the Replicator Log is on `disk2`. The two data volumes are prepared and the prepared plexes lie on `disk2`. In this scenario the Volume Replicator snapshots fails (provided force option has not been used) because the `disk2` on which the snapshot volumes need to be created is a part of an RVG, as it contains the Replicator Log of the RVG.

# About the snapshot operation

Using the snapshot feature, you can create the snapshots of all the data volumes in the RVG at a single point-in-time by breaking off the mirrors from the data volumes. These snapshots are a copy of the data at a single point-in-time. Therefore, if the snapshot for one of the volumes fails, the entire snapshot operation fails.

You can create the snapshots with appropriate prefixes so that they can be identified easily. This is especially useful if you want to reattach the snapshot volume back to its original volume using the snapback operation. If the volumes have multiple snapshots, you can choose the snapshots that need to be reattached with the help of their prefixes.

Before creating snapshots, ensure that every data volume in the RVG has a snapshot mirror associated with it. This can be done using the prepare operation. This operation creates and attaches a snapshot mirror (prepared plex) to the original volume and automatically synchronizes the mirror with the original volume. Only after the resynchronization is complete are the prepared plexes ready for snapshot operations.

**Note:** Trying to create snapshots using the prepared plexes when the resynchronization of these plexes is still in progress fails the snapshots.

For information about using the Volume Replicator snapshot operation from the graphical user interface, See “Creating snapshots for the data volumes” on page 182.

For information about using the Volume Replicator snapshot operation from the command line, See “Creating snapshots for data volumes in an RVG” on page 283.

## Conditions for creating disk group split friendly snapshots

For successful Volume Replicator snapshot operation on an RVG, it is required that each data volume in this RVG is prepared and the prepared plex satisfies the condition for disk group split friendly snapshots. For creating disk group split friendly snapshots, the prepared plex must lie on a disk that does not contain any type of plex belonging to data volume or the Replicator Log of any RVG with the exception of the prepared plexes of the data volumes of this RVG (RVG on which the snapshot operation is carried out).

If the prepared plex has been appropriately created for each data volume in the RVG, the Volume Replicator snapshot operation will snapshot each data volume using these plexes. If the operation cannot find such a plex, it fails with a summary report, which details the name of the data volume and the prepared plex which could not satisfy the condition and the disks on which the plexes lie.

### Forcing the snapshot operation

If each data volume in the RVG has a prepared plex that is associated with it then you can force the snapshot operation for that RVG, even if snapshot ready plexes do not satisfy the requirements for RVG friendly disk group split operation. The snapshot operation completes successfully irrespective of whether the conditions that are mentioned are satisfied. Even if the snapshots are successfully created, performing a subsequent disk group split operation may not succeed when the force option is used.

## About the snapback operation

Volume Replicator snapback operation reattaches the plexes of the snapshot volumes back to the original data volumes in the RVG. Even if the snapback operation fails for one or more volumes, it continues to snapback the remaining volumes. This is unlike the snapshot operation. After the operation completes successfully for some of the volumes, appropriate error messages with names of the volumes for which the snapback operation failed along with the reasons for the failure, are displayed.

The default action of the snapback operation is to resynchronize the snapshot plex with the contents of the original volume. However, if the data on the original volume becomes unavailable due to corruption or some software error, you need to recover the lost data. This can be done by performing the snapback operation with the option of resynchronizing the original volume with the contents from the snapshot plex.

For information about using the Volume Replicator snapback operation from the graphical user interface, See “Reattaching the snapshots back to the original volumes” on page 183.

For information about using the Volume Replicator snapback operation from the command line, See “Reattaching the snapshot volumes back to the data volumes in an RVG” on page 284.

Creating snapshots works on any type of file system and should be used when a point-in-time copy of volume is required. Otherwise, you can also create the data volumes with mirrors and break-off the mirrors to create a separate volume which is a point-in-time copy of the data.

Refer to the *Storage Foundation Administrator's Guide*.

# About Synchronized Snapshots

Storage Foundation (SFW) FlashSnap feature integrates with the Microsoft Volume Shadow Copy Service (VSS) to provide support for taking snapshots of Microsoft Exchange storage groups and SQL Server databases. This feature creates snapshots of all volumes that are associated with an Exchange storage group without taking the storage group's databases offline or disrupting the email flow. Similarly, it takes snapshots of all SQL database volumes, without taking the database offline. Volume Replicator leverages the SFW capability to take component snapshots and integrate it with the IBC messaging to create synchronized snapshots of the Exchange storage group and SQL database component on the Primary and Secondary. The synchronized snapshot on the Secondary can then be used to recover the data up to a certain consistent point-in-time quickly, in the case of a disaster at the Primary.

## How Volume Replicator creates synchronized snapshots

The VSS snapshot utility creates snapshots (snapshot set) of all or specified volumes in the Exchange storage group or SQL database component. You can take the snapshots even when the application accesses these volumes.

For Volume Replicator to be able to associate the volumes in a storage group or a database with an RVG, ensure that the following conditions are as satisfied:

- A separate RVG is created for each Exchange storage group or SQL database.
- All the volumes in a storage group or the database are grouped under the same RVG.

Before you take a snapshot, the volumes in the required storage group on the Primary and the Secondary hosts must be prepared for the operation. The VSS snapshot operation uses the VSS service to quiesce the application and take a snapshot, after which it resumes the application. Before you resume the application, it sends an IBC message to the Secondary. The Secondary host is programmed to check for IBC messages at preset intervals, so that it can receive the IBC when it arrives.

IBC messages are typically used to ensure application-level consistency within an RVG. When the IBC arrives on the Secondary, it reads the message and freezes the replication so that the volumes do not change. The Secondary then completes the snapshot operation based on the information it has received through the IBC message.

The synchronous snapshots are initiated on the Primary and then on the Secondary at the same point of data consistency. An XML file containing the information about the volume metadata is maintained on the Primary and is used when while you reattach the snapshots.

You can either use `vxsnap`, the command line option, or the VSS Snapshot wizard to create the required synchronous snapshots.

For more information using the `vxsnap` command, See “Creating Synchronized Snapshots” on page 294.

For information about using the VSS wizard, See “Creating synchronized snapshots using the VSS Snapshot wizard ” on page 184.

Volume Replicator also provides a VSS Snapshot Scheduler wizard that enables you to set up a schedule for automating the snapback refresh process for synchronized snapshots. At the scheduled time for the snapshot, the snapshot volumes are automatically reattached, resynchronized, and then split again. The `VxSchedService.exe` scheduler service that maintains the schedule runs in the background.

For more information about using the VSS Snapshot Scheduler wizard, See “Creating schedules for synchronized snapshots” on page 188.

# Understanding Bunker replication

Volume Replicator supports different modes of replication; synchronous and asynchronous. You can use these modes of replication to obtain a complete Disaster Recovery (DR) solution by maintaining additional synchronous Secondaries at a location closer to the Primary.

The synchronous mode of replication enables you to replicate data to an additional Secondary DR site that is located closer to the Primary. That is, in the case of a disaster at the Primary site, it should be possible to start business from the Secondary site without any loss of data, using the synchronous additional Secondary. However, if the additional Secondary is at least 300 miles away from the Primary site, there may be some network write latency, which degrades the input or output performance of the application. In this case the Recovery Time Objective (RTO) depends on the amount of time you need to recover. For example, if the data needs 2 hours to recover, the RTO is 2 hours. In addition, you also have the overhead of maintaining an additional Secondary site. The asynchronous mode of replication does not incur network write latency. During normal operations, the data on the additional Secondary site may not be up-to-date. If a disaster occurs, it is possible that some of the data may not be available at the disaster recovery site and thus zero RPO is not achieved. Besides, maintaining an additional Secondary can result in additional cost overheads.

## About Bunker replication

Any update is first written to the Replicator Log before it is written to the data volumes. Bunker replication maintains a copy of the Primary Replicator Log on a node at a site close to the Primary that is known as the Bunker node. This copy of the Replicator Log can then be used to bring the Secondary up-to-date if there is a disaster at the Primary site.

## Advantages of Bunker replication

The Bunker node requires additional storage only for the Bunker Replicator Log as it does not contain any data volumes in the RVG. By default, the replication to the Bunker node is performed using synchronous override mode to provide zero RPO.

Bunker replication combines the advantages of synchronous and asynchronous replication, to achieve zero RPO and limited or required RTO, without the overhead of maintaining two complete copies of your data on additional Secondary sites. The Bunker feature also allows the flexibility to choose between RPO or RTO depending on your specific requirements. Ideally, the Bunker Replicator Log should be at a site far away to not be within the same disaster zone as the Primary site, yet close enough to not impede the synchronous update of the Bunker Replicator Log.

## How Bunker replication differs from normal replication

Bunker replication can be performed using an IP network or using direct connectivity to the storage through Fibre Channel (FC) or iSCSI. Thus, when connecting to the storage directly from the Primary, you do not need to maintain a physical host at the Bunker site during normal replication.

When replication is performed over IP, the Primary node sends updates to the Bunker node that is located at a site away from the Primary site. The Bunker node logs the updates to the Bunker Replicator Log. If replication is set directly to the storage that is located at the Bunker site, then the disk group containing the Bunker Replicator Log is imported on the Primary node and the updates to the Primary Replicator Log and the Bunker Replicator Log are performed almost in parallel. This helps to reduce the latency to a minimum and in turn improves performance.

While disaster recovery is the Primary advantage of using Bunker replication, the Bunker replication can also be used as an intermediary location for storing updates if the replication to the Secondary gets disrupted due to non-availability of network bandwidth.

## Bunker node workflow during normal operations

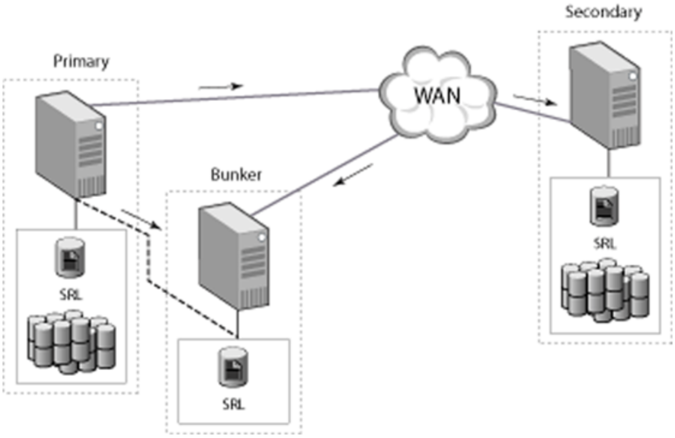
Under normal operating conditions, application writes are logged to the Primary Replicator Log and synchronously replicated to the Bunker node and any other synchronous Secondary sites. By default, the replication to the Bunker node is in the synchronous override mode. Thus, in the case of proper network availability the replication happens in synchronous mode. However, if the network becomes unavailable, replication to the Bunker Secondary happens asynchronously. During normal replication, the Bunker node functions as a Secondary. However, if a disaster occurs at the Primary, the Bunker node must be converted to a Primary and the data in its Replicator Log can be used to bring the Secondary up-to-date.

A write is completed to the application as soon as it is logged to the Primary Replicator Log, the Bunker Replicator Log, and the other synchronous Secondary Replicator Logs. Volume Replicator asynchronously writes the data to the Primary data volume and sends it to the asynchronous Secondary site. When the Secondary acknowledges the writes, the Replicator Log header is updated to indicate the status of the Secondary.

In a typical asynchronous replication setup, the network bandwidth is provisioned for average application write rate. Therefore, in the case of high write rates, the Bunker Replicator Log may contain some writes that are considered complete by the application but are still to be applied to the asynchronous Secondary. The network bandwidth for synchronous replication must therefore be provisioned for peak application write rate. The Replicator Log protection (`srlprot`) for the RLINK between the Primary and Bunker is set to off, by default. If for some reason the Primary replicator overflows for this RLINK, then the RLINK is detached.

See Table 2-4 on page 52.

**Figure 2-5** Bunker setup



# Using the Bunker node for disaster recovery

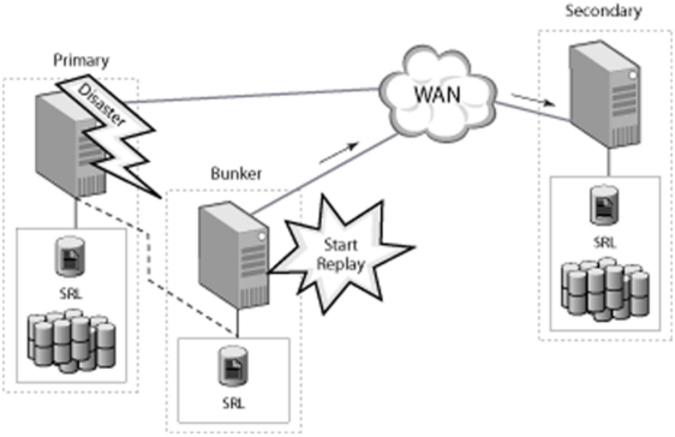
If the Primary site fails, the Secondary needs to take over the role of the Primary. However, the asynchronous Secondary may be behind the Primary. That is, there may be some writes that are completed to the application but have not yet reached the Secondary data volumes; these writes are stored in the Replicator Log on the Bunker node.

To recover from a disaster on the Primary, you can use the Replicator Log on the Bunker node to update the Secondary. If the Bunker storage was directly connected to the Primary when it crashed, then you must import the disk group on the Bunker Secondary. Activate the Bunker and start replication from Bunker node to Secondary.

After all of the pending writes are transferred to the Secondary, the Secondary is as up-to-date as the Primary. The Secondary can take over the role of Primary, with no data loss.

See Table 2-4 on page 52.

**Figure 2-6** The Bunker setup after a failure at the Primary site



Bunker replication enables you to balance the Recovery Point Objective (RPO) with the Recovery Time Objective (RTO) depending on your specific needs. In the case of a disaster, completely replaying the Bunker Replicator Log to the Secondary provides zero RPO. However, if your required RTO is less than the time required to complete replay of data from the Bunker Replicator Log to the Secondary, then you can choose to stop the replay after some time to recover as much data as possible within the required RTO. If the Secondary is far behind the Primary at the time of the disaster, then the time that is required to recover the complete data (RTO) could be large.

Using Bunker replication, you can stop the replay after a period of time to recover as much data as possible within a target RTO. For example, if your Secondary is 2 hours behind the Primary, you can replay the full Bunker Replicator Log to achieve zero RPO but your RTO could then be about 2 hours. If you require an RTO of 1 hour, you could begin Bunker replay and then stop the replay after 1 hour. You can also perform a normal Secondary take over, without replaying the Bunker at all, if you need the application to be immediately available (RTO is zero). In this case, the writes to the Bunker Replicator Log that have not yet been transferred to the Secondary are lost.

**Note:** The Bunker can act as a Secondary to receive updates from the Primary, or it can act as a Primary to send updates to the Secondary during replay. However, it cannot perform both roles at the same time, and therefore, does not serve as a relay between a Primary and another Secondary.

After the Secondary has been updated (either the Bunker replay has completed or the target RTO is reached and the Bunker replay has been stopped), the Secondary can take over the Primary role. If you plan to continue using the new Primary, then the Bunker for the original Primary cannot be used as a Bunker for the new Primary. You must configure another suitable host near the new Primary as a Bunker for the new Primary.

# Understanding Volume Replicator Support for TCP Multi-Connection

To achieve better network throughput, multiple TCP Connections have been introduced with this release of Volume Replicator. TCP does not perform well in Long Fat Networks (LFNs) which has high latency and high-bandwidth. Due to TCP's flow control nature, factors like window size limit, slow recovery from losses, Round-Trip Time (RTT) estimation, and slow start does not allow single TCP connection to saturate the network completely. As a result, optimum network throughput is not achieved when Volume Replicator replicates in the TCP mode.

## Advantages of TCP Multi-Connection

As the Round-Trip Time (RTT) between network grows, the amount of data that can flow across a TCP stream goes down. TCP gets hung up waiting for the acknowledgment (ACK) packets and the transfer rate goes down. One way to handle this is to make use of parallel connections that yield faster throughput for each RLINK. This way rather than waiting for the acknowledgments from a single stream you can have multiple ACKs going across.

Replicating through multiple TCP connection for each RLINK enables the maximum utilization of high latency and high-bandwidth networks. Single TCP connection usually fails to use the entire bandwidth. To enable optimum use of bandwidth available for each RLINK, Volume Replicator establishes multiple TCP connections to the Secondary. Multiple connections improve the overall replicating performance of Volume Replicator.

# About Volume Replicator compression

Compression feature enables Volume Replicator to send data in a compressed form from a Primary to a Secondary host. It reduces network bandwidth consumption by Volume Replicator. This feature is particularly useful in scenarios where there is low availability of bandwidth or where the bandwidth is shared among several applications. Purchasing an external compression software or hardware for data transfer can prove costly. Hence, compression feature in Volume Replicator is a cost-effective alternative in such scenarios.

Compression option can be enabled on a per RLINK basis either through the CLI or GUI. If both sides have compression enabled, the Primary site generates the compressed data during sending of any updates. At any time, the user can disable or enable the compression feature without stopping the replication.

General functionality constraints for Volume Replicator compression are as shown. Data should not be sent in compressed form in the following cases:

- If either the Primary or Secondary RLINK does not have compression enabled
- If the compressed data size is greater than the uncompressed data size
- If the memory for keeping the compressed data could not be allocated on the Primary side

# About Volume Replicator memory monitoring and control support

This feature enables Volume Replicator to monitor and control Non-Paged Pool memory (NPP) usage.

During replication, Volume Replicator uses the NPP memory for the following operations:

- Stabilizing an incoming write from an application

  Volume Replicator makes a copy of the application writes in VOLIOMEM pool as soon as a new write arrives. The pool gets memory from the Non-Paged Pool system memory.
- Reading back the data from Replicator Log or data volumes

  Volume Replicator may read back the data from the Replicator Log (in case of behind Secondary) or from the data volumes (in case of DCM mode replication) to send that data to Secondary. Buffer for both of these scenarios is allocated from the READBACK memory pool. The pool gets memory from the Non-Paged Pool system memory.
- Holding the incoming updates on Secondary

  Volume Replicator on Secondary stores the incoming writes from Primary in NMCOM pool. The pool gets memory from the Non-Paged Pool system memory. Among the three pools that are described, the VOLIOMEM pool is used by SFW as well as for serving mirrored volume writes and few other operations. Volume Replicator exclusively maintains and uses the READBACK and NMCOM pools.

## Advantages of memory monitoring

Non-Paged Pool (NPP) memory may get depleted due to large consumption of the memory pool by Volume Replicator. This is especially true for customers having a /3GB switch and running either a Microsot SQL or Exchange Server. When the NPP memory gets depleted, the application either starts giving error or stops responding.

# About Volume Replicator Graphs

Volume Replicator Graphs are used for displaying statistical information in the VEA GUI.

Volume Replicator Graphs display the following statistics:

- The bandwidth that each RLINK in an RDS uses

  For bandwidth usage, a graph of the data sent per second kilobits (Kb) is plotted against time. This graph is updated every five seconds. The bandwidth limit that is set on the RLINK is also displayed for every rlink graph. Bandwidth usage can be monitored both in the Online as well as the Historic mode. The graph file can be saved as a CSV or PNG file.

- The Non-Paged Pool (NPP) memory usage by SFW

  Volume Replicator and SFW use the VOLIOMEM, NMCOM, and READBACK memory pools. The NPP usage graph plots the allocated and max values for each of these three pools. This graph gets updated every 5 seconds and displays the memory usage in kilobytes (KB) against time.

See “Obtaining statistical information through Volume Replicator Graphs” on page 222.

## General functionality constraints for Volume Replicator Graphs in a clustered environment

In a multinode Primary cluster, if historic data collection is enabled on an RLINK and the storage group is moved to another node, you may need to explicitly Start Historic Data Collection on the new node. Also, it would not be possible to merge the collected data on the old and the new node.

Chapter 3

# Setting up replication

This chapter includes the following topics:

## About setting up replication

This chapter guides you through the process for setting up an RDS, which is the most important step to get replication started. Data is replicated from a Primary node, where the application is running, to one or more Secondary nodes. The primary node and the secondary node could be present in an on-premise or an on-cloud data center. An RVG on the Primary node, and the corresponding RVG on the Secondary nodes, make up an RDS. Volume Replicator uses the Replicator Log to keep track of pending writes.

You must first set up an RDS and start replication before you can perform any other Volume Replicator operations. After setting up an RDS you may want to perform other tasks such as monitoring the replication and changing configuration settings.

**Note:** If your deployment involves setting up replication in over cloud, then you must first set up the cloud infrastructure and then begin with the replication setup.

See “Best practices for setting up replication” on page 75.

See “About monitoring replication” on page 114.

See “About administering Volume Replicator” on page 146.

Most of the tasks that have been discussed in this chapter and in the following chapters can be performed using the VEA or from the command line interface.

See “About using the command line interface” on page 227.

**Note:** Within this document, any reference to a Secondary node implies all the Secondary nodes. Thus, for the operations that need to be performed on the Secondary, it is implied for all the Secondary hosts, unless otherwise specified.

# Best practices for setting up replication

Certain best practices can be used when setting up replication.

**Note:** The Volume Replicator Advisor (VRAdvisor), a tool to collect and analyze samples of data, can help you determine the optimal size of the Replicator Log.

Some best practices regarding setting up replication are as follows:

- Create the Primary data volumes with drive letters. Plan the size and layout of the data volumes based on the requirement of your application. You must configure the Primary and Secondary data volumes with the same name.
- If you create the Replicator Log volume manually, then make sure that you do not assign a drive letter to the Replicator Log volume. Veritas recommends that you create the Replicator Log volume as a mirrored volume. For better performance, Replicator Log should be a mirrored-striped volume.
- To improve write performance configure the data volumes and the Replicator Log volume on different physical disks.
- Veritas recommends that you create a Replicator Log volume of the same size on the Primary and Secondary. Size the Replicator Log appropriately to avoid overflow.
- As Volume Replicator uses Replicator Log volumes, Veritas recommends that you do not format these volumes with any file system. When a volume is assigned for use as a Replicator Log volume, the existing file system and data is lost.
- Plan the bandwidth of the network to be used, based on your requirement.
- If you plan to replicate data over cloud, ensure that you satisfy the following network requirements to establish connectivity between the on-premise and on-cloud networks, or in between the same or different cloud regions:

  - A non-overlapping IP address space for network subnets is available on the primary and the secondary datacenters.
  - A virtual network (VNet) is set up in Microsoft Azure network.
  - The Volume Replicator **DHCP_IP_Support** tunable is set to True. This tunable determines whether or not DHCP IPs can be used while configuring Volume Replicator.

    DHCP IP addressess are assigned to the virtual machines that are created in an Azure environment. To identify the DHCP IP addresses, DHCP_IP tunable must be set to True.

    Run the following command line to enable the tunable:

    ```
    vxtune dhcp_ip_support true
    ```
  - To set up replication from an on-premise datacenter to an on-cloud datacenter, a virtual private network (VPN) gateway and a local network gateway is set up in the cloud network. This enables to set up a communication tunnel between the machines that are located in the on-premise datacenter to the machines that are located in the on-cloud datacenter.

    Alternatively, you may configure Microsoft ExpressRoute to establish a connection between datacenters.

    For more details refer to Microsoft documentation.
  - To set up replication between any two regions on cloud, a virtual private network (VPN) gateway is set up over the virtual network, and a connection is established between the VPN gateways for the virtual machines to communicate with each other.
  - To set up replication within a same region on cloud, a VNet Peering is set up between the two virtual networks in which the machines are configured.

    Alternatively, you may set up a VPN gateway between the two virtual networks.

**Note:** While provisioning storage for setting up replication in Microsoft Azure environment, do not configure multiple network paths to the storage disks. Microsoft Azure does not support storage configuration with multiple network paths.

- You can choose to use either the UDP, TCP, or STORAGE protocol for network communication between the Primary and Secondary during replication. You can use the STORAGE protocol only for a Bunker Secondary if the storage on the Bunker Secondary is visible from the Primary host.

- Make sure that the Secondary volumes are not formatted and no other application uses these volumes. If you use these volumes for replication, all the original data on these volumes is lost.
- Avoid using RDS and RVG names with special characters with the exception of the hyphen (-) or an underscore (_). The names can have a maximum of 31 characters and can begin with a hyphen (-) or underscore (_) character.
- If you want to replicate encapsulated volumes, ensure that the volume names and sizes on the Secondary are the same as the corresponding volume names and sizes on the Primary. The volume name is auto generated while encapsulating the volume.
- Enable the ports and services that are used for inbound and outbound communication.
  For a list of required ports and services,
  See "InfoScale ports and services" on page 425.

# Security considerations for Volume Replicator

Volume Replicator operations can be performed directly from the VEA or using the CLI. To understand the concept of the local (originator) host and remote (target) host better, you need to understand how the configuration-specific operations are performed. Using VEA, any operation on a Volume Replicator host to which the VEA is not directly connected, is performed as a remote operation.

You can perform the operations on various Volume Replicator objects which include RVG, RDS, replicated volumes, and the RLINKs (Secondaries). Some Volume Replicator operations involve more than one host as a part of their operations. Before you execute such an operation, Volume Replicator first validates whether the originator host is allowed to execute the specified operation on the target hosts. If not, the specified operation fails. This validation process is referred to as the security check. Local operations for the hosts that are connected to VEA require local administrative privileges.

## Configuring the VxSAS Service

Volume Replicator provides you with a VxSAS Wizard that enables you to configure the VxSAS service across multiple hosts at the same time. Many of the Volume Replicator commands require the VxSAS service logon account to be the same across different hosts, for the commands to run successfully. This wizard enables you to configure the same user name and password for the VxSAS service on multiple hosts, with ease. This wizard can also configure the VxSAS service logon account for all the hosts in a Cluster Server (VCS) or Microsoft Cluster as a group.

This means that if a single node in a cluster is selected then all the nodes that are part of that cluster gets selected automatically.

Certain considerations should be taken into account before configuration of VxSAS.

They are as follows:

- For a VCS cluster setup, the hosts are displayed as a part of the VCS cluster setup on the **Host Selection** panel, if the local host on which you have invoked the VxSAS wizard belongs to the VCS cluster.
  If the local host on which you have invoked the VxSAS wizard and the secure remote cluster, are part of the same domain.
- For any other VCS cluster, if it is configured as a non-secure cluster, then that cluster does not show up in the **Host Selection** panel and the hosts under the cluster are shown as independent hosts.
- If you have a Microsoft Cluster setup, then, the host display may not indicate that it is a part of a cluster if:
  The host on which you invoke VxSAS is not a part of the same domain as the Microsoft Cluster nodes.
  You have not logged on as domain administrator on the host from which the VxSAS service configuration wizard is invoked.
  The host on which the VxSAS service configuration wizard is invoked is not part of the same subnet on which the cluster nodes are present.

---

**Note:** If you have chosen Japanese as the language of installation, the VxSAS wizard does not launch automatically, after the first restart post-installation. In this case, you must launch the wizard manually.

---

**To configure the VxSAS service**

1. Launch the VVR Security Service Configuration Wizard manually from **Start > All Programs > Veritas > Veritas Storage Foundation > Configuration Wizards > VVR Security Service Configuration Wizard** or, on Windows 2012 operating systems, from the **Apps** menu in the **Start** screen.

   Optionally, run `vxsascfg.exe` from the command prompt.

2. The **Welcome** panel appears.

   This panel displays important information that is useful as you configure the VxSAS service. Read the information that is provided on the **Welcome** panel and click **Next**.

3 The **Account Information** panel appears.

Complete this panel as follows:

**Account name (domain\account)** Enter the administrative account name in the **Account name** field.

**Password** Specify a password in the **Password** field.

If you have already configured the VxSAS service for one host that is intended to be a part of the RDS, then make sure that you specify the same user name and password when configuring the VxSAS service on the other hosts.

After providing the required information, click **Next**.

4 Select the required domain to which the hosts that you want to configure belong, from the **Domain Selection** panel.

**Selecting Domains** The **Available Domains** pane lists all the domains that are present in the Windows network neighborhood.

Select the required domain by moving the appropriate name from the **Available Domains** pane to the **Selected Domains** pane, either by double-clicking it or using the arrow button.

**Adding a Domain** If the domain name that you require is not displayed, then add it by using the **Add Domain** button. This displays a dialog that lets you specify the domain name. Click **Add** to add the name to the **Selected Domains** list.

After specifying the domain click **Next**.

5 Select the required hosts from the **Host Selection** panel.

Complete this panel as follows:

| | |
|---|---|
| **Selecting Hosts** | The **Available Hosts** pane lists the hosts that are present in the specified domain.<br>Select the required host by moving the appropriate name from the **Available Hosts** list to the **Selected Hosts** list, either by double-clicking it or using the arrow button. Use the Shift key with the up or down arrow keys to select multiple hosts. |
| **Adding a Host** | If the host name you require is not displayed, then add it using **Add Host** option. In the **Add Host** dialog specify the required host name or IP in the **Host Name** field. Click **Add** to add the name to the **Selected Hosts** list. |

After you select the host name, click **Configure** to proceed with configuring the VxSAS service.

6 After the configuration completes, the **Configuration Results** panel is displayed. If the operation is successful then the **Status** column displays the appropriate message to indicate that the operation was successful.

If the operation was not successful then the **Status** column displays the appropriate message along with the error message.

- This panel displays the status as failed and the corresponding details on why the account update failed. It also displays the possible reasons for failure and recommendations on getting over the failure.
  Click **Back** to change any information you had provided earlier.
- When you configure the VxSAS service for Volume Replicator in a firewall setup, the VxSAS wizard may not be able to configure the computers that are across the firewall, although the **Host Selection** dialog may list these nodes. In this case, configure the VxSAS service locally on the computers that are across the firewall.
- Click **Finish** to exit the wizard.

# Enabling NAT support for Volume Replicator

Network Address Translation (NAT) involves translating the Internet Protocol address (IP address) used within one network to an IP address that is known within another network.

To enable Volume Replicator in a NAT setup use the host name for configuring Volume Replicator by adding the host name and its NAT address to the `hosts` file, only if the host uses a NAT. If the Primary uses NAT you need to make the corresponding entry for the Primary host name and its NAT address on the Secondary, but you do not need to do this on the Primary, if the Secondary IP is visible from the Primary. However, if the Secondary also uses NAT then you must make an entry for the Secondary host name and its NAT address, on the Primary.

For the hosts within the Volume Replicator configuration, if even one of the hosts in the RDS is under a NAT, then it is best to set up NAT support for all the hosts.

See "Tuning Volume Replicator" on page 312.

For example, if both the Primary and Secondary are under a NAT setup, perform the following tasks in the given order, to ensure that replication happens correctly:

- On the Primary, add an entry for the Secondary host name and its NAT IP address in the hosts file present at the following location:

  ```
  <systemroot>\system32\drivers\etc\hosts
  ```

- On the Secondary, add an entry for the Primary host name and its NAT IP address in the hosts file present in the following location:

  ```
  <systemroot>\system32\drivers\etc\hosts
  ```

- While creating the RDS, use the host names instead of IP addresses. This automatically maps to the NAT IP address using the entries in the hosts file. The replication is now enabled across NAT.

# Setting up replication using the Setup Replicated Data Set wizard

You can configure and set up the replication by performing certain tasks.

To configure and set up replication, the tasks should be performed in the following order:

- Create the Primary RVG.
- Add a Secondary to the RVG.
- Synchronize the Secondary and start Replication

You can set up an RDS on the Primary host and one Secondary host, using the Setup Replicated Data Set Wizard. You can add more Secondaries later, using the Add Secondary wizard.

The Setup Replicated Data Set wizard requires only the disk group with the data volumes to be created on the Primary host. This wizard enables you to create the

Replicator Log volume for the Primary as you create the RDS. It can also create the same configuration on the Secondary host. However, if you have created the required disk group, the data volumes and the Replicator Log volumes on all the hosts beforehand, then the wizard proceeds with creating RDS without displaying options for creating volumes.

Volume Replicator also provides some advanced options that enable you to specify some additional replication properties. The following sections discuss these properties.

## Prerequisites for setting up the RDS

Before creating an RDS, check whether your setup meets the following prerequisites:

- Verify that the intended Primary host is connected to VEA, if you configure the RDS from a remote client or from a host that is not the Primary.
- Verify that you set the IP preference, whether Volume Replicator should use IPv4 or IPv6 addresses, before you configure replication. The default setting is IPv4.

  When you specify host names while you configuring replication, Volume Replicator resolves the host names with the IP addresses associated with them. This setting determines which IP protocol Volume Replicator uses to resolve the host names.

  Use Veritas Enterprise Administrator (VEA) (Control Panel > VVR Configuration > IP Settings tab) to set the IP preference.
- Verify that the data volumes and Replicator Log volume that intended to be a part of the RDS are not of the following types, as Volume Replicator does not support the following types of volumes:
  - Storage Foundation (software) RAID-5 volumes
  - Volumes with the Dirty Region Log (DRL)
  - Volumes with a comma in their names
  - Secondary volume of a size smaller or greater than that on the Primary
  - Volume that is under replication

    For the Replicator Log volume, in addition to these types, make sure that the volume does not have a DCM.

# Creating a Replicated Data Set (RDS)

You can create the Replicated Data Set (RDS) in the following way.

**To create the replicated data set**

1. In the tree in the left pane, right-click the **Replication Network** node and select **Setup Replicated Data Set**.
2. Read the information about the **Welcome** panel and click **Next**.
3. Complete the Enter names for the Replicated Data Set and Replicated Volume Group panel as follows:

| | |
|---|---|
| Replicated Data Set name | Enter a name for the RDS. |
| Replicated Volume Group name | Enter a name for the RVG. The same name is used for the Primary and Secondary RVG. |
| Primary Host | By default the local host is selected. To specify a different host name, make sure that the required host is connected to the VEA console and select it in the **Primary Host** list. If the required host is not connected to the VEA, it does not appear in the list. In that case, use the VEA console to connect to the host. |

   Click **Next**.

4. Select the dynamic disk group and volumes to be replicated as follows.

| | |
|---|---|
| Dynamic Disk Group | Select the appropriate dynamic disk group from the list. Multiple disk groups cannot be added in an RDS. |
| Select Volumes | Choose the required data volumes from the table by selecting the check boxes for the volumes. To select all the volumes, select the check box in the top left corner of the Select Volumes table.<br>To select multiple volumes, press the Shift or Control key while using the Up or Down arrow key.<br>By default, adds DCM logs with mirrored plexes for all selected volumes. If the disk space is inadequate for creating a DCM with mirrored plexes, a single plex is created. |

   Click **Next**.

5. Complete the Select or create a volume for Replicator Log panel by choosing one of the following:
   - To select an existing volume, select the volume in the table and click **Next**.
   - If you have not created a volume for the Replicator Log or want to create a new one, click **Create Volume**. Complete the information about the **Create Volume** dialog box as follows:

| | |
|---|---|
| Name | Enter a name for the volume. |
| Size | Enter a size for the volume. |
| Layout | Select the appropriate volume layout. |
| Disks Selection | If you want Volume Replicator to select the disks for the Replicator Log, choose **Select disks automatically**.<br>If you want to choose specific disks from the Available disks pane for the Replicator Log, choose **Select disks manually**. Either double-click on the disks or click **Add** to move the disks into the Selected disks pane. |

Click **OK**. The volume is created and displayed in the **Replicator Log** panel. Click **Next**. The summary panel appears.

6 Review the information on the summary panel. Click **Back** if you want to change any information.

Click **Create Primary RVG** to create the RVG.

7 After the Primary RVG has been created successfully, Volume Replicator displays the following message:

```
RDS with Primary RVG has been created successfully. Do you want
 to add Secondary host to this RDS for replication now?
```

8 On the **Specify Secondary host for replication** panel, enter the name or IP address of the Secondary host. Click **Next**. If the Secondary host is not connected to VEA, the wizard tries to connect it when you click **Next**. Wait till the connection process is complete and then click **Next** again.

See "Adding a Secondary host" on page 153.

- If the disk group with the required data volumes and the Replicator Log volume as on the Primary host does not exist on the Secondary, Volume Replicator displays a message. Read the message carefully.
  - The option to automatically create the disk group and the associated volumes on the Secondary host is available only if the required number of disks of the same type, having the same or a larger amount of space as on the Primary, are available on the Secondary. Otherwise, the wizard enables you to create the disk group and the volumes manually.
  - Click **Yes** to automatically create the disk group, data volumes, and Replicator Log. Any available disks are automatically chosen for creating the disk group on the Secondary host.

- Click **No** to manually create the disk group, data volumes, and Replicator Log. Complete the Create Dynamic Disk Group on Secondary host panel. If the dynamic disk group as on the Primary has already been created on the Secondary, then this panel does not appear.

  Complete the information on this panel as follows:

  | | |
  |---|---|
  | Create cluster group | Choose this option only if you need to create clustered disk groups. Select the required disks from the Available disks pane. Either double-click on the disks or click **Add** to move the disks into the Selected disks pane. To select all the available disks, choose the **Add All** option. |
  | Create Dynamic Disk Group | Click **Create Dynamic Disk Group** to proceed with creating the disk group. A disk group with the same name as that on the Primary is created. |

  After the disk group has been created, click **Next**. The Volume Information on connected hosts panel appears.

  Complete this panel as described in step 9.

- If a disk group, without any data volumes or Replicator Log, as on the Primary host exists on the Secondary, Volume Replicator displays a message. Read the message carefully.
  - The option to automatically create the volumes on the Secondary host is available only as follows: If the disks that are part of the disk group have the same or a larger amount of space as on the Primary and enough space to create volumes with the same layout as on the Primary. Otherwise, the wizard enables you to create the required volumes manually.
  - Click **Yes** to automatically create the data volumes and the Replicator Log.

    After the configuration has been automatically created on the Secondary, proceed to step 10.
  - Click **No** to create the data volumes and the Replicator Log manually, using the Volume Information on connected hosts panel.

**9** The Volume Information on connected hosts panel displays information about the availability of volumes on the Secondary nodes, if the Primary and Secondary hosts are connected to the VEA.

This panel does not appear if all the required volumes that are available on the Primary host are also available on the Secondary hosts.

If the required data volumes or the Replicator Log volume have not been created on the Secondary host, the panel displays the appropriate message against the volume name on the Secondary. Create the required volumes as follows:

- For each required volume that is not created, click **Create Volume**.
- The Create Volume dialog verifies the information about the Primary host and displays the volume name and the size.
  Complete the information on this panel as follows:

  | | |
  |---|---|
  | Name | Displays the name that is specified for the Primary volume. |
  | Size | Displays the size that is specified for the primary volume. |
  | Layout | Lets you specify the volume layout. Select the appropriate volume layout depending on your requirement. |
  | Disks Selection | Enables you to specify the disk selection method.<br>Select the **Select disks automatically** option if you want Volume Replicator to select the disks.<br>Select the **Select disks manually** option to use specific disks from the Available disks pane for creating the volume. Either double-click on it or select **Add** to move the disks into the Selected disks pane. |

  Click **OK** to create the required volume.
- Repeat the steps for each of the data volumes and Replicator Log that has not been created.
- After all volumes are created, the volume information panel is updated. Click **Next**.

If the required volumes are created but are not eligible for replication, the reason for non-eligibility is indicated against the volume name.

See “Setting up replication using the Setup Replicated Data Set wizard” on page 81.

The Volume Information on connected hosts panel enables the appropriate option to convert a non-eligible volume to a Volume Replicator acceptable format.

Complete the information on this panel as follows:

Recreate Volume

This option is enabled if the required data volume is available on the Secondary, but is of a size greater than the Primary volume.

Clicking this option displays a message that prompts you to confirm that you want to recreate the volume.

**Warning:** This operation first deletes the volume resulting in loss of the data that already exists on the volumes.

Choose **Yes** to recreate the volume using the Create Volume dialog.

Remove DRL

This option is enabled if the required data volume is available on the Secondary but has a DRL. Clicking this option displays a message that prompts you to confirm that you want to remove the log. Click **Yes** to confirm the removal of DRL.

Remove DCM

This option is enabled if the required Replicator Log volume is available on the Secondary but has a DCM log. Clicking this option displays a message that prompts you to confirm if you want to remove the log. Click **Yes** to confirm the removal of the DCM log.

Expand Volume

This option is enabled if the required data volume is available on the Secondary but is of a smaller size than the Primary volume. Clicking this option displays a message that prompts you to confirm that you want to grow the volume.

Click **Yes** to grow the volume to the required size.

After you have converted the non-eligible volumes to a Volume Replicator acceptable format, click **Next**.

If the volume on the Secondary is already a part of another RDS, the wizard does not let you proceed. If you want to use the same volume, you must either remove the corresponding Primary volume from the Primary RVG or delete the other RDS.

**10** Complete the **Edit replication settings** panel to specify basic and advanced replication settings for a Secondary host as follows:

- To modify the default values for the basic settings, select the required value from the drop-down list for each property, as follows:

  Primary side IP

  Displays the IP address on the Primary that is to be used for replication. If there is more than one IP address available for replication, you can choose the one that you want to use from the drop-down list. If the required IP address is not displayed in the list, edit the field to add the IP address.

Secondary Side IP

Displays the IP address on the Secondary that is to be used for replication, if the Secondary is connected to VEA. If there is more than one IP address available for replication, you can choose the one that you want to use from the drop-down list. If the required IP address is not displayed in the list, edit the field to add the IP address.

See “Changing replication settings for an RDS” on page 173.

Replication Mode

Select the required mode of replication; Synchronous, Asynchronous, or Synchronous Override. The default is synchronous override.

**Note:** If the Secondary is set to the synchronous mode of replication and is disconnected, the Primary data volumes with NTFS or ReFS file systems may be displayed as MISSING.

Replicator Log Protection

The **AutoDCM** is the default mode for the Replicator Log overflow protection when all the volumes in the Primary RVG have a DCM log. The DCM is enabled when the Replicator Log overflows.

The **DCM** option enables the Replicator Log protection for the Secondary host when the Replicator Log overflows, and the connection between the Primary and Secondary is lost. This option is available only if all the data volumes under the Primary RVG have a DCM Log associated with them.

The **Off** option disables Replicator Log Overflow protection.

In the case of the Bunker node. Replicator Log protection is set to Off, by default. Thus, if the Primary RLINK overflows due to the Bunker RLINK, then this RLINK is detached.

The **Override** option enables log protection. If the Secondary node is still connected and the Replicator Log is about to overflow, the writes are stalled until 5% or 20 MB of space (whichever is lesser) becomes available in the Replicator Log.

If the Secondary becomes inactive due to disconnection or administrative action then Replicator Log protection is disabled, and the Replicator Log overflows.

The **Fail** option enables log protection. If the log is about to overflow the writes are stalled until 5% or 20 MB of space (whichever is lesser) becomes available in the Replicator Log. If the connection between Primary and Secondary RVG is broken, then, any new writes to the Primary RVG are failed.

Primary RLINK Name

This option enables you to specify a Primary RLINK name of your choice. If you do not specify any name, Volume Replicator assigns a default name.

Secondary RLINK Name

This option enables you to specify a Secondary RLINK name of your choice. If you do not specify any name, Volume Replicator assigns a default name.

- To proceed without modifying the advanced replication settings, click **Next**. The Start Replication panel appears.
  Proceed to step 11.
- To specify advanced replication settings, click **Advanced**.
  Complete the Advanced Replication Settings panel as follows:

Latency Protection

By default, latency protection is set to Off. When this option is selected the High Mark Value and the Low Mark Value are disabled. Select the Fail or Override option to enable Latency protection.

See "Latency protection—`latencyprot` attribute" on page 52.

The Override option behaves like the Off option when the Secondary is disconnected and behaves like the Fail option when the Secondary is connected.

High Mark Value

This option is enabled only when Latency Protection is set to Override or Fail. It specifies the maximum number of pending updates by which the Secondary can be behind the Primary. The default value is 10000, but you can specify the required limit.

To ensure that latency protection is most effective the difference between the high and low mark values must not be very large.

Low Mark Value

This option is enabled only when Latency Protection is set to Override or Fail. When the updates in the Replicator Log reach the High Mark Value, then the writes to the Primary continue to be stalled until the number of pending updates on the Replicator Log falls back to the Low Mark Value. The default value is 9950, but you can specify the required limit.

Protocol

UDP/IP is the default replication protocol. Choose TCP/IP or UDP/IP for a regular Secondary. If the storage on the Bunker Secondary is directly accessible from the Primary use the STORAGE protocol, otherwise use TCP/IP or UDP/IP.

**Note:** If the replication protocol for the Bunker Secondary has been set to STORAGE, you can change it only by deleting and recreating the Bunker with the required protocol; UDP/IP or TCP/IP. You cannot change the STORAGE protocol using the Change Replication Settings option.

| | |
|---|---|
| Packet Size(Bytes) | Default is 1400. Choose the required packet size for data transfer from the drop-down list. The default unit for the packet size is Bytes. You can set the packet size only if the protocol is UDP/IP.<br>Some firewalls do not support packet sizes greater than 1400 bytes. To replicate across such a firewall, use the default packet size to make sure all the Volume Replicator operations function as required. You can also set the packet size to 1300 by selecting from the list. The minimum packet size that you can specify is 1100 bytes.<br>**Note:** If you need to set a value for packet size different from the value that is provided in the list, use the command line interface.<br>See "About using the command line interface" on page 227. |
| Bandwidth | By default, Volume Replicator uses the maximum available bandwidth.<br>To control the bandwidth thatVolume Replicator uses for replication, choose **Specify Limit**, and then specify the bandwidth limit in the field provided. The default unit is Megabits per second (Mbps) and the minimum allowed value is 1 Mbps. |
| Enable Compression | Enable this option if you want to enable compression for the Secondary host. |

After completing the Advanced Replication Settings panel, click **OK**. The wizard returns to the **Edit Replication Settings** panel. Click **Next**. The **Start Replication** panel appears.

11 Choose the appropriate option from the **Start Replication** panel as follows:

- To add the Secondary and start replication immediately, select the Start Replication with one of the following options:

| | |
|---|---|
| Synchronize Automatically | For an initial setup, use this option to synchronize the Secondary and start the replication. This setting is the default.<br>When this option is selected, Volume Replicator by default performs intelligent synchronization to replicate only those blocks on a volume that the file system uses. If required, you can disable intelligent synchronization.<br>**Note:** Intelligent synchronization is applicable only to volumes with the NTFS and ReFS file systems and not to raw volumes or volumes with FAT file systems. |

| | |
|---|---|
| Synchronize from Checkpoint | If you want to use this method, then you must first create a checkpoint.<br>See "Using backup and checkpoint" on page 60.<br>If the Primary data volumes have a considerable amount of data, you may first want to synchronize the Secondary for existing data using the backup-restore method with checkpoint. After the restore is complete, use the Synchronize from Checkpoint option to start replication from checkpoint; this operation synchronizes the Secondary with the writes that happened when backup-restore was in progress. |

- To add the Secondary without starting replication, deselect the **Start Replication** option. You can start replication later by using the **Start Replication** option from the Secondary RVG right-click menu.
  Click **Next** to display the **Summary** panel.

12 Review the information on the **Summary** panel.

Click **Back** to change any information you had specified or click **Finish** to add the Secondary to the RDS and exit the wizard.

# Setting up the Bunker RVG for replication

You can add a Bunker RVG to an existing RDS without interrupting the replication. Each Bunker node can support replay to one or more Secondaries. Multiple Bunker nodes can be associated with each Primary. A Primary host with multiple Bunker nodes is useful if a disaster occurs on a Bunker node, while replaying to the Secondary. In that case, the second Bunker node can take care of replaying the rest of the pending data to the Secondary. You do not need to have a dedicated network bandwidth between the Bunker node and the Secondary, as the connection is used only during the recovery process after a disaster.

On the Bunker node, create the Bunker RVG with only the Replicator Log volume and no data volumes. Make sure that appropriate RLINKs from the Bunker to the Primary and Secondary nodes, and vice versa, exist.

## Prerequisites for setting up Bunker RVG

There are some pre-requisites that you need to follow before setting up a Bunker RVG.

The pre-requisites are as follows:

- Verify that sufficient storage is available on the Bunker node for creating the Replicator Log.
- Verify that IP connectivity from the Primary to the Bunker node exists.
- Verify that IP connectivity from Bunker to the Secondary node exists.
- Verify that iSCSI or FC connectivity from the Primary to the Bunker storage exists, for a storage Bunker.

## Best practices for creating the Bunker RVG

Certain practices should be followed when you create a Bunker RVG.

Best practices for creating a Bunker RVG are as follows:

- The Bunker RVG must contain only the Replicator Log and no data volumes.
- The Bunker Replicator Log must be of the same size and the same name as the Primary Replicator Log. Adding the Bunker to the RDS fails if the Bunker Replicator Log is not of the same size as the Primary Replicator Log. In the case of a storage Bunker, the Replicator Log name may be different.
- Veritas recommends that you do not replicate to the Bunker using asynchronous mode as the Bunker node may not be up-to-date at all times. By default, replication to the Bunker node is in the synchronous override mode.

## Adding the Bunker RVG to the RDS

This section guides you through the process of creating the Bunker RVG, establishing the required RLINKs and starting replication using the Add Bunker option. You can also do this using the vxrds addBunker command.

**Note:** Adding the Bunker RVG fails if the Replicator Log sizes differ. The Bunker Replicator Log must be of the same size and the same name as the Primary Replicator Log.

See “Adding a Bunker node” on page 235.

You can either choose to create the RVG on the Bunker node through the Add Bunker wizard or using the command line options.

### To create and add a Bunker RVG to an RDS

1. Click on the required RDS under the Replication Network node and select the **Add Bunker** option from the RDS right-click menu.
2. Read the information on the **Welcome** panel of the **Add Bunker** wizard and click **Next**.

   Complete the Specify Bunker Host for Replication panel as follows:

   Bunker Host — Specify the name or IP of the Bunker host in the provided field. Even if the storage on the Bunker host is directly accessible to the Primary, you must still provide the name of the host that you may plan to use if a disaster occurs.

   Add Bunker with Storage protocol — Select this check box only if the storage on the Bunker node is directly accessible from the Primary, that is, the storage is shared between the Primary and Bunker Secondary. Make sure that the disk group which you plan to use for creating the Bunker RVG is imported on the Primary node.

   You can then use the Storage protocol to replicate to the Bunker node across Fibre Channel (FC) or iSCSI.

   Bunker Diskgroup — This option is enabled for selection only if you have selected the Add Bunker with STORAGE protocol. In this case you can choose to have a different disk group name for the Bunker RVG. Otherwise, the same disk group name as on the Primary is used.

   Click **Next**. If the specified host is not connected to VEA, the wizard tries to connect it when you click **Next**. When prompted, enter the connection information in the provided fields. Wait till the connection process is complete and then click **Next** again.

3. If the disk group with the required Replicator Log volume as on the Primary host does not exist on the Bunker Secondary, you can create the disk group and the required volumes through the Create Dynamic Disk Group on Secondary host panel. If the Dynamic Disk group that is the same as that on the Primary has already been created on the Bunker Secondary, then this panel does not appear.

   Complete the Create Dynamic Disk Group on Bunker host panel as follows:

   | | |
   |---|---|
   | Create cluster group | Choose this option only if you need to create a clustered disk group. Select the required disks from the Available disks pane. Either double-click on it or use the Add option to move the disks into the Selected disks pane. To select all the available disks, use the Add All option. |
   | Create Dynamic Disk Group | Click **Create Dynamic Disk Group** to proceed with creating the disk group. A disk group with the same name as that on the Primary is created. |

   After the disk group has been created, click **Next**.

4. The Volume Information on connected hosts panel appears. This panel displays information about the availability of Replicator Log volume on the Bunker Secondary node.

   This panel does not appear if the required Replicator Log volume that is the same as that on the Primary is available on the Bunker Secondary host.

   - Because the Replicator Log volume is not created, the **Create Volume** option is enabled. Click this option to create the required Replicator Log volume on the Bunker Secondary.
   - The Create Volume dialog automatically displays the Replicator Log volume name and the size after verifying the information about the Primary host. Complete the information on this panel as follows:

     | | |
     |---|---|
     | Name | Displays the name for the volume in the Name field. This is the same as that specified for the Primary volume. |
     | Size | Displays the size of the volume in the Size field. This is the same as that specified for the Primary volume. |
     | Layout | Specify the volume layout. Select the appropriate volume layout depending on your requirement. |

| | |
|---|---|
| Disks Selection | Enables you to specify the disk selection method.<br>Select the **Select disks automatically** option if you want Volume Replicator to select the disks.<br>Select the **Select disks manually** option to use specific disks from the Available disks pane for creating the volume. Either double-click on it or select **Add** to move the disks into the Selected disks pane. |

After verifying the information, click **OK** to create the required volume. You are taken back to the **Volume Information on connected hosts** panel.

- After the Replicator Log volume has been created, the volume information panel is updated to display the Replicator Log volume on the Primary and Bunker Secondary host. Click **Next**.

5 Complete the **Edit replication settings** panel to specify the basic and advanced replication settings. The required settings are exactly similar to the **Edit replication settings** panel on the RDS wizard.

For details, See 4 on page 157.

6 Choose the appropriate option from the **Start Replication** panel as described below.

To add the Bunker Secondary and start replication immediately, check Start Replication with the following options:

| | |
|---|---|
| Synchronize Automatically | For an initial setup, use this default option to synchronize the Bunker Secondary and start replication.<br>If you add the Bunker RVG to a setup that already has Secondary hosts, then this option checks for the position of Secondary that lags behind the most and updates the Bunker RVG, appropriately. |
| Synchronize from Checkpoint | This option is not supported for a Bunker RVG. |

- To add the Bunker Secondary without starting replication clear the **Start Replication** option. You can start replication later by using the Start Replication option from the Secondary RVG right-click menu. Click **Next** to display the Summary panel.
- Review the information on the **Summary** panel. Click **Back** to change any information you had specified or click **Finish** to add the Bunker Secondary to the RDS and exit the wizard.

Chapter 4

# Using the VEA Console for Volume Replication Operations

This chapter includes the following topics:

## About performing Volume Replicator operations in the VEA console

This chapter explains how you can get started with using the VEA to perform the Volume Replicator operations and also how you can manage the Volume Replicator objects.

The VEA console is a Java-based Graphical User Interface (GUI) that consists of a server and a client. The server runs on a host that runs SFW and Volume Replicator. Volume Replicator is integrated with Storage Foundation and it provides its graphical user interface through VEA. This graphical user interface enables you to configure, monitor, and administer Volume Replicator in a distributed environment, that is, if you perform a task on an RDS or RVG, the task is performed on all the hosts in that RDS. You can thus use VEA to manage Volume Replicator objects on multiple hosts.

Using the VEA console, you can remotely administer and monitor products using its framework. Volume Replicator extends this remote administration feature for administering an entire RDS spanned across multiple hosts.

Volume Replicator provides a Graphical User Interface (GUI), a WebGUI interface as well a command line interface to perform Volume Replicator operations. The graphical user interface for Volume Replicator is provided through Veritas Enterprise Administrator (VEA).

# Features of VEA console

You can use the VEA to administer Volume Replicator objects on local or remote computers. The VEA server must be running on all the hosts in the Replicated Data Set (RDS).

VEA provides the following features:

- Ease of Use
  The task-based user interface provides access to tasks through Volume Replicator menus. Administrators can easily navigate, configure, and administer Volume Replicator, browse through the objects on the system or view detailed information about a specific object.
- Remote Administration
  Administrators can perform Volume Replicator administration remotely or locally. VEA offers wizards to guide you through complex configuration operations, such as Creating a Replicated Data Set, and so on.
- Navigation
  The tree-like organization of Volume Replicator objects facilitates easy navigation. Context-based menus provide easy access to operations.
- Mechanism for Notifying Users
  Users can configure rules using the Rule Manager to receive SNMP notifications or Email notifications of any alerts or events that are related to Volume Replicator.
- Multiple views of objects
  The VEA presents a tree view that organizes the Volume Replicator objects under a node called Replication Network. For each selected Volume Replicator object in the tree view there is an object view that displays detailed information about it.

- Monitoring Replication

  The monitor view enables you to monitor the replication activity in the replicated data set to which it belongs.

- Displaying Alerts

  The lower pane of the VEA displays alerts when the Console tab is selected. This view provides a detailed listing of alerts for the Volume Replicator operations that are performed.

# Launching the VEA console

The Veritas Enterprise Administrator (VEA) console is a graphical console that can be used to configure, monitor, and administer Volume Replicator in a distributed environment. The following sections provide information about how you can use the VEA console. For details, see the complete VEA help that is available by clicking the Help option from the VEA console.

From a Windows client, you can start VEA from the Start menu, or from the command line.

To invoke VEA from the Start menu, click **Start > All Programs > Veritas > Veritas Storage Foundation > Veritas Enterprise Administrator** or, on Windows 2012 operating systems, from the **Apps** menu in the **Start** screen.

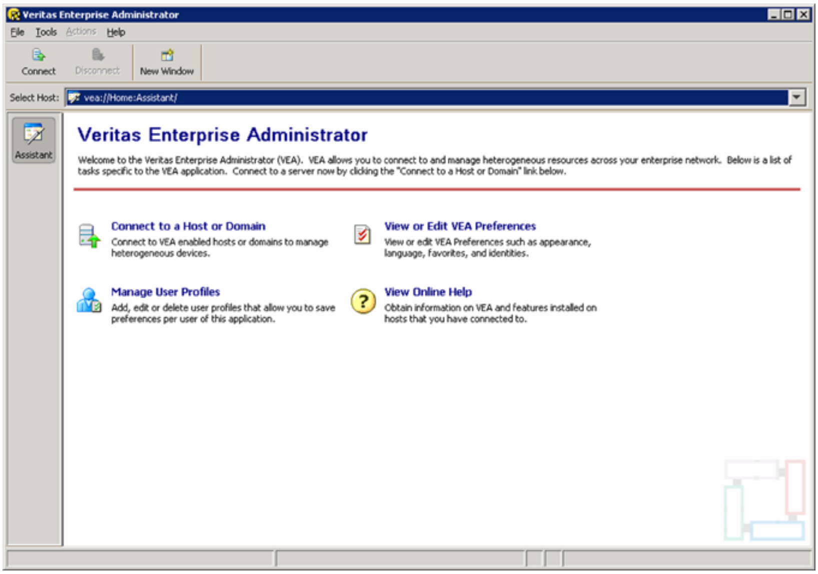
To invoke VEA from the command line, start the VEA client by running `vea.exe` from its installed location, such as `C:\Program Files\Veritas\Veritas Object Bus\bin`.

When you start the VEA client from the command line, the following options are available:

| | |
|---|---|
| `-v` | Shows the version of client console. |
| `-host` | Specifies the host to connect to. If the user account for the host is already stored, these are used; otherwise, you are prompted for your user account. |

The Veritas Enterprise Administrator default screen appears.

**Figure 4-1** VEA console



Veritas Enterprise Administrator

Welcome to the Veritas Enterprise Administrator (VEA). VEA allows you to connect to and manage heterogeneous resources across your enterprise network. Below is a list of tasks specific to the VEA application. Connect to a server now by clicking the "Connect to a Host or Domain" link below.

Connect to a Host or Domain
Connect to VEA enabled hosts or domains to manage heterogeneous devices.

View or Edit VEA Preferences
View or edit VEA Preferences such as appearance, language, favorites, and identities.

Manage User Profiles
Add, edit or delete user profiles that allow you to save preferences per user of this application.

View Online Help
Obtain information on VEA and features installed on hosts that you have connected to.

# Managing connections

The system host typically has multiple Veritas products installed on it. To be able to use the products, the client console needs to connect to the system through an authentication channel.

Storage Foundation and High Availability Solutions can access host machines simultaneously in different ways. The following topics provide the details:

- See “Connecting to a host” on page 100.
- See “Disconnecting from a host ” on page 101.
- See “Reconnecting hosts at startup” on page 102.
- See “Using history to view recent connections” on page 102.
- See “Managing favorites” on page 102.
- See “Adding a host to the favorites ” on page 103.
- See “Removing a host from the favorites ” on page 103.
- See “Switching connections” on page 103.

# Connecting to a host

You can connect to all hosts that are intended to be a part of the RDS, from VEA, and perform all the Volume Replicator operations on them.

After you have started Storage Foundation on one host, you can connect to additional hosts from the same session. Each host machine must be running the Veritas Enterprise Administrator service.

**Note:** This task requires a host machine name, user name, domain name, and password. Only users with appropriate privileges can run Storage Foundation and High Availability Solutions.

**Note:** If User Access Control (UAC) is enabled, then you cannot log on to VEA GUI with an account that is not a member of the Administrators group, such as a guest user. This happens because such user does not have the "Write" permission for the "Veritas" folder in the installation directory (typically, `C:\Program Files\Veritas`). As a workaround, an OS administrator user can set "Write" permission for the guest user using the Security tab of the "Veritas" folder's properties.

You can connect to a host in one of the following ways:

**To connect to a host**

1. To connect to a host:
   - Click **File > Connect**.
   - Click **Connect** toolbar icon.
   - Click the **Connect to a Host or Domain** task that is displayed on the **Home** panel.
2. Complete the Connect dialog box as follows:
   - Host Name
     Enter the name of the system to be administered. (For example, in case of Volume Replicator, both vcsvc and the VEA service must be running on this system.
   - Use Browse to locate the system to be administered.
   - Clicking Browse displays the Browse dialog box. The Browse dialog box includes the Favorites and Network tabs. You may select a host from the Favorites and Network tab.

3 When the default user account is not set, the **Connect as** option displays **No default user account** as the default user account. When the default user account is set the radio button displays the same to connect to the system.

If there is no default account identity but the host has been connected to some other user account, it is displayed in the dialog box.

4 Select **Connect using a different user account** to connect to the system using any other user account other than the default user account.

5 Click **Connect** to log on.

The Connecting to:<machine name/machine IP> dialog box is displayed.

Complete the Connecting to:<machine name/machine IP> dialog box as follows:

| | |
|---|---|
| Username | Enter your logon name. Only users with appropriate privileges can access Storage Foundation on the specified system. (The service is already running on the host.) |
| Password | Enter your password for the system to be administered. |
| @Domain | Select a domain (If any) from the drop-down list. The list contains the domains that the host is part of. |
| Save Password | Select this check box to save the password on your computer |
| Set this as the default user account for this profile | Select this check box to set the current account information as the default user account for this profile. The saved user account can be viewed by clicking **Pick** or by clicking **Security** tab in the `Preferences` panel. |

Alternatively, the user can select an already saved user account by clicking Pick. When you have provided all necessary information in the dialog box, click **OK**. The new host appears in the object tree in the main window.

After you connect to the required hosts, VEA displays the Replication Network object in the **Select Host** field. Click on **Replication Network** to view the Volume Replicator objects. The VEA console provides a single graphical interface to view and manipulate all the SFW objects and Volume Replicator objects on your system. You must first connect to at least the local node so that Replication Network node is available.

## Disconnecting from a host

The disconnect procedure disconnects a host machine from the current VEA session. When a host machine is disconnected, Storage Foundation cannot administer that system until a new connection is made. To restore access to a disconnected host machine, you must reconnect to the host.

**To disconnect from the host using the option from the right-click menu**

1. Right-click the host in the **System** pane.
2. Select **Disconnect** from the pop-up menu.
3. A confirmation dialog appears. Click **Yes** to end the remote connection.
4. The host-related views disappear.

**To disconnect from the host using the Disconnect icon from the toolbar**

1. Click **Disconnect**.

   The Disconnect dialog box is displayed. This dialog box displays the list of connected hosts.
2. Select the host to be disconnected.
3. Click **OK** to disconnect the host.

   or

   Alternatively, the disconnect dialog box can also be accessed from the File menu. Select **File > Disconnect** to display the **Disconnect** dialog box.

## Reconnecting hosts at startup

By default, hosts in the **Favorites** list are reconnected at startup. You can disable the default by disabling the **Reconnect At Logon** option.

**Reconnecting hosts at startup**

1. Select **Tools > Preferences > Connection** tab.

   **Favorites** displays the hosts that have been added to the list as favorites.
2. Select the host and click the **Reconnect At Logon** column for the host.
3. Select **No**.

   A host in the **Favorites** is not reconnected at startup if the **Reconnect At Logon** option is set to **No**.

## Using history to view recent connections

VEA displays the list of hosts recently connected in the connection dialog.

## Managing favorites

Favorites is a convenient way to organize and connect to hosts that you need frequently. It enables listing your favorite hosts for quick viewing. Favorites contains a list of hosts that are connected to by default at the startup of VEA if the user account is saved for them. If the user account is not saved for a particular host, then this information is prompted for at the time of connection.

You can manage Favorites by adding and removing hosts from the Favorites lists. See the instructions on how to add and remove a host from the Favorites list.

To access favorites:

- Select **Tools > Preferences > Connection**.

# Adding a host to the favorites

You can add a host to the favorites list.

**To add a host to the Favorites list**

1. Right-click on any of the connected host nodes.
2. Choose **Add to Favorites** from the pop-up menu.
3. Select the **Tools > Preferences > Connection** tab to verify whether the host has been added to the **Favorites** list.

   The hosts that have been added as Favorites are displayed in the list.

# Removing a host from the favorites

You can remove a host from the Favorites list.

**To remove a host from the Favorites list**

1. Select **Tools > Preferences**.

   The Preferences dialog is displayed.
2. Click the **Connection** tab.

   Select the host(s) to be removed from the Favorites list.
3. Click **Remove**.

   The specified host is removed from Favorites and is not displayed.

# Switching connections

VEA follows a host-based approach. Only one host can be viewed in a particular window at a given point of time. For viewing multiple hosts, you can use the New Window feature to launch multiple windows.

If you want to view a different host that you have already connected to, you can do so in the following ways:

- By switching Connections using the URL bar

- By selecting **View > Connection > <machine name or machine IP>**

# Layout of the VEA console

This section explains the Panes with respect to Volume Replicator. Refer to the VEA online Help for more information. It can be accessed by selecting Contents from the Help menu in the VEA GUI.

The VEA display console can be divided into the following view areas:

- Navigation View, on the left, which uses the tree structure
- Details View, on the right, which uses the table structure
- Status Pane at the bottom, which includes the Console and Task Views.

The following sections describe the VEA console layout in detail:

- See “Performing tasks related to views” on page 104.
- See “Selecting objects” on page 105.
- See “Left pane or navigation view (tree view)” on page 106.
- See “Right pane or details view (tabular view)” on page 106.
- See “Status pane” on page 107.
- See “URL bar ” on page 108.
- See “Perspectives” on page 109.
- See “Menu bar and toolbar” on page 110.

## Performing tasks related to views

You can perform various tasks that are related to views.

**To Browse Objects in the Tree View**

1. Expand or collapse the hierarchy under a particular object node in the tree.
2. Click the plus sign (+) or minus sign (-) icon next to that node.
3. Alternatively, you can use the down arrow, up arrow, and right arrow keys to browse the tree using the keyboard.

**To display the objects in an object group listed in the object tree**

1. Browse to the object group and then select the object.
2. Select the object by clicking the object group or browsing to the object group and pressing the **Enter** key. All objects that belong to the selected object group appear in a tabular view on the right.

   If a non-group or leaf object that does not contain any objects is selected, then the properties of the object are displayed instead of the contained objects.

**To sort the objects in tabular view by a specific property**

1. Click the appropriate property column header.
2. To reverse the sort order, click the column heading again.

**To resize a table column**

1. Place the pointer over the line that divides the column headings.
2. Press and hold the mouse button to drag the column divider to the desired position.

**To resize the left pane (tree) and right pane (tabular view)**

1. Place the pointer over the vertical splitter.
2. Press and hold the mouse button to drag the splitter to the desired position.

## Selecting objects

To select multiple objects, hold down the Control key while selecting the objects. The objects that you select in this way do not have to be adjacent.

You can select a single, or range of adjacent objects in the following ways.

**To select a single object**

1. Click the object or browse to the object
2. Press the **Enter** key.

**To select a range of adjacent objects**

1. Select the first object, then hold down the **Shift** key while selecting the last object in the range.
2. You can also select multiple adjacent objects by pressing and holding the mouse button while dragging the pointer over the desired objects.

## Left pane or navigation view (tree view)

The left pane displays a collapsible and expandable tree view. After you connect to a host, and select Replication Network from the URL bar, VEA populates the tree view with the related Volume Replicator objects. Each object can be related to other objects and you can see related objects in the tree hierarchy. You can view detailed information about an object that is shown in the tree view by selecting it and viewing properties through the right-click context menu or by selecting the object and invoking the File > Properties menu. To view the contents of each of the tree nodes in the left pane, expand it by clicking on the (+) symbol. Alternatively, you can use down arrow, up arrow and right arrow keys to browse the tree using the keyboard. You can then right-click on each object to view required operations.

The Replication Network node displays the list of RDS on the connected hosts.

Expand the RDS node to see the following:

- Primary RVG under the selected RDS
- Secondary RVGs under the selected RDS

## Right pane or details view (tabular view)

The right pane displays detailed tabulated information about the objects that are selected in the left pane tree view. When the Replication Network node is selected, the right pane displays all the RDSs present under this node in a tabular format.

Click Replication Network (+) in the URL bar and select Replication Network node in the tree view to expand the tree and view all the RDSs under it. If you select an RDS under Replication Network node, the detailed information about it is displayed in the right pane in a tabular format. A vertical bar separates the right and the left pane. This bar can be dragged to the right or left thus enabling you to modify the right and left pane display area.

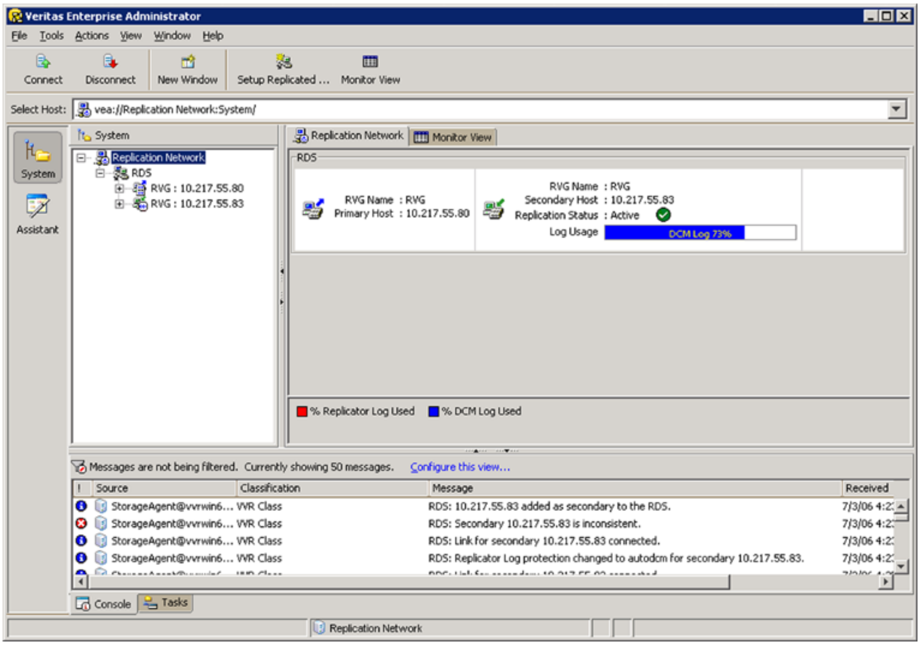
To view information about the specific objects, click on the objects that are displayed under each RDS node in the left pane tree view. Limited properties of the object are shown in the columns of the tabular view. You can set custom preferences on the layout and size of these columns by using the column setting functionality. You can view more information about an object that is shown in the tabular view by selecting it and viewing properties through the right-click context menu or by selecting the object and invoking the File > Properties menu.

The right pane displays two tabs, the <Object> view and the Monitor View. The object view tab is context sensitive to the object that is selected in the left tree view. When the Replication Network node is selected, the object view displays information about the replication activity for all the RDSs under it. When an RDS or any object under that RDS is selected the object view displays information for that RDS.

Depending on the selected object, the tab name changes accordingly. For example, if the RDS is selected, the tab becomes the Replicated Data Set tab. The Monitor View tab displays the information that enables you to monitor replication.

See “Interpreting the information in the Volume Replicator views” on page 115.

**Figure 4-2** Replication objects properties in Monitor view



Select Host: vea://Replication Network:System/

System
- Replication Network
  - RDS
    - RVG : 10.217.55.80
    - RVG : 10.217.55.83

RDS

RVG Name : RVG
Primary Host : 10.217.55.80

RVG Name : RVG
Secondary Host : 10.217.55.83
Replication Status : Active ✓
Log Usage: DCM Log 73%

🟥 % Replicator Log Used 🟦 % DCM Log Used

Messages are not being filtered. Currently showing 50 messages. Configure this view...

| ! | Source | Classification | Message | Received |
|---|---|---|---|---|
| ℹ | StorageAgent@vvrwin6... | VVR Class | RDS: 10.217.55.83 added as secondary to the RDS. | 7/3/06 4:2 |
| ⊗ | StorageAgent@vvrwin6... | VVR Class | RDS: Secondary 10.217.55.83 is inconsistent. | 7/3/06 4:2 |
| ℹ | StorageAgent@vvrwin6... | VVR Class | RDS: Link for secondary 10.217.55.83 connected. | 7/3/06 4:2 |
| ℹ | StorageAgent@vvrwin6... | VVR Class | RDS: Replicator Log protection changed to autodcm for secondary 10.217.55.83. | 7/3/06 4:2 |

Replication Network

The views display the properties of the selected replication object. You can also view the properties by clicking on the required replication object and selecting the Properties option from the right-click menu. For example, select RDS and right-click. Select Properties from this menu to view the RDS properties.

# Status pane

The Status pane occupies the bottom part of the VEA client window. It includes the Console View and Task View tabs which are present in the lowermost left corner of the VEA console.

## Console view

The Console view displays the listing of recent messages for the connected hosts using a distinct table interface. You can view the messages when you select the Console tab at the bottom left corner of the window. For each message listing, you see information about the severity of the message, the date and time that the message was received, the message description, its source, and its classification.

Select the row to expand it and display additional information about the message. This information includes an event description, recommended action, and user-defined properties. This makes it easier to read the description of the message.

Double-click the message or press the Enter key on the message to open the **Console Message Details** dialog to display more properties.

The **Console Message Details** dialog provides details about each message. You can copy the contents to the system Clipboard so that you can use it later for support calls.

The filtering functionality has been introduced to enable you to filter the alerts based on the source, classification, and severity.

Clicking the "Configure this view..." link at the top of the Console View window displays the **Preferences** dialog box with the Console View tab. In the **Console View** tab, you can change the message buffer and filter settings. You can also select **Tools > Preferences** to configure the filter settings.

### Tasks view

The Tasks View displays the start time of the task along with the object name for which the task was fired. Click on the **Tasks** tab to display information for tasks.

## URL bar

VEA offers the option of using the URL bar to reduce the complexity of tree view. It displays the currently selected object's location in the tree. You can also change the active host. Every new connection is added as an entry to the URL bar, and you can manage only one system in one window at a time.

You can change the active connection by selecting it from the URL bar box. Alternatively, you can select the View > Connection menu to change the active connection. All top-level nodes appear in the URL bar.

See the relevant product documentation for more details about which features are available from the URL bar.

The format of the URL in the URL bar is as follows:

```
vea://<host name>:<perspective>/<path of the selected object
in the navigational view>
```

# Perspectives

VEA has introduced the concept of perspectives to separate distinct aspects of a connected system. A perspective is a filtered view of a system that exposes only certain operations and objects on that system. For example, the Logs perspective display only the Event and Task logs.

Assistant is another perspective that provides you with a list of the most common tasks on a host or a domain. You can select and perform your tasks on objects, without the need to know the objects. It is a task-based approach to perform the job at hand instead of an object-based approach.

The Control Panel is a perspective. It displays the configuration-related tasks available on the system to which you have connected. You can switch perspectives by selecting the appropriate option from the Perspective Bar displayed on the left side of the VEA window. You can also select a perspective using the **View > Perspective** menu.

The Assistant and Logs perspectives are displayed only on connecting to a VEA host.

## Control Panel

The Control Panel can be used to view and modify application settings. The Control Panel is available as a perspective in the **Perspective** bar on the left pane of the VEA console. It displays configuration-related tasks available on the host or system to which you are connected. You can select the Control Panel using the **View** > **Perspective** > **Control Panel** menu.

### Changing the IPv6 preference through the Control Panel

When you configure replication, if you specify host names for the Primary or the Secondary systems, Volume Replicator resolves the host names to the IP addresses associated with them. The IP setting determines which IP protocol Volume Replicator uses to resolve the host names. Before you proceed with configuring the replication, you must set the IP preference depending on the IP protocol to use.

**To change System NPP and Volume Replicator NPP values and IP preference through the Control Panel**

1. Select the Control Panel using the **View** > **Perspective** > **Control Panel** menu.
2. Select the **StorageAgent** to display the Volume Replicator Configuration icon on the right pane of the VEA console.
3. Double-click the **VVR Configuration** icon.

   VVR Configuration dialog box is displayed.

4 On the **IP Settings** tab, check the **Prefer IPv6 Settings** check box if you want to use IPv6 addresses for replication.

When this option is checked, Volume Replicator resolves the host names to IPv6 addresses.

This option is cleared by default, which means that Volume Replicator resolves host names to IPv4 addresses by default.

5 Click **OK** to confirm the settings and close the window.

## Menu bar and toolbar

The top portion of the VEA has the menu bar which includes the File, Action, Tools, and Help options. Below that is the toolbar. The toolbar displays some options that you may need to use very frequently. Of these, the Connect, Disconnect and New Window options are always available on the toolbar and can be used to connect to the required hosts.

However, the additional options on the toolbar are sensitive to the object that you have selected. When you select Replication Network or any object under this node, the Setup Replicated Data Set and Monitor View tool buttons appear on the toolbar. The monitor view is a constant menu option that is available for all Volume Replicator objects.

# Accessing the Volume Replicator options

Using VEA, you can access various Volume Replicator options.

See “Menu bar options” on page 110.

See “Toolbar options” on page 112.

## Menu bar options

This section briefly describes the menus available under the menu options for Volume Replicator. Note that the menu options are sensitive to the object that is selected. Depending on the object that is selected in the left tree, some of the options in the menu change.

See “File menu” on page 111.

See “Tools menu” on page 111.

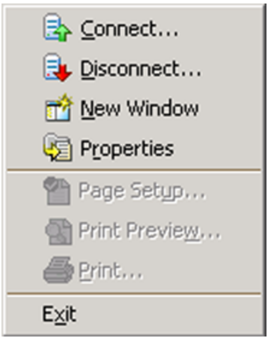
See “Actions menu” on page 112.

See “Toolbar options” on page 112.

## File menu

The File menu displays the following options. Some of these options are also available from the toolbar and are represented by the icons that have been displayed alongside the options.

**Figure 4-3** File menu



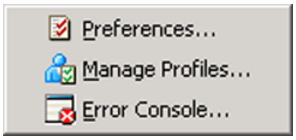
File menu contains the following options:

- Select **Connect** to connect to the hosts where the Volume Replicator server is installed.
- Select **Disconnect** to disconnect the host.

## Tools menu

Tools menu contains the options which are represented by the icons that have been displayed alongside the options.

**Figure 4-4** Tools menu



The tools menu displays the following options:

- Select Preferences option to set any specific preferences for the VEA console display. You can use the Volume Replicator Monitor View tab in the **Preferences** dialog to customize the monitor view.
- The Manage Profiles option enables users running VEA on the same system to maintain their own preferences, connection history, and favorites.

For more information about setting up the user profiles, refer to the online Help that is available from the VEA windows Help option. Select Contents from the Help menu. The Help window appears. From the Select help set drop-down list, select Veritas Enterprise Administrator (VEA) > Getting Started with VEA.

- Select Error Console to display the error messages, if any.

## Actions menu

The options that are available under the **Actions** menu are context sensitive to the object that is selected under the Replication Network node. For example, if the Primary RVG is selected, then the Actions menu lists the Primary RVG tasks as shown in the following menu.

The following options are commonly displayed across all the Actions menu:

- Select Refresh to refresh the VEA view if the view did not get updated after you performed some task.
- Select Rescan to display the Volume Replicator objects if they did not get refreshed after you performed some task.
- Select the Monitor View to display the Monitor View window.
- Click **Add Secondary**.
- Add Bunker
- Add Volume
- Delete Replicated Data Set.

## Toolbar options

To enable you to perform some frequently used tasks quickly some of the options are made available on the toolbar and represented by icons.

The following table summarizes the icons available on toolbar menu and corresponding Volume Replicator tasks.

**Table 4-1** Icons and corresponding Volume Replicator tasks

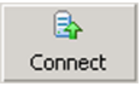
| Icon | Description |
|---|---|
|  | Click this icon to connect to the required hosts from the VEA. Note that although you can connect to the hosts from a VEA client, the host must have Volume Replicator installed and VEA server running. |

**Table 4-1** Icons and corresponding Volume Replicator tasks *(continued)*

| Icon | Description |
|---|---|
| | Click this icon to disconnect the specified hosts from the VEA console. |
| | Click this icon to open up another window, which duplicates everything in the main window. You can simultaneously browse different sections of the System tree in this window without having to launch another instance of the VEA GUI. You can then select the other host that you want to manage from the URL bar. This feature enables browsing and comparing of the objects that are found in different parts of the tree. |
| | You can directly click on this icon to create a replicated data set once you have finished creating the required disk groups and volumes on the Primary host. |
| | Use this icon to display the monitor view.<br>See “Interpreting the information in the monitor view” on page 134. |

# Exiting the VEA client

Before closing the VEA, you can disconnect all hosts. If you have not disconnected all hosts, VEA displays a message asking whether it is okay to disconnect the hosts.

To close the VEA, select **File > Exit**. Alternatively, you can select the close (x) icon from the top right corner of the VEA.

# Monitoring replication

This chapter includes the following topics:

## About monitoring replication

This chapter discusses the methods that you can use to monitor replication. This enables you to ensure that the replication happens correctly and also detect any problems, up front. Volume Replicator provides the Monitor View option both from the toolbar and the Menu bar.

See "Interpreting the information in the monitor view" on page 134.

VEA also provides context-sensitive object views, which can be used to obtain complete information about each of the selected objects. Each view displays detailed information about the selected object and the states (if any) that are associated with it.

# Interpreting the information in the Volume Replicator views

Using Volume Replicator, you can view information about the Volume Replicator objects. This section provides information about how you can display views for different Volume Replicator objects and interpret the information that is displayed in each view.

See "Viewing RDS information" on page 116.

See "Viewing information about the Primary RVG" on page 122.

See "Viewing information about the Secondary RVG" on page 125.

See "Viewing information about the Primary data volume" on page 129.

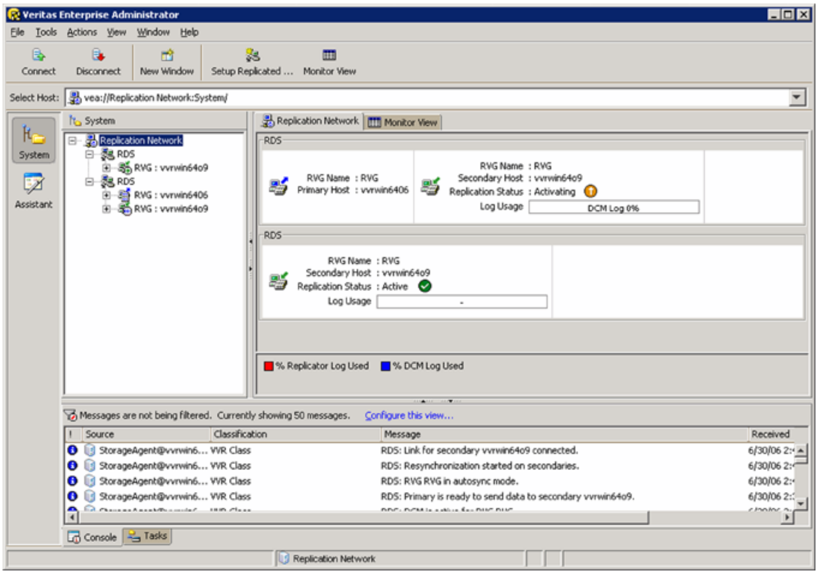
See "Viewing the Replicator Log volume information " on page 129.

See "Viewing information about the Secondary data volume " on page 131.

## Viewing all the RDSs on the host

Select the Replication Network node. The list of all the RDSs present on the connected hosts are displayed in the right pane.

**Figure 5-1** VEA console: list of RDS present on the connected host



Select Host: vea://Replication Network:System/

RVG Name : RVG
Primary Host : vvrwin6406

RVG Name : RVG
Secondary Host : vvrwin6409
Replication Status : Activating
Log Usage DCM Log 0%

RVG Name : RVG
Secondary Host : vvrwin6409
Replication Status : Active
Log Usage -

Messages are not being filtered. Currently showing 50 messages.

| ! | Source | Classification | Message | Received |
|---|---|---|---|---|
| ℹ | StorageAgent@vvrwin6... | VVR Class | RDS: Link for secondary vvrwin6409 connected. | 6/30/06 2: |
| ℹ | StorageAgent@vvrwin6... | VVR Class | RDS: Resynchronization started on secondaries. | 6/30/06 2: |
| ℹ | StorageAgent@vvrwin6... | VVR Class | RDS: RVG RVG in autosync mode. | 6/30/06 2: |
| ℹ | StorageAgent@vvrwin6... | VVR Class | RDS: Primary is ready to send data to secondary vvrwin6409. | 6/30/06 2: |

Each row in the right pane includes a complete summary information about all the RDSs present on the host.

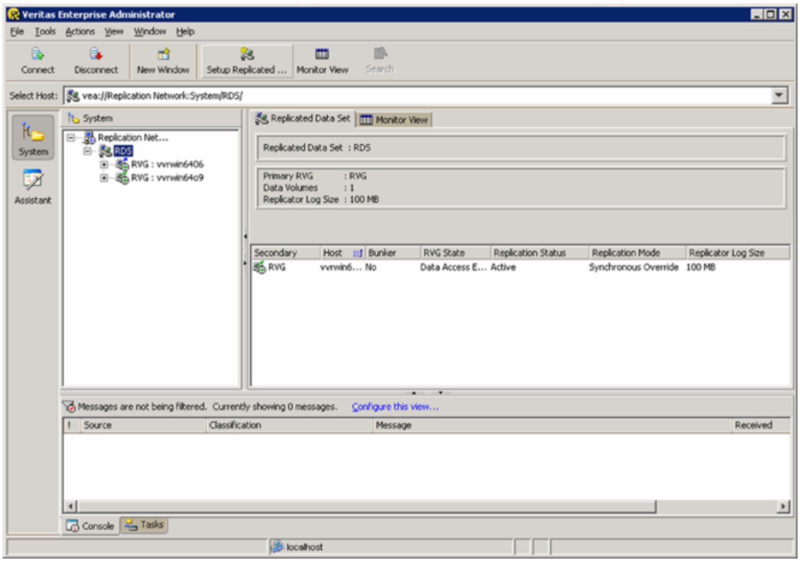
The information is as follows:

- Name of the RVG
- Name or IP address of the Primary host
- Name or IP address of the Secondary host
- Replication status
- Log usage details

# Viewing RDS information

Select the required RDS. The Replicated Data Set view is displayed in the right pane as shown below.

**Figure 5-2** VEA console: RDS view



Select Host: vea://Replication Network:System/RDS/

Replicated Data Set : RDS

Primary RVG : RVG
Data Volumes : 1
Replicator Log Size : 100 MB

| Secondary | Host | Bunker | RVG State | Replication Status | Replication Mode | Replicator Log Size |
|---|---|---|---|---|---|---|
| RVG | vvrwin6... | No | Data Access E... | Active | Synchronous Override | 100 MB |

Messages are not being filtered. Currently showing 0 messages. Configure this view...

| ! | Source | Classification | Message | Received |
|---|---|---|---|---|

This view displays information about the RVGs in the selected RDS. After this, a tabular structure displays the information about the Secondary RVG. If you click the **Monitor View** tab when the RDS object is selected in the left pane, the right pane displays statistical information about the replication activity, for the selected RDS.

The following table describes the information that is displayed about RVGs in the selected RDS.

**Table 5-1** Displayed information about RVGs in the selected RDS

| Field Name | Description |
|---|---|
| Primary RVG | Displays the name of the Primary RVG. |
| Data Volumes | Displays the number of data volumes that are associated with the RVG. |
| Replicator Log Size | Displays the size of the Replicator Log volume in the Primary RVG. |

The following table describes the information that is displayed about the Secondary RVGs, which are a part of the selected RDS.

**Table 5-2** Field names and corresponding descriptions of Secondary RVGs

| Field Name | Description |
|---|---|
| Secondary RVG | Displays the name of the Secondary RVG. |
| Host | Displays the IP address or name of the Secondary host that belongs to the selected RDS. |
| RVG State | Displays the state of the Secondary RVG.<br>See "RVG states" on page 117. |
| Replication Status | Displays the current status of the replication.<br>See "Replication status" on page 120. |
| Replication Mode | Displays the current mode of replication. The different modes are, synchronous, asynchronous, and synchronous override.<br>See "Modes of replication" on page 28. |
| Replicator Log Size | Displays the size of the Replicator Log volume in the Secondary RVG. |

## RVG states

There are icons that represent all Volume Replicator objects. Of these, only the RVG icon changes to represent the current state of the RVG.

The following table explains the different RVG states. The icons column lists the various icons that are used to represent each state on the Primary and Secondary. The Primary and Secondary columns indicate the validity of the state for each of these hosts. The command line interface column represents the equivalent of the GUI states on the command line, that is, the output of the `vxprint -l` command.

**Table 5-3** Volume Replicator object icons

| Primary Icon | Secondary Icon | State | Description |
|---|---|---|---|
| ▶ | ▶ | `Data Access Enabled`<br>CLI States: `ACTIVE`<br>Valid for Primary and Secondary | Indicates that the data volumes under the RVG are enabled for Input/Output, that is, these volumes can be used for writing and reading data. |
| 🟥 | 🟥 | `Data Access Disabled`<br>CLI States: `CLEAN`<br>Valid for Primary and Secondary | Indicates that the data volumes under the RVG are disabled for Input/Output and volumes are unavailable for reading or writing data. |
| ↗ | ↗ | `Failed`<br>CLI States: `fail`<br>Valid for Primary and Secondary | The failed flag is set if the incoming Input/Output cannot be written to the underlying data volumes due to some problem with the data volumes. |
|  | ⟳ | `Autosynchronizing`<br>CLI States: `autosync`<br>Valid for Secondary only | Indicates that Automatic Synchronization has started. |
|  | ⏸ | `Resynchronization Paused`<br>CLI States: `resync_paused`<br>Valid for Secondary only | Indicates that resynchronization is paused. |
|  | ◀◀ | `Resync Started`<br>CLI States: `resync_started`<br>Valid for Secondary only | Indicates that resynchronization is in progress. |

**Table 5-3** Volume Replicator object icons *(continued)*

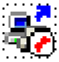
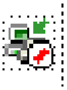
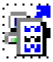
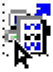
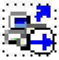
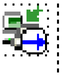
| Primary Icon | Secondary Icon | State | Description |
|---|---|---|---|
| | | `Inconsistent`<br>CLI States: `inconsistent`<br>Valid for Secondary only | This state is displayed only for the Secondary RVG, when the data on the Secondary volumes is inconsistent with respect to Primary RVG.<br>The Secondary may become inconsistent when the resynchronization or autosynchronization is in progress.<br>The Secondary may also become inconsistent when the RVG goes into `Failed` state. |
|  |  | `Replicator Log Header Error`<br>CLI States: `srl_header_err`<br>Valid for Primary and Secondary | This error is encountered when attempts to access the header section of Replicator Log are unsuccessful. |
|  | | `DCM Active`<br>CLI States: `dcm_logging`<br>(only in case the Replicator Log overflows)<br>Valid for Primary only | Indicates that the DCM is in use, either due to autosynchronization, resynchronization, fast-failback logging, or Replicator Log overflow. |
|  | | `Fast-failback Logging`<br>CLI States: `failback_logging`<br>Valid for Primary only | Indicates that Volume Replicator logs new updates to the Primary using the DCM logging. |
|  |  | `No Replicator Log`<br>CLI States: `passthru`<br>Valid for Primary and Secondary | This state is encountered when the Replicator Log is not associated with the RVG. |

**Table 5-3** Volume Replicator object icons *(continued)*

| Primary Icon | Secondary Icon | State | Description |
|---|---|---|---|
| | | `Primary Replicator Log error`<br>CLI States: `passthru`<br>Valid for Primary only | This state is encountered if the Primary receives Input/Output error when attempting to read from or write to its log volume. |
| | | `Checkstarted`<br>CLI States: `awaiting_checkend`<br>Valid for Primary only | Indicates that the checkpoint has started and awaits checkend. |
| | | `Not Recovered`<br>CLI States: `needs_recovery`<br>Valid for Primary and Secondary | This state is encountered if the RVG does not recover automatically after a system restart. |
| | | `Acting as Secondary`<br>CLI States: `acting_Secondary`<br>Valid for Primary only | Indicates that the original Primary RVG is currently the acting Secondary as part of the fast-failback process. Writes to the data volumes in this RVG are disabled irrespective of whether the RVG is started or stopped. |

## Replication status

The Replication Status column displays the current status of replication, that is, the state of the RLINK (Secondary).

The following table describes the Secondary RVG icons.

**Table 5-4** Replication status for Secondary RVG icons

| Icon | Status | Description | Command Line Interface States |
|---|---|---|---|
| | `ACTIVE` | Indicates that the replication is in an active state and also the Primary and the Secondary configuration are correct. | `ACTIVE` |

**Table 5-4** Replication status for Secondary RVG icons *(continued)*

| Icon | Status | Description | Command Line Interface States |
|---|---|---|---|
| | `Activating` | Indicates that the Primary and Secondary RLINK for the RVG in consideration is attached but not yet connected. | `attached`<br>`disconnected` |
| | `Secondary Paused` | Indicates that the Secondary has been paused, however, the connection between Primary and Secondary is maintained. In this state, the data is written only to the Primary and is not sent to the Secondary.<br>Only after the Secondary has been resumed, can all the data on the Replicator Log be sent to the Secondary. | `Secondary_paused` |
| | `Primary Paused` | Indicates that the Secondary has been paused from the Primary.<br>**Note:** When pause is effected from the Primary, the Secondary gets disconnected. It can get reconnected only after a Resume operation is performed. | `Primary_paused` |
| | `Inactive` | Indicates one of the following conditions:<br>■ No RLINK has been created and associated for the concerned RVG<br>■ The Primary and Secondary RLINKs for the concerned RVG are not attached | `STALE` |
| | `Failed` | Indicates that the Secondary RLINK is in a `Failed` state. | `FAILED` |
| | `Secondary Replicator Log Error` | This error is encountered when the Secondary receives an Input/Output error when attempting to read from or write to its log volume. | `Secondary_log_err` |

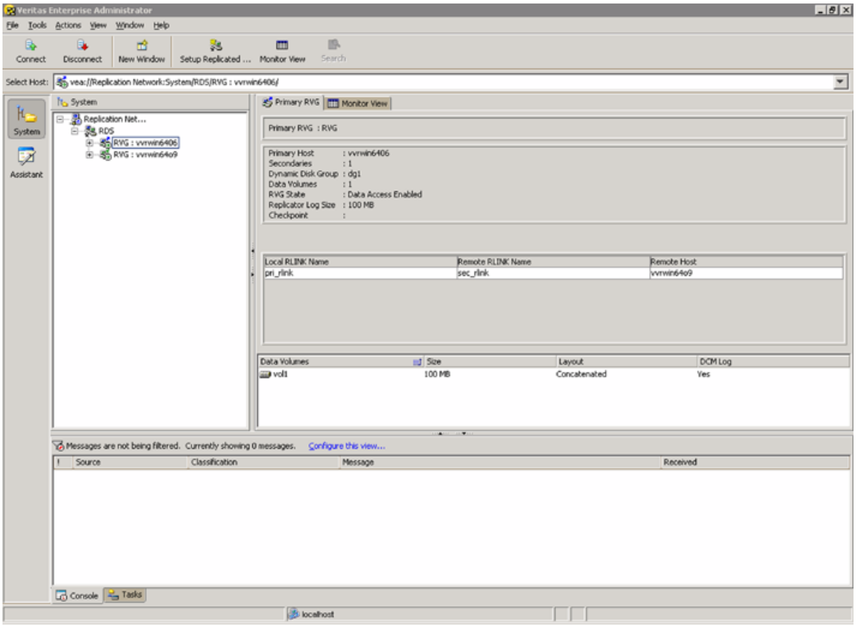
**Table 5-4** Replication status for Secondary RVG icons *(continued)*

| Icon | Status | Description | Command Line Interface States |
|---|---|---|---|
| | `Configuration Error` | Indicates one of the following conditions:<br>▪ The size of the data volumes on Primary RVG is not the same as the Secondary RVG volumes.<br>▪ The Secondary RVG does not have same number of volumes as compared to that on the Primary RVG.<br>▪ The names of volumes that are associated with the Primary RVG do not match those associated with the Secondary. | `Secondary_config_err` |

# Viewing information about the Primary RVG

Select the Primary RVG from the left pane. The right pane displays information about the Primary RVG and the associated data volumes.

**Figure 5-3** VEA console: Primary RVG information



Select Host: vea://Replication Network:System/RDS/RVG : vvrwin6406/

Primary RVG : RVG

```
Primary Host        : vvrwin6406
Secondaries         : 1
Dynamic Disk Group  : dg1
Data Volumes        : 1
RVG State           : Data Access Enabled
Replicator Log Size : 100 MB
Checkpoint          :
```

| Local RLINK Name | Remote RLINK Name | Remote Host |
|---|---|---|
| pri_rlink | sec_rlink | vvrwin64o9 |

| Data Volumes | Size | Layout | DCM Log |
|---|---|---|---|
| vol1 | 100 MB | Concatenated | Yes |

Messages are not being filtered. Currently showing 0 messages. Configure this view...

| ! | Source | Classification | Message | Received |
|---|---|---|---|---|

The Primary RVG view displays the Primary host name or IP address, the number of Secondary hosts, the number of data volumes, the RVG State, Replicator Log size and the checkpoints. The Primary RVG view also displays the detailed information about the data volumes that are associated with the Primary RVG.

The following table describes the Primary RVG view.

**Table 5-5** Primary RVG information

| Displayed field | Description |
|---|---|
| Primary Host | Displays the IP address or host name of the Primary host. |
| Secondaries | Displays the number of Secondary hosts in the RDS. |
| Dynamic Disk Group | Displays the name of the dynamic disk group, whose volumes are a part of the RVG.<br>If the RVG is part of a clustered disk group, then the disk group name is displayed with a `Cluster` tag against it. |
| Data Volumes | Displays the number of data volumes that are present in the RVG. |

**Table 5-5** Primary RVG information *(continued)*

| Displayed field | Description |
|---|---|
| RVG State | Displays the state of the RVG.<br>See "RVG states" on page 117. |
| Replicator Log Size | Displays the size of the Replicator Log. |
| Checkpoint | Displays the Primary RVG checkpoint that has already been started, but not yet ended. |

The following table describes the RLINK information that is displayed for a selected RVG.

**Table 5-6** RLINK Information in the Primary RVG View

| Displayed field | Description |
|---|---|
| Local RLINK Name | Displays the name of the local RLINK. If you specify a name for the RLINK when you create it then that name is displayed. Otherwise, the default name that Volume Replicator specifies is displayed. |
| Remote RLINK Name | Displays the name of the remote RLINK. If you specify a name for the RLINK when you create it then that name is displayed. Otherwise, the default name that Volume Replicator specifies is displayed. |
| Remote Host | Displays either the name or the IP of the remote host, depending on how the RLINK is configured. If the RLINK is configured using the host name then the name is displayed. |

The following table describes the Primary RVG data volume fields that are displayed.

**Table 5-7** Primary RVG data volume information

| Displayed field | Description |
|---|---|
| Data Volumes | Displays the names of the data volumes that are associated with the RVG. |
| Size | Displays the size of the data volumes. |

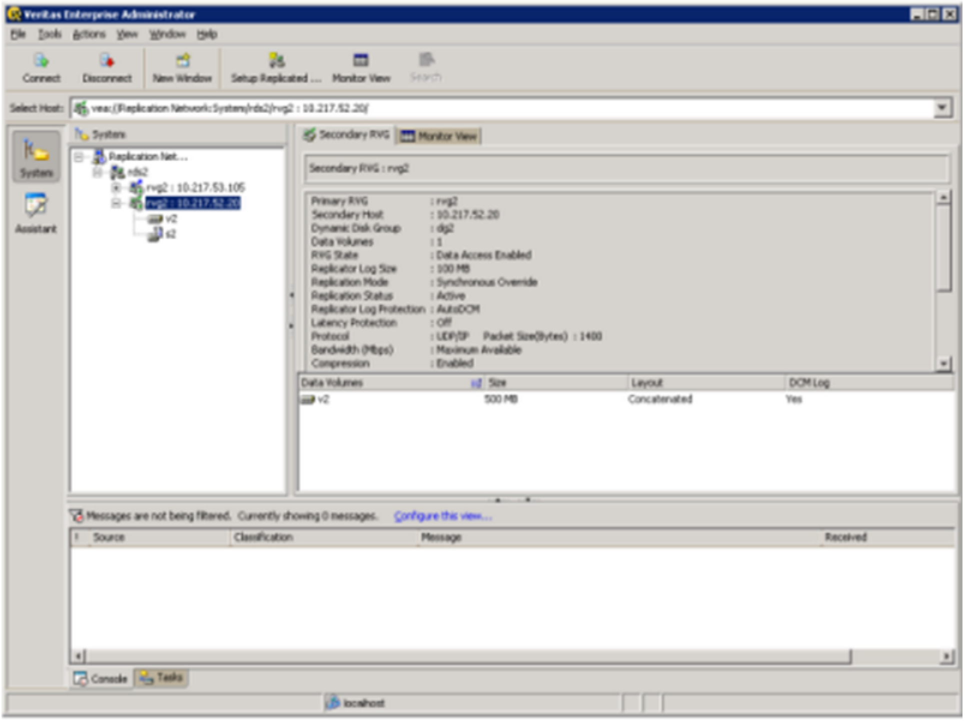
**Table 5-7** Primary RVG data volume information *(continued)*

| Displayed field | Description |
|---|---|
| Layout | Displays the type of volume layout, that is:<br>■ Concatenated<br>■ Mirrored Concatenated<br>■ Striped<br>■ Mirrored Striped<br>■ Mixed<br>For more information about the volume layout, see *Storage Foundation Administrator's Guide*. |
| DCMLog | Displays whether the DCM log is present. Valid values are:<br>**Yes**: indicates that the volume has a DCM log.<br>**No**: indicates that the volume does not have a DCM log. |

## Viewing information about the Secondary RVG

To view information about the Secondary RVG, from the tree view in the left pane, expand the Replication Network node to view the RDSs on that host. Expand the required RDS node to select the appropriate Secondary RVG from the tree view of the left pane.

The right pane displays information about the Secondary RVG. The Secondary RVG view is similar to the Primary RVG view, except that it displays some additional information.



Secondary RVG : rvg2

```
Primary RVG                 : rvg2
Secondary Host              : 10.217.52.20
Dynamic Disk Group          : dg2
Data Volumes                : 1
RVG State                   : Data Access Enabled
Replicator Log Size         : 100 MB
Replication Mode            : Synchronous Override
Replication Status          : Active
Replicator Log Protection   : AutoDCM
Latency Protection          : Off
Protocol                    : UDP/IP    Packet Size(Bytes) : 1400
Bandwidth (Mbps)            : Maximum Available
Compression                 : Enabled
```

| Data Volumes | Size | Layout | DCM Log |
|---|---|---|---|
| v2 | 500 MB | Concatenated | Yes |

Messages are not being filtered. Currently showing 0 messages. Configure this view...

| ! | Source | Classification | Message | Received |
|---|---|---|---|---|

Clicking the Secondary RVG tab in the right pane displays the following information in the upper part of the VEA window.

The following table describes the Secondary RVG fields.

**Table 5-8** Secondary RVG information

| Displayed Field | Description |
|---|---|
| Primary RVG | Displays the name of the Primary RVG. |
| Secondary Host | Displays the host name or IP address of the Secondary that is used for replication. |
| Dynamic Disk group | Displays the name of the dynamic disk group, whose volumes are a part of the RVG.<br>If the RVG is part of a clustered disk group, then the disk group name is displayed with a `Cluster` tag against it. |
| Data Volumes | Displays the information about the number of data volumes that are associated with the Secondary RVG. |

**Table 5-8** Secondary RVG information *(continued)*

| Displayed Field | Description |
|---|---|
| RVG State | Displays the state of the RVG.<br>See "RVG states" on page 117. |
| Replicator Log Size | Displays the size of the Replicator Log. |
| Replication Mode | Displays the current mode of replication. The different modes are, synchronous, asynchronous, and synchronous override.<br>See "Modes of replication" on page 28. |
| Replication Status | Displays the current status of replication.<br>See "Replication status" on page 120. |
| Replicator Log Protection | Displays the value that has been set for Replicator Log protection, that is, `Autodcm`, `DCM`, `Off`, `Fail`, or `Override`.<br>See "Replicator Log overflow protection—`srlprot` attribute" on page 48. |
| Latency Protection | Displays the value that has been set for Latency protection, that is, `OFF`, `FAIL`, `Override`.<br>See "Latency protection—`latencyprot` attribute" on page 52. |
| Protocol | Displays the protocol that Volume Replicator uses for sending data from Primary to Secondary during replication. UDP/IP is the default replication protocol. However, you can use either UDP/IP or TCP/IP.<br>Displays STORAGE in the case of a Bunker Secondary where the storage on the Bunker Secondary is directly accessible from the Primary and STORAGE protocol has been used. |
| Packet Size (Bytes) | Displays the size of the packet that is used to send the data to Secondary when the UDP protocol is used. |
| Bandwidth (Mbps) | Displays the bandwidth that Volume Replicator uses. The default unit is Megabits per second (Mbps) and the minimum allowed value is 1 Mbps.<br>If no value has been specified Volume Replicator uses the available bandwidth by default. In this case this field displays the `Maximum Available` value. |
| Compression | Specifies whether compression is enabled or disabled. |

**Table 5-8** Secondary RVG information *(continued)*

| Displayed Field | Description |
|---|---|
| Replication Time Lag | Displays the exact number of hours, minutes, and seconds by which the Secondary is behind the Primary. The current time on the Primary is also displayed. Note that this field is displayed when the Primary becomes unavailable. This information helps you to decide which Secondary should take over the Primary role in a setup with multiple Secondaries, when a disaster occurs. |

After this, the Secondary RVG view displays information about the RLINKs that are configured for the selected Primary RVG. This is similar to the Primary RLINK information.

The following table describes the fields that display the Secondary RVG data volume information.

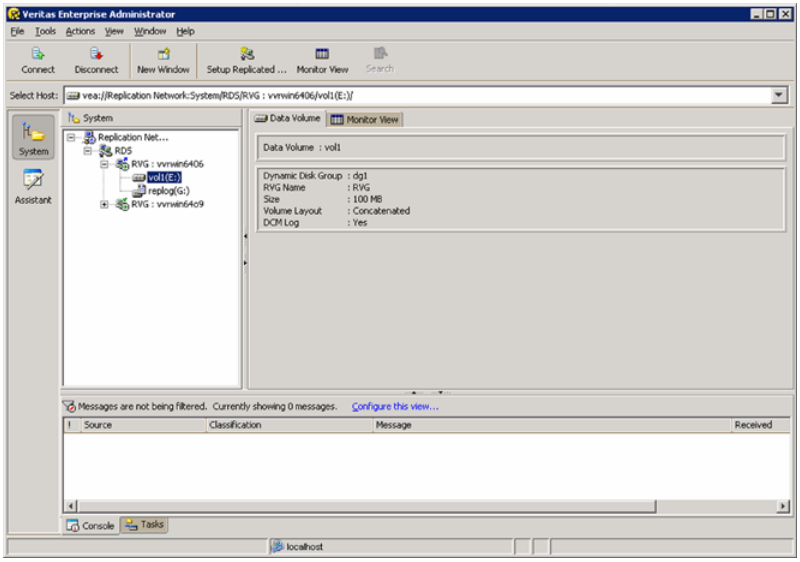
**Table 5-9** Secondary RVG data volume information

| Displayed Field | Description |
|---|---|
| Data Volumes | Displays the names of the data volumes that are associated with the RVG. |
| Size | Displays the size of the data volumes. |
| Layout | Displays the type of volume layout, that is:<br>■ Concatenated<br>■ Mirrored Concatenated<br>■ Striped<br>■ Mirrored Striped<br>■ Mixed<br>For more information about the volume layout, see *Storage Foundation Administrator's Guide*. |
| DCMLog | Displays whether the DCM log is present. Valid values are:<br>■ Yes<br>Indicates that the volume has a DCM log.<br>■ No<br>Indicates that the volume does not have a DCM log. |
| Primary Volume | Displays the name of the corresponding Primary data volume. |

# Viewing information about the Primary data volume

Select the required Primary data volume from the Replication Network tree view, in the left pane. The right pane displays the data volume view with all the related information. This information is similar to the data volume information that is displayed in the lower part of the Primary RVG view.

**Figure 5-4** VEA console: Primary data volume information



Select Host: vea://Replication Network:System/RDS/RVG : vvrwin6406/vol1(E:)/

Data Volume : vol1

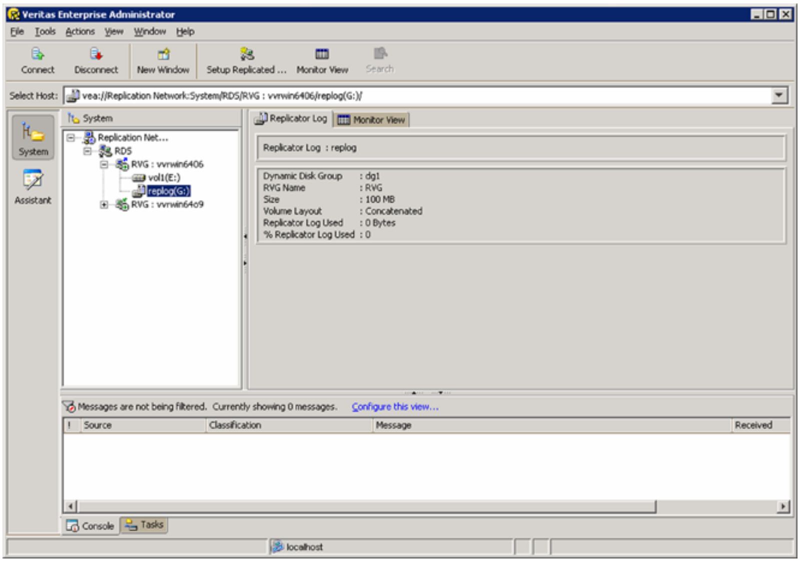
Dynamic Disk Group : dg1
RVG Name : RVG
Size : 100 MB
Volume Layout : Concatenated
DCM Log : Yes

Messages are not being filtered. Currently showing 0 messages. Configure this view...

# Viewing the Replicator Log volume information

Select the Replicator Log volume from the expanded view of the Replication Network node in the left pane. The right pane displays the Replicator Log volume with all the information related to Replicator Log volume.

**Figure 5-5** VEA console: Replicator Log volume Information



Select Host: vea://Replication Network:System/RDS/RVG : vvrwin6406/replog(G:)/

- Replication Net...
  - RDS
    - RVG : vvrwin6406
      - vol1(E:)
      - replog(G:)
    - RVG : vvrwin64o9

Replicator Log : replog

```
Dynamic Disk Group      : dg1
RVG Name                : RVG
Size                    : 100 MB
Volume Layout           : Concatenated
Replicator Log Used     : 0 Bytes
% Replicator Log Used   : 0
```

Messages are not being filtered. Currently showing 0 messages.

As the Replicator Logs on the Primary and corresponding Secondary hosts have the same properties, the Replicator Log volume view is similar for the Primary and the corresponding Secondary.

The following table describes the Replicator Log volume view.

**Table 5-10** Replicator Log volume view

| Displayed Field | Description |
|---|---|
| Dynamic Disk Group | Displays the name of the disk group to which the Replicator Log volume belongs. |
| RVG Name | Displays the name of the RVG to which the Replicator log is associated. |
| Size | Displays the size of the Replicator log in appropriate units. |
| Volume Layout | Displays the type of volume layout, that is:<br>■ Concatenated<br>■ Mirrored Concatenated<br>■ Striped<br>■ Mirrored Striped<br>■ Mixed<br>For more information about the volume layout, see *Storage Foundation Administrator's Guide*. |

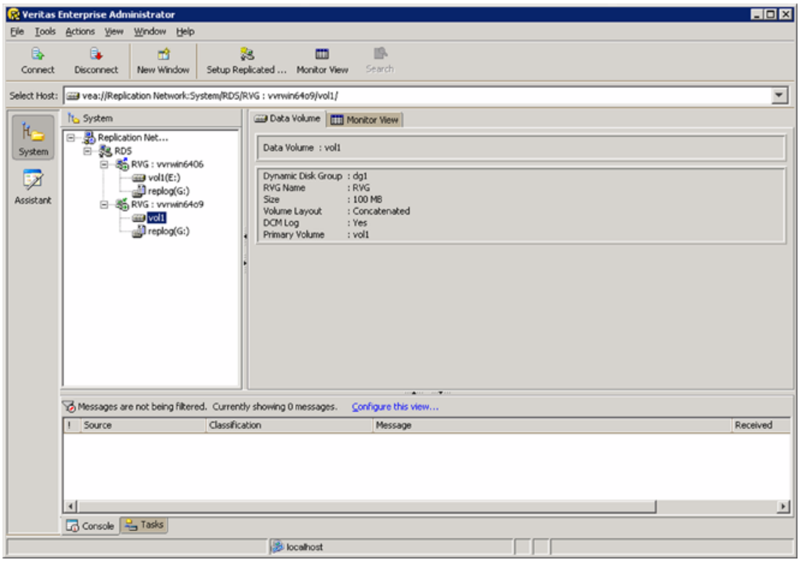
**Table 5-10** Replicator Log volume view *(continued)*

| Displayed Field | Description |
|---|---|
| Replicator Log Used | Displays the amount of the total allocated space that the Replicator Log uses, in appropriate units. |
| % Replicator Log Used | Displays the percentage of the total Replicator Log space that has been used. |

## Viewing information about the Secondary data volume

Select the required Secondary data volume from the Replication Network node in the tree view of the left pane. The right pane displays the data volume view with all the related information, which is similar to the data volume information that is displayed in the lower part of the Secondary RVG view.

**Figure 5-6** VEA console: Secondary data volume information



Select Host: vea://Replication Network:System/RDS/RVG : vvrwin64o9/vol1/

Data Volume : vol1

Dynamic Disk Group : dg1
RVG Name : RVG
Size : 100 MB
Volume Layout : Concatenated
DCM Log : Yes
Primary Volume : vol1

Messages are not being filtered. Currently showing 0 messages.

# Monitoring replication using the VEA console

Volume Replicator provides you with the Monitor View that enables you to monitor the replication progress. The monitor view, which is a tabular view, gives you a complete picture of the replication activity within the replication network and can be invoked in a separate window by using the Monitor view option.

The following topics describe the tasks that help you to use the Monitor view effectively and to obtain the required information:

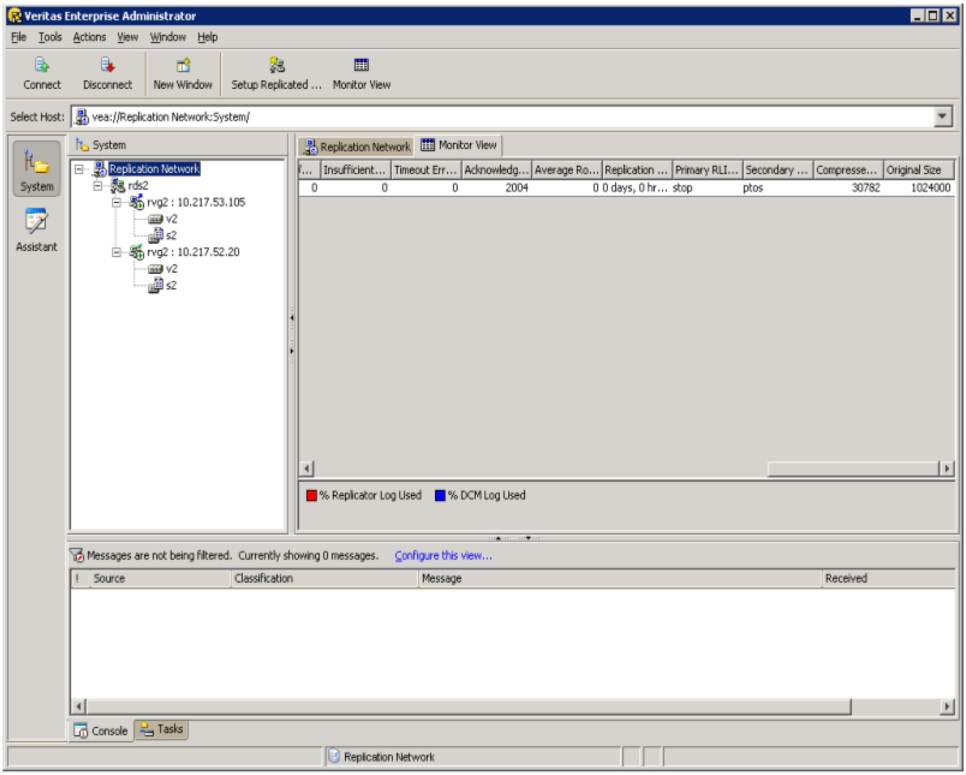
- See "Displaying the monitor view" on page 132.
- See "Specifying preferences for the monitor view" on page 133.
- See "Interpreting the information in the monitor view" on page 134.

## Displaying the monitor view

Volume Replicator provides the option to display the monitor view from the Menu bar and the toolbar. The right pane also displays the Monitor View option in every view. When Replication Network node is selected, the **Monitor View** tab displays the information about replication activity for all the available RDSs in the right pane. When you select an RDS or any node under that RDS, the Monitor View tab in the right pane displays information about the replication activity only for that RDS.

The monitor view that can be invoked from the Actions menu on the toolbar, displays the statistical information for all the RDSs under Replication Network node, in a separate window.

Select the Monitor View tab that is displayed in the right pane when Volume Replicator objects are selected. You can toggle the right pane view between the specific object view and the Monitor view. Alternatively, you can also select the Monitor View option from the Actions menu. The monitor view is displayed in a separate window.



Select Host: vea://Replication Network:System/

System
- Replication Network
  - rds2
    - rvg2 : 10.217.53.105
      - v2
      - s2
    - rvg2 : 10.217.52.20
      - v2
      - s2

| /... | Insufficient... | Timeout Err... | Acknowledg... | Average Ro... | Replication ... | Primary RLI... | Secondary ... | Compresse... | Original Size |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 2004 | 0 | 0 days, 0 hr... | stop | ptos | 30782 | 1024000 |

Messages are not being filtered. Currently showing 0 messages. Configure this view...

| ! | Source | Classification | Message | Received |
|---|---|---|---|---|

Replication Network

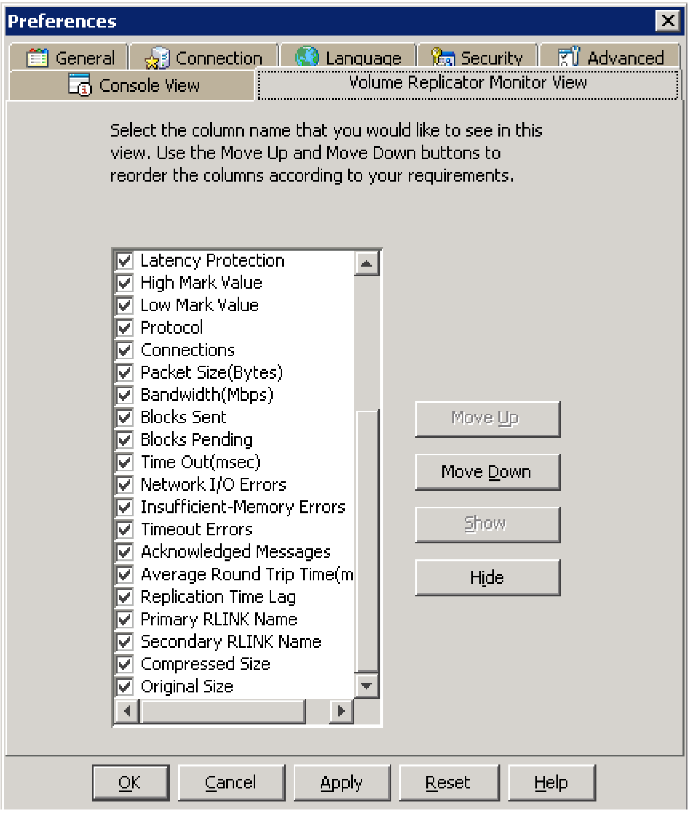
The Monitor View provides statistical information in a tabular format which is organized across rows. The Monitor View has scroll bars to help you move across the screen from left to right and vice versa. You can also choose to display only those columns that you require by specifying preferences.

## Specifying preferences for the monitor view

Although the monitor view allows a user to scroll across the length and breadth of the view, it may be helpful to display only the required columns. Based on the fields that you select in the **Preferences** dialog, the appropriate columns are displayed. By default, the monitor view displays all the available columns.

**To choose the columns that you want to display in the monitor view**

1. Select **Tools > Preferences**. The Preferences dialog box is displayed.
2. Select the **Volume Replicator Monitor View** tab. The Preferences dialog now displays a list of column names.



Select the column name that you would like to see in this view. Use the Move Up and Move Down buttons to reorder the columns according to your requirements.

- ☑ Latency Protection
- ☑ High Mark Value
- ☑ Low Mark Value
- ☑ Protocol
- ☑ Connections
- ☑ Packet Size(Bytes)
- ☑ Bandwidth(Mbps)
- ☑ Blocks Sent
- ☑ Blocks Pending
- ☑ Time Out(msec)
- ☑ Network I/O Errors
- ☑ Insufficient-Memory Errors
- ☑ Timeout Errors
- ☑ Acknowledged Messages
- ☑ Average Round Trip Time(m
- ☑ Replication Time Lag
- ☑ Primary RLINK Name
- ☑ Secondary RLINK Name
- ☑ Compressed Size
- ☑ Original Size

Select the column names that you want to display in the Monitor View by clicking on the check boxes beside each field. Note that if you want to display the default fields, click **Reset**, and then click **OK**. The **Monitor** view is displayed with the default fields selected.

3 Use the Move Up and Move Down options to position the columns in the monitor view display according to your requirement.

4 Click **OK** to confirm the changes.

## Interpreting the information in the monitor view

The information that is displayed in the monitor view helps you to understand and track the replication progress. The following sections describe how you can interpret the information in the monitor view to obtain the required statistics. It also explains how to interpret and understand the error conditions.

The information fields that are displayed in the Monitor view correspond to those in the output of the `vxrlink stats` command and the `vxrlink status` command. The `vxrlink stats` command is used to obtain the network statistics when you work with command line options whereas the Monitor view can be used when you work with the graphical user interface.

See "Checking replication performance using `vxrlink stats` " on page 139.

See "Displaying the RLINK status" on page 267.

Each row in the monitor view represents information for an RDS, and displays information such as the Primary RVG for the RDS, Log Usage by the Primary RVG and the Secondary hosts that are associated with the RDS. If there are multiple Secondary hosts, then each of them is listed in a separate row. The Monitor View has scroll bars that help you to move across the window from left to right and vice versa. You can change the width of the columns by dragging the column separators to suit your requirements. If the host has multiple RDSs then the monitor view displays information for all the RDSs.

The following tables explain the columns that are displayed in the monitor view. They have been grouped according to the purpose they serve.

## Configuration information

The columns that are described in the following table can be used to obtain the complete configuration information without having to go through the individual views that are provided for each object. Each of the columns provides information about a specific Volume Replicator object within an RVG.

**Table 5-11** Obtaining configuration information

| Name | Description |
|---|---|
| Replicated Data Set | Displays the name of the current RDS. If there are multiple RDSs, then the information for each of the RDSs is displayed in the Monitor view. In this case, the Monitor view has a list of RDSs in this column. |
| Primary RVG | Displays the Primary RVG name. If there are multiple RDSs then this column lists the name of the Primary RVG within each RDS. |
| Secondary | Displays the name of the Secondary RVG, corresponding to the Primary RVG. If there are multiple RDSs, then the RVG information for each RDS is displayed. If the RDS has multiple Secondaries, then, the information for each of these Secondaries is displayed in a separate row. |
| Replication Mode | Displays the current mode of replication. |

**Table 5-11** Obtaining configuration information *(continued)*

| Name | Description |
|---|---|
| Latency Protection | Displays the current Latency Protection setting for the Secondary RVG. |
| High Mark Value | Displays the maximum number of units by which the Secondary can lag behind. |
| Low Mark Value | Displays the value to which the latency protection must be reset, once it reaches the high mark value. Incoming writes are stalled until this value is reached. |
| Replicator Log Protection | Displays the current setting for the Replicator Log Protection, for the Secondary RVG. |
| Protocol | Displays the protocol that Volume Replicator uses for sending data from Primary to Secondary, during replication. UDP/IP is the default replication protocol. However, Volume Replicator can use either UDP/IP or TCP/IP.<br>Displays STORAGE in the case of a Bunker Secondary where the storage on the Bunker Secondary is directly accessible from the Primary and STORAGE protocol has been used. |
| Connections | Displays the number of TCP connections when replication is carried out in the TCP/IP mode. |
| Packet Size (Bytes) | Displays the packet size that has been specified for transferring the data from the Primary to the Secondary. The packet size is displayed in bytes when the replication is carried out in the UDP/IP mode. |
| Bandwidth (Mbps) | Displays the maximum bandwidth that Volume Replicator can use when replicating to the Secondary. The default unit is Megabits per second (Mbps). |
| Primary RLINK Name | Displays the name of the Primary RLINK. If you had specified a name when setting up the RDS, that name is displayed. Otherwise, the default name is displayed. |
| Secondary RLINK Name | Displays the name of the Secondary RLINK. If you had specified a name when setting up the RDS that name is displayed. Otherwise, the default name is displayed. |
| Compressed Size | Displays the data size after compression. |
| Original Size | Displays the original data size. |

## Log usage information

Monitoring the Replicator Log usage can be very useful during normal replication and especially when there is a high rate of writes to the Primary RVG volumes. This information is available by viewing the display in the Log Usage column. This column displays both the Replicator Log and DCM log usage, separately. It displays the Log usage as a percentage value. Note that when the Replicator Log is 80 percent full an alert message is displayed in the bottom pane. The message displays the name of the RDS with the Secondary host name for which the log is full. A red progress bar indicates the Replicator Log usage.

This column also displays the DCM log usage, which is indicated by a blue progress bar, along with a percentage value of the usage. The DCM log is used for autosynchronization or resynchronization, when the Replicator Log overflows or for fast-failback logging after Takeover. After the Secondary is fully synchronized, the DCM log usage display changes over to Replicator Log usage.

**Table 5-12** Obtaining log usage information

| Name | Description |
|---|---|
| Log Usage | Displays the percentage of the log used. The ToolTip that appears when you move the mouse pointer over this field indicates whether the display is for Replicator Log Usage or DCM Log Usage. |

## Obtaining replication status information

The following table explains the fields of the Monitor view that can be used to obtain replication status information.

**Table 5-13** Replication status and RVG states

| Name | Description |
|---|---|
| Primary RVG State | Displays the current state of the Primary RVG.<br>See "RVG states" on page 117. |
| Secondary RVG State | Displays the current state of the Secondary RVG.<br>See "RVG states" on page 117. |
| Replication Status | Display the current replication status.<br>See "Replication status" on page 120. |

## Obtaining statistical information

The **Monitor** view enables you to obtain replication statistics with the help of the information that is displayed in the following columns. Each of these columns provide important statistical information that you can use to interpret the current state of replication.

The following table describes information pertaining to replication statistics.

**Table 5-14** Obtaining information about replication statistics

| Name | Description |
|---|---|
| Acknowledged Messages | Indicates the number of messages that the Secondary has received, the acknowledgment for which has already been sent to the Primary. The acknowledgement is sent for every packet that is received. |
| Average Round-Trip Time | Displays the time that is required for the average round-trip of the message in milliseconds, that is, the message is sent and acknowledged only when the Secondary fully receives it.<br>This is dynamically calculated, and may vary based on the various factors such as the network bandwidth, the packet size, and processing capabilities of the hosts. |
| Blocks Sent | Displays the number of blocks that have already been sent to the Secondary RVG. One block consists of 512 bytes. |
| Blocks Pending | Displays the number of blocks that are pending, that is, they have not yet been sent to the Secondary RVG and are queued onto the Replicator Log. |
| Replication Time Lag | Displays the exact number of hours, minutes, and seconds by which the Secondary is behind the Primary. This is the difference between the time when the latest update that arrived on the Primary and the time when the last update that arrived on the Primary and was acknowledged by the Secondary. The time for each update is noted when it is written to the Primary Replicator Log.<br>If the Replication Time Lag is zero then this indicates that the Secondary is up-to-date. If the Replication Time Lag displays a value then it indicates that the Secondary is behind the Primary. |

## Interpreting error information

The Monitor view enables you to obtain different error statistics with the help of the information that is displayed in various columns. Each of these error conditions points to a specific problem.

The following table explains the fields of the Monitor view that can be used to obtain error information.

**Table 5-15** Obtaining information about Error Conditions

| Name | Description |
|---|---|
| Network I/O Errors | Indicates the number of network errors that occurred, which affected replication. |
| Insufficient Memory Errors | This error is primarily reported on the Secondary when the Secondary cannot handle a particular packet due to insufficient memory, which in turn may affect replication. In most cases however, built in flow control manages this problem automatically. |
| Time-out Errors | Indicates the number of time-out errors that affects replication. Time-out errors may occur for reasons such as, dropped packets, or unacknowledged packets due to which the Primary does not receive acknowledgement within the specified time period. |

# Checking replication performance using `vxrlink stats`

The `vxrlink stats` command reports detailed information about replication statistics, which can be used to assess network problems. This information about the network performance can be used to determine the optimum network configuration for efficient use of system resources. The `vxrlink stats` command can be executed only from the Primary. The **Monitor view** option from the VEA is the parallel for the `vxrlink stats` command output in the GUI and is available both from the Primary and Secondary.

See "Displaying the network statistics for the RLINK" on page 264.

See "Monitoring replication using the VEA console" on page 131.

**Note:** All the statistics that the `vxrlink stats` command displays, are reinitialized when the replication is restarted, either because of a user command or because of the network or server outage.

The following table describes the output of the `vxrlink stats` command.

**Table 5-16** `vxrlink stats` command output: Information Messages

| Field Name | Description |
|---|---|
| `#` | Displays the number of messages transmitted. |
| `Blocks` | Displays the number of blocks that are transmitted to the Secondary RVG. One block consists of 512 bytes. |
| `RT (msec)` | Displays the average round-trip time. |
| `Delays` | Displays the delay that Volume Replicator introduces while sending the packets, if it is flow controlled. Usually, delays are introduced when there are errors on the link or the outstanding bytes for flow control have been exceeded for a single message. |

The following table describes the output of the `vxrlink stats` command.

**Table 5-17** `vxrlink stats` command output: Error Information

| Field Name | Description |
|---|---|
| `Timeout` | Displays the number of time-out errors. A time-out error occurs when an acknowledgement for a message is not received from the remote host within the computed time-out period. The time-out period is automatically adjusted for optimum performance based on round-trip time (RT). |
| `Stream` | Displays the errors that occur while sending the updates on the network, which can include errors due to insufficient memory, errors returned by the underlying protocol driver and so on. |
| `Memory` | Displays the number of memory errors. Memory errors generally occur when the Secondary is unable to store the out of order packets that it receives. One reason for this may be because the Secondary has insufficient buffer space to handle incoming messages or the earlier messages still have some packets pending. This can be fixed by increasing the `NMCOM_POOL_SIZE` tunable on the Secondary. |

The following table describes how the flow control reacts to the errors that are displayed for `vxrlink stats` command.

Table 5-18 `vxrlink stats` command output: Flow control

| Field Name | Description |
|---|---|
| `NW Bytes` | Displays the number of bytes that can be transmitted without flow controlling and introducing any intervening delays.<br>If an RLINK does not experience network errors, Volume Replicator steadily increases the NW Bytes to permit more data to be transmitted. If an RLINK experiences network error, Volume Replicator tries to perform flow control by reducing this number. The minimum value is 5000 bytes. |
| `NW Delays` | Displays the delay that Volume Replicator may introduce while sending the packets, if it was flow controlled. Usually, delays are introduced when there are errors on the link or the outstanding bytes for flow control have been exceeded for a single message. |
| `Timeout` | Displays the current time-out value in milliseconds. This value is computed dynamically. If an acknowledgement for a message is not received from the remote host within this value, the message is considered lost and is retransmitted. |

## Identifying the most up-to-date Secondary

The `vxrlink updates` command enables you to identify the most up-to-date Secondary in a Volume Replicator configuration. The `vxrlink updates` command can be issued only on a Secondary.

See "Identifying the most up-to-date Secondary" on page 269.

You can also identify the most up-to-date Secondary through the VEA, by checking the value that is displayed for the Replication Time Lag property in the Secondary RVG view.

See "Viewing information about the Secondary RVG" on page 125.

See "Interpreting the information in the monitor view" on page 134.

# Analyzing Volume Replicator performance

You can now analyze the Volume Replicator performance through the performance monitor (perfmon), which is a utility that the Windows operating system provides. This utility can be launched by executing the `perfmon` command.

To be able to monitor the Volume Replicator performance, the performance objects that have been added to perfmon are as follows:

- Volume Replicator Memory

- Volume Replicator Remote hosts

Each of these performance objects includes a set of performance counters, which are used for logging the Volume Replicator performance-related information. For logging information you must create the log file with the required parameters. To do this, right-click the **Counter Log** from the tree view and select **New Log Settings** form the menu that appears.

For more information about using the performance monitor, refer to the help that is available from the Help button on the performance monitor console.

**Note:** When setting the properties for a new log file on a system running Windows Server, you must specify an account with administrative privileges to run the log. Otherwise, the log file fails to gather the required information.

The Volume Replicator Memory object includes the parameters available with the `vxmemstat` command, whereas the Volume Replicator remote hosts object includes a combination of parameters available with the `vxrlink stats` command and the `vxrlink status`.

See "Checking replication performance using `vxrlink stats` " on page 139.

See "Displaying the RLINK status" on page 267.

The Volume Replicator objects can be viewed using the different graphical view options that `perfmon` provides.

The following types of Volume Replicator objects can be viewed:

- Volume Replicator Remote Host object
  Volume Replicator creates the Volume Replicator Remote host object.
- Volume Replicator Memory Object
  Volume Replicator creates the Volume Replicator Memory Object.

**Table 5-19** Performance object counters and their descriptions for Volume Replicator Remote Host object

| Performance Counter Names | Description |
|---|---|
| Data Transmitted (KB) | The amount of data that is successfully transmitted to the remote host.<br>The following table lists the performance object counters with their descriptions for Volume Replicator Remote Host object. |
| DCM Usage (%) | Indicates the percentage of DCM that is used, based on the number of bits marked in the DCM log. |

**Table 5-19** Performance object counters and their descriptions for Volume Replicator Remote Host object *(continued)*

| Performance Counter Names | Description |
|---|---|
| Delays | The total amount of delay that has been introduced so far after flow control was enforced. |
| Flow Control NW Bytes | Number of bytes which can be transmitted without imposing flow control measures. |
| Flow Control NW Delay | The delay that is introduced while sending data, so as to enforce flow control. |
| Flow Control Timeout | Indicates a dynamically computed time-out value for the acknowledgement of a message that has already been sent. If no acknowledgement is received within the time-out period, then retransmission is attempted. |
| Lost Packets | Displays the rate at which replication data packets are lost. |
| Memory Errors | Displays the errors due to insufficient memory. |
| Round Trip Time (msec) | Displays the average round-trip time that is required for transmitting and acknowledging the replication messages. |
| SRL Requests | Displays the number of updates pending on the Replicator Log. |
| Stream Errors | Displays the errors due to insufficient bandwidth. |
| Used SRL (%) | Displays the percentage of the Replicator Log used for recording updates or the writes that need to be replicated. |

The following table lists performance counters with their descriptions for Volume Replicator Memory object.

**Table 5-20** Performance counters and description associated with Volume Replicator Memory Object

| Field Name | Description |
|---|---|
| Allocated `NMCOM` Pool (KB) | Memory that the Secondary allocates to hold the updates that are received from the Primary. |
| Allocated `READBACK` Memory Pool (KB) | Memory that is allocated for holding updates after reading them from the Replicator Log. |

**Table 5-20** Performance counters and description associated with Volume Replicator Memory Object *(continued)*

| Field Name | Description |
|---|---|
| Allocated VOLIO Memory Pool (KB) | Memory that is allocated by the Primary to hold the updates for replicating them. |
| Used NMCOM Pool (KB) | Displays the currently used portion of the allocated NMCOM Pool. |
| Used READBACK Pool (KB) | Displays the currently used portion of the allocated READBACK Pool. |
| Used VOLIO Memory Pool (KB) | Displays the currently used portion of the allocated VOLIO memory Pool. |
| WaitQ for VOLIO Memory | Displays the number of updates waiting for free memory in the VOLIO Memory Pool. |

# Monitoring alerts to interpret error conditions

The console or the lower pane of the VEA displays alerts on the Volume Replicator related tasks when you select the Console tab from the lowermost left corner of the VEA console. The alerts can be classified as information messages, warnings, or errors and are identified by the icon that is displayed beside the alert.

Reading the console when performing the various Volume Replicator related tasks helps you to understand whether the current task that you perform is progressing as required.

# Handling Volume Replicator events

VEA provides the option to set up rule based monitoring in response to events. Volume Replicator supports this feature and you can set up rules to detect conditions or the events that you want to monitor. The rules that are created include the actions that are performed when the product detects specified conditions. You can use the Rule Manager to set up configurations for the SNMP server and the default senders. For more information about setting up the SNMP refer to the online Help that is available from the VEA console's Help option. Select Contents from the Help menu. The Help window appears. From the Select help set drop-down list select the Optional Rules for Handling Events.

You can use variables to provide meaningful information about the alerts you monitor. Each variable is based on an alert attribute.

The list of alert attributes that are common to all the Volume Replicator messages for which the SNMP traps are generated are as follows:

- Alert Severity
- Alert Message
- Recommended Action
- Friendly Alert Name
- RDS Name

The following table summarizes alert attributes for Volume Replicator messages.

**Table 5-21** Alert attributes

| Attributes | Description |
|---|---|
| Alert severity | The severity of the alert.<br>Following are the severity values:<br>■ critical - 1<br>■ error - 2<br>■ warning - 3<br>■ informational - 4 |
| Alert message | The message that has been defined for the alert. You can define a different message for every alert. |
| Recommended action | The recommended action that has been suggested for the alert. |
| Friendly Alert Name | A name that has been provided to make the alert easy to understand. |
| RDS Name | Specifies the RDS name for which the specified event has occurred. |
| RVG Name | RVG name that is associated to the <*RDS Name*> for which the event has occurred. |
| Secondary Host Name | The name of the Secondary host for which the event has occurred. This can be used only for some messages. |
| Primary Host Name | The name of the Primary host for which the event has occurred. This can be used only for some messages. |
| SRL Usage | The percentage of the Replicator Log that has already been used. Once the Replicator Log is 80% full an alert message is automatically generated. |

Chapter 6

# Administering Volume Replicator

This chapter includes the following topics:

## About administering Volume Replicator

This chapter describes the tasks that enable you to administer the RDS, RVG, Replicator Log, and the data volumes using the VEA GUI.

The tasks specific to a Volume Replicator object are available from the right-click menu that appears when the object is selected from the Actions menu. Within this document the tasks have been grouped according to the function they perform. You can use the Properties option that is available from the object > right-click menu to view the properties of each object. For example, the **Properties** option that is available on the RDS right-click menu displays the properties of the RDS.

---

**Note:** For some operations, Volume Replicator checks the volumes and locks them before proceeding any further. The operations that require the volumes to be locked are Disable Data Access, Migrate, and Takeover.

---

Most of the tasks that can be performed using VEA menus can also be performed using the command line options.

See “About using the command line interface” on page 227.

The following sections describe the procedure to perform each of the tasks using the VEA menus.

# Modifying the configuration

This section describes tasks such as, adding new volumes and Secondary hosts to the existing configuration, which you can perform to effect configuration changes. These tasks affect the RDS as a whole and in turn affect replication.

The tasks described in this section are as follows:

- Adding volumes. For information, See “Adding volumes” on page 147.
- Adding a Secondary host. For information, See “Adding a Secondary host” on page 153.
- Administering the RVG. For information, See “Administering the RVG” on page 163.

# Adding volumes

Using this option, you can add additional volumes to an RDS even when replication is in progress. This command associates a volume to all the RVGs of the RDS. Note that the **Add Volume** wizard provides you with the option to create volumes on the Secondary host, corresponding to those on the Primary, if they are not already created. However, you can also choose to create the volumes on the Secondary hosts beforehand and then use this wizard to add the volumes to the RDS.

The options available on this wizard vary depending on whether you have created the volumes on the required hosts.

**Note:** When you create the volumes on the Secondary host, to prevent these volumes from being mounted, Veritas recommends that you do not assign a drive letter to these volumes. Otherwise, the file system on the volume may try to mount these volumes and report an error that the volume is corrupt because the data on the volume changes due to replication.

## Prerequisite for adding data volumes to an RDS

Verify that the volumes to be added to the RDS have already been created on the Primary host. By default, Volume Replicator adds the Data Change Map (DCM) log to all volumes that are selected to be a part of the RDS. If the disk space available is not adequate for creating DCM with mirrored plexes, then, Volume Replicator creates DCM with a single plex.

Although, you can add the data volume to the RDS even when replication is in progress, there is no way to synchronize the newly added volumes using Volume Replicator. Veritas recommends that you synchronize the data volumes first, using the methods such as Backup and Restore and then add them to the RDS.

See “Setting up replication using the Setup Replicated Data Set wizard” on page 81.

**To add data volumes to an RDS**

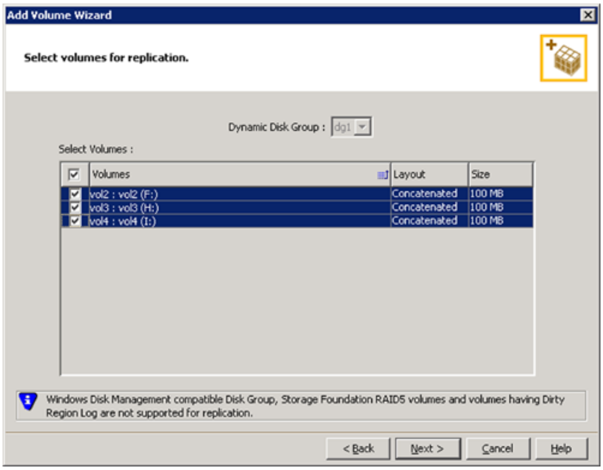
1. Select the required RDS node from the tree display in the left pane and select the **Add Volume** option from the RDS right-click menu. A message box appears.

   Read the information that is provided in the message box carefully. To proceed with adding new volumes, click **Yes**.

2. On the **Welcome** panel of the Add Volume wizard click **Next**.

   If VEA is not connected to the Primary, the wizard tries to connect to it. Wait till the connection process is complete and then click **Next** again.

3 Complete the Select volumes for replication panel as follows to specify the data volumes that you want Volume Replicator to replicate.



Add Volume Wizard

Select volumes for replication.

Dynamic Disk Group : dg1

Select Volumes :

| ☑ | Volumes | Layout | Size |
|---|---|---|---|
| ☑ | vol2 : vol2 (F:) | Concatenated | 100 MB |
| ☑ | vol3 : vol3 (H:) | Concatenated | 100 MB |
| ☑ | vol4 : vol4 (I:) | Concatenated | 100 MB |

Windows Disk Management compatible Disk Group, Storage Foundation RAID5 volumes and volumes having Dirty Region Log are not supported for replication.

Complete the information on this panel as follows:

Dynamic Disk Group

This field displays the disk group that the Primary RDS uses.

Select Volumes

Choose the required data volumes from the table by selecting the check boxes for the volumes. To select all the volumes select the check box present in the top left corner of the Select Volumes table.

You can also select multiple volumes using the Up or Down arrow key, while holding down the Shift or Control keys.

If you have created snapshot volumes then these volumes are also available for selection.

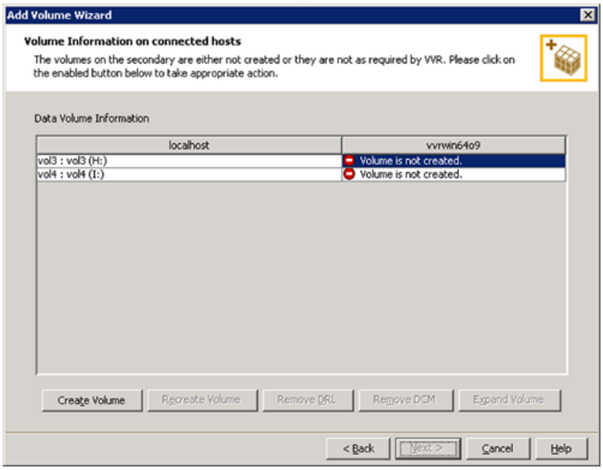
After specifying the required information, click **Next**.

If VEA is not connected to the Secondary hosts, the wizard tries to connect them. Wait till the connection process is complete and then click **Next** again.

4 The Volume information about connected hosts panel appears. This panel displays information about the availability of volumes on the Secondary nodes, if the Primary and Secondary hosts are connected to VEA.

This panel does not appear if all the required volumes that are available on the Primary host are also available on the Secondary hosts.

**5** If the required disk group has been created, but the data volumes and the Replicator Log have not been created on the Secondary host, then the panel displays the appropriate message against the volume name on the Secondary.



Add Volume Wizard

**Volume Information on connected hosts**

The volumes on the secondary are either not created or they are not as required by VVR. Please click on the enabled button below to take appropriate action.

Data Volume Information

| localhost | vvrwin64o9 |
|---|---|
| vol3 : vol3 (H:) | Volume is not created. |
| vol4 : vol4 (I:) | Volume is not created. |

- Because the volume is not created, the **Create Volume** option is enabled. Click this option to create the required volumes on the Secondary host, corresponding to those on the Primary.
- The **Create Volume** dialog automatically displays the volume name and the size after verifying the information about the Primary host.

  Complete the information as follows:

  Name

  Displays the name for the volume in the **Name** field. This is the same as that specified for the Primary volume.

  Size

  Displays the size of the volume in the **Size** field. This is the same as that specified for the Primary volume.

  Layout

  Specify the volume layout. Select the appropriate option depending on your requirement.

  Disks Selection

  Enables you to specify the disk selection method.

  Select the **Select disks automatically** option if you want Volume Replicator to select the disks.

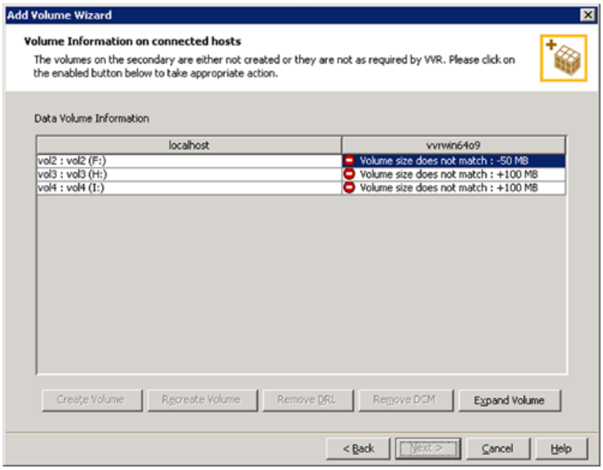
  Select the **Select disks manually** option to use specific disks from the Available disks pane for creating the volume. Either double-click on it or select the **Add** option to move the disks into the Selected disks pane.

After verifying the information click **OK** to create the required volume. You are then taken back to the **Volume information about connected hosts** panel.

Repeat the above steps for data volumes and Replicator Log that has not been created.

- After all the volumes have been created, the volume information panel is updated to display the volumes on the Primary and Secondary host.
- Click **Next**.

**6** If the required disk group and the volumes have been created but these volumes are not eligible for replication, then the reason for non-eligibility is indicated against the volume name.



Add Volume Wizard

**Volume Information on connected hosts**

The volumes on the secondary are either not created or they are not as required by VVR. Please click on the enabled button below to take appropriate action.

Data Volume Information

| localhost | vvrwin64o9 |
|---|---|
| vol2 : vol2 (F:) | ⊖ Volume size does not match : -50 MB |
| vol3 : vol3 (H:) | ⊖ Volume size does not match : +100 MB |
| vol4 : vol4 (I:) | ⊖ Volume size does not match : +100 MB |

The Volume information on connected hosts panel enables the appropriate option to convert a non-eligible volume to a Volume Replicator acceptable format.

Complete the information on this panel as follows:

Recreate Volume

This option is enabled if the required data volume is available on the Secondary, but is of a size greater than the Primary volume.

Clicking this option displays a message that prompts you to confirm that you want to recreate the volume.

Choose **Yes** to recreate the volume using the Create Volume dialog.

**Note:** This operation first deletes the volume resulting in loss of the data that already exists on the volumes.

Remove DRL

This option is enabled if the required data volume is available on the Secondary but has a DRL. Clicking this option displays a message that prompts you to confirm that you want to remove the log. Click **Yes** to confirm the removal of DRL.

Remove DCM

This option is enabled if the required Replicator Log volume is available on the Secondary but has a DCM log. Clicking this option displays a message that prompts you to confirm if you want to remove the log. Click **Yes** to confirm the removal of DCM log.

Expand Volume

This option is enabled if the required data volume is available on the Secondary but is of a smaller size than the Primary volume. Clicking this option displays a message that prompts you to confirm that you want to grow the volume.

Click **Yes** to grow the volume to the required size.

After you have converted the non-eligible volumes to a Volume Replicator acceptable format, click **Next**.

7 The Summary panel of the Add Volume wizard appears. Review the information on this panel.

Click **Back** to change any information or click **Finish** to add the specified volumes to the RDS and exit the wizard.

# Adding a Secondary host

Using this option, you can add a Secondary host to the existing Volume Replicator configuration, that is, an RDS and synchronize the Secondary volumes with the Primary data volumes. Before adding the new Secondary host to an existing RDS, you can either choose to create the data volumes on that Secondary host with the same names and sizes as those on the Primary, or you can do it from the Add Secondary wizard. Note that the options on the wizard panels change depending on whether you have created the disk group, the data volumes, and the Replicator Log volume.

**Note:** You can specify only one Secondary host at a time.

**To add a Secondary host**

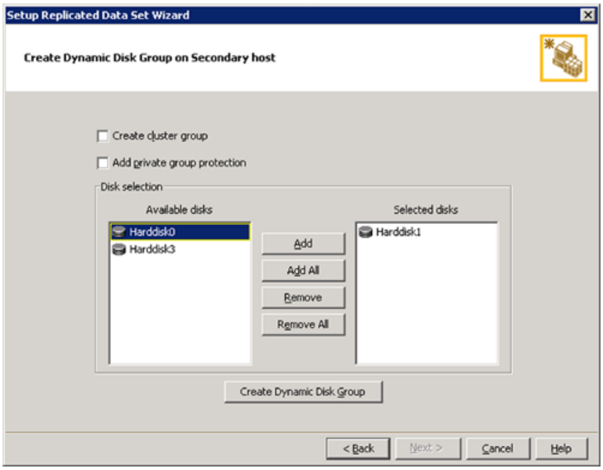
1 Select the **Add Secondary** option from the RDS right-click menu. On the **Welcome** panel click **Next**.

2 The Specify Secondary host for replication panel appears. Enter the name or IP address of the Secondary host in the Secondary Host field. Click **Next**. If the Secondary host is not connected to VEA, the wizard tries to connect it when you click **Next**. Wait till the connection process is complete and then click **Next** again.

- If the disk group with the required data volumes and the Replicator Log volume as on the Primary host does not exist on the Secondary, Volume Replicator displays a message. Read the message, carefully.

  The option to automatically create the disk group, and the associated volumes on the Secondary host is available only if the required number of disks of the same type, having the same or a larger amount of space as that on the Primary is available on the Secondary. Otherwise, the RDS setup wizard enables you to create the required disk group and the volumes manually.

- Click **Yes** to automatically create the disk group, data volumes, and the Replicator Log. When you click **Yes** any available disks are automatically chosen for creating the disk group on the Secondary host.
- Click **No** to manually create the disk group with data volumes and the Replicator Log on the Secondary host. Complete the Create Dynamic Disk Group on Secondary host panel. If the Dynamic Disk group as on the Primary has already been created on the Secondary, then this panel does not appear.



Setup Replicated Data Set Wizard

Create Dynamic Disk Group on Secondary host

☐ Create cluster group

☐ Add private group protection

Disk selection

| Available disks | Selected disks |
|---|---|
| Harddisk0<br>Harddisk3 | Harddisk1 |

Complete the information on this panel as follows:

Create cluster group
: Choose this option only if you need to create clustered disk groups. Select the required disks from the Available disks pane. Either double-click on the host name or click the **Add** option to move the disks into the Selected disks pane. To select all the available disks, choose the **Add All** option.

Create Dynamic Disk Group
: Click **Create Dynamic Disk Group** to proceed with creating the Disk group. A disk group with the same name as that on the Primary is created.

After the disk group has been created, click **Next**. The Volume Information on connected hosts panel appears.

Complete this panel as described in step 3.

If only a disk group without any data volumes or Replicator Log, as on the Primary host, exists on the Secondary, then Volume Replicator displays a message. Read the message, carefully.

The option to automatically create the volumes on the Secondary host, is available only if the disks that are part of the disk group have either the same or a larger amount of space as that on the Primary or enough space to create volumes with the same layout as on the Primary.

Otherwise, the RDS setup wizard enables you to create the required volumes manually.

- Click **Yes** to automatically create the Secondary data volumes and the Replicator Log on the Secondary host. After the configuration has been automatically created on the Secondary, proceed to step 4.
- Click **No** to create the Secondary data volumes and the Replicator Log manually, using the Volume Information on connected hosts panel. Complete this panel as described in step 3.

**3** The Volume Information on connected hosts panel appears. This panel displays information about the availability of volumes on the Secondary nodes, if the Primary and Secondary hosts are connected to VEA.

This panel does not appear if all the required volumes that are available on the Primary host, are also available on the Secondary hosts.

- If the required disk group has been created but the data volumes and the Replicator Log have not been created on the Secondary host, then the panel displays the appropriate message against the volume name on the Secondary.
  Because the volumes have not been created the **Create Volume** option is enabled. Click this option to create the data volumes and the Replicator Log volume on the Secondary host.
- The Create Volume panel automatically displays the volume name and the size after verifying the information about the Primary host. Complete the information on this panel as follows:

| | |
|---|---|
| Name | Displays the name for the volume. This is the same as that specified for the Primary volume. |
| Size | Displays the size for the volume. This is the same as that specified for the Primary volume. |
| Layout | Specify the volume layout. Select the appropriate option depending on your requirement. |

Disks Selection

Enables you to specify the disk selection method.

You can select the following:

- Enable the **Thin Provisioned Only** check box to ensure that the Replicator Log volume is created only on Thin Provisioned (TP) disks.

  **Note:** Note: The check box remains disabled if the disk group does not have any TP disks.

  If this option is selected along with the **Select disks automatically** option, then the Replicator Log volume is created only on TP disks. However, if you enable this check box along with **Select disks manually** option, then the user can select only TP disks from **Available Disks**.

  For more information about Thin Provisioning refer to the *Storage Foundation Administrator's Guide*.
- Choose the **Select disks automatically** option if you want Volume Replicator to select the disks.
- Choose the **Select disks manually** option to use specific disks from the Available disks pane for creating the volumes. Either double-click on it or select the **Add** option to move the disks into the Selected disks pane.

After verifying the information click **OK** to create the required volume. You are then taken back to the **Volume Information on the connected host**s panel.

Repeat the above steps for each of the volumes that has not been created, including the data volumes and Replicator Log.

- After all the volumes have been created, the volume information panel is updated to display the available volumes on the Primary and Secondary host. Click **Next**.
- If the required disk group and the volumes have been created but these volumes are not eligible for replication, then the reason for non-eligibility is indicated against the volume name.

  See "Setting up replication using the Setup Replicated Data Set wizard" on page 81.

  The Volume Information on connected hosts panel enables the appropriate option to convert a non-eligible volume to a Volume Replicator acceptable format.

  Complete the information on this panel as follows:

Recreate Volume

This option is enabled if the required data volume is available on the Secondary, but is of a size greater than the Primary volume.

Clicking this option displays a message that prompts you to confirm whether you want to recreate the volume.

Choose **Yes** to recreate the volume using the Create Volume dialog. Note that this operation first deletes the volume resulting in the loss of the data that already exists on the volumes.

Remove DRL

This option is enabled if the required data volume is available on the Secondary but has a DRL. Clicking this option displays a message that prompts you to confirm that you want to remove the log. Click **Yes** to confirm the removal of DRL.

Remove DCM

This option is enabled if the required Replicator Log volume is available on the Secondary but has a DCM log. Clicking this option displays a message that prompts you to confirm if you want to remove the log. Click **Yes** to confirm the removal of DCM log.

Expand Volume

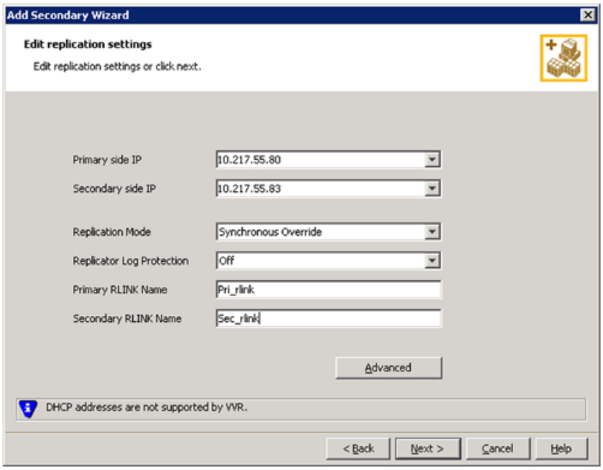
This option is enabled if the required data volume is available on the Secondary but is of a smaller size than the Primary volume. Clicking this option displays a message that prompts you to confirm that you want to grow the volume.

Click **Yes** to grow the volume to the required size.

After you have converted the non-eligible volumes to a Volume Replicator acceptable format, click **Next**. The Edit replication settings panel appears.

If the volume on the Secondary is already a part of another RDS, you cannot proceed. If you want to use the same volume, you must either remove the corresponding Primary volume from the Primary RVG or delete the other RDS.

4 Complete the **Edit replication settings** panel to specify basic and advanced replication settings for a Secondary, as follows:



Add Secondary Wizard

Edit replication settings

Edit replication settings or click next.

Primary side IP: 10.217.55.80

Secondary side IP: 10.217.55.83

Replication Mode: Synchronous Override

Replicator Log Protection: Off

Primary RLINK Name: Pri_rlink

Secondary RLINK Name: Sec_rlink

DHCP addresses are not supported by VVR.

- To modify each of the default values listed on this panel, select the required value from the drop-down list for each property. If you do not want to modify basic properties, then the replication can be started with the default values when you click **Next**.

  Complete the following:

Primary side IP

Displays the IP address on the Primary that is to be used for replication. If there is more than one IP address available for replication, you can choose the one that you want to use from the drop-down list. If the required IP address is not displayed in the list then edit the field to add the IP address.

Secondary Side IP

Displays the IP address on the Secondary that is to be used for replication, if the Secondary is connected to VEA. If there is more than one IP address available for replication, you can choose the one that you want to use from the drop-down list. If the required IP address is not displayed in the list then edit the field to add the IP address.

If you need to modify the IP addresses used for replication after creating the RDS, you can do it using the Change Replication Settings option.

See “Changing replication settings for an RDS” on page 173.

Replication Mode

Select the required mode of replication; **Synchronous**, **Asynchronous**, or **Synchronous Override**. The default is synchronous override.

See “Modes of replication” on page 28.

Replicator Log Protection

The AutoDCM is the default mode for the Replicator Log overflow protection when all the volumes in the Primary RVG have a DCM log. The DCM is enabled when the Replicator Log overflows.

The DCM option enables the Replicator Log protection for the Secondary host when the Replicator Log overflows, and the connection between the Primary and Secondary is lost. This option is available only if all the data volumes under the Primary RVG have a DCM Log associated with them.

The Off option disables Replicator Log Overflow protection.

The Override option enables log protection. If the Secondary node is still connected and the Replicator Log is about to overflow then the writes are stalled until a predetermined amount of space, that is, 5% or 20 MB (whichever is lesser) becomes available in the Replicator Log.

If the Secondary becomes inactive due to disconnection or administrative action then Replicator Log protection is disabled, and the Replicator Log will overflow.

The Fail option enables log protection. If the log is about to overflow the writes are stalled until a predetermined amount of space, that is, 5% or 20 MB (whichever is lesser) becomes available in the Replicator Log. Note that the writes are stalled only as long as the Secondary is connected. If the connection between Primary and Secondary RVG is broken, then, any new writes to the Primary data volumes are failed.

See “Replicator Log overflow protection—`srlprot` attribute” on page 48.

Primary RLINK Name

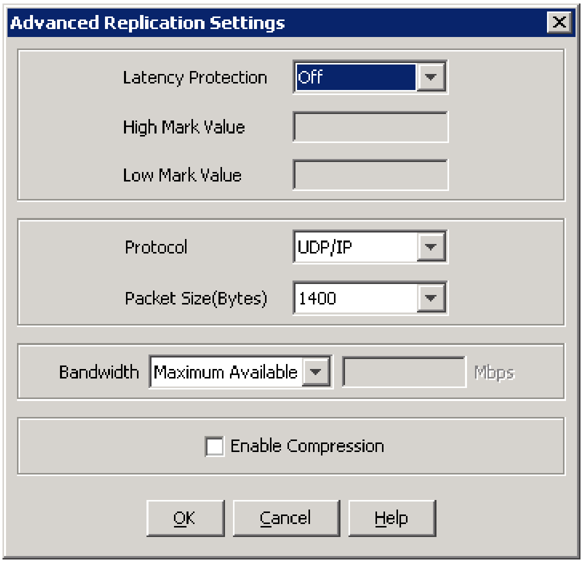
This option enables you to specify a Primary RLINK name of your choice. If you do not specify any name then Volume Replicator assigns a default name.

Secondary RLINK Name

This option enables you to specify a Secondary RLINK name of your choice. If you do not specify any name then Volume Replicator assigns a default name.

Click **Next** to start replication without any advanced settings. Proceed to step 5.

- Click **Advanced** to specify the advanced replication settings.



Advanced Replication Settings

Latency Protection: Off

High Mark Value:

Low Mark Value:

Protocol: UDP/IP

Packet Size(Bytes): 1400

Bandwidth: Maximum Available Mbps

□ Enable Compression

Complete the Advanced Replication Settings panel as follows:

Latency Protection

By default, latency protection is set to Off. When this option is selected the High Mark Value and the Low Mark Value are disabled. Select the **Fail** or **Override** option to enable Latency protection.

See “Latency protection—`latencyprot` attribute” on page 52.

This Override option behaves like the Off option when the Secondary is disconnected and behaves like the Fail option when the Secondary is connected.

High Mark Value

This option is enabled only when Latency Protection is set to Override or Fail. It specifies the maximum number of pending updates by which the Secondary can be behind the Primary. The default value is 10000, but you can specify the required limit.

To ensure that latency protection is most effective the difference between the high and low mark values must not be very large.

Low Mark Value

This option is enabled only when Latency Protection is set to Override or Fail. When the updates in the Replicator Log reach the High Mark Value, then the writes to the Primary continue to be stalled until the number of pending updates on the Replicator Log falls back to the Low Mark Value. The default value is 9950, but you can specify the required limit.

Protocol

UDP/IP is the default replication protocol. Choose TCP/IP or UDP/IP for a regular Secondary. If the storage on the Bunker Secondary is directly accessible from the Primary use the STORAGE protocol, otherwise use TCP/IP or UDP/IP.

**Note:** If the replication protocol for the Bunker Secondary has been set to STORAGE then you can change it only by deleting and recreating the Bunker with the required protocol; UDP/IP or TCP/IP. You cannot change the STORAGE protocol using the Change Replication Settings option.

Packet Size(Bytes)

Default is 1400. Choose the required packet size from the drop-down list. The default unit for the packet size is Bytes. You can set the packet size only if the protocol is UDP/IP.

Some firewalls do not support packet sizes greater than 1400 bytes. If you replicate across such a firewall, then use the default packet size to make sure all the Volume Replicator operations function as required. You can also set the packet size to 1300 by selecting from the list. The minimum packet size that you can specify is 1100 bytes.

**Note:** If you need to set a value for packet size different from that provided in the list then you can do this by using the command line interface.

See “About using the command line interface” on page 227.

Bandwidth

By default, Volume Replicator uses the maximum available bandwidth.

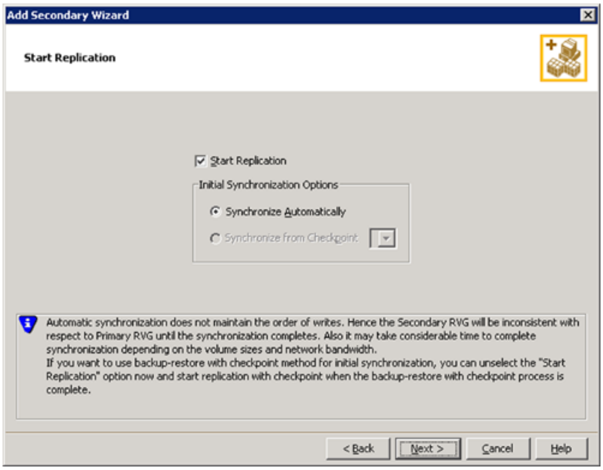
To control the bandwidth that Volume Replicator replication uses, choose Specify Limit, and then enter the bandwidth limit in the field provided. The default unit is Megabits per second (Mbps) and the minimum allowed value is 1 Mbps.

Enable Compression

Select this check box to enable compression for the Secondary host.

After completing the **Advanced Replication Settings** panel, click **OK**. You are taken back to the **Edit Replication Settings** panel. Click **Next**. The **Start Replication** panel appears.

**5** Choose the appropriate option from the **Start Replication** panel as described below:



Start Replication

☑ Start Replication

Initial Synchronization Options

● Synchronize Automatically

○ Synchronize from Checkpoint

Automatic synchronization does not maintain the order of writes. Hence the Secondary RVG will be inconsistent with respect to Primary RVG until the synchronization completes. Also it may take considerable time to complete synchronization depending on the volume sizes and network bandwidth.
If you want to use backup-restore with checkpoint method for initial synchronization, you can unselect the "Start Replication" option now and start replication with checkpoint when the backup-restore with checkpoint process is complete.

To add the Secondary and start replication immediately select **Start Replication** with one of the following options:

Synchronize Automatically

For an initial setup, then use this option to synchronize the Secondary and start replication. This is the default.

When this option is selected, Volume Replicator by default performs intelligent synchronization to replicate only those blocks on a volume that the file system uses. If required, you can disable intelligent synchronization.

See “Disabling the SwiftSync feature” on page 171.

**Note:** Intelligent synchronization is applicable only to volumes with the NTFS and ReFS file systems and not to raw volumes or volumes with FAT file systems.

Synchronize from Checkpoint

If you have considerable amount of data on the Primary data volumes then you may first want to synchronize the Secondary for existing data using the backup-restore method with checkpoint. After this completes use the **Synchronize from Checkpoint** option to start replication from checkpoint to synchronize the Secondary with the writes that happened when backup-restore was in progress.

- To add the Secondary without starting replication unselect the **Start Replication** option. You can start replication later by using the **Start Replication** from the Secondary RVG right-click menu.

Click **Next** to display the Summary panel.

6 Review the information on the **Summary** panel.

Click **Back** to change any information that you had specified or click **Finish** to add the Secondary to the RDS and exit the wizard.

# Administering the RVG

You can perform various RVG operations, of which some can be performed on both the Primary and Secondary RVG, whereas the others are specific to either the Primary or the Secondary RVG.

The following topics describe the tasks that you can perform to administer an RVG:

- See “Enabling or disabling data access to the RVG data volumes” on page 163.
- See “Expanding the data volumes” on page 164.
- See “Expanding the Replicator Log” on page 165.
- See “Shrinking the data volumes” on page 166.
- See “Adding or removing the DCM logs from the data volumes” on page 167.
- See “Resynchronizing the Secondary hosts” on page 169.
- See “Associating or dissociating the Replicator Log volume” on page 169.

## Enabling or disabling data access to the RVG data volumes

The user or the application can write data to the data volumes only if the data access is enabled for the volumes. This operation prepares the volumes to receive the writes from the application. The disable data access operation prevents the user or application from writing any data to the data volumes.

The enable data access operation first tries to lock all the volumes under the RVG and fails if it is unable to lock the volume because of the following reasons:

- Some application or file handles are still open on the volume.
  The disable data access operation requires that no application should use those volumes.
- The volume drive letter is currently being accessed through the explorer.
- The drive letter is active in the command prompt.

This option is available from the right-click menu of the Primary and Secondary RVG, and is a toggle option. If the RVG already has the data access enabled, then, the menu displays the Disable Data Access option. Otherwise, the menu displays the Enable Data Access option.

See “Enabling data access (Starting the RVG)” on page 285.

See “Disabling data access (stopping the RVG)” on page 286.

---

**Note:** If Volume Replicator is configured in a VCS or Microsoft Cluster and the cluster resource for this RVG exists, then Volume Replicator fails the Disable Data Access operation as this can cause the resource to fail.

---

Use the `vxrvg dismount` command to verify whether Disable Data Access operation succeeds.

See “Administering the RVGs using the `vxrvg` command” on page 272.

---

**Note:** If the data access to the Primary RVG is disabled, the Primary data volumes with NTFS or ReFS file systems may be displayed with a status as MISSING. To view these volumes enable data access to the Primary RVG and use the **Actions > Rescan** option from the VEA.

---

**To enable data access**

1. Select the Primary RVG and right-click. Select the **Enable Data access** option from the menu that appears.
2. The **Enable Data Access** dialog box appears.

   Click **Yes** to enable data access to the Primary RVG. Click **No** to cancel the operation.

## Expanding the data volumes

Using this option, you can increase the size of the data volumes to a specified value, across the RDS. The new volume size can be specified in sectors, kilobytes (KB), megabytes (MB), gigabytes (GB) or terabytes (TB) and cannot exceed the maximum size that the volume can be grown to.

---

**Note:** Trying to expand the volumes when replication is active in the Synchronous mode fails. To expand the volume, temporarily change the mode of replication to Asynchronous or Synchronous Override. After you have finished expanding the volume you can switch back to the synchronous mode of replication.

---

**To expand the data volumes**

1. Select the Primary data volume or the Secondary data volume and right-click. Select the **Expand Volume** option from the menu that appears.
2. The **Expand Volume** dialog box is displayed.
   - Specify the new size for the volume in the **New Size** field.
   - Select the unit for the volume size from the drop-down list.
3. Click **OK** to expand the volumes across the RDS.

# Expanding the Replicator Log

The Replicator Log must be large enough to meet the constraints. However, these constraints can change with the changes in the business needs, application write rate, available network bandwidth, and so on. As a result, it becomes necessary to redetermine the appropriate size of the Replicator Log. This section describes how to expand the Replicator Log on the Primary.

See "Sizing the Replicator Log" on page 43.

Before expanding the Replicator Log, verify that there is enough free space in the disk group in which the Replicator Log resides, by checking the Properties for the disks in the disk group through the VEA disk group view. Also, verify that the RVG host whose Replicator Log we are resizing is connected to VEA.

**Note:** Veritas recommends that size of Replicator Log volume should be same on all hosts within an RDS.

**To expand the Replicator Log on the Primary**

1. Select the volume that is used as the Replicator Log from the Volumes node in the tree view. Right-click and select the **Expand Volume** option. Specify the new value for the Replicator Log size in the New volume size field and click **OK**.
2. Alternatively, you can also select the Replicator Log volume from the Primary RVG. Right-click and select the **Expand Volume** option. Specify the new value for the Replicator Log in the **New Size** field and click **OK**.

   The Expand volume option resizes the Replicator Log on the Primary as well as the associated Bunker nodes.

# Shrinking the data volumes

You can decrease or shrink the size of a data volume across the Replicated Data Set (RDS) using the online volume shrink feature. This feature is helpful in reclaiming unused space to better use your resource.

The new volume size can be specified in Sectors, kilobytes (KB), megabytes (MB), gigabytes (GB) or terabytes (TB), and the specified value must be less than the maximum size of the volume.

The feature calculates the amount of space that can be freed from the volume to create a new smaller volume size. The size of a volume after the shrink volume operation is approximately the difference of the current volume size and the number of maximum reclaimable bytes. The new volume size is displayed in the Veritas Enterprise Administrator (VEA) GUI.

## Before shrinking a data volume

Consider the following before shrinking a data volume:

- Before performing the volume shrink operation, you must install the KB 2615327 hotfix from Microsoft.
- If the combined length of the volume name and disk group name is more than 9 characters, then you must install the KB 2619083 hotfix from Microsoft before shrinking the volume.
- Online volume shrink is not supported on Volume Replicator Secondary hosts, Storage Replicator Log (SRL), non-NTFS, and read-only volumes, and volumes on which a task is performed.
- For RDS configurations with only one Secondary host, the IBC messaging facility is used while shrinking the Secondary volume.
- For RDS configurations with more than one Secondary hosts, the RLINKs must be up-to-date before you perform a volume shrink operation. This is required because when the file system is shrunk during this operation, it may move some data clusters while defragmenting the volume and generate a large amount of I/O. Because of this, the RLINKs may not be up-to-date after the file system shrink, and the volume shrink operation may fail.
- In some cases, the Replicator Log overflows because of heavy I/Os during a volume shrink or defragmentation operation. Because of this, the volume shrink operation does not happen and, therefore, you may have a volume of the size greater than the file system at the Primary. In such cases, retry the volume shrink operation when the I/O is low after growing the file system by using the `vxvol growfs` command. For information about the command, refer to the *Storage Foundation Administrator’s Guide*.

### Shrinking a data volume

Perform the following steps to shrink a data volume.

**To shrink a data volume**

1 Right-click the data volume that you want to shrink, and select **Shrink Volume**.

2 The **Shrink Volume** dialog box is displayed.

Specify the new size for the volume in the **New Size** box, and then select the unit for the volume size from the drop-down list.

3 Click **OK** to shrink the volumes across the RDS.

**Note:** After the volume shrink operation completes, the existing RVG and RLINK checkpoints are deleted. A message prompts you to confirm the same.

## Adding or removing the DCM logs from the data volumes

By default, Volume Replicator adds DCM logs to all the volumes that are part of the RVG. The DCM log is used for automatically synchronizing the Secondary, when a new Secondary is added to the RDS. If the Replicator Log overflows when the Replicator Log protection has been set to DCM or AutoDCM then the DCM logs are used for resynchronizing the Secondary. The DCM log is also used for fast-failback logging and resynchronizing the original Primary when it comes up after a disaster.

If the RVG is part of a cluster setup, then from the VEA you must connect to the host which is the cluster virtual server by using the virtual name or address that was used when configuring the server.

If a volume has a DCM log, then the right-click menu displays only the Remove DCM Log option. However, if the volume does not have a DCM log then the Add DCM Log option is available.

**Note:** The Add DCM Log or Remove DCM Log option is available only if the hosts to which the volumes belong is connected to VEA.

**To remove the DCM log**

1 Select the data volume and right-click. Select the **Remove DCM Log** option from the menu that appears.

2 The Remove DCM Log dialog box appears.

Click **Yes** to Remove the DCM Log from the selected volume. Click **No** to cancel the operation.

This option is a toggle and only if the volume has a DCM log is the Remove DCM Log option displayed.

**To add a DCM log**

1 Select the data volume and right-click. Select the **Add DCM Log** option from the menu that appears.

2 The Add DCM Log dialog box appears.

Click **Yes** to add the DCM Log from the selected volume. Click **No** to cancel the operation.

This option is a toggle and only when the volume does not contain a DCM log, the Add DCM Log option displayed.

## Adding or removing the DCM logs for all volumes in an RVG

If the Replicator Log protection is not set to DCM or AutoDCM, then you can remove the DCM for all the volumes in the RVG.

**To add or remove the DCM log for all the volumes in the RVG**

1 Click the RVG. The right pane displays the Primary or Secondary RVG view depending on the RVG that you have selected.

The RVG information in the right pane is followed by a display of the list of volumes. Select all the required volumes using the Up or Down arrow keys keeping the Shift key pressed.

2 Right-click and select the **Add DCM Log** or **Remove DCM Log** from the menu that appears.

3 The Add or Remove DCM Log dialog box appears.

Click **Yes** to Add or Remove the DCM Log for the selected volumes. Click **No** to cancel the operation.

# Resynchronizing the Secondary hosts

If the Replicator Log overflows when log protection is set to DCM or AutoDCM, then, the writes to the Primary RVG are tracked on the DCM log. To start sending these writes that are tracked on the DCM log to the Secondary, you need to perform the Resynchronize Secondaries operation.

**Note:** To use this option, the Secondary must be connected to the Primary, that is they must be able to communicate with each other.

If the Primary RVG is part of cluster setup, you must connect to the host which is the cluster virtual server by using the virtual name or IP address that was used when configuring the server.

**Note:** The Secondary is inconsistent from the time the resynchronization starts and until it is completed.

**To resynchronize the Secondaries**

1. Select the Primary RVG and right-click. Select **Resynchronize Secondaries** option from the menu that appears.
2. In the **Resynchronize Secondaries** dialog box, click **Yes** to resynchronize the Secondary hosts with the Primary node. Click **No** to cancel the operation.

# Associating or dissociating the Replicator Log volume

By default, Volume Replicator does not let you create an RDS without a Replicator Log. All the RVGs in the RDS must have a Replicator Log.

However, you may later choose to dissociate the existing Replicator Log by using the option from the Replicator Log right-click menu. In that case you can use the Dissociate Replicator Log option to dissociate the Replicator Log from the RVG. Note that replication is not possible without a Replicator Log. This is one of the most important components that are required for replication to occur. The Associate Replicator Log option is available for selection only if the Replicator Log for an RVG has been removed, otherwise, at all times this option is unavailable.

## Associating the Replicator Log with the RVG

To associate the Replicator Log with the RVG you must be connected to the host through VEA. If the RVG is part of cluster setup, you must connect to the cluster virtual server by using the virtual name or IP address that was used when configuring the cluster.

The method to associate the Replicator Log on the Primary or the Secondary host is the same as described below.

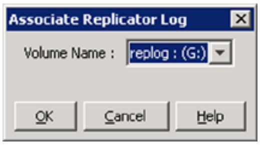
See "Setting up replication using the Setup Replicated Data Set wizard" on page 81.

**Note:** The Associate Replicator Log menu option is available only if the VEA is connected to the host of the selected RVG.

**To associate the Replicator Log**

1 Click on the RVG. Select **Associate Replicator Log** option from the right-click menu.

2 The Associate Replicator Log dialog box appears. Click the Volume Name arrow to see a list of volumes that are available to be selected and are part of the same Dynamic Group as RVG. If the required volume is not listed, then the volume may not satisfy the eligibility criteria.



Associate Replicator Log

Volume Name : replog : (G:)

3 Select the volume that you want to use as the Replicator Log.

4 Click **OK** to Associate the Replicator Log. On successful completion, the Replicator Log volume is displayed under the appropriate RVG in the VEA tree.

## Dissociating the Replicator Log volume on an RVG

This option is available for selection only when Replicator Log is associated with the RVG.

**To dissociate the Replicator Log**

1 Select the Replicator Log volume and right-click. Select **Dissociate Replicator Log** option from the menu that appears.

2 The Dissociate Replicator Log dialog box appears.

Click **Yes** to disassociate the Replicator Log. Click **No** to cancel the operation.

The method to dissociate the Replicator Log on the Primary and the Secondary host is the same.

# Administering replication

This section describes the tasks that enable you to administer replication.

- See “Disabling the SwiftSync feature” on page 171.
- See “Starting replication through the VEA console” on page 172.
- See “Stopping replication using the VEA console” on page 173.
- See “Changing replication settings for an RDS” on page 173.
- See “Managing checkpoints” on page 177.
- See “Pausing replication using Volume Replicator” on page 178.
- See “Converting the Primary to a Secondary” on page 180.
- See “Migrating the Primary role within an RDS” on page 181.
- See “Creating snapshots for the data volumes” on page 182.
- See “Recovering the RVG” on page 194.
- See “Restoring the Secondary” on page 194.

## Disabling the SwiftSync feature

By default, Volume Replicator is enabled to perform intelligent synchronization, which means that Volume Replicator replicates only those data blocks on the volumes that the application uses. However, if you want Volume Replicator to replicate all the data blocks then you must disable intelligent synchronization.

For information about how to edit the registry, refer to the Help topic "Changing Keys and Values" in Registry Editor `Regedit.exe` or the "Add and Delete Information in the Registry" and "Edit Registry Data" Help topics in `Regedt32.exe`. Make sure that you back up the registry before you edit it. After changing the registry, make sure that you update your Emergency Repair Disk (ERD).

**Note:** Using the Registry Editor incorrectly can cause serious problems that may require you to reinstall your operating system. Thus, before you edit the registry, make sure that you understand how to restore it, if a problem occurs. For information about how to do this, refer to the "Restoring the Registry" Help topic in Regedit.exe or the "Restoring a Registry Key" Help topic in Regedt32.exe.

**To disable intelligent synchronization**

1. Open the registry editor using the command, `regedit`.
2. Navigate to the following location:

   ```
   HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\vxio\VVRParams\
   ```

3. Set the DWORD `SwiftSync` to a value 0. By default, this DWORD value is set to 1, indicating that the intelligent synchronization support is enabled.

## Starting replication through the VEA console

Using this option, you can start replication whenever required, if it was not done when creating the RDS or adding a Secondary host. This option is available from the Secondary RVG right-click menu.

**Note:** Intelligent synchronization is applicable only to volumes with NTFS and ReFS file systems and not to raw volumes or volumes with FAT file systems.

See "Disabling the SwiftSync feature" on page 171.

**To start replication from the VEA console**

1. Select the Secondary RVG and right-click on it. Select **Start Replication** from the menu that appears. The Start Replication dialog box appears.
   - Choose the **Synchronize Automatically** option to synchronize the Secondary data volumes with the Primary using the DCM log. This may take a considerable amount of time depending on the volume sizes and network bandwidth.
     When this option is selected, Volume Replicator by default performs intelligent synchronization to replicate only those blocks of volumes that the file system on a given volume uses. If required, you can disable intelligent synchronization.
   - Choose the **Synchronize from Checkpoint** option to start replication from the precreated RVG checkpoint marker on the Primary Replicator Log. If the RVG checkpoints are not available, then **Synchronize Automatically** is the default option.
2. Click **OK** to start the replication. Click **Cancel** to quit the operation.

   Before using the Synchronize from Checkpoint option, the backup that is associated with the checkpoint must be restored on the Secondary volumes.

# Stopping replication using the VEA console

The stop replication option is available only on selecting the Secondary RVG. When this operation is performed the connection between the Primary and Secondary RVG is broken.

**To stop replication using the VEA console**

1. Select the Secondary RVG and right-click. Select the **Stop Replication** option from the menu that appears.
2. The **Stop Replication** dialog box appears.

   Note that if you restart replication after it has been stopped, you may require to synchronize the Secondary volume again, if the Primary volumes had changed. The message elaborates this. Read the information that is provided in the **Stop Replication** dialog box, carefully.

   Click **Yes** to stop replication or click **No** to cancel the operation.

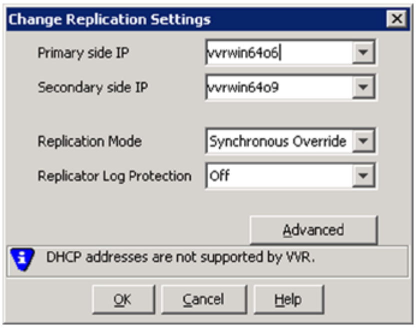
# Changing replication settings for an RDS

This option enables you to modify the replication settings that were specified when creating the RDS. It provides a basic as well as advanced set of options. You can choose to proceed with only the basic replication settings or specify the advanced properties based on your specific requirements.

**To change replication settings**

1. Select the **Change Replication Settings** from the Secondary RVG right-click menu. The Change Replication Settings dialog box appears.

   - To modify each of the basic properties that are listed on this panel, select the required value from the drop-down list for each property.



Change Replication Settings

Primary side IP: vvrwin64o6

Secondary side IP: vvrwin64o9

Replication Mode: Synchronous Override

Replicator Log Protection: Off

DHCP addresses are not supported by VVR.

   Complete the information on this panel to specify basic and advanced replication settings for a Secondary as follows:

Primary side IP

Displays the IP address on the Primary that is to be used for replication. If there is more than one IP address available for replication, you can choose the one that you want to use from the drop-down list. If the required IP address is not displayed in the list then edit the field to add the IP address.

Secondary Side IP

Displays the IP address on the Secondary that is to be used for replication, if the Secondary is connected to VEA. If there is more than one IP address available for replication, you can choose the one that you want to use from the drop-down list. If the required IP address is not displayed in the list then edit the field to add the IP address.

Replication Mode

Select the required mode of replication; Synchronous, Asynchronous, or Synchronous Override. The default is synchronous override.

See “Modes of replication” on page 28.

Replicator Log Protection

The AutoDCM is the default mode for the Replicator Log overflow protection when all the volumes in the Primary RVG have a DCM log. The DCM is enabled when the Replicator Log overflows.

The DCM option enables the Replicator Log protection for the Secondary host when the Replicator Log overflows, and the connection between the Primary and Secondary is lost. This option is available only if all the data volumes under the Primary RVG have a DCM Log associated with them.

The Off option disables Replicator Log Overflow protection.

The Override option enables log protection. If the Secondary node is still connected and the Replicator Log is about to overflow then the writes are stalled until a predetermined amount of space, that is, 5% or 20 MB (whichever is lesser) becomes available in the Replicator Log.

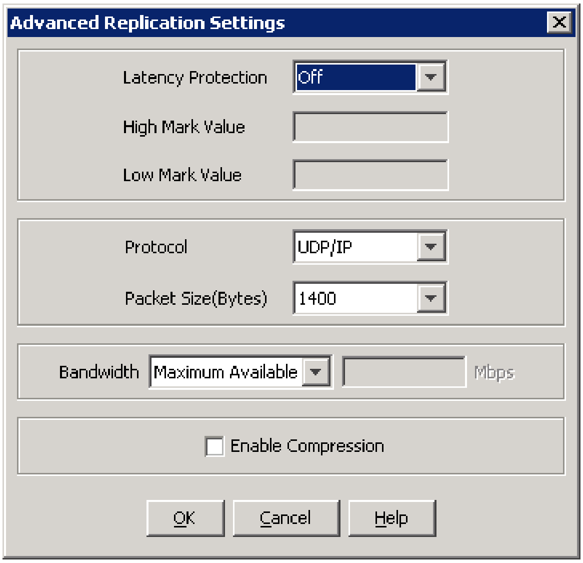
If the Secondary becomes inactive due to disconnection or administrative action then Replicator Log protection is disabled, and the Replicator Log will overflow.

The Fail option enables log protection. If the log is about to overflow the writes are stalled until a predetermined amount of space, that is, 5% or 20 MB (whichever is lesser) becomes available in the Replicator Log. If the connection between Primary and Secondary RVG is broken, then, any new writes to the Primary RVG are failed.

See “Replicator Log overflow protection—`srlprot` attribute” on page 48.

Click **OK** to start replication without any advanced settings.

2 Click **Advanced** to specify the advanced replication settings.



Advanced Replication Settings

Latency Protection: Off

High Mark Value:

Low Mark Value:

Protocol: UDP/IP

Packet Size(Bytes): 1400

Bandwidth: Maximum Available Mbps

☐ Enable Compression

Complete the Advanced Replication Settings panel as follows or proceed to the next step:

Latency Protection

By default, latency protection is set to Off and the High Mark Value and the Low Mark Value are disabled. Select the Fail or Override option to enable Latency protection.

See “Latency protection—`latencyprot` attribute” on page 52.

This Override option behaves like the Off option when the Secondary is disconnected and behaves like the Fail option when the Secondary is connected.

High Mark Value

This option is enabled only when Latency Protection is set to Override or Fail. It specifies the maximum number of pending updates by which the Secondary can be behind the Primary. The default value is 10000, but you can specify the required limit.

To ensure that latency protection is most effective the difference between the high and low mark values must not be very large.

Low Mark Value

This option is enabled only when Latency Protection is set to Override or Fail. When the updates in the Replicator Log reach the High Mark Value, then the writes to the Primary continue to be stalled until the number of pending updates on the Replicator Log falls back to the Low Mark Value. The default value is 9950, but you can specify the required limit.

Protocol

UDP/IP is the default replication protocol. Choose TCP/IP or UDP/IP for a regular Secondary. If the storage on the Bunker Secondary is directly accessible from the Primary use the STORAGE protocol, otherwise use TCP/IP or UDP/IP.

**Note:** If the replication protocol for the Bunker Secondary has been set to STORAGE then you can change it only by deleting and recreating the Bunker with the required protocol; UDP/IP or TCP/IP. You cannot change the STORAGE protocol using the Change Replication Settings option.

Packet Size (Bytes)

Default is 1400. Choose the required packet size from the drop-down list. The default unit for the packet size is Bytes.

Some firewalls do not support packet sizes greater than 1400 bytes. If you replicate across such a firewall, then use the default packet size to make sure all the Volume Replicator operations function as required. The minimum packet size that you can specify is 1100 bytes.

**Note:** If you need to set a value for packet size different from that provided in the list then you can do this by using the command line interface.

See “About using the command line interface” on page 227.

Bandwidth

By default, Volume Replicator uses the maximum available bandwidth.

To control the bandwidth that Volume Replicator uses for replication, choose Specify Limit from the drop-down list, and then specify the bandwidth limit in the field provided. The default unit is Megabits per second (Mbps) and the minimum allowed value is 1 Mbps.

Enable Compression

Select this check box to enable compression for the Secondary host.

3 Click **OK** to change the Replication settings. Click **Cancel** to cancel the operation.

# Managing checkpoints

Checkpoints are markers inserted into the Replicator Log. You can create two types of checkpoints, RVG checkpoints and RLINK checkpoints. The RVG checkpoints are created directly on the Primary RVG by using the Start Checkpoint option. The Start Checkpoint and End Checkpoint can be used to set markers and take backups of the Primary with the aim of synchronizing the Secondary. You can then ship this backup to the Secondary and apply it to the Secondary data volumes. After the backup is applied, start replication using the Synchronize from Checkpoint option by selecting the corresponding checkpoint. You can preserve a maximum of 72 checkpoints at any point-in-time.

On the Secondary however, you can create the checkpoints when pausing the Secondary. You may then want to take a backup of the Secondary volumes. This checkpoint is used to insert a marker in the Primary Replicator Log to indicate the point when replication was paused for taking the backup. Performing a Resume operation on the Secondary resumes the Secondary. If the Secondary data volumes fail at a later point-in-time, then you can apply the backup to the Secondary data volumes and then use the Restore with checkpoint option to synchronize the Secondary with all the writes that had come in after the corresponding checkpoint. Thus, the RLINK checkpoints are very useful if you need to restore the data on the Secondary volumes.

**To create an RVG checkpoint**

1. Select the **Start Checkpoint** option from the Primary RVG right-click menu. The Start Checkpoint dialog box appears.

   Enter a checkpoint string to be used as a marker in the Replicator Log. The checkpoint string can have a maximum length of 19 characters.

2. Click **OK**. This checkpoint is marked on the Primary Replicator Log and is used as a marker for synchronizing the Secondary after the backup has been restored. The Primary RVG View displays the checkpoint string that you had specified.

## Ending the checkpoint

Use this option to end the checkpoint. This option must be performed after you have completed taking the backup of the Primary data volumes. It marks the end of the backup on the Replicator Log.

**To end the Primary checkpoint**

1 Select the Primary RVG and right-click. Select the **End Checkpoint** option from the menu that appears.

2 The End Checkpoint dialog box appears.

Click **Yes** to end the checkpoint. Click **No** to cancel the operation.

**To delete the Primary checkpoint**

1 Select the Primary RVG and right-click. Select the **Delete Checkpoint** option from the menu that appears.

2 The Delete Checkpoint dialog box appears.

Click **Yes** to delete the checkpoint. Click **No** to cancel the operation.

# Pausing replication using Volume Replicator

Volume Replicator provides the option to pause replication from the Primary as well as the Secondary host. However, there is some difference in the way the pause is effected. Note that pausing the Secondary from the Primary or from the Secondary, effectively results in pausing replication. If the pause was initiated from the Primary host, the Secondary gets disconnected. After a resume operation is performed, the Secondary gets connected automatically and the pending updates from the Replicator Log are sent to Secondary. However, in the case of the Secondary initiated pause, you can specify a checkpoint that marks the point when the Secondary was paused on the Primary Replicator Log. You can take a backup of the Secondary and then resume replication. In future if the Secondary data volumes fail then you can apply the backup to the Secondary data volumes and use the **Restore** with checkpoint option to synchronize the Secondary with all the writes that had come in after the checkpoint. When you perform a **Restore**, the Secondary is updated only with updates from the checkpoint.

**Note:** Because the replication gets paused due to Primary initiated pause or the Secondary initiated pause, Veritas recommends that the pause operation should be applied only for a short period of time.

## Notes on pausing the replication

Certain features of the pause replication operation are explained below.

They are as follows:

- In the paused state of replication, as long as the Replicator Log is not full, the write-order is preserved.

- Prolonged periods of pause can cause the Replicator Log to overflow if there have been writes to the Primary. It is therefore necessary to ensure that the Replicator Log is configured such that it can hold all the writes to the Primary data volumes until replication is resumed.
- Secondary pause ensures that the connection to the Primary is maintained. However, the Primary pause causes the Secondary to disconnect.
- The Secondary can be paused with a checkpoint and a backup of the Secondary data can be taken. This backup can be used to restore the Secondary if the Secondary fails.

## Pausing Secondary from the Primary

Use this option to pause the Secondary node, from the Primary node. This option is generally used when you want to perform some maintenance tasks on the Primary node such as network configuration. When you pause the Secondary from the Primary the Secondary gets disconnected. In this case the replication status is displayed as `Primary Paused` in the Secondary RVG view.

**To pause Secondary from Primary**

1. Select the **Pause Secondaries from Primary** option from the Primary RVG right-click menu. The Pause Secondary dialog box appears.
2. Select the required Secondary from the list that appears when you click **Secondary Host**.
3. Click **OK** to pause the specified Secondary.

   The Secondary RVG view displays the state as `Primary Paused`.

## Resuming the Secondary host from Primary

Use this option to continue the replication on the Secondary node after the pause operation.

**To resume the Secondary host from Primary**

1. Select the **Resume Secondary from Primary** option from the Primary RVG right-click menu. The Resume Secondary dialog box appears.
2. If there are multiple Secondary hosts, select the required Secondary from the list that appears when you click **Secondary Host** list.
3. Click **OK** to resume replication on the specified Secondary.

   The Secondary RVG view displays the replication status as `Active`.

### Pausing the Secondary host from the Secondary

The Secondary host may need to be paused when you want to take a Secondary backup or for performing some maintenance task. In the Secondary initiated pause the connection between Primary and Secondary is maintained. Also, with the Secondary initiated pause, you can specify a checkpoint to track the location from where the replication is paused.

In the case of the Secondary initiated pause, the replication status is displayed as `Secondary paused` in the Secondary RVG view.

After finishing with the backup or other such activities that are required to be performed when the Secondary is paused, resume the Secondary. This option is a toggle option.

**Note:** It is not mandatory to specify a checkpoint and you can choose to pause without specifying a checkpoint.

**To pause the Secondary RVG**

1. Select the **Pause Secondary** option from the Secondary RVG right-click menu.

   The Pause Initiated by Secondary dialog box appears.

2. Specify the checkpoint string in the **Checkpoint** field and click **OK**.

## Converting the Primary to a Secondary

Use this option to convert the original Primary to a Secondary after performing a takeover without fast-failback logging. After the takeover operation is performed, an existing Secondary takes the role of the new Primary. If the original Primary comes up, use the Make Secondary option to change the role of a Primary to a Secondary. Thus, when the original Primary becomes available again, it can be made a Secondary to the new Primary.

Veritas recommends that you perform the Start Replication operation with the **Synchronize Automatically** option on the converted Secondary to bring the RDS to a consistent state after Make Secondary operation.

**Note:** Although the Make Secondary option is available on the original Primary as well as the new Primary, make sure that you perform this operation only on the original Primary.

**To convert the Primary to a Secondary**

1. Select the **Make Secondary** option from the Primary RVG right-click menu. The Make Secondary dialog box appears.
2. The IP address or host name of all the original Secondary hosts is displayed in the box. Select the host name or IP address of the host name that you intend to use as the new Primary, from the list that appears when you click the list option.
3. Click **OK** to make the original Primary a Secondary.

   If the `RVGPrimary` resource is configured for the selected Primary RVG, then Volume Replicator does not allow the Make Secondary operation to succeed as this can cause the resource to get into a faulted state.

## Migrating the Primary role within an RDS

Use this option to switch the Primary and Secondary roles between two hosts, within the RDS. This option is generally used for planned moves. For example, the Primary may need to undergo some maintenance tasks. The migration operation first disables data access to the Primary and Secondary RVGs. This operation then tries to lock all the volumes under RVG and then checks if the Secondary is up-to-date.

If a disaster occurs at the Primary node it is an unplanned situation. In this case the Take Over option is used.

**Note:** Veritas recommends using the `vxrvg dismount` command to verify whether the migrate operation succeeds.

See “Dismounting data volumes” on page 279.

See “Taking over the Primary role using the fast-failback option” on page 201.

The disable data access operation fails if it is unable to lock the volume due to any one of the following reasons:

- Some application or file handles are still running on the volume hence it cannot be locked. The disable data access operation requires that no application should use those volumes.
- The volume drive letter is accessed through an explorer.
- The volume drive letter is active in the command prompt.

**To migrate the Primary role**

1. Select the **Migrate** option from the Primary RVG right-click menu. The **Migrate** dialog box appears.

2. Select the required Secondary host from the Secondary Name option list. Click **OK** to migrate the Primary role to the Secondary. The Primary and Secondary roles are interchanged.

   If the `RVGPrimary` resource is configured for the selected Primary RVG, then Volume Replicator does not allow the Migrate operation to succeed as this can cause the resource to get into a faulted state.

   After migration, the replication to new Secondary becomes active. For all the other Secondary hosts, delete the existing RVGs and add them as Secondary hosts of the new Primary.

## Creating snapshots for the data volumes

Use this option to create a snapshot for each data volume in an RVG. Before creating the snapshots make sure that the volumes have been prepared by using the Storage Foundation Prepare operation. This operation creates snapshot mirrors (prepared plexes) for the specified data volumes.

For more information about the Prepare operation refer to the *Storage Foundation Administrator's Guide*.

After creating the prepared plexes for the all data volume in the RVG proceed with the following steps.

See "Enabling data access (Starting the RVG)" on page 285.

**To create snapshots for the data volumes**

1. Select **Snapshot** from the Primary RVG right-click menu. The **SnapShot** dialog box is displayed.

   Specify an appropriate prefix for the snapshot volume in the **Prefix for snapshot volume names** field. The snapshot volume names have the naming format as follows: *<prefix>-<volume name>*

   The total length of the snapshot volume name cannot exceed 18 characters including the prefix and the dash (-) character.

2. Click **OK** to create the snapshots for all the data volumes in the RVG.

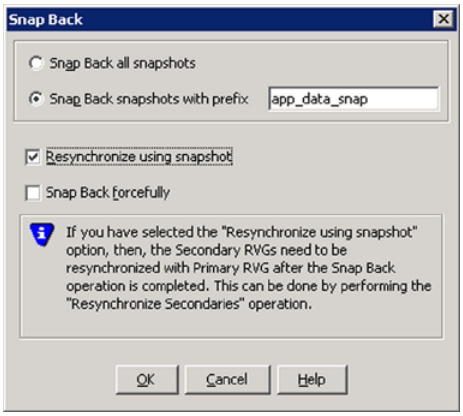
## Reattaching the snapshots back to the original volumes

Use this option to reattach the snapshots back to the original data volumes in an RVG. You can choose to attach all the snapshot volumes or the snapshot volumes with specific prefixes.

**Note:** After the snapshots are attached back to the original volumes, the contents of the original data volume remain unchanged. However, you can choose to resynchronize the original volumes from the snapshot volumes. In this case, the source volumes are updated with the contents of the snapshot volumes.

**To reattach the snapshots back to the original volumes**

1 Select **Snapback** from the Primary RVG right-click menu. The Snapback dialog box is displayed.



Snap Back

○ Snap Back all snapshots

● Snap Back snapshots with prefix: app_data_snap

☑ Resynchronize using snapshot

□ Snap Back forcefully

If you have selected the "Resynchronize using snapshot" option, then, the Secondary RVGs need to be resynchronized with Primary RVG after the Snap Back operation is completed. This can be done by performing the "Resynchronize Secondaries" operation.

2 Select one of the following options:

- Click **Snap Back all snapshots** to reattach all the snapshots back to their original volumes.
- Click **Snap Back snapshots with prefix** to reattach only the snapshot volumes with the specified prefixes back to their original volumes. Specify the required prefix in the field provided. The two options that are described above are mutually exclusive.

  If you add a snapshot volume to an RVG, then Snap Back operation cannot be performed using that volume. In this case, first remove the volume from the RVG before you perform a Snap Back operation.
- Select **Resynchronize using snapshot** to resynchronize the data in the original volumes with the data on the snapshot volumes.

Note that performing the snapback operation using the Resynchronize using snapshot option causes the checkpoint information to be lost.

- Select **Snap Back forcefully** to forcefully snapback the snapshot volumes even if the original volumes are in use. This option can be used with both the snapback options.

**3** Click **OK** to reattach the snapshots, back to the original volumes under the RVG, depending on the specified option.

## Creating synchronized snapshots using the VSS Snapshot wizard

SFW provides support for creating snapshots for the Microsoft Exchange storage groups and the SQL Server databases. FlashSnap integrates with the Microsoft Volume Shadow Copy Service (VSS) to let you create snapshots of all volumes that tare associated with an Exchange storage group or SQL database component without taking the databases offline. Volume Replicator further integrates the VSS snapshot feature with the IBC messaging to enable synchronized snapshots on the Primary and Secondary.

The VSS Snapshot wizard integrates with VSS to quiesce the databases of an Exchange Server storage group or SQL Server databases and then simultaneously snapshot the volumes in the Exchange or SQL components across the Primary and Secondary hosts. VSS then reactivates the database immediately after the snapshots are created. This quiescing, supported by Exchange Server at the storage group level and SQL at the database level, allows for Microsoft-supported and guaranteed persistent snapshots of your data.

A snapshot of a storage group or the database can be reattached and resynchronized to match the current state of the storage group or the database. An XML file to store the volume snapshot metadata is created on the Primary as a part of the snapshot operation.

**Note:** The VSS Restore GUI operations are not supported for synchronized snapshots. You need to use either the `vxassist snapback` or `vxsnap reattach` command to resynchronize the source volumes from the snapshot volume.

**Note:** Synchronized restore on Secondary is not supported.

When creating synchronized snapshots, the wizard verifies that the Secondary satisfies some preset conditions; there are some checks in place to validate this.

## About snapshot naming convention on the Secondary

The volume name by convention can have a maximum of 18 characters, of which one is an underscore (_), that leaves 17 characters. On the Secondary, the snapshots are named uniquely according to a specific naming convention so that the snapshots can be easily associated with the specific source volumes, if we want to reattach them later. The last 10 characters of the XML file that is created on the Primary and the last seven characters of the original volume name separated by an underscore are used as the volume name. This name is unique to every snapshot. For example, if the XML file name is `xmlfilename` and the volume name is `datavol` then the Secondary snapshots are named as `datavol_mlfilename`.

Because the XML file name is used for creating a unique snapshot name identifier, Veritas recommends that you have a unique string in the last 10 characters of the XML file name.

**Note:** Veritas recommends that for creating a unique snapshot name identifier, the last seven characters of the volume name in a Secondary disk group should be unique. Failure to follow the naming convention can result in some volumes on the Secondary not getting snapshotted.

**Note:** You can use VSS to snapshot only the read/write volumes. The resulting VSS snapshot is read-only.

Refer to the *Storage Foundation Administrator's Guide* for additional information about VSS snapshots.

See "Example 4: Using synchronized snapshots to restore data" on page 328.

## Creating synchronized snapshot sets

Creating a snapshot is a two-step process. The first step is to prepare the volumes for the snapshot to create snapshot mirrors attached to all the original volumes in the specified Exchange storage group or SQL database component. Depending on the size and number of volumes, the process of synchronizing the new snapshot mirrors with the original production volumes can take a long time. The second step uses the VSS Snapshot wizard to create the snapshot set (snapshot backup set) by detaching the snapshot mirrors from the original volumes and creating separate on-host snapshot volumes as well as an XML file to store the Exchange or SQL and the corresponding snapshot volume metadata.

Once a snapshot set has been created, it can be reattached and resynchronized with the original volumes using either the VSS Snapback wizard or the `vxsnap` command.

See “Creating snapshots using the `vxsnap` command” on page 292.

See “Example 4: Using synchronized snapshots to restore data” on page 328.

## Prerequisites for creating synchronized snapshot sets

Before you create a synchronized snapshot you need to follow certain prerequisites

They are as follows:

- Exchange or SQL as required, has been configured on the system.
- RVG volumes include the all the volumes as in the Exchange storage group or the SQL database.
- At least one RLINK to the Secondary exists.
- RVG with same name as on Primary exists on the Secondary.
- Volumes have been prepared.

**To create the snapshot set using the VEA console snapshot option**

1. From the VEA console URL bar, select the <*host name*> which is the system where the production volumes and snapshot mirrors are located, as the active host.
2. Expand the system node, the Storage Agent node, and the VSS Writers node.
3. Select one of the following depending on the application for which you create the snapshot:
   - For Exchange, right-click **Microsoft Exchange Writer** and click **VSS Exchange Snapshot**.
   - For SQL, right-click **Microsoft SQL Writer** and click **VSS SQL Snapshot**.
4. In the wizard, review the Welcome panel and click **Next**.
5. Specify the snapshot set parameters as follows and then click **Next.**

   Complete this panel as follows:

| | |
|---|---|
| Select Component for snapshot operation | Select the appropriate component that you have created, for the snapshot set.<br>If you create snapshots for Exchange, select the storage group.<br>If you create snapshots for SQL, select the database. |

Snapshot set

Enter a name for the snapshot set. The snapshot set metadata XML file is stored under this name.

To change the XML file location, use a text editor to create a text file named `redirect.txt`. This text file should contain a single text line specifying the full path to the location of the XML file, for example, `G:\BackupSets`. Save the `redirect.txt` file in the default directory `C:\Program Files\Veritas\Veritas Volume Manager\VSSXML`.

Select snapshot type

You can specify that snapshots be created as either a Full backup or Copy backup type.

**Full Backup** is typically used for backup to tape or other storage media. It does the following:

- Creates a copy of the selected component
- Only for Exchange, runs Eseutil to check for consistency before you truncate the logs
- Truncates the transaction logs

**Copy** is typically used for Quick Recovery. It creates a copy of the storage group, but does not truncate the transaction logs.

**For Exchange**: optionally check **Run Eseutil** with the **Copy** option to check the snapshot for consistency.

**For SQL**: Either type can be used to restore a database. However, if you want to replay logs in SQL Server as part of restoring a database, a Full backup needs to have been created earlier. When replaying logs, you can replay from the time of the last Full backup. A Copy backup does not affect this sequence of log replay and therefore is often used as an "out of band" copy for purposes such as testing or data mining.

6 In the **Change Attributes** panel, optionally, change the attributes for the snapshot volumes and click **Next**.

Snapshot Volume Label

Displays the read-only label for the snapshot volume.

Drive Letter

Optionally, click a drive letter and select a new choice from the drop-down menu.

Plex

Optionally, click a plex and select a new choice from the drop-down menu.

7. On the **Synchronized Snapshot** panel, select the Secondary hosts for which you want to create synchronized snapshots. Either double-click on the host name or click the **Add** option to move the host into the Selected Secondary Host pane. To select all the available hosts, click the **Add All** option. The VSS wizard creates synchronized snapshots on all the selected Secondary hosts.

   This panel is displayed only in an environment using Volume Replicator. Otherwise, you are directly taken to the **Schedule Information** panel.

8. Review the specifications of the snapshot set and click **Finish**.

## Creating schedules for synchronized snapshots

You can use the VSS Snapshot Scheduler wizard to add a snapshot schedule. The scheduling capability automates the process of refreshing snapshot sets simultaneously on the Primary and Secondary nodes. At the time that is scheduled for the snapshot, the snapshot volumes are automatically reattached, resynchronized, and then split again. Once configured and applied, a scheduler service `VxSchedService.exe` maintains the schedule.

If the Secondary host initially satisfies the required conditions but during execution of the synchronized snapshot operation some of the checks fail, then the command does not fail, but proceeds with creating the snapshots on the Primary host.

The wizard then logs an event with an appropriate error code, which can be viewed through the Event Viewer.

---

**Note:** The VSS Snapshot Scheduler wizard does not prepare the snapshot mirror. Prepare the snapshot mirror on the Primary and Secondary hosts with the `prepare` command before running the VSS Snapshot Scheduler wizard.

---

**To schedule a snapshot for a selected component**

1. From the VEA console URL bar, select the *<host name>* which is the system where the production volumes and snapshot mirrors are located, as the active host.

2. In the tree view expand the system node, the Storage Agent node, and the VSS Writers node.

3. Select one of the following depending on the application for which you want to create the snapshot:

   - For Exchange, right-click **Microsoft Exchange Writer** and click **VSS Exchange Snapshot**.
   - For SQL, right-click **Microsoft SQL Writer** and click **VSS SQL Snapshot**.

4 In the **Welcome** panel, review the information and click **Next**.

5 On the **Select Component** panel, specify the snapshot set parameters as follows and then click **Next**.

Complete this panel as follows:

Select component for snapshot operation

Select the component for the snapshot set.

To create snapshots for Exchange, select the appropriate storage group.

To create snapshots for SQL, select the appropriate database.

Snapshot set

Enter a name for the snapshot set. The snapshot set metadata XML file is stored under this name, with the prefix "VM_".

The XML file is stored by default in the directory that is shown on the screen.

To change the XML file location, use a text editor to create a text file named `redirect.txt`. This text file should contain a single text line specifying the full path to the location of the XML file, for example, `G:\BackupSets`. Save the `redirect.txt` file in the default directory `C:\Program Files\Veritas\Veritas Volume Manager\VSSXML`.

Select snapshot type

Select the snapshot type.

Full Backup is typically used for backup to tape or other storage media. It does the following:

- Creates a copy of the selected component
- Only for Exchange, runs Eseutil to check for consistency before you truncate the logs
- Truncates the transaction logs

Copy is typically used for Quick Recovery. It creates a copy of the storage group, but does not truncate the transaction logs.

For Exchange: optionally check Run Eseutil with the Copy option to check the snapshot for consistency.

For SQL: Either type can be used to restore a database. However, if you want to replay logs in SQL Server as part of restoring a database, a Full backup needs to have been created earlier. When replaying logs, you can replay from the time of the last Full backup. A Copy backup does not affect this sequence of log replay and therefore is often used as an "out of band" copy for purposes such as testing or data mining.

6 In the **Change Attributes** panel, optionally change the attributes for the snapshot volumes and click **Next**.

Complete this panel as follows:

| | |
|---|---|
| Snapshot Volume Label | Displays the read-only label for the snapshot volume. |
| Drive Letter | Optionally, click a drive letter and select a new choice from the drop-down menu.<br>The drive letters that are specified may not be available when the snapshot is taken. When this occurs, the snapshot operation is performed, but no drive letters are assigned. |
| Plex | Optionally, click a plex and select a new choice from the drop-down menu. |

7 On the **Synchronized Snapshot** panel, select the Secondary hosts for which you want to create synchronized snapshots. Either double-click on the host name or click the **Add** option to move the host into the **Selected Secondary Hosts** pane. To select all the available hosts, click the **Add All** option. The VSS wizard creates synchronized snapshots on all the selected Secondary hosts.

This panel is displayed only in an environment using Volume Replicator. Otherwise, you are directly taken to the **Schedule Information** panel.

8 In the **Schedule Information** panel, on the **General Options** tab, you need to the following.

Complete the information as:

Name of this schedule

Enter a unique name for the snapshot set schedule. This name identifies the snapshot schedule if you later want to view information about the snapshot status. A default name consists of the VSS writer name, the component name, and a numbered suffix that increments with each schedule.

Description of this schedule

Optionally, enter a description to help you identify the schedule when you view information about the snapshot status.

Start Time

The time of the day to begin taking snapshots

End Time

The time of day to end taking snapshots.

If a snapshot is in progress it is completed but a new one is not started after the end time.

Schedule takes effect on

The date on which the specified schedule takes effect. The default is the current date.

Restart task every

The interval between snapshots, in minutes.

For example, if the interval is 360 minutes and you schedule a snapshot start time of 12:00 P.M. and an end time of 7:00 P.M., the snapshot occurs twice. If no interval is specified the snapshot occurs once.

Every

Enable the **Every** option to have the snapshot schedule continue to occur. Otherwise the schedule applies only for one day.

Specify the number of days before restarting the snapshot schedule.

For example, 1 day would mean that the schedule takes effect daily, 2 days would mean every other day.

Start On

If you enable the Every option, specify the starting date.

Pre Command

Optionally, specify the full path of a command script to run before the scheduled snapshot occurs.

Post Command

Optionally, specify the full path of a command script to run after the snapshot is complete.

9 To specify additional schedule days or dates, make selections on the following tabs:

| | |
|---|---|
| Days of Week | Select one or more days on one or more weeks of the month.<br>You can click a button at the top of the column to select the entire column or a button to the left of a row to select the entire row. For example, clicking **First** schedules the snapshots to occur on the first occurrence of all the week days for the month. |
| Days of Month | Select one or more days of the month. You can also check the Last Day check box to schedule the snapshot for the last day of each month. |
| Specific Dates | Select one or more specific dates to include in or to exclude from the schedule.<br>Excluding a date takes precedence over days scheduled on the other tabs. For example, if you schedule every Monday on the **Days of Week** tab, and you exclude Monday October 9 on the **Specific Dates** tab, the snapshots are not taken on October 9. |

If two schedules overlap for the same snapshot set, only one snapshot is taken. For example, if you select every Thursday plus the last day of the month, and the last day of the month occurs on Thursday, then only one snapshot is taken on Thursday.

10 Click **Next**.

11 Review the snapshot set and schedule details and click **Finish**.

## Displaying the status of the scheduled synchronized snapshot

If a scheduled snapshot fails for some reason, the scheduler process attempts to rerun it. You may want to verify that scheduled snapshots completed successfully. From the VEA console, you can view snapshot results and other information about scheduled snapshots.

**To view a scheduled snapshot status**

1 From the VEA console URL bar, select the `<host name>` that is the system where the production volumes and snapshot mirrors are located, as the active host.

2 In the tree view, expand the system node, the Storage Agent node, and the VSS Writers node.

3 Right-click the snapshot schedule name and click **Job History**.

4 In the **Job History** dialog box, view the schedule information.

You can sort listed schedules by clicking the column headings. The Status column shows if the snapshot completed successfully.

## Reattaching synchronized snapshots

The VSS Snapback wizard reattaches and resynchronizes existing shadow copy set so that it matches the current state of its original Exchange storage group or the SQL database. This can be done simultaneously on the Primary and Secondary nodes if you have created synchronized snapshots. The wizard is available in the context menu of the VSS Writer object.

**To snapback a snapshot set**

1 Close the database application GUI and all Explorer windows, applications, consoles (except the VEA console), or third-party system management tools that may access the snapshot set.

2 From the VEA console URL bar, select the *<host name>* which is the system where the production volumes and snapshot mirrors are located, as the active host.

3 Expand the system node, the Storage Agent node, and the VSS Writers node.

4 Right-click the writer node of the application and click **VSS Snapback**.

5 Review the Welcome panel and click **Next**.

6 Select the snapshot set you want to snapback and click **Next**.

The XML metadata file contains all required information that is needed to snapback the snapshot set, including the names of the database and transaction logs volumes. Click the appropriate header to sort the list of available files by File Name or Creation Time. This file is deleted after the snapback operation has completed successfully.

7 If a message appears that indicates some volumes have open handles, confirm that all open handles are closed and then click **Yes** to proceed.

8 Verify that the snapback specifications are correct and click **Finish**.

## Deleting a synchronized snapshot schedule

If the snapshot schedule that you created is no longer required, you can delete it.

**To delete a schedule from the VEA**

1. From the VEA console URL bar, select the VEA <host name> that is the system where the production volumes and snapshot mirrors are located, as the active host.
2. In the tree view, expand the system node, the Storage Agent node, and the VSS Writers node.
3. Click **Scheduled Tasks**. The scheduled snapshots are listed in the right pane details view.
4. Right-click the name of the snapshot schedule and click **Delete Schedule**.

## Recovering the RVG

Use this option to recover the Primary RVG. This is especially useful if the Primary system becomes unavailable due to some problem resulting in some updates in the Replicator Log that could not be written on to the Primary data volumes. After the system is restarted, generally Volume Replicator updates the data volumes with all the pending updates from the Replicator Log.

However, if the Primary RVG could not be recovered automatically after the system was restarted, the Recover option can be used.

**To recover the Primary RVG**

1. Select **Recover** from the Primary RVG right-click menu. The Recover Primary RVG dialog box appears.
2. Click **Yes** to recover the Primary RVG. Click **No** to cancel the operation.

## Restoring the Secondary

Use this option to restore the Secondary when its replication status is displayed as Failed.

During active replication if you find that the data on the Secondary is inconsistent or corrupted then you can rollback the Secondary to a known good state with the help of the restore feature. You can first restore the Secondary volumes from the backup and then restore the Secondary from a known Secondary or RLINK checkpoint that is associated with the backup.

To restore the Secondary, it is essential that the Replication status of the Secondary must be in the `Failed` state. Hence, if the replication status is `Active` then you can forcefully fail the Secondary before restoring it. If there are writes on the Primary node that have not been copied to the Secondary node, then the Restore operation displays a message stating that there may be a loss of writes temporarily. The Restore option enables you to reestablish the link between the Primary and the

Secondary and then writes the data on to the Secondary. After the restore operation completes, the Secondary is up-to-date. This option is enabled only if any checkpoints are available for the selected Secondary.

**To restore the Secondary**

1. Select **Restore** from Secondary RVG right-click menu or select the **Restore** option from the toolbar. The **Restore Replicated Volume Group** dialog box appears.

   Select the **Confirm this operation on the Secondary** option to forcefully fail the Secondary. The **OK** option is enabled only when you select this option. Click **OK** to proceed.

2. The **Restore Replicated Volume Group** dialog box appears.
3. Specify the checkpoint from which to update the Secondary data volumes in the **Checkpoint** field by selecting the appropriate one from the list.
4. Click **OK** to restore the connection between the Primary and the Secondary nodes and synchronize the Secondary data volumes from the required checkpoint.

# Administering Bunker replication

Volume Replicator provides some specific tasks to administer Bunker replication and the Bunker RVG. These tasks are available from the Bunker RVG right-click menu. Most of these tasks are similar to the tasks available for a normal RVG.

The following topics describe the tasks to administer Bunker replication and RVG:

- See “Stopping the replication” on page 195.
- See “Pausing Secondary” on page 196.
- See “Changing replication settings for Bunker RVG” on page 196.
- See “Associating or dissociating the Replicator Log” on page 196.
- See “Activate Bunker” on page 197.
- See “Deleting the Bunker Secondary” on page 198.

## Stopping the replication

The Stop Replication option is a toggle and is similar to the same operation for a regular Secondary.

See “Stopping replication using the VEA console” on page 173.

# Pausing Secondary

The Bunker Secondary host may need to be paused when you want to take a backup or for performing some maintenance task.

In a pause that is initiated from the Bunker Secondary, the connection between Primary and Bunker Secondary is maintained and the replication status for the Bunker Secondary is displayed as `Secondary Paused` in the Secondary RVG view. After finishing with the required task, resume the Secondary.

**To pause and resume the Bunker Secondary RVG**

1. Select the **Pause Secondary** option from the Bunker Secondary RVG right-click menu.

   This is a toggle option.

   **Note:** You cannot specify a checkpoint for a Bunker Secondary.

2. To resume the Secondary, select the **Resume Secondary** option from the Bunker Secondary RVG right-click menu.

# Changing replication settings for Bunker RVG

This option enables you to modify the replication settings that were specified when adding the Bunker RVG to the RDS. It provides basic as well as an advanced set of options. You can choose to proceed with only the basic replication settings or specify the advanced properties based on your specific requirements. The options are similar to those for a normal Secondary.

See “Changing replication settings for an RDS” on page 173.

# Associating or dissociating the Replicator Log

For a Bunker RVG, Volume Replicator requires you to create an RDS with only the Replicator Log. If you dissociate the Replicator Log using the Dissociate Replicator Log option, you can add it back using the Associate Replicator Log option.

**Note:** The Associate Replicator Log menu option is available only if the VEA is connected to the host of the selected RVG.

## Dissociating the Replicator Log volume on Bunker RVG

This option is available for selection only if the Replicator Log is associated with the RVG.

**Note:** Replication is not possible without a Replicator Log as this is one of the most important components that is required for replication to occur.

**To dissociate the Replicator Log**

1. Click to select the Replicator Log volume in the Bunker RVG and select the **Dissociate Replicator Log** option from the menu that appears.
2. Because the replication needs to be stopped before the Replicator Log can be dissociated, a warning message is displayed. Click **Yes** to continue.

## Associating the Replicator Log with Bunker RVG

To associate the Replicator Log with the Bunker RVG make sure VEA is connected to the Bunker host. If the RVG is part of cluster setup, you must connect to the cluster virtual server by using the virtual name or IP address that was used when configuring the cluster.

If you use a storage Bunker setup then during the regular operations the Bunker RVG is imported on the Primary node. To associate the Replicator Log you must first be connected to the Primary node. If a disaster has occurred at the Primary then you need to import the Bunker disk group on the Bunker node. In this case you must first connect to the Bunker node to be able to see the Bunker RVG.

**To associate the Replicator Log**

1. Click and select **Associate Replicator Log** option from the RVG right-click menu.
2. The Associate Replicator Log dialog box appears. The appropriate Replicator Log volume with the same name and size as that of the Primary, is displayed in the field. If there are multiple volumes, select the appropriate volume from the **Volume Name** drop-down list.
3. Click **OK** to Associate the Replicator Log. On successful completion, the Replicator Log volume is displayed under the appropriate RVG in the VEA tree.

# Activate Bunker

This option is available from the Bunker RVG right-click menu and is enabled only if the Primary host becomes unavailable. When a disaster occurs at the Primary host, before performing a takeover on the Secondary, you may want to make sure that all the updates on the original Primary are available on the Secondary. You can do this by activating the Bunker RVG, converting the Bunker Secondary to a Bunker Primary and then replaying all the pending updates that did not reach the Secondary. After the replay completes, you can choose to deactivate the Bunker and convert it back to a Bunker Secondary and perform takeover on the up-to-date Secondary or restore the original Primary if it becomes available again.

See "Updating the Secondary from the Bunker" on page 200.

After the replay of pending updates from the Bunker Primary to the Secondary completes and the Secondary RLINK status is up-to-date, it is ready for takeover.

**To activate the Bunker**

1. Select the **Activate Bunker** option from the Bunker RVG right-click menu. The Bunker Secondary gets converted to a Bunker Primary.

   When a Primary becomes unavailable due to a disaster or is down for some maintenance, the Activate Bunker option is enabled on the Bunker Secondary.

2. Now select **Start Replication** on the Secondary host to replay all the pending updates from the Bunker Primary to the Secondary. Check the status of the Secondary using the `vxrlink updates` command and verify that the status is up-to-date.

**To deactivate the Bunker**

1. Stop replication to the Secondary by selecting the **Stop Replication** option.

2. Select the **Deactivate Bunker** option from the Bunker RVG right-click menu. The Bunker Primary is converted back to a Bunker Secondary.

## Deleting the Bunker Secondary

The procedure to delete the Bunker Secondary RVG is similar to the procedure that is used for the Primary or Secondary RVG.

**To delete the Bunker Secondary RVG**

1. Select the **Delete Secondary RVG** from the Secondary RVG right-click menu.

2. Depending on the current state of replication the appropriate message is displayed.

   - If the replication has already stopped then the dialog displays a confirmation message asking you if you want to proceed with deleting the Bunker Secondary RVG. Click **Yes** to delete the Secondary RVG. Click **No** to cancel the operation.

   - If the replication to the Bunker Secondary RVG in the RDS is active, it must be stopped before deleting the Secondary RVG. Otherwise, the Delete Secondary dialog displays the following message:

```
To delete the Bunker Secondary, replication must be stopped.
Are you sure you want to stop the replication and delete
 the Bunker Secondary?
```

Click **Yes** to delete the Secondary RVG. Click **No** to cancel the operation.

- If Volume Replicator is configured in a VCS or Microsoft Cluster and the cluster resource for this Bunker RVG exists, then Volume Replicator fails the Delete Secondary operation as this can cause the resource to fail. In such a situation, first delete the cluster resource before deleting the Bunker Secondary.

# Performing disaster recovery operation

In the case of a Primary failure or if the Primary needs to be brought down for some maintenance tasks, the role of the Primary can be taken over by the Secondary. When the original Primary becomes available again you may want to failback to the original Primary. The fast-failback feature enables you to do this quickly and efficiently as it performs incremental synchronization, for only the changed data. This feature uses the DCM of the data volumes of the new Primary to keep track of the changed content and the new content. This process of logging on the DCM after takeover is called failback logging or fast-failback logging.

You can perform the Takeover operation with fast-failback by using the fast-failback logging option on one of the Secondaries. After the Takeover operation is complete, you can start the applications on the new Primary. All the subsequent writes from the applications running on the new Primary are then tracked on the DCM of the new Primary. However, if there are any updates on the Primary that did not reach the Secondary, these may be lost.

## Using the Bunker node to update the Secondary

If your setup is configured for Bunker replication and a disaster occurs at the Primary site, you can use the Bunker node to update the Secondary. Because replication to Bunker node is synchronous, the writes that are written to the Primary are simultaneously written to the Bunker, therefore the Bunker node does not lag behind and enables zero RPO. Before you start the replay, activate the Bunker node to convert it to a Bunker Primary. Then start replication on the Secondary so that any pending updates that did not reach the Secondary are sent to the Secondary from the Bunker node. After all the updates have been sent to the Secondary, you can verify the status of the Secondary using the `vxrlink status` command.

**Note:** If the Primary Replicator Log has overflowed for a Secondary, or if the Secondary is inconsistent because it is resynchronizing, you cannot use the corresponding Bunker Replicator Log to recover the Secondary. Because the Bunker node does not have data volumes, it cannot use DCM to track overflows. By default, the Replicator Log protection for the RLINK between the Primary and the Bunker is set to off.

After all the updates have been sent to the Secondary, you can stop replication and then perform takeover on the up-to-date Secondary. Before takeover, you must deactivate the Bunker to convert it back to a Bunker Secondary. If you plan to continue using the original Secondary as a Primary, you cannot use the Bunker of the original Primary as a Bunker to the new Primary. You must configure a new Bunker host.

## Resynchronizing the original Primary when it becomes available

After the original Primary becomes available again it discovers that one of its Secondaries has taken over as the new Primary and it starts acting as a Secondary. Synchronize the original Primary with the new Primary by playing back the DCM. This synchronization can be started manually or automatically depending on the options that were specified during takeover. The RVG volumes on the original Primary provide read-only access permissions to the applications. Perform the resynchronize operation to start the DCM replay if you have not chosen the option to start it automatically during the takeover operation. At the start of the DCM replay, the original Primary becomes a Secondary and starts receiving the missing updates.

You can then continue to use the current setup after takeover as is, or, you can complete the failback process by using the Migrate operation.

## Updating the Secondary from the Bunker

Use the Bunker node to replay all the pending updates that did not reach the Secondary host. To do this, you must first activate the Bunker node and then start replication on the Secondary.

**Note:** You can also choose not to replay the Bunker Replicator Log after a disaster at the Primary if you want zero RTO. However, in this case the pending updates that were present on the Bunker Replicator Log are lost.

**Note:** As the Bunker Replicator Log does not store Primary checkpoints, it does not support attaching or resuming the Secondary from a checkpoint.

See “Viewing all the RDSs on the host” on page 115.

**To update the Bunker node from Secondary**

1 Select the **Activate Bunker** option from the Bunker RVG right-click menu.

This converts the Bunker RVG to a Primary, that is from the receiving mode (Secondary) to the replicating mode (Primary). Note that at any point-in-time the Bunker RVG can only be in either the receiving mode or the sending mode, but not both.

This option needs to be selected only once, even if you update multiple Secondaries.

2 Select the **Start Replication** option from the Secondary RVG right-click menu to start replication from the Bunker node.

This command switches the RLINK on the Secondary that points to the original Primary to point to the Bunker node which is now the Primary and begins replaying the Bunker Replicator Log.

If you have more than one Secondary using the same Bunker, repeat this step for each Secondary.

3 Monitor the status of the replication from Bunker to Secondary using the Monitor view.

4 When the replay is complete, verify that the Secondary is up-to-date using the `vxrlink status` command.

5 Select the **Stop Replication** option from the Secondary RVG right-click menu to stop replication to the up-to-date Secondary.

You can stop the replication before the replay is finished, for example, if the Primary is restored or depending on your RTO.

6 Convert the Bunker back to a Secondary Bunker by selecting the **Deactivate Bunker** option from the Bunker RVG right-click menu.

After using the Bunker for replay, if it is no longer needed, deactivate the Bunker. Make sure that you deactivate the Bunker only after all the replays from the Bunker have been stopped.

The Secondary is now up-to-date and can take over as a Primary.

## Taking over the Primary role using the fast-failback option

The takeover procedure enables you to convert a consistent Secondary to a Primary. This is very useful when the Primary experiences unscheduled downtimes or is destroyed because of a disaster and cannot be recovered immediately.

If the RVG is a part of cluster setup, you must connect to the host which is the cluster virtual server by using the virtual name or IP address that was used when configuring the server.

For zero RPO you must ensure that the Secondary is up-to-date before the takeover. If you have configured a Bunker RVG, before takeover, you can update the Secondary from the Bunker host.

See “Updating the Secondary from the Bunker” on page 200.

After takeover, the original Secondary becomes the new Primary. You can now add new Secondary hosts or the existing Secondary hosts to the new Primary. However, if the original Primary becomes available again, then you may want to failback the Primary role back to the original Primary. This can be done using failback logging or without it.

## Performing takeover with fast-failback

When performing takeover with fast-failback, the DCM log is used for logging the incoming writes on the new Primary. It is therefore necessary that the Secondary data volumes must have a DCM log.

### Prerequisites for takeover with fast-failback

To use the takeover with fast-failback option, there are certain prerequisites.

If you want to perform takeover with the fast-failback option, you need to do the following:

- Verify that the Secondary data volumes have DCM logs.
- Verify that the Secondary is attached or the replication status of the Secondary is displayed as `Activating`.
- Verify that the original Primary can be recovered and made available after the failure, if you want to failback to the original Primary.
- Verify that the new Primary can connect to the original Primary.

**To take over the Primary role using fast-failback**

1. Select the Secondary RVG and right-click. Select the **Take Over** option from the menu that appears.
2. The **Take Over** dialog box is displayed.
   - By default, the **Enable Fast-Failback Logging** option is selected if the data volumes have DCM logs associated with them. You can use this option to perform takeover with fast-failback logging.
     The DCM is activated for fast-failback logging and the new incoming writes are marked on the DCM of the new Primary.

If the replication status of Secondary RVG was `Inactive` when the Primary failed, then the **Enable Fast-Failback Logging** option is unavailable for selection. In this case you can perform **Take Over** without using fast-failback logging.

- Select the **Synchronize Automatically** option if you want the new Primary and the original Primary to get synchronized automatically, after the original Primary recovers.

  If you have not selected this option, the original Primary, after it recovers is in the `Acting as Secondary` state. To synchronize this original Primary with the new Primary use the **Resynchronize Secondaries** option from new Primary RVG's right-click menu. When the resynchronization starts, the original Primary which was in the `Acting as Secondary` state is converted to a Secondary of the new Primary. The new Primary now starts replaying the DCM to update the Secondary with the writes that were written to the DCM.

3 Click **OK** to proceed with takeover. Click **Cancel** to cancel the operation.

## Performing takeover without using fast-failback

To perform takeover without using the fast-failback option, perform the following procedure:

**To take over the Primary role without using fast-failback**

1. Select the **Take Over** option from the Secondary RVG right-click menu. The **Take Over** dialog box is displayed.
2. If you do not want to use the **Enable Fast-Failback Logging** option, clear the check box, and click **OK** to perform Take Over without the fast-failback logging.

   After takeover is complete, to add the Secondary hosts of the original Primary as Secondary hosts of the new Primary, delete the existing RVGs of the original Secondary hosts and then add them as a part of the new Primary.
3. If you have chosen to perform the Take Over operation without using fast-failback logging and the original Primary becomes available again, convert it to a Secondary using the **Make Secondary** option. Then resynchronize the original Primary with the new Primary using the **Synchronize Automatically** option. Depending on the size of the data volume, this may take quite a while.

## General notes on take over operation

For the takeover operation to be successful, you need to take the following into consideration.

Some considerations are as follows:

- If the original Primary has multiple Secondary hosts, and the RLINKs between every pair of Secondaries have not been created, then, after migrating the Primary role to one of the Secondaries or performing takeover on one of the Secondaries, all the remaining Secondaries in the RDS become orphaned. You must manually delete these Secondaries and then again add them as Secondaries to the new Primary.

  However, if you have created RLINKs between each pair of Secondaries in the RDS, then after a migrate or takeover operation, use the following steps to add the orphaned Secondaries back in the RDS:

  - On each orphaned Secondary host, detach the RLINK on this orphan Secondary pointing to the original Primary (the Primary host before migrate or takeover).
  - The orphan Secondaries join the RDS of the new Primary. Now, start replication with Automatic Synchronization on each of these orphans.

- After the original Primary host becomes available again, you may want to failback to this host. To do this, first synchronize the original Primary with the new Primary, and then migrate the Primary role back to the original Primary. If you had not deleted the Secondary RVGs of the original Primary hosts, then after a migrate operation you need not perform an Add Secondary operation to add the Secondaries back to the original Primary. However, as the replication to these Secondaries is stopped or is inactive, you must start replication to these Secondaries and synchronize them with the Primary.
- After performing a takeover with fast-failback, Veritas recommends that you do not detach the RLINKs on the original Primary using the `vxrlink det` command or convert the original Primary to a Secondary using the Make Secondary option. However, if you do perform these operations, you must perform a complete synchronization of the original Primary with the new Primary.

## Performing takeover in a multiple Bunker setup

Depending on your requirements, you can choose to have multiple Bunker nodes for a Primary. If one of the Bunker nodes crashes during the replay, you can synchronize the Secondaries from an alternative Bunker node.

Multiple Bunker nodes are also useful if you want to avoid a single point of failure due to a Bunker node crashing. If you have multiple Bunker nodes, check the status of the Bunker nodes using the `vxrlink status` command to find out the most up-to-date node, before you perform replay. This is necessary if any of the Bunker nodes are replicated to, asynchronously. The rest of the procedure to recover from a disaster is same as that for a single Bunker node.

See “Performing disaster recovery operation” on page 199.

# Deleting Volume Replicator objects

This section describes the tasks that are involved in deleting the Volume Replicator objects.

- See "Removing data volumes" on page 205.
- See "Deleting the replicated data set" on page 206.
- See "Deleting the Primary RVG" on page 206.
- See "Deleting the Secondary RVG" on page 207.

## Removing data volumes

This option is used to remove the data volumes from the selected Primary RVG and the corresponding volume from the Secondary RVG within the same RDS.

**To remove the data volumes**

1. Select the Primary data volume or the Secondary data volume and right-click. Select the **Remove Volume** option from the menu that appears.
2. The Remove Volume dialog box is displayed.

   Click **Yes** to delete the data volume from the Primary and Secondary RVG within the RDS. Click **No** to cancel the operation.

### Understanding the remove data volume behavior in different scenarios

The remove data volume operation has different output for different scenarios.

The Remove data volume behavior in different scenarios is as follows:

- Consider the scenario where the RDS setup has a Primary with multiple Secondary RVGs. Removing a data volume from any one of the RVGs removes it from all the RVGs. This does not require the replication to be stopped.
  To proceed click **Yes**, when the following confirmation message is displayed.

  ```
  Are you sure you want to remove the volume?
  ```

- Consider the scenario where the RDS setup has Primary and Secondary RVG, and replication is active. However, due to a network disconnection, if only the Primary RVG is available on the Primary host, then trying to remove the Primary data volume can cause the Secondary data volume to go out of synchronization. A dialog box with the following message appears. To proceed click **Yes**.

```
Since replication is active, there may be outstanding writes
present in the Primary Replicator Log. Removing the Primary data
volumes can cause the corresponding Secondary data volumes to be
out of synchronization. Are you sure you want to remove the
Primary data volumes?
```

- Consider a scenario where the RDS has a Primary and Secondary RVG, and the replication is active. However, due to network disconnection if only the Secondary RVG is available in the RDS, then removing the Secondary data volume pauses the replication with a configuration error, when the connection between Primary and Secondary is established. A dialog box with following message appears. To proceed click **Yes**.

```
To remove the Secondary data volume, replication must be
stopped. Are you sure you want to stop the replication and remove the
data volume?
```

To avoid this error condition, stop replication before removing the volume.

# Deleting the replicated data set

To delete an RDS perform the following steps:

**To delete the RDS**

1. Click on the RDS and select **Delete Replicated Data Set** from the right-click menu. The Delete Replicated Data Set dialog box appears.
2. Click **Yes** to delete the RDS. Click **No** to cancel the operation.

   If Volume Replicator is configured in a VCS or Microsoft Cluster and the cluster resource for this RVG exists, then Volume Replicator fails the Delete Replicated Data Set operation, as this can cause the resource to fail.

# Deleting the Primary RVG

Use this option to delete the Primary RVG from the RDS.

**Note:** If you are connected only to the Primary node, deleting the Primary RVG removes the entire RDS from the VEA tree.

**To delete the Primary RVG**

1. Select **Delete Primary** from the Primary RVG right-click menu. The Delete Primary dialog box appears.
2. Depending on the current state of replication the appropriate message is displayed in the dialog box.
   - If replication is already stopped then the dialog displays a confirmation message asking you if you want to proceed with deleting the Primary RVG. Click **Yes** to delete the Primary RVG. Click **No** to cancel the operation.
   - If replication to any of the Secondary RVGs in the RDS is active, it must be stopped before you delete the Primary RVG. Otherwise, the Delete Primary dialog displays the following confirmation message:

     ```
     To delete the Primary RVG, replication to all the Secondary
     hosts must be stopped. Are you sure you want to stop the
     replication and delete the Primary RVG?
     ```

   - If Volume Replicator is configured in a VCS or Microsoft Cluster and the cluster resource for this RVG exists, then Volume Replicator fails the Delete Primary operation as this can cause the resource to fail. In such a situation, first delete the cluster resource before deleting the RVG.
     To proceed, click **Yes**. Click **No** to cancel the operation.

## Deleting the Secondary RVG

The procedure to delete the Secondary RVG is similar to the one for the Primary RVG.

**To delete the Secondary RVG**

1. Select the Secondary RVG and right-click. Select **Delete Secondary RVG** from the menu that appears.
2. The Delete Secondary dialog box appears. Depending on the current state of replication the appropriate message is displayed in the dialog box.
   - If the replication is already stopped then the dialog displays a confirmation message asking you if you want to proceed with deleting the Secondary RVG. Click **Yes** to delete the Secondary RVG. Click **No** to cancel the operation.
   - If replication to the Secondary RVG in the RDS is active, it must be stopped before you delete the Secondary RVG. Otherwise, the Delete Secondary dialog displays the following confirmation message:

```
To delete the Secondary, replication must be stopped.
Are you sure you want to stop the replication
and delete the Secondary?
```

- If Volume Replicator is configured in a VCS or Microsoft Cluster and the cluster resource for this RVG exists, then Volume Replicator fails the Delete Secondary operation as this can cause the resource to fail. In such a situation, first delete the cluster resource before deleting the Secondary.

  To proceed, click **Yes**. Click **No** to cancel the operation.

# Accessing data on Secondary host

You can access data on the Secondary while replication is active by creating volumes with mirrors and breaking off the mirrors or by taking snapshots of all the data volumes in the RVG.

**Note:** A valid license for Storage Foundation FlashSnap must be present on all the systems on which you want to use the snapshot operations.

See "Enabling data access (Starting the RVG)" on page 285.

Once the mirrors are broken off, these are stand-alone volumes and the data on them can be accessed for read-write operations. The advantage with snapshot volumes is that these volumes are associated to the original volume and you can reattach them back to the original volume.

## Creating a mirror break-off

Breaking a mirror takes away a redundant mirror (or plex) of a volume and assigns it another drive letter. The data on the new volume is a snapshot of the original volume at the time of breaking. Breaking off a plex of the mirrored volume does not delete the information, but it does mean that the plex that is broken off no longer mirrors information from the other plex or plexes in the mirrored volume.

For further details on mirror break off, refer to the *Storage Foundation Administrator's Guide*.

**Note:** To create the mirror break-off, the volumes must have been created with mirrored plexes.

**To create mirror break-offs**

1. Right-click on the volume from which you want to break a mirror. Click **Mirror** from the menu that comes up, and then select **Break** from the submenu. The Break Mirror dialog box appears.
2. Select the mirror you want to break off from the Break Mirror dialog box. Choose whether or not to assign a drive letter to the broken-off volume. You can assign a specific letter by selecting from the list or you can accept the default.

   Click **OK** to break-off the mirror.

   This mirror break-off volume gives you the data on the Secondary data volume up to the point before the mirror break-off operation was performed.

## Creating snapshots

To create snapshots you need to prepare the data volumes.

For more information about the FlashSnap feature refer to the *Storage Foundation Administrator’s Guide*.

---

**Note:** Wait until the snap plexes are completely synchronized before creating the snapshot.

---

**To prepare the data volume**

1. In the VEA Tree view for Volumes, right-click the data volume that you want to access.
2. Select the **Snap > Prepare** option from the menu that appears.

**To create a snapshot of the data volume**

1. In the VEA tree view for Volumes, right-click the data volume that you want to access.
2. Select **Snap > Snap Shot** option from the menu that appears.

   This snapshot volume provides you with the data on the Secondary data volume up to the point before snapshot operation was performed.

# Performing automated system recovery (ASR)

This section describes the Automated System Recovery (ASR) feature available in Microsoft Windows Server 2003 and the process to save and restore Volume Replicator configuration.

- See “Automated system recovery (ASR) overview” on page 210.

- See "Volume Replicator support for ASR" on page 211.
- See "ASR recovery process " on page 211.

## Automated system recovery (ASR) overview

Automated System Recovery (ASR) is a disaster recovery feature that is part of the Microsoft Windows Server 2003 operating system. ASR extends the functionality of a traditional backup and restore application by providing an external mechanism to save information about the system state and configuration, including Volume Replicator specific information. ASR captures and stores the information to a floppy disk and tape or other backup media. Information that is saved includes the system registry, critical Windows files, and volume configuration information, including the number and type of partitions as well as file system and drive letter information. If a disaster or other event causes a computer to reach an unusable state, ASR can be used to restore a system to a bootable state and prepare it for data recovery. Volume Replicator supports ASR on systems running Microsoft Windows Server 2003 and any backup and restore application specifically designed to support ASR, such as Backup Exec, NetBackup, or the Backup Utility that is packaged with Microsoft Windows Server 2003.

An ASR backup should be performed after the system is initially configured and repeated whenever there is a change in the system configuration. Examples of such changes include adding of new volumes into an existing RVG, or creating a new RVG, or installation of a patch or service pack.

**Warning:** As part of the ASR backup process, Storage Foundation saves the configuration information only of those dynamic disk groups which are currently imported on the system running the ASR backup. For example, in a cluster setup, configuration information about cluster dynamic disk groups currently imported on the node which is backed up is saved, but cluster dynamic disk groups currently owned by other nodes is not saved.

**Note:** ASR attempts to make the target system bootable and recovers the original disk and volume and RVG configuration where possible. Sometimes it may not be possible to recover all of the disk, volume, and RVG configuration. In this case, ASR attempt to create a bootable system and allow the administrator to manually reconstruct the disk and volume and RVG configuration.

## Volume Replicator support for ASR

During an ASR backup several files are created including asr.sif, asrpnp.sif and setup.log. The following section describes the files that are created and the type of Volume Replicator information that is stored in each.

The `asr.sif` (ASR State Information File) stores system name, host ID, and other system state information and contains a number of subsections that store specific types of information.

To save dynamic disk group, volume and RVG configuration information, Volume Replicator uses the following subsections:

- InstallFiles

  This subsection lists the set of files that are needed to perform the recovery of the dynamic disk groups, volumes, and RVG configuration. It also contains information about the original installation media where these files are located. ASR uses this section during the text-only mode of recovery to prompt the user to insert the correct media and to copy the listed files to the requested location.

- Commands

  Contains the commands to execute the re-installation of Volume Replicator and to reconstruct the original dynamic disk groups, volumes, and RVG configuration during the GUI mode of a system recovery.

- VXVMASR.VOLUMECONFIG

  Contains the configuration information for all the dynamic disk groups, volumes, and RVGs on the system.

  Note that Manual edits to the `asr.sif` file may result in an invalid ASR backup and cause a Recovery operation to fail.

The `asrpnp.sif` and `setup.log` files are used to store the PNP state and the system configuration information respectively. Together with the `asr.sif` file they create a complete picture of the layout and configuration of your system.

## ASR recovery process

For a complete description of the ASR Recovery process, see the documentation that accompanies your backup and recovery application.

The recovery process begins by booting the repaired or replacement system from the operating system CD, and then pressing F2 to begin the text-only mode of the ASR recovery process. This is followed by a prompt to insert the floppy disk that is created during the ASR backup process.

During the text-only mode you are prompted to insert the Storage Foundation CD as well as the CDs from your backup and recovery application and any other third-party applications that participate in the ASR recovery process. At the end of the text-only mode of recovery, the system performs an automatic restart. You may have to remove any floppy disks or CDs in order for the system to continue to the GUI mode of setup by booting through the hard disk.

The system restarts into GUI mode and the ASR Recovery process continues automatically. In the event of a failure, on-screen directions guide you.

**Warning:** If there is a failure related to Volume Replicator during this phase, Veritas recommends that you retrieve and save all the error and trace logs when you are provided the opportunity to do so. You may not have access to these diagnostic files later as the system may not be in a bootable state if the error encountered is critical in nature. The error and trace logs can be found in the `<systemroot>\repair` folder.

After the successful completion of the GUI mode, you are again prompted to restart your system.

Following this final restart, your system is recovered and you are ready to begin the process of data recovery. The RVG configuration is restored, however, the Secondary is detached. You need to resynchronize the Secondary hosts. If any RVG was stopped, that is, the data access was disabled at the time of backup, then the RVG is started, that is, data access is enabled after restore.

## Considerations when restoring a Secondary with a healthy Primary

If you restore a Secondary that has a healthy Primary, using ASR, then after recovery, ASR creates the Secondary RVG, but it is detached from the RDS. You can then add the RVG back into the RDS but this may result in the Secondary being inconsistent.

As the Secondary may not necessarily hold all the valid data on RVG volumes, you are recommended to do the following:

- Stop replication to the Secondary from the healthy Primary before performing the ASR recovery
- Avoid adding the recovered Secondary into the RDS
  However, the safest method to restore replication is to start replication with the **Synchronize Automatically** or Checkpoint option. Before doing this make sure that you have restored the block level backup that you had taken after creating a checkpoint on the Primary.

## Microsoft Cluster recovery

This section describes the general process for ASR recoveries with Microsoft Cluster. Refer to your backup and recovery application documentation and related Microsoft articles for further information.

There are two types of recoveries that may occur within a Microsoft Cluster set-up, node restore and cluster restore.

A node restore is necessary when a single node of a cluster has failed. In this case, the shared disks failover to another node, but the local node needs to be recovered using the ASR backup. The recovery process is similar to the general process previously described, the system configuration recreated except that the disks that failed over to another node are inaccessible to the local node during the ASR Recovery. After the ASR Recovery process is complete, the node should restart and automatically join the cluster.

A cluster restore is necessary when a cluster with a single node running fails. In this case, since there is no node available for failover, the disk containing the quorum information needs to be restored. The quorum information is saved during the ASR Backup process, but is not automatically copied to the quorum disk during the ASR Recovery process. Instead the quorum information must be manually restored using the resource kit utility clustrest.exe. Following this, a system restart should be forced. The single node cluster boots and begins operating properly.

# Alternative methods to synchronize the Secondary faster

The earlier sections have described in detail the various features of Volume Replicator along with the disaster recovery procedures. This section describes the alternate methods that can be used to synchronize the Secondary faster.

The methods are explained with the help of a sample configuration that is described in this section where two Volume Replicator hosts are located at two different, geographically remote locations. Take for example, London and Seattle, the Primary being at London and the Secondary at Seattle.

Veritas recommends that you use the **Synchronize Automatically** option when you start the replication initially, to make sure that the Secondary is completely synchronized with the Primary. Although Volume Replicator would ensure the integrity of data under all circumstances, trying to synchronize the Secondary over a WAN may become restricted, due to problems such as network errors, rate of application writes or bandwidth availability.

To enable faster synchronization, you can use one of the methods that are described in the following sections to minimize the time required. However, one requirement when you use these methods is that the replication status of Secondary must be Inactive, that is, the RLINKs are detached. You can verify this by using the vxprint -PVl *<rvg>* command.

See "About using the command line interface" on page 227.

The methods that are given below are described using the following sample Volume Replicator setup. Note that the host names are indicative of the locations where the host exists.

Sample setup to synchronize the Secondary faster:

For Primary host london, do the following:

| | |
|---|---|
| vvr_dg | Disk Group |
| vvr_rvg | Primary RVG |
| rlk_seattle_vvr_rvg | Primary RLINK to Secondary seattle |
| host ip | 10.212.80.251 |
| vvr_dv01 | Primary data volume #1 |
| vvr_dv02 | Primary data volume #2 |
| vvr_srl | Primary Replicator Log volume |

For Secondary host seattle, do the following:

| | |
|---|---|
| vvr_dg | Disk Group |
| vvr_rvg | Secondary RVG |
| rlk_london_vvr_rvg | Secondary RLINK to Primary london |
| host ip | 10.256.88.126 |
| vvr_dv01 | Secondary data volume #1 |
| vvr_dv02 | Secondary data volume #2 |
| vvr_srl | Secondary Replicator Log volume |

## Method 1: Moving the Secondary RVG disk group on to a spare server within the same LAN as the Primary

For example, consider the following scenario in a sample setup where the Primary host name is london and the Secondary host name seattle. The host cambridge is a spare server that exists on the same LAN as the Primary host london.

**To move the Secondary RVG disk group on to a spare server**

1. On the Secondary host seattle:
   - Select the `vvr_dg` disk group and right-click.
   - Select the **Deport Dynamic Disk Group** option from the menu that appears, to deport the disk group `vvr_dg` on which Secondary RVG `vvr_rvg` is created.
2. Physically ship the disks under this disk group to the system `cambridge` that is present on the same LAN as that of Primary host: london.

   The host `cambridge` is a spare server that exists on the same LAN as the Primary host `london`.
3. On the host cambridge:
   - Select the `vvr_dg` disk group and right-click.
   - Select the **Import Dynamic Disk Group** option from the menu that appears. Import the disk group `vvr_dg` on which Secondary RVG `vvr_rvg` exists on another host by selecting the **Clear host ID** option.
4. Considering that the host IP of `cambridge` is 10.212.80.252 change the RLINK IP addresses using the commands:

   ```
   On Primary london
   vxrlink set remote_host=10.212.80.252 rlk_seattle_vvr_rvg
   On Secondary Cambridge
   vxrlink set local_host=10.212.80.252 rlk_london_vvr_rvg
   ```
5. On host `cambridge`:

   Select the **Start Replication** operation to start the replication using the **Synchronize Automatically** option. The operation is much faster over a LAN as compared to that over a WAN.
6. After synchronization is complete,
   - On host london

     ```
     vxrlink pause rlk_seattle_vvr_rvg
     vxrvlink set remote_host=10.256.88.126 rlk_seattle_vvr_rvg
     ```

- On host cambridge

  ```
  vxrlink pause rlk_london_vvr_rvg
  vxrvlink set local_host=10.256.88.126 rlk_london_vvr_rvg
  ```

7 Deport the disk group from host `cambridge` and ship the disks back to the original Secondary host `seattle`. Now, import the disk group on host `seattle`. Import the disk group on another host by selecting the **Clear host ID** option.

8 Resume the RLINKs that were paused in step 6.

## Method 2: Using snapshots for synchronizing the Secondary data volumes

Consider the following scenario where you need to synchronize the Secondary data volumes using snapshots on Primary host london and Secondary host seattle respectively.

**To synchronize the Secondary data volumes using snapshots on Primary host london**

1 Prepare the volumes under the required RVG. Ensure that the new snap plex is created on independent disks. Also make sure that the disks are clean disks. To prepare the volumes, run the following command:

```
vxassist -g vvr_dg prepare vvr_dv01 <disk name>
vxassist -g vvr_dg prepare vvr_dv02 <disk name>
```

The disk name can be obtained by running the `vxvol volinfo` command.

2 Start a checkpoint using the Start Checkpoint option from the Primary RVG right-click menu or using the command:

```
vxrvg -g vvr_dg -c checkpt1 checkstart vvr_rvg
```

3 Use the **Snap Shot** option from the Primary RVG right-click menu to create snapshots of all the volumes in the RVG or using the command:

```
vxrvg -g vvr_dg -P snap snapshot vvr_rvg
```

The snapshot volumes are `snap_vvr_dv01` and `snap_vvr_dv02`. Note that the snapshot is created on disks different from the original volumes.

4 Split the disk group using the **split dynamic disk group by volumes** option to create a new disk group `temp_dg`, containing the snapshot volumes.

5. End the checkpoint using the End Checkpoint option from the Primary RVG right-click menu or using the command:

   ```
   vxrvg -g vvr_dg checkend vvr_rvg
   ```

6. Deport the disk group `temp_dg`.

7. Physically ship the disks that comprise the `temp_dg` to the Secondary host `seattle`.

**To synchronize the Secondary data volumes using snapshots on Secondary host seattle**

1. Import the disk group `temp_dg` on which the snapshot volumes are present by selecting the **Clear host ID** option.

   If you cannot see the newly added disk group `temp_dg` perform a rescan operation using the Rescan option from the Actions menu.

2. Perform a disk group join operation by selecting the **Join Dynamic Disk Group** from the disk group right-click menu.

   This adds the new `temp_dg` with the snapshot volumes to the `vvr_dg` disk group.

3. If a volume with the same name `vvr_dv01` and `vvr_dv02` as that on the Primary exists on the Secondary disk group *vvr_dg* then:

   - Stop replication on the Secondary RVG
   - Dissociate the volumes from the RVG:

     ```
     vxrvg -g vvr_dg -r vvr_rvg dis vvr_dv01vxrvg -g
      vvr_dg -r vvr_rvg dis vvr_dv02
     ```

   - Delete the volumes vvr_dv01 and vvr_dv02 since we recreate it from the Primary snapshot volumes.

## Renaming Volumes on the Secondary host

If a volume with the same name as that on the Primary host does not exist on the Secondary then you need to rename the volumes. This can be done either using the VEA GUI or CLI.

**Note:** Using the `-f` option without caution can cause data corruption as the Secondary may sometime miss the writes that may have been present on the Replicator Log but did not reach the Secondary. As there is no way of knowing whether the Replicator Log had some pending writes that have not reached Secondary, use this option only when the Secondary is completely up-to-date with the Primary.

**To rename the volumes on the Secondary host seattle**

1 Prepare the volumes `snap_vvr_dv01` and `snap_vvr_dv02` by running the command

```
vxassist -g vvr_dg prepare snap_vvr_dv01 <disk name>
vxassist -g vvr_dg prepare snap_vvr_dv02 <disk name>
```

2 Perform the snapshot operation using the following command.

```
vxassist -g vvr_dg snapshot snap_vvr_dv01 vvr_dv01
vxassist -g vvr_dg snapshot snap_vvr_dv02 vvr_dv02
```

3 From the VEA GUI, expand the Volumes node in the tree view.

4 Right-click the desired data volume node.

5 Select **Change Volume Internal Name** from the context menu.

A dialog box appears to rename the volume.

6 Enter the new internal name of the volume.

7 Click **OK** to complete the operation.

or

Open a command window and run the following command:

```
vxedit [-g DynamicDiskgGroupName] [-f] rename <OldName><NewName>
```

8 Associate the volumes to the RVG:

```
vxrvg -g vvr_dg assoc vvr_rvg vvr_dv01vxrvg -g
 vvr_dg assoc vvr_rvg vvr_dv02
```

9 Select the **Start Replication** option from the Secondary RVG right-click menu. The Start Replication menu appears. Select the **Synchronize from Checkpoint** option to synchronize the Secondary from the checkpoint `checkpt1` you had created.

10 Verify that the replication state is `Active` by checking the `Replication Status` field in the right pane of the Secondary RVG view or using the command:

```
vxprint -PVl vvr_rvg
```

If you do not have a license for FlashSnap, then use one of the following methods:

- Use the Synchronize from Checkpoint option to synchronize the Secondary. Copy the required data from the block-level backup and then restore it on the Secondary.
- Use the Synchronize from Checkpoint option to synchronize the Secondary by using the mirrored plexes as a block-level backup.

## Method 3: Using mirrored plexes to synchronize the Secondary

Another method to synchronize the Secondary without using the **Synchronize Automatically** option is by using the mirrored plexes. Consider that on host london the data volumes vvr_dv01 and vvr_dv02 are created with mirrored plexes. The mirrored plexes are always synchronized with the source volumes.

**Note:** Veritas recommends creating mirrors for each volume on separate disks so as to avoid problems or issues when you perform a mirror-breakoff. Also make sure that the disks you use are clean disks.

**To synchronize the Secondary using mirrored plexes on Primary host london**

1 Create additional mirrors for the volumes `vvr_dv01` and `vvr_dv02` under the RVG using the Mirror > Add option from the <volume-name> right-click menu.

2 Start a checkpoint using the Start Checkpoint option from the Primary RVG right-click menu or using the command:

```
vxrvg -g vvr_dg -c checkpt1 checkstart vvr_rvg
```

3 Break off the mirrors from the volumes: `vvr_dv01` and `vvr_dv02` using the Break Mirror option.

4 In the **Break Mirror** dialog box select the mirror that needs to be broken based on the disk on which they are located. The dialog lists the different disks. Select the one on which the mirror exists.

- Specify a drive letter for the mirror-breakoff.
  Click **OK**.

5 The mirror-breakoff is created with the default name on the specified disk.

6 Split the disk group to create a new disk group `temp_dg` with the mirror-breakoff volumes.

7 End the checkpoint using the End Checkpoint option from the Primary RVG right-click menu or using the command:

```
vxrvg -g vvr_dg checkend vvr_rvg
```

8 Deport the disk group `temp_dg`.

9 Physically ship the disks that contain the mirror-breakoff volume to the Secondary host `seattle`.

**To synchronize the Secondary using mirrored plexes on Primary host seattle**

1 Import the disk group `temp_dg` on which the mirror-breakoff volumes are present by selecting the **Clear host ID** option.

If you cannot see the newly added disk group `temp_dg` perform a rescan operation using the Rescan option from the Actions menu.

2 Perform a disk group join operation by selecting the **Join Dynamic Disk Group** from the disk group right-click menu.

This adds the new `temp_dg` with the mirror-breakoff volumes to the `vvr_dg` disk group.

3 If a volume with the same name vvr_dv01 and vvr_dv02 as that on the Primary exists on the Secondary disk group vvr_dg then:

- Stop replication on the Secondary RVG.
- Dissociate the volumes from the RVG by running the following command:

```
vxrvg -g vvr_dg -r vvr_rvg dis vvr_dv01vxrvg -g vvr_dg -r
 vvr_rvg dis vvr_dv02
```

Delete the volumes vvr_dv01 and vvr_dv02 since you would need to recreate it from the Primary mirror-breakoff volumes.

## Renaming mirror-breakoff volumes on the Secondary host

If a volume with the same name as that on the Primary host does not exist on the Secondary then you need to rename the volumes. This can be done either using the VEA GUI or CLI.

**Note:** Using the `-f` option without caution can cause data corruption as the Secondary may sometime miss the writes that may have been present on the Replicator Log but did not reach the Secondary. As there is no way of knowing whether the Replicator Log had some pending writes that have not reached Secondary, use this option only when the Secondary is completely up-to-date with the Primary.

**To rename the mirror-breakoff volumes on the Secondary host seattle perform the following steps**

1 Identify the volume names from the volume view, with the help of the volume's drive letter. Assuming that the default name is default_vol01 and devault_vol02, prepare these volumes:

```
vxassist -g vvr_dg prepare default_vol01 <disk name>
vxassist -g vvr_dg prepare devault_vol02 <disk name>
```

The disk name can be obtained by running the `vxvol volinfo` command.

2 Perform the snapshot operation using the following command.

```
vxassist -g vvr_dg snapshot default_vol01 vvr_dv01
vxassist -g vvr_dg snapshot devault_vol02 vvr_dv02
```

3 From the VEA GUI, expand the Volumes node in the tree view.

4 Right-click the desired data volume node.

5 Select **Change Volume Internal Name** from the context menu.

A dialog box appears.

6 Enter the new internal name of the volume.

7 Click **OK** to complete the operation through the VEA.

or

Open a command window and run the following command:

```
vxedit [-g DynamicDiskgGroupName] [-f] rename <OldName><NewName>
```

8 Associate the volumes to the RVG by running the following commands

```
vxrvg -g vvr_dg assoc vvr_rvg vvr_dv01vxrvg -g vvr_dg
 assoc vvr_rvg vvr_dv02
```

9. Select the Start Replication option from the Secondary RVG right-click menu. The Start Replication menu appears. Select the Synchronize from Checkpoint option to synchronize the Secondary from the checkpoint *checkpt1* you had created.
10. Verify that the replication state is Active by checking the Replication Status field in the right pane of the Secondary RVG view or using the command:

    ```
    vxprint -PVl vvr_rvg
    ```

# Obtaining statistical information through Volume Replicator Graphs

Volume Replicator Graph is used to view the Non-Paged Pool (NPP) memory statistics and RLINK bandwidth usage for an RDS. The online graph shows bandwidth usage information in real time while the historic graph displays historical information about the RLINK bandwidth usage.

Volume Replicator Graph can be used to obtain the following information:

- Bandwidth usage by RLINKs in an RDS
- Non-paged Pool (NPP) memory that SFW uses

## Graph types and usage

On the basis of functionality, we can have the following types of graphs for Volume Replicator:

- Non-Paged Pool (NPP) memory usage graph
  Volume Replicator and SFW use the VOLIOMEM, NMCOM, and READBACK memory pools. The NPP usage graph plots the allocated and max values for these memory pools. The NPP graph gets updated every 5 seconds and displays the memory usage in kilobytes (KB) against time. Volume Replicator uses the VOLIOMEM pool for buffering writes sent to the RLINKs. The Secondary uses the NMCOM pool for buffering the incoming writes which later get written to the Secondary data volume. READBACK pool is used during DCM replay and for RLINKs that are behind
- Online bandwidth graph
  Online graphs plot the rate (in Kbps) at which Volume Replicator sends data to the Secondary. The send rate is calculated and plotted for each RLINK separately. The data gets updated every five seconds. You can save the data using the **File > Save** option. The data can be saved as a PNG image or a CSV

file. The saved file can be opened through the VEA GUI using the Open Graph option available on the Replication Network node.

- Historic bandwidth graph
  When a Secondary host is added to an RDS, Volume Replicator automatically starts collecting the bandwidth usage for the Secondary. The data can be displayed through the Historic Bandwidth Usage option available on the RDS. Currently, the file that is used for storing historic bandwidth information can grow up to 20 MB
- Saved graph
  Online or the historic bandwidth graph can be saved using the **File > Save** option. The data is saved as a CSV file. This file can be later opened using the Open Graph option available on the Replication Network node in the VEA GUI.

# Viewing statistical information using Volume Replicator Graph

This section You provides information to view and use Volume Replicator Graph functionality:

- See “Viewing online bandwidth usage for an RLINK” on page 223.
- See “Viewing Historic bandwidth usage for an RLINK” on page 224.
- See “Viewing Volume Replicator Non-Paged Pool (NPP) Memory Graph” on page 224.
- See “Saving an Online or Historic bandwidth usage graph” on page 225.
- See “Re-opening a saved CSV graph file” on page 225.
- See “Starting or stopping the Historic Bandwidth Data Collection” on page 226.

## Viewing online bandwidth usage for an RLINK

For getting the online RLINK bandwidth usage for all the RLINKs in an RDS, right-click the RDS object in the VEA GUI tree and select the appropriate option. The online graph can be saved as a CSV or PNG file. Graphs that are saved as .csv file can be reopened in the VEA GUI.

See “Saving an Online or Historic bandwidth usage graph” on page 225.

**Note:** A PNG file cannot be opened in the VEA GUI for viewing. You may require a graphics viewer to view this file.

To collect real-time or online bandwidth usage for RLINKs in an RDS, you need to do the following:

**To collect the Online bandwidth usage for an RLINK**

1. From the VEA GUI's replication network tree, right-click the appropriate RDS for which you want to view the online RLINK bandwidth usage. The **Bandwidth Usage** window appears. Online bandwidth usage graphs get updated or refreshed every five seconds.
2. Click **File > Exit** to close the graph file.

## Viewing Historic bandwidth usage for an RLINK

The historic bandwidth usage statistics is enabled on an RDS by default. After the historic bandwidth collection is enabled for an RLINK, select **View Historic Data Graph** from the VEA GUI. Select **File > Refresh** from Historic graph menu to refresh the Historic bandwidth graph.

See "Starting or stopping the Historic Bandwidth Data Collection" on page 226.

**To view the Historic Data Graph**

1. Right-click the RDS and select **View Historic Bandwidth Usage** from the shortcut menu that appears. The Bandwidth Usage (Historic) window appears and graph for historic bandwidth usage for RLINKs in the selected RDS is displayed. or Alternatively, you can also select **Actions > View Historic Bandwidth Usage** from the menu bar.
2. Click **File > Exit** to close the graph.

## Viewing Volume Replicator Non-Paged Pool (NPP) Memory Graph

To view the NPP memory usage, do the following:

**To view Volume Replicator NPP memory usage graph**

1. From the VEA GUI, right-click **Replication Network** and select **View Memory Usage** from the menu that appears. The Non-Paged Pool Memory Usage window appears.
2. To choose a particular host, select the appropriate host name from the **Select Host** drop-down menu that is displayed in the center of the NPP memory usage graph.

   Only hosts through which you are connected through the VEA GUI are shown in the drop-down list.

3. You can select the **Consolidated** check box that is displayed at the bottom of the graph to view consolidated NPP usage for NMCOM, READBACK, and VOLIOMEM memory Pools collectively.

   To view memory usage for the NMCOM, VOLIOMEM, or READBACK pools respectively, enable any of the other three check boxes that are displayed at the bottom of the graphing window. For example, if you enable NMCOM, the NMCOM memory usage graph is displayed.

4. Click **File > Exit** to close the graph.

## Saving an Online or Historic bandwidth usage graph

The Online as well as Historic bandwidth usage graph can be saved as a .csv or .png file.

See “Re-opening a saved CSV graph file” on page 225.

**To save the RLINK bandwidth usage graph**

1. From the VEA GUI, select **View Bandwidth Usage** in case of an online line graph. For Historic graph, select **View Historic Bandwidth Usage**. The **Bandwidth Usage** window appears.
2. Click **File > Save** to save the Online or Historic bandwidth usage graph as a .csv or .png file.

## Re-opening a saved CSV graph file

The saved graph file that is saved as .csv can be opened through the VEA GUI.

**Note:** A graph file that is saved in the PNG format cannot be opened through the VEA GUI. Use a graphic viewer to view such files.

To open the saved CSV graph file, you need to do the following:

**To re-open a saved CSV graph file**

1. From the VEA GUI, right-click **Replication Network** and select **Open Graph** from the shortcut menu or alternatively, you can select **Actions > Open Graph** from the menu bar.
2. In the **Open** list, locate and select the file that you want to open. Click **Open**. The selected graph file is displayed.
3. Click **File > Exit** to close the graph file.

## Starting or stopping the Historic Bandwidth Data Collection

The Start and Stop Historic Data Collection option is available from the right-click menu of a Secondary RVG and is a toggle option.

---

**Note:** In a multinode Primary cluster, if historic data collection is enabled on an RLINK and the storage group is moved to another node, then data that is collected on the old and the new node cannot be merged.

---

Historic bandwidth usage data for an RLINK can also be collected through the CLI using the `vxrlink startstats` and `vxrlink stopstats` command.

See "Starting the Historic Bandwidth Data Collection using the CLI" on page 271.

To start or stop the historic bandwidth data collection for an RLINK, you need to do the following:

**To start and stop the Historic Data Collection**

1. From the VEA GUI, select and right-click the **Secondary RVG**. Select **Start Historic Data Collection** from the shortcut menu. The data is collected every five seconds and is collected until the historic data collection is stopped.
2. To view the Historic bandwidth usage graph, right-click the appropriate RDS and select **View Historic Bandwidth Usage** shortcut menu option.

   See "Viewing Historic bandwidth usage for an RLINK" on page 224.
3. To stop historic bandwidth data collection, right-click the Secondary RVG and select **Stop Historic Data Collection** from the shortcut menu.

Chapter 7

# Using the command line interface

This chapter includes the following topics:

## About using the command line interface

The Volume Replicator Command Line Interface (CLI) provides you with a set of commands with various options that can be used from the command line. The command line interface is for an advanced level user for writing scripts and batch files.

If User Access Control (UAC) is enabled, run the Volume Replicator commands in the “Run as administrator” mode even if the logged-on user belongs to the local administrators group. Alternatively, log on as an Administrator (default administrator account) to perform the tasks.

The following table lists the commands with a brief description on each command.

**Table 7-1** Volume Replicator commands

| Command | Description |
|---|---|
| `vxrds` | Performs the administrative tasks on the Replicated Data Set (RDS).<br>See “Administering the RDS using the `vxrds` command” on page 230. |
| `vxrlink` | Performs the Volume Replicator related operations on the RLINKs.<br>See “Performing RLINK Operations using the `vxrlink` command” on page 253. |
| `vxrvg` | Performs the Volume Replicator related operations on the Replicated Volume Groups (RVGs).<br>See “Administering the RVGs using the `vxrvg` command” on page 272. |
| `vxprint` | Displays the complete or partial information about the replicator objects from the Volume Replicator configurations.<br>See “Displaying information using the `vxprint` command” on page 287. |
| `vxsnap` | Creates multiple snapshots at the same time and synchronized snapshots between the Primary and Secondary.<br>See “Creating snapshots using the `vxsnap` command” on page 292. |
| `vxmemstat` | Displays the memory statistics for Volume Replicator.<br>See “Displaying memory statistics using the `vxmemstat` command” on page 296. |
| `vxvol` | Performs the volume-specific operations.<br>See “Administering replicated volumes using the `vxvol` command” on page 298. |
| `vrport` | Displays and modifies the port values used by Volume Replicator.<br>See “Displaying and changing replication ports using the `vrport` command” on page 301. |

**Table 7-1** Volume Replicator commands *(continued)*

| Command | Description |
|---|---|
| `vxedit` | Enables you to edit the information that is associated with the Volume Replicator objects.<br>See "Administering the RVG using the `vxedit`" on page 306. |
| `vxassist` | Enables you to add or remove DCM logs for replicated volumes and grow the volumes, especially the Replicator Log volume.<br>See "Administering the RVG using the `vxassist` command" on page 309. |
| `vxtune` | Displays and modifies the Volume Replicator tunable values.<br>See "Tuning Volume Replicator" on page 312. |
| `vradmin` | Performs the administrative tasks on the RDS and is similar to the `vxrds` command. This command is supported to maintain parity with the Volume Replicator UNIX commands. |

# Conventions for command line syntax

This topic describes the conventions for the command line syntax in this CLI section.

The conventions for CLI syntax are as follows:

- Any parameter that is optional for the command syntax has a square bracket ([]) around it. For example:
  `[-P]` or `[<rlink>]`
- Required command words and parameters for the command do not have square brackets around them. For example:
  `vxrlink` or `<rlink>`
- Command words and parameters that are typed as shown in the command syntax are displayed in the Courier bold font, for example:

  ```
  vxrlink make or [-P]
  ```

- Parameters that the user needs to specify are displayed in Courier Italic font and have angle brackets around them. For example, <*diskgroup_name*>. They are placeholders for information the user is required to specify.
- The pipe (|) character is a separator that allows two or more choices for a given parameter. The user can use any one of the choices for the command. For example,

  ```
  [-f | -c <checkpoint> | -a]
  ```

- Help for any command is available if you type a hyphen followed by a question mark (-?) after the command. To view additional information about each of the parameters, type the command and the parameter followed by the -?. For example, `vxrds addsec -?` displays the information about the addsec parameter and the supported options.

# Administering the RDS using the `vxrds` command

The `vxrds` command helps you to perform the various administrative tasks on the Replicated Data Set (RDS). These tasks are performed by using the specific keywords with the `vxrds` command.

The following table lists the keywords that can be used with the `vxrds` command.

**Table 7-2** `vxrds` command keywords

| Keyword | Function |
|---|---|
| `activatebunker` | Activates the Bunker RVG to take over the Primary role when the original Primary becomes unavailable.<br>See "Activating the Bunker RVG" on page 233. |
| `addsec` | Creates and adds Secondary RVG to an RDS.<br>See "Activating the Bunker RVG" on page 233. |
| `addvol` | Associates the specified volume to all RVGs in the RDS.<br>See "Adding an existing volume to the RDS" on page 235. |
| `addBunker` | Adds a Bunker to an existing RDS without interrupting replication from the Primary to the Secondary.<br>See "Adding a Bunker node" on page 235. |
| `changeip` | Changes the host name or IP address of the Primary and Secondary RLINKs that are part of an RDS.<br>See "Changing the host name or IP" on page 236. |
| `createpri` | Creates a Primary RVG on the local system.<br>See "Creating the Primary RVG" on page 237. |
| `deactivatebunker` | Deactivates the Bunker to convert the Bunker Primary back to a Bunker Secondary.<br>See "Deactivating the Bunker RVG" on page 238. |

**Table 7-2** `vxrds` command keywords *(continued)*

| Keyword | Function |
|---|---|
| `delBunker` | Delete the Bunker RVG from the RDS configuration.<br>See "Deleting the Bunker node" on page 238. |
| `delsec` | Removes the Secondary RVG from the RDS.<br>See "Deleting the Secondary" on page 239. |
| `delpri` | Deletes the Primary RVG and the corresponding RDS if no Secondary is configured for the RDS.<br>See "Deleting the Primary" on page 239. |
| `delvol` | Dissociates the volume from all the RVGs in an RDS.<br>See "Dissociating data volumes" on page 240. |
| `fbsync` | Resynchronizes the original Primary with the new Primary once it becomes available after the takeover with fast-failback.<br>See "Resynchronizing a failed Primary with the new Primary" on page 240. |
| `makesec` | Converts the existing Primary RVG to a Secondary RVG after takeover.<br>See "Converting a Primary to a Secondary" on page 241. |
| `migrate` | Migrates the Primary RVG of the RDS to the specified Secondary host.<br>See "Migrating the Primary to a Secondary" on page 242. |
| `pauserep` | Pauses the replication to the specified Secondary.<br>See "Pausing replication using the vxrds pauserep command" on page 243. |
| `printrvg` | Displays the complete or partial information about all the RVGs in the RDS.<br>See "Displaying the RDS" on page 244. |
| `resizevol` | Depending on the new volume size specified, the command either grows or shrinks the volume uniformly across the RDS.<br>See "Resizing the data volumes" on page 245. |
| `resizesrl` | Grows the Replicator Log volume uniformly across the Primary and the Bunker Secondary.<br>See "Growing the Replicator Log volume" on page 246. |

**Table 7-2** `vxrds` command keywords *(continued)*

| Keyword | Function |
|---|---|
| `resumerep` | Resumes the replication to the specified Secondary.<br>See “Resuming replication after pausing” on page 247. |
| `resync` | Resynchronizes the Secondary in case the Replicator Log overflows and DCM is activated.<br>See “Resynchronizing the Secondary” on page 247. |
| `set` | Sets the replication attributes on the Secondary and Primary hosts.<br>See “Setting replication attributes” on page 247. |
| `startrep` | Starts the replication to the specified Secondary.<br>See “Starting replication using the `vxrds startrep` command” on page 250. |
| `stoprep` | Stops the replication to the specified Secondary.<br>See “Stopping replication using the `vxrds stoprep` command” on page 252. |
| `takeover` | Converts the Secondary RVG to Primary RVG of the RDS.<br>See “Taking over the Primary role using the `vxrds takeover` command” on page 252. |

The following table lists the options that are available with the `vxrds` command:

**Table 7-3** Available options for `vxrds` command

| Options | Description |
|---|---|
| `-autofba` | Enables the takeover with automatic resynchronization. The `-autofb` and `-N` options are mutually exclusive. |
| `-a or-autosync` | Starts the replication with autosynchronization. This option is used to attach the Secondary to the Primary as a part of the `startrep` command. |
| `-b A` | This option is used to replay the pending updates from the Bunker Primary to synchronize the Secondary host when the Bunker acts as Primary. |
| `-c <checkpoint>` | This option can be used with the `startrep` command to indicate the checkpoint with which the Secondary needs to be attached. |

**Table 7-3** Available options for `vxrds` command *(continued)*

| Options | Description |
|---|---|
| `-clean` | The `-clean` option deletes the RVG and the RLINKs associated with the RVG. |
| `-e <extended stats>` | The`-e <extended stats>` option is used for diagnostic or analytical purposes. |
| `-f` | You can use this option with the `vxrds delpri` command to force the Primary RVG to be deleted even if the data volumes are used.<br>You can use this option with the `startrep` command to make it behave like the `-forceatt` option. The `-f` option when used with the `-r` option performs the same function as the `-autofb` option.<br>**Note:** This `-f` option can cause data corruption because the Secondary may miss the writes that may have been present on the Replicator Log but did not reach the Secondary. Because there is no way of knowing whether the Replicator Log had some pending writes that have not yet been sent to the Secondary, use this option only if you know that the Secondary is completely up-to-date. |
| `-forceatt` | Forces the attach of a Secondary, assuming that the Primary and Secondary are synchronized. |
| `-F` | Enables Failback logging when taking over the role of the Primary after the original Primary becomes unavailable due to a disaster or some other problems. To ensure successful failback logging make sure that:<br>■ All the volumes of the new Primary have DCM logs.<br>■ The RLINK to the original Primary from the new Primary is attached. |
| `-g <diskgroup>` | Specifies the disk group for the various operations. |
| `-N` | Disables failback logging when performing takeover. |
| `-wait` | Ensures that the `vxrds fbsync` and `vxrds resync` command waits until the resynchronization completes. |

# Activating the Bunker RVG

Use the `vxrds activatebunker` command to activate the Bunker RVG to take over the Primary role when the original Primary becomes unavailable.

After the Bunker RVG has been converted to a Primary, you can start replication to the Secondary host using the `vxrds startrep` command with the `-b` option to replay all the pending updates. When the updates have been replayed and the status of the Secondary is up-to-date, you can either perform a takeover to convert the Secondary to a Primary, or restore the original Primary, if it becomes available again.

Syntax for `vxrds activatebunker` command:

```
vxrds [-g <diskgroup>] activatebunker <local_rvg>
```

Example:

```
vxrds -g vvrdg -b activatebunker rvg
```

## Creating and adding a Secondary RVG

Use the `vxrds addsec` command to create a Secondary RVG with the same name as the Primary RVG and add it to the RDS, to which the RVG belongs. The `addsec` command associates the existing data volumes and Replicator Log volumes on the Secondary node with the newly created Secondary RVG. It creates and associates Primary and Secondary RLINKs with the Primary and Secondary RVG.

Before using the `addsec` command, ensure that the data volumes and Replicator Log volume with the same name as that on the Primary node are present on the Secondary node.

By default, the `addsec` command sets the replication protocol to be UDP. This command can be executed from the Primary or any existing Secondary host in the RDS. If an RDS contains only a Primary host, then `addsec` must be executed from the Primary host.

**Note:** Do not run this command from the Secondary host that is added to the RDS. Also make sure that the volumes that you add to the Secondary RVG do not have a DRL.

Syntax for `vxrds addsec` command

```
vxrds [-g <diskgroup>] addsec <local_rvg> <pri_host> <sec_host>
 [attribute=value..]
```

Example

```
vxrds -g vvrdg addsec rvg pri_host sec_host
vxrds -g vvrdg addsec rvg pri_host sec_host prlink=rlk_to_sec
 \srlink=rlk_to_pri
```

## Adding an existing volume to the RDS

Use the `vxrds addvol` command to associate an existing volume as a data volume to the RDS, to which the RVG belongs. This command associates an existing volume with the corresponding RVGs on all the hosts in the RDS. The volume must already be present on all the hosts and must have the same name and size. Note that if the RDS contains a Bunker RVG then the `vxrds addvol` command ignores the Bunker RVG.

See "Alternative methods to synchronize the Secondary faster" on page 213.

See "Displaying or setting ports for `vxrsyncd`" on page 305.

**Note:** Volume Replicator does not synchronize the newly added volume, and replication starts from that point onwards.

Syntax for `vxrds addvol` command:

```
vxrds [-g<diskgroup>] addvol <local_rvg> <volume>
```

You can also perform difference-based synchronization using the `vxrsync` command as:

```
vxrds -g vvrdg addvol rvg volume
```

## Adding a Bunker node

Use the `vxrds addbunker` command to add a Bunker to an existing RDS without interrupting replication from the Primary to the Secondary.

On the Bunker node, create only the Bunker Replicator Log volume. You do not require to create the data volumes. Make sure that the Bunker Replicator Log is of the same size and has the same name as the Primary Replicator Log. The `vxrds addbunker` command takes care of creating the Bunker Secondary RVG and establishing the required RLINKs. To create the RLINKs with the names of your choice between the Primary and Bunker Secondary RVG use the `vxrds addbunker` command with the `prlink` and `srlink` attributes.

**Note:** Adding the Bunker RVG fails if the Replicator Log sizes differ. The Bunker Replicator Log must be of the same size and the same name as the Primary Replicator Log.

Syntax `forvxrds addbunker` command command:

```
vxrds [-g <diskgroup>] [-bdg <diskgroup>] addBunker
 <local_rvg><pri_host> <Bunker_host> [attribute=value..]
```

Example:

```
vxrds -g dg1 -bdg Bunker_dg addBunker local_rvg london
 london1protocol=storage
```

where `dg1` is the Primary disk group and `Bunker_dg` is the Bunker disk group.

The following table describes the attributes that you can specify with `vxrds addbunker` command.

**Table 7-4** Attributes for `vxrds addbunker` command

| Attributes | Description |
|---|---|
| `protocol` | Specifies the protocol to be used for replication between the Primary and Bunker Secondary.<br>If the storage on the Bunker Secondary is directly accessible from the Primary use the STORAGE protocol, otherwise use TCP/IP or UDP/IP.<br>**Note:** If the replication protocol for the Bunker Secondary has been set to STORAGE, then you can change it only by deleting and recreating the Bunker with the required protocol; UDP/IP or TCP/IP. You cannot change the STORAGE protocol using the Change Replication Settings option. |
| `-bdg <diskgroup>` | Specifies the Bunker disk group in a storage Bunker set up that is used for creating the Bunker RVG. |
| `prlink` | Creates the Primary RLINK with the specified name. |
| `srlink` | Creates the Bunker Secondary RLINK with the specified name. |

## Changing the host name or IP

Use the `vxrds changeip` command for changing the host name or the IP of the Primary and Secondary RLINKs used for replication.

```
vxrds [-g <diskgroup>] changeip <local_rvg><sec_host>\[attribute=value...]
```

The argument *sec_host* is the name of the Secondary host as displayed by the `vxrds printrvg` command and is not optional.

The `vxrds changeip` command changes the host name or IP address of Primary and Secondary RLINKs as specified by the `newpri` and `newsec` attributes. These attributes are of the form `attribute=value`.

You can also use the `vxrds set` command to perform the same operation.

See “Setting replication attributes” on page 247.

Syntax for `vxrds changeip` command

```
vxrds [-g <diskgroup>] changeip <local_rvg> <sec_host> \
[attribute=value...]
```

Example

```
vxrds -g vvrdg changeip rvg sec_host newpri=10.212.20.102
vxrds -g vvrdg changeip rvg sec_host newsec=10.212.20.105
```

**Note:** The `vxrds changeip` command can be used to set the Primary and Secondary replication IPs to non-existent IPs that are resolvable to a valid name.

The following table describes the attributes that you can specify with the `vxrds changeip` command.

**Table 7-5** Attributes for `vxrds changeip` command

| Attributes | Description |
|---|---|
| `newpri` | Name or IP address of the Primary RLINK to be used for replication. This can be used to set a specific IP for replication if the host has multiple IP addresses. |
| `newsec` | Name or IP address of the Secondary RLINK to be used for replication. This can be used to set a specific IP for replication if the host has multiple IP addresses. |

## Creating the Primary RVG

Use the `vxrds createpri` command to create a Primary RVG using the attributes that are available with the command. Before using the `createpri` command, use the `vxassist` command to create the data volumes and Replicator Log volumes with the required layout. Run the `createpri` command on the host that you configure as the Primary host of the new RDS.

Syntax for `vxrds createpri` command

```
vxrds -g <diskgroup> createpri <rvg_name> [attribute=value...]
```

Example

```
vxrds -g vvrdg createpri rvg vols=dv1,dv2 srl=rep_log rds=rds
```

The following table describes the attributes that can be specified with `vxrds createpri` command.

**Table 7-6** Attributes for `vxrds createpri` command

| Attributes | Description |
|---|---|
| `vols` | Specifies a comma-separated list of the data volumes. |
| `srl` | Specifies the volume name that needs to be used as the Replicator Log volume. |
| `rds` | Specifies the RDS name that needs to be associated to the RVG.<br>**Note:** If you do not specify the RDS name then the RVG name is considered as the RDS name. |

# Deactivating the Bunker RVG

Use the `vxrds deactivatebunker` command to deactivate the Bunker RVG after the replay of pending updates to the Secondary completes. After the replay completes, use the `vxrds stoprep` command to stop replication to the Secondary and then deactivate the Bunker to convert it back to a Bunker Secondary. You can now perform takeover on the up-to-date Secondary.

Syntax for `vxrds deactivatebunker` command

```
vxrds [-g <diskgroup>] deactivatebunker <local_rvg>
```

Example

```
vxrds -g vvrdg deactivatebunker rvg
```

# Deleting the Bunker node

Use the `vxrds delbunker` command to remove the Bunker node from the RDS. The operation that the `vxrds delbunker` command performs is irreversible.

**Note:** Before removing a Bunker, you must stop replication to the specified Bunker, using the `vxrds stoprep` command.

Syntax for `vxrds delbunker` command

```
vxrds [-g <diskgroup>] [-f] [-clean] delBunker
 <local_rvg><Bunker_host>
```

Example

```
vxrds -g vvr_dg -clean delBunker vvr_rvg london1
```

## Deleting the Secondary

Use the `vxrds delsec` command to delete the Secondary RVG on the host that the `sec_host` parameter specifies, from the RDS to which the RVG belongs. Use the same name that is displayed for the Secondary host in the output of the `vxrds printrvg` command. The `delsec` command dissociates the data volumes and the Replicator Log volume from the Secondary RVG, and deletes the Secondary RVG. The data volumes and Replicator Log volumes are not deleted from the SFW configuration.

**Note:** If Volume Replicator is configured in a VCS or Microsoft Cluster and the cluster resource for this RVG exists, then Volume Replicator fails the delete Secondary operation if the Secondary that you want to delete is part of this configuration.

Before running the `delsec` command, stop replication to the Secondary host.

Use the `vxrds stoprep` command to stop replication.

**Note:** This command dissociates the RLINKs, data volumes, and Replicator Log from the Secondary RVG. Use the `-clean` option to delete the dissociated RLINKs.

Syntax for `vxrds delsec` command

```
vxrds [-g<diskgroup>] [-f][-clean] delsec <local_rvg> <sec_host>
```

Example

```
vxrds -g vvrdg -clean delsec RVG sec_host
```

## Deleting the Primary

The `vxrds delpri` command deletes the Primary RVG, thereby deleting the corresponding RDS. This command only dissociates the data volumes and the Replicator Log from the Primary RVG. It does not delete the data volumes and the Replicator Log from the SFW configuration.

The `vxrds delpri` command fails if the Primary RVG to be deleted still has one or more Secondaries. Use the `vxrds delsec` command to delete all of its Secondaries before using the `vxrds delpri` command to delete the Primary RVG.

Use the `vxrds delpri` command with the `-f` option to forcefully delete the RVG even when the data access is enabled and the Primary data volumes are used. However, this can result in some data loss.

**Note:** If Volume Replicator is configured in a VCS or Microsoft Cluster and the cluster resource for this RVG exists, then Volume Replicator fails the delete Primary operation.

Syntax for `vxrds delpri` command:

```
vxrds [-g <diskgroup>] [-f] delpri <local_rvg>
```

Example

```
vxrds -g vvrdg delpri RVG
```

## Dissociating data volumes

Use the `vxrds delvol` command to dissociate the data volume from the RDS, to which the RVG belongs. The volumes are not physically deleted from the SFW configuration. Before using the `vxrds delvol` command make sure that the Secondary is up-to-date or you stop the replication to the Secondary. Use the `-f` option to forcefully delete the volumes, even if they are in use. Note that if the RDS contains a Bunker RVG then the `vxrds delvol` command ignores the Bunker RVG.

Syntax for `vxrds delvol` command

```
vxrds [-g<diskgroup>] [-f] delvol <local_rvg> <volume>
```

Example

```
vxrds -g vvrdg delvol rvg volume
```

## Resynchronizing a failed Primary with the new Primary

Use the `vxrds fbsync` to resynchronize the failed Primary with the new Primary after it becomes available again after the takeover operation. This command uses failback logging on the new Primary to synchronize data volumes on the failed Primary with the data volumes on the new Primary.

**Note:** This command can be run only if failback logging was enabled during takeover.

See "Taking over the Primary role using the `vxrds takeover` command" on page 252.

In the failback logging mode, Volume Replicator uses Data Change Maps (DCM) to track the changes happening on the new Primary while the original Primary is not available.

When the original Primary recovers, it needs to be synchronized with the new Primary by playing back the DCM on the new Primary. To receive the missing updates, the original Primary must first be converted to a Secondary. The `vxrds fbsync` command synchronizes the original Primary with the new changes on the new Primary by replaying the DCMs.

**Note:** The data on the Secondary data volumes is inconsistent for the duration of the replay.

Use the `vxrds fbsync` command with the `-wait` option to make sure that the command waits until the resynchronization completes.

**Note:** Do not use the `vxrds fbsync` command if the `-autofb` option was used at the time of the takeover.

Syntax for `vxrds fbsync` command

```
vxrds [-g <diskgroup>] [-wait] fbsync <rvg>
```

Example

```
vxrds -g vvrdg -wait fbsync rvg
```

## Converting a Primary to a Secondary

Use the `vxrds makesec` command to convert the specified Primary RVG to a Secondary RVG and associate it to the RDS, to which the RVG belongs. The `oldsec_hostname` parameter specifies the name of the host which is now the new Primary.

You must run this command only after you perform the Takeover operation.

**Note:** Make sure that you run the `vxrds makesec` command only on the original Primary.

If the original Primary RVG is part of a VCS cluster and the `RVGPrimary` resource exists, then Volume Replicator fails the `vxrvg makesec` command for this RVG.

Syntax for `vxrds makesec` command

```
vxrds [-g <diskgroup>] makesec <local_rvg> <oldsec_hostname>
```

Example

```
vxrds -g vvrdg makesec rvg host
```

## Migrating the Primary to a Secondary

Use the `vxrds migrate` command to interchange the Primary role with the specified Secondary. The `new_Primary_hostname` parameter specifies the Secondary host. The Primary role can only be migrated when the Secondary is active, consistent, and up-to-date. The data volumes in the RDS must be inactive, that is, the applications that are involved in replication must be stopped before you run the `vxrds migrate` command. After the migration is complete, the Primary and Secondary roles are interchanged.

If the original Primary has multiple Secondary hosts, and the RLINKs between every pair of Secondaries have not been created, then, after migrating the Primary role to one of the Secondaries or performing takeover on one of the Secondaries, all the remaining Secondaries in the RDS become orphan. You must manually delete these Secondaries and then again add them as Secondaries to the new Primary.

However, if RLINKs have been created between each pair of Secondaries in the RDS, then following steps can be used after migrate or takeover operation to add the orphaned Secondaries back in the RDS.

Syntax for `vxrds migrate` command

```
vxrds [-g <diskgroup>] migrate <local_rvg> <new_Primary_hostname>
```

Example

```
vxrds -g vvrdg migrate rvg sec_host
```

**To migrate the Primary to a Secondary**

1. On each orphaned Secondary host, detach the RLINK on this orphan Secondary pointing to the original Primary (the Primary host before migrate or takeover).
2. The orphan Secondaries join the RDS of the new Primary. Now, start replication with Automatic Synchronization on each of these orphans.
3. Alternatively, you can use the `vxrsync` utility to bring the Secondaries up-to-date. To do this, Start a checkpoint using the Start Checkpoint option. Use the `vxrsync` utility to perform difference-based synchronization to the Secondaries from the new Primary host.

   After the synchronization completes, End the checkpoint using the End Checkpoint option.

   Select the **Synchronize from Checkpoint** option to start replication from checkpoint to synchronize the Secondary with the writes that happened when `vxrsync` was in progress

   Because the RLINKs for the other Secondary hosts are still associated you do not need to use the `vxrds addsec` command to add the existing Secondary hosts to the new Primary after the migrate or takeover operation.

   You can choose to perform Automatic Synchronization or difference-based synchronization depending on the amount of the data that exists on the volumes. For example, if you have large volumes, but the actual data on it is very small, then Automatic Synchronization can be used as the `intellisync` option synchronizes only those bits of data that changed. However, if you have large amounts of data with comparable changes then the `vxrsync` difference-based synchronization is a better option.

   If the RVG is part of a VCS cluster and the `RVGPrimary` resource for the Primary RVG exists, then Volume Replicator fails the `vxrvg migrate` command for this RVG.

# Pausing replication using the vxrds pauserep command

Use the `vxrds pauserep` command to pause the replication to the specified Secondary host in the RDS, to which the RVG belongs. This command can be used only for a Primary initiated pause.

The `sec_host` parameter specifies the name of the Secondary host as displayed in the output of `vxrds printrvg` command.

Syntax for `vxrds pauserep` command

```
vxrds [-g <diskgroup>] pauserep <local_rvg> <sec_host>
```

Example

```
vxrds -g vvrdg pauserep rvg sec_host
```

# Displaying the RDS

Use the `vxrds printrvg` command to display the list of RVGs on all the hosts in the RDS, to which the RVG belongs. The host name, the RVG name, and the disk group are displayed for each host in the RDS. If the RDS consists of a Bunker RVG, then in addition to the Primary and Secondary RVG, information about the Bunker RVG is also displayed. If the RVG parameter is not specified, all the RDSs on the local host are displayed. Use the `-g <diskgroup>` option to display all the RDSes in a particular disk group.

The `-l` option displays information in a long format. This format displays additional information such as the data volume count, the Replicator Log name, RLINK name for each RVG in the RDS, its state, and the replication mode.

The `vxrds printrvg` command output resembles

```
Replicated Data Set : rds
Primary :
        Hostname : pri_host <localhost>
        RvgName  : rvg
        DgName   : vvrdg
Secondary :
        Hostname : sec_host
        RvgName  : rvg
        DgName   : vvrdg
```

The `vxrds printrvg` command output in case of a setup that has Bunker RVG resembles

```
Primary :
        Hostname : pri_host <localhost>
        RvgName  : rvg
        DgName   : vvrdg
Secondary :
        Hostname : sec_host
        RvgName  : rvg
        DgName   : vvrdg
Bunker(Secondary) :
        Hostname : bunker_host
        RvgName  : rvg
        DgName   : vvrdg
```

Syntax for `vxrds printrvg` command

```
vxrds [-g<diskgroup>] [-l] printrvg [<local_rvg>]
```

Example:

```
vxrds -g vvrdg printrvg rvg
```

# Resizing the data volumes

You can decrease or shrink the size of a data volume using the online volume shrink feature. The `vxrds resizevol` command is helpful in reclaiming unused space to better use your resource.

## Before resizing a data volume

Consider the following before shrinking a data volume:

- Before performing the volume shrink operation, you must install the KB 2615327 hotfix from Microsoft.
- If the combined length of the volume name and disk group name is more than 9 characters, then you must install the KB 2619083 hotfix from Microsoft before shrinking the volume.
- Online volume shrink is not supported on Volume Replicator Secondary hosts, Storage Replicator Log (SRL), non-NTFS, and read-only volumes, and volumes on which a task is performed. For resizing the SRL volumes, use the `vxrds resizesrl` command.
- For RDS configurations with only one Secondary host, the IBC messaging facility is used while shrinking the Secondary volume.
- For RDS configurations with more than one Secondary hosts, the RLINKs must be up-to-date before you perform a volume shrink operation. This is required because when the file system is shrunk during this operation, it may move some data clusters while defragmenting the volume and generate a large amount of I/O. Because of this, the RLINKs may not be up-to-date after the file system shrink, and the volume shrink operation may fail.
- In some cases, the Replicator Log overflows because of heavy I/Os during a volume shrink or defragmentation operation. Because of this, the volume shrink operation does not happen and, therefore, you may have a volume of the size greater than the file system at the Primary. In such cases, retry the volume shrink operation when the I/O is low after growing the file system by using the `vxvol growfs` command. For information about the command, refer to the *Storage Foundation Administrator’s Guide*.

### Shrinking a data volume

Use the `vxrds resizevol` command to grow or shrink the size of the specified data volume across the Replicated Data Source (RDS), that is, the specified volume gets resized uniformly on all the nodes in the RDS. You can grow NTFS, ReFS, or raw volumes. However, you can shrink only NTFS or raw volumes.

You can specify the length parameter in units of Mega Bytes (MB), Giga Bytes (GB) or Tera Bytes (TB). If you do not specify any suffix, such as K for KB, M for MB, G for GB and T for TB after the length parameter, then it is taken as MB by default.

See "Growing the Replicator Log volume" on page 246.

**Note:** The `length` parameter is used to specify the new size you want to grow the volume to. This command does not require you to stop replication before growing the volumes. However, the additional size by which you have grown the volume on all the hosts are not synchronized automatically by Volume Replicator after growing.

Syntax for the `vxrds resizevol` command:

```
vxrds [-g <diskgroup>] resizevol <local_rvg> <volume> <length>
```

Example

```
vxrds -g vvrdg resizevol rvg volume 100M
```

## Growing the Replicator Log volume

Use the `vxrds resizesrl` command to grow the size of the Replicator Log volume to the specified length, uniformly across the Primary and Bunker host on the RDS. Do not use the `vxassist growby` command to resize the Replicator Log as this causes the replication to pause. You must therefore use either the `vxrds resizesrl` or `vradmin resizesrl` command to resize the Replicator Log uniformly across the Bunker and Primary host.

You can specify the length parameter in units of Mega Bytes (MB), Giga Bytes (GB) or Tera Bytes (TB). If you do not specify any suffix, such as K for KB, M for MB, G for GB and T for TB after the length parameter, then it is taken as MB by default.

Syntax for `vxrds resizesrl` command

```
vxrds [-g <diskgroup>] [-f] resizesrl <local_rvg> <length>
```

Example

```
vxrds -g vvrdg resizesrl rvg 200M
```

## Resuming replication after pausing

Use the `vxrds resumerep` command to resume the replication to the Secondary host in the RDS, to which the RVG belongs. The `sec_host` parameter is the name of the Secondary host that is displayed in the output of the `vxrds printrvg` command. The Primary RLINK must be in PAUSED state to resume replication.

Syntax for `vxrds resumerep` command:

```
vxrds [-g <diskgroup>] resumerep <local_rvg> <sec_host>
```

Example:

```
vxrds -g vvrdg resumerep rvg sec_host
```

## Resynchronizing the Secondary

Use the `vxrds resync` command for synchronizing the Secondary data volumes when the Replicator Log has already overflowed and the log protection is set to DCM. This command replays the DCM to synchronize the Secondary hosts. To resynchronize the Secondary hosts the RLINK must be in the CONNECTED state or the Secondary must be ACTIVE.

**Note:** When DCM logs are used to synchronize the data, the Secondary is inconsistent until the synchronization process completes.

Use the `vxrds resync` command with the `-wait` option to make sure that the command waits until the resynchronization completes.

Syntax for `vxrds resync` command

```
vxrds [-g <diskgroup>] [-wait] resync <local_rvg>
```

Example

```
vxrds -g vvrdg resync rvg
```

## Setting replication attributes

Use the `vxrds set` command sets the specified attributes on the Secondary RLINK and corresponding Primary RLINK.

Syntax for `vxrds set` command

```
vxrds [-g <diskgroup>] set <local_rvg> <sec_host> attribute=value...
```

Example

```
vxrds -g vvrdg set rvg sec_host synchronous=override srlprot=dcm
```

The following table describes the different attributes that you can set for `vxrds set` command.

**Table 7-7** Attributes for `vxrds set` command

| Attribute | Description |
|---|---|
| `synchronous` | Specifies the mode of replication. This attribute can be set to the following values:<br>`synchronous=off` for asynchronous mode of replication.<br>`synchronous=override` for synchronous override mode of replication.<br>Set `synchronous=fail` for synchronous mode of replication. |
| `srlprot` | Enables or disables log protection. The data volumes must have a DCM log for `srlprot` to be set to DCM or AutoDCM. This attribute can be set to the following values:<br>`srlprot=autodcm` enables log protection. The DCM logs are used to synchronize the data when the Replicator Log overflows, even when the Primary and Secondary are connected.<br>`srlprot=dcm` enables log protection. The DCM logs are used to synchronize the data if the Replicator Log overflows, when the Primary and Secondary are disconnected.<br>`srlprot=override` enables log protection. If the Secondary is still connected and the Replicator Log is about to overflow then the writes are stalled until a predetermined amount of space, that is, 5% or 20 MB (whichever is lesser) becomes available in the Replicator Log. The log protection is automatically disabled if the Secondary becomes inactive due to a disconnection or administrative action, and Replicator Log will overflow.<br>`srlprot=fail` enables log protection. If the log is about to overflow the writes are stalled until a predetermined amount of space, that is, 5% or 20 MB (whichever is lesser) becomes available in the Replicator Log. If the connection between Primary and Secondary RVG is broken, then, any new writes to the Primary RVG are failed.<br>`srlprot=off` disables log protection.<br>See “Replicator Log overflow protection—`srlprot` attribute” on page 48. |

**Table 7-7** Attributes for `vxrds set` command *(continued)*

| Attribute | Description |
|---|---|
| `latencyprot` | Enables or disables latency protection. This attribute can be set to the following values:<br>`latencyprot=off` disables latency protection.<br>`latencyprot=override` enables latency protection. However, latency protection is automatically disabled if the RLINK becomes inactive due to a disconnection or administrative action.<br>`latencyprot=fail` enables latency protection.<br>See “Latency protection—`latencyprot` attribute” on page 52. |
| `latency_high_mark` | Specifies the maximum number of outstanding requests that are allowed when latency protection is enabled. |
| `latency_low_mark` | Specifies a value such that when the writes are stalled, the number of outstanding requests must drop to this value before latency protection can be disabled. |
| `pri_host` | Name or IP address of the Primary host. This can be used to set or modify a specific IP for replication if the host has multiple IP addresses. |
| `sec_host` | Name or IP address of the Secondary host. This can be used to set or modify a specific IP for replication if the host has multiple IP addresses. |
| `packet_size` | Specifies the size of packets in which data can be sent through the network during replication.<br>**Note:** Some firewalls do not support packet sizes greater than 1400 bytes. If you replicate across such a firewall, then use the default packet size to make sure all the Volume Replicator operations function as required or you can choose to set it to a packet size of 1100 bytes.<br>If you specify a value smaller than 1100 bytes then it is automatically rounded off to 1100 bytes. Similarly, if you specify a value greater than 64400, it is automatically rounded off to 64400 bytes.<br>Within the range of 1100 to 1400 bytes you can choose to specify any value in multiples of four. If the value you specify is not a multiple of four it is automatically rounded off to the next higher value that is a multiple of four.<br>From 1400 onwards any packet size that you specify is rounded off to the next multiple of 1400. |

**Table 7-7** Attributes for `vxrds set` command *(continued)*

| Attribute | Description |
|---|---|
| `bandwidth_limit` | Specifies a value that can be used to control the bandwidth that Volume Replicator needs to use for replication. If this attribute is not specified Volume Replicator uses the entire available bandwidth for replication. To disable bandwidth throttling, set this attribute to `none`. Note that the specified bandwidth value must be at least 1 `Mbps` (Megabits per second). You can specify the value in units of `Kbps`, `Mbps`, `Gbps`, or `bps`. The default is Kbps. If no value is specified then bandwidth throttling is disabled. |
| `protocol` | Specifies the protocol to be used for replication between the Primary and Secondary. Specify TCP or UDP.<br>If the setup includes a Bunker Secondary and replication is over IP, the protocol can be set to UDP or TCP. The default is UDP.<br>If the storage at the Bunker Secondary is directly accessible from the Primary, for example, DAS or SAN, use the STORAGE protocol, otherwise use TCP/IP or UDP/IP<br>**Note:** If the replication protocol for the Bunker Secondary has been set to STORAGE then you can change it only by deleting and recreating the Bunker with the required protocol; UDP/IP or TCP/IP. You cannot change the STORAGE protocol using the Change Replication Settings option. |
| `compression` | Specifies whether compression is enabled or disabled and takes the value of `true` and `false` respectively. |

## Starting replication using the `vxrds startrep` command

Use the `vxrds startrep` to start the replication to the normal or Bunker Secondary host in the RDS, to which the RVG belongs. The `sec_host` parameter is the name of the Secondary host that is displayed in the output of the `vxrds printrvg` command.

The `startrep` command attaches the Secondary host to the Primary to establish a communication link and start replication.

If the Primary becomes unavailable due to a disaster, then use the

`vxrds startrep` command with the `-b` option to start replication from the Bunker Primary to the Secondary. In this scenario the `vxrds startrep` command also switches the RLINKs to point to the Bunker Primary instead of the original Primary.

See “Activating the Bunker RVG” on page 233.

Syntax for `vxrds startrep` command

```
vxrds [-g <diskgroup>] -c <checkpoint>| -f | -forceatt |-autosync|
|-a|-b startrep <local_rvg> <sec_host>
```

Example

```
vxrds -g vvrdg -forceatt startrep rvg sec_host
```

The following table describes the attributes that you can specify with the `vxrds startrep` command.

**Table 7-8** Attributes for `vxrds startrep` command

| Option | Description |
|---|---|
| `-autosync` or `-a` | Use this option to automatically synchronize the Secondary data volumes. |
| `-b <startrep>` | Use this option to replay pending updates from Bunker Primary to Secondary. |
| `-c <checkpoint>` | Use this option to attach the Secondary to the Primary with the specified checkpoint<br>**Note:** This option is not supported when synchronizing a Bunker RVG with the Primary RVG. |
| `-forceatt` or `-f` | Use this option to start replication if the Secondary data volumes contain exactly the same data as the Primary data volumes and therefore there is no need to synchronize the Secondary data volumes.<br>If the data volumes are not synchronized then the `-f` option can cause data corruption as replication is started immediately and the Secondary may miss the writes that may have been present on the Replicator Log but did not reach the Secondary. Because there is no way of knowing whether the Replicator Log had some pending writes that have not yet been sent to the Secondary, use this option only if you know that the Secondary is completely up-to-date.<br>**Note:** This option is not supported when synchronizing a Bunker RVG with the Primary RVG. |

These attributes are mutually exclusive and only one of these options can be used with the `startrep` command. The function of these options is similar to the function of the same options available with the `vxrlink att` command.

See "Attaching a Secondary" on page 256.

## Stopping replication using the vxrds stoprep command

Use the vxrds stoprep command to stop the replication to the normal or Bunker Secondary host in the RDS, to which the RVG belongs. The sec_host parameter is the name of the Secondary host that is displayed in the output of the vxrds printrvg command. The stoprep command stops the replication by detaching the RLINKs on the Secondary and the Primary host.

Syntax for vxrds stoprep command

```
vxrds [-g <diskgroup>] stoprep <local_rvg> <sec_host>
```

Example

```
vxrds -g vvrdg stoprep rvg sec_host
```

## Taking over the Primary role using the vxrds takeover command

Use the vxrds takeover command to enable the Secondary host to take over the Primary role. When the automatic failback feature is enabled using the -autofb option, the original Primary automatically becomes the Secondary, after it is available again.

The takeover command must be run on the Secondary host in the RDS and it works only when the Primary host in the RDS is down or not reachable from the Secondary due to some problems such as, network failure or HBA failure, or due to a disaster. After takeover the Secondary RVG is converted to a Primary RVG. However, the original Primary must become available again for the fast-failback to work successfully.

**Note:** The Secondary must be consistent for this command to work.

Syntax for vxrds takeover command

```
vxrds [-g <diskgroup>] [[-F] [-autofb]] | [-N] takeover<local_rvg>
```

For example, to perform takeover with failback logging and Automatic Synchronization run the vxrds takeover command as follows:

```
vxrds -g vvrdg -autofb takeover rvgvxrds -g vvrdg -f takeover rvg
```

To perform takeover without the failback logging option, run the vxrds takeover command as follows

```
vxrds -g vvrdg -N takeover rvg
```

## Performing RLINK Operations using the `vxrlink` command

RLINK objects are associated with RVGs. Each RLINK on a Primary RVG represents the communication link from the Primary RVG to a corresponding Secondary RVG. An RLINK on a Secondary RVG represents the communication link from the Secondary RVG to the corresponding Primary RVG.

An RLINK reads data from the Replicator Log volume and sends it to the Secondary. All the RLINKs in an RVG share the Replicator Log volume, and each RLINK reads data at its own rate. An update is removed from the Replicator Log volume when all the RLINKs have successfully sent the update to the Secondary.

The `vxrlink` command along with its keywords and options can be used to perform the Volume Replicator operations on the RLINKS.

The following table lists the keywords that are available with the `vxrlink` command and their respective descriptions.

**Table 7-9** Keywords available for `vxrlink` command

| Keyword | Description |
|---|---|
| `assoc` | Associates an RLINK to an RVG.<br>See "Associating a Secondary" on page 256. |
| `att` | Attaches an RLINK to an RVG.<br>See "Attaching a Secondary" on page 256. |
| `cplist` | Displays the list of currently available Secondary checkpoints.<br>See "Displaying the list of Secondary checkpoints" on page 257. |
| `checkdelete` | Deletes the specified Secondary checkpoint.<br>See "Deleting the Secondary checkpoint" on page 257. |
| `det` | Detaches an RLINK from an RVG.<br>See "Detaching an RLINK" on page 257. |
| `dis` | Disassociates an RLINK from an RVG.<br>See "Dissociating an RLINK" on page 258. |
| `make` | Creates an RLINK.<br>See "Creating new RLINK" on page 258. |
| `pause` | Pauses an RLINK.<br>See "Pausing the RLINK" on page 260. |

**Table 7-9** Keywords available for `vxrlink` command *(continued)*

| Keyword | Description |
|---|---|
| `recover` | Recovers an RLINK.<br>See “Recovering the RLINK” on page 261. |
| `restore` | Restores an RLINK.<br>See “Restoring the RLINK” on page 261. |
| `resume` | Resumes an earlier paused RLINK.<br>See “Resuming the RLINK” on page 262. |
| `rm` | Deletes an RLINK with the given name.<br>See “Removing the RLINK” on page 262. |
| `set` | Sets the attributes of the specified RLINK.<br>See “Setting the RLINK attributes” on page 262. |
| `stats` | Displays the network statistics for the specified Secondary.<br>See “Displaying the network statistics for the RLINK” on page 264. |
| `status` | Displays the replication status for a specific Secondary.<br>See “Displaying the RLINK status” on page 267. |
| `updates` | Displays the ID of the latest update the Secondary received and the number of updates by which the Primary is ahead.<br>See “Identifying the most up-to-date Secondary” on page 269. |
| `verify` | Verifies the specified RLINK or all the RLINKs in the RVG for configuration errors.<br>See “Verifying the RLINK” on page 270. |
| `startstats` | Verifies the bandwidth usage by RLINKs in an RDS by starting the historic bandwidth data collection in the form of a graph file.<br>See See “Starting the Historic Bandwidth Data Collection using the CLI” on page 271. |
| `stopstats` | Verifies the bandwidth usage by RLINKs in an RDS by stopping the historic bandwidth data collection in the form a graph file.<br>See See “Stopping the Historic Bandwidth Data Collection using the CLI” on page 272. |

The following table lists the options that are available with the `vxrlink` command.

**Table 7-10** Options available for the `vxrlink` command

| Option | Description |
|---|---|
| `-a` | Automatically attaches and synchronizes the Secondary data volumes. Optionally used with the `att` command on the Primary.<br>**Note:** The autosync operation proceeds only if all the data volumes in an RDS or Primary have DCM logs and if the RLINK is able to connect to the Secondary. |
| `-c <checkpoint>` | Attaches an RLINK which is consistent up to the point indicated by the checkpoint string. The `-c` option can be used with the:<br>- `vxrlink restore` command to indicate from where to start the restore operation.<br>- `vxrlink checkdelete` command to specify the checkpoint that needs to be deleted.<br>- `vxrlink pause` command to mark a point at which a backup of the Secondary has been taken.<br>- `vxrlink att` command to indicate the point from were to start the synchronization when attaching the RLINK. |
| `-f` | Forces the attach of an RLINK to an RVG to succeed, even though the `-a` or `-c <checkpoint>` option was not specified.<br>**Note:** This `-f` option can cause data corruption because the Secondary may miss the writes that may have been present on the Replicator Log but did not reach the Secondary. Because there is no way of knowing whether the Replicator Log had some pending writes that have not yet been sent to the Secondary, use this option only if you know that the Secondary is completely up-to-date. |
| `-g <diskgroup>` | Specifies the local disk group for the operation. |
| `-i <interval>` | Displays the network statistics or replication status at the specified intervals in seconds specified by this option. |
| `-r <rvg>` | Specifies the name of the RVG with which the RLINK is associated. If the RVG name is not specified, the RLINK is examined to retrieve the name of the associated RVG. |
| `-t <timestamp>` | This option specifies the number of lines in the output after which the timestamp is displayed. |
| `-T` | Displays the actual difference in time by which the Secondary is behind. |

**Table 7-10** Options available for the `vxrlink` command *(continued)*

| Option | Description |
|---|---|
| `-w` | Forces a Secondary RLINK into the FAIL state. Used only in special circumstances such as the Secondary online backup. The RLINK status is displayed as inconsistent. |

## Associating a Secondary

Use the `vxrlink assoc` command to associate an RLINK with an RVG. Alternatively, the association can be specified when you use the `vxrvg make` command to create the RVG.

Syntax for `vxrlink assoc` command

```
vxrlink [-g<diskgroup>] assoc <rvg> <rlink>
```

Example

```
vxrlink -g vvrdg assoc rvg rlink
```

## Attaching a Secondary

Use the `vxrlink att` command to attach one or more RLINKs to an RVG. The RLINK must already be associated with the RVG before the attach. An RLINK on the Secondary can be attached at any time to indicate that it is ready for use as a Secondary RLINK. For the attach to succeed, the `remote_host`, `remote_dg`, and `remote_rlink` attributes must be set on both the Primary and the Secondary. These can be set during RLINK creation (using the `vxrlink make` command) or with the `vxrlink set` command.

The attach fails if the Primary RVG does not have a Replicator Log volume associated with it.

**Note:** Ensure that the data volumes on the Secondary are also of the same name and size as on the Primary for attach to succeed.

When attaching the RLINK you must specify the `-c <checkpoint>`, `-f`, or `-a` option.

Syntax for `vxrlink att` command

```
vxrlink -a|-b -c <checkpoint>|-f [-g <diskgroup>] [-r <rvg>] \
att <rlink>
```

Example

```
vxrlink -g vvrdg -a att rlink
```

## Displaying the list of Secondary checkpoints

Use the `vxrlink cplist` command to display the list of currently available Secondary checkpoints. Any checkpoint from this list can be used for restoring the corresponding Secondary. This command can be run either from the Primary or Secondary host.

Syntax for `vxrlink cplist` command

```
vxrlink [-g <diskgroup>] [-r <rvg>] cplist <rlink>
```

Example

```
vxrlink -g vvrdg -r rvg cplist rlink
```

## Deleting the Secondary checkpoint

Use the `vxrlink checkdelete` command to delete the specified Secondary (RLINK) checkpoint. This command must be run on the Primary and you must specify the Primary RLINK to the required Secondary.

See "Pausing the RLINK" on page 260.

Syntax for `vxrlink checkdelete` command

```
vxrlink [-g <diskgroup>] -c <checkpoint> checkdelete <rlink>
```

## Detaching an RLINK

Use the `vxrlink det` command to detach an RLINK from the Primary or Secondary RVG.

**Note:** After the RLINKs have been detached, synchronize all the Secondary volumes completely, before reattaching them.

Syntax for `vxrlink det` command

```
vxrlink [-g <diskgroup>] [-r <rvg>] det <rlink>
```

Example

```
vxrlink -g vvrdg -r rvg det rlink
```

# Dissociating an RLINK

Use the `vxrlink dis` command to dissociate an RLINK from the RVG to which it is associated. This cannot be executed if the RLINK is currently attached.

Syntax for `vxrlink dis` command

```
vxrlink [-g <diskgroup>] [-r <rvg>] dis <rlink>
```

Example

```
vxrlink -g vvrdg -r rvg dis rlink
```

# Creating new RLINK

Use the `vxrlink make` command to create a new RLINK based on the attributes that have been specified.

Syntax for `vxrlink make` command

```
vxrlink -g <diskgroup> make <name> attribute=value
```

Example

```
vxrlink -g vvrdg make rlk_sechost synchronous=override \
local_host=prihost remote_host=sec_host remote_dg=vvrdg \
remote_rlink=rlk_prihost srlprot=off latencyprot=fail \
latency_high_mark=10000 latency_low_mark=9950 protocol=TCP
```

The following table lists the attributes that can be specified for the `vxrlink make` command.

**Table 7-11** Attributes for the `vxrlink make` command

| Attribute | Description |
|---|---|
| `synchronous` | Indicates the mode in which the RLINK should operate; synchronous, asynchronous, or synchronous override mode. The attribute can be set to one of the following values:<br>■ Set `synchronous=off` for asynchronous mode.<br>■ Set `synchronous=override` for synchronous override mode.<br>■ Set `synchronous=fail` for synchronous mode. |
| `local_host` | Sets the name or IP address of the local host. |
| `remote_host` | Sets the name or IP address of the remote host. |
| `remote_dg` | Sets the name of the remote disk group. |

**Table 7-11** Attributes for the vxrlink make command *(continued)*

| Attribute | Description |
|---|---|
| remote_rlink | Sets the name of the remote RLINK. |
| latencyprot | Indicates whether latency protection is enabled for the RLINK. The attribute can have one of following values:<br>■ Set latencyprot=off to disable latency protection.<br>■ Set latencyprot=override to enable latency protection. It is automatically disabled if the RLINK becomes inactive due to a disconnection or administrative action.<br>■ Set latencyprot=fail to enable latency protection.<br>See "Latency protection—latencyprot attribute" on page 52. |
| srlprot | Enables or disables log protection. The data volumes must have a DCM log for srlprot to be set to DCM or AutoDCM. This attribute can be set to the following values:<br>srlprot=autodcm enables log protection. The DCM logs are used to synchronize the data when the Replicator Log overflows, even when the Primary and Secondary are connected.<br>srlprot=dcm enables log protection. The DCM logs are used to synchronize the data if the Replicator Log overflows, when the Primary and Secondary are disconnected.<br>srlprot=override enables log protection. If the Secondary is still connected and the Replicator Log is about to overflow then the writes are stalled until a predetermined amount of space, (that is, 5% or 20 MB (whichever is lesser) becomes available in the Replicator Log. The log protection is automatically disabled if the Secondary becomes inactive due to a disconnection or administrative action, and Replicator Log will overflow.<br>srlprot=fail enables log protection. If the log is about to overflow the writes are stalled until a predetermined amount of space, that is, 5% or 20 MB (whichever is lesser) becomes available in the Replicator Log. If the connection between Primary and Secondary RVG is broken, then, any new writes to the Primary RVG are failed.<br>srlprot=off disables log protection.<br>See "Replicator Log overflow protection—srlprot attribute" on page 48. |
| latency_high_mark | Specifies the maximum number of outstanding requests that are allowed when latency protection is enabled. |

**Table 7-11** Attributes for the vxrlink make command *(continued)*

| Attribute | Description |
|---|---|
| latency_low_mark | Specifies a value such that, when the writes are stalled because the number of outstanding write requests is higher than latency_high_mark value, then, the number of outstanding requests must drop to this value before latency protection is disabled. |
| packet_size | Specifies the size of packets in which data can be sent over the network during replication.<br>See "Setting replication attributes" on page 247. |
| bandwidth_limit | Specifies a value that can be used to control the bandwidth that Volume Replicator needs to use for replication. If this attribute is not specified,Volume Replicator uses the entire available bandwidth for replication. To disable bandwidth throttling, set this attribute to none. Note that the specified bandwidth value must be at least 56 Kbps. You can specify the value in units of Kbps, Mbps, Gbps, or bps. The default is Kbps. If no value is specified then bandwidth throttling is disabled. |
| protocol | Specifies the protocol to be used for replication between the Primary and Secondary. Specify TCP or UDP. |
| bunker | Specifies the bunker flag. The value can be either true or false. |
| bunker_target | Specifies the bunker_target flag. The value can be either true or false. |
| Compression | Specifies whether compression is enabled or disabled and as such takes the value of true or false respectively. |

# Pausing the RLINK

Use the vxrlink pause command to pause updates to the RLINK until you run the vxrlink resume command. New updates are logged while the RLINK is paused, and are applied once the RLINK is resumed.

On the Primary host, if the DCM is replayed on the RLINK to be paused, the replay pauses until the Secondary is resumed.

On the Secondary host, the - c <checkpoint> option is valid and can be used to mark a point at which a backup of the Secondary has been taken. This checkpoint is later used for restoring the Secondary. To delete this checkpoint you can use the vxrlink checkdelete command.

See "Deleting the Secondary checkpoint" on page 257.

**Note:** The `-w` option is used on the Secondary host for pausing the Secondary in special cases, to force the Secondary RLINK into the FAIL state. You may need to do this before restoring the Secondary from an online backup.

Syntax for `vxrlink pause` command

```
vxrlink [-c <checkpoint>|-w] [-g <diskgroup>] [-r <rvg>] pause
<rlink>
```

Example

```
vxrlink -g vvrdg -r rvg pause rlink
```

## Recovering the RLINK

Use the `vxrlink recover` command if the output of the `vxprint -l <rlink>` command displays the `needs_recovery` flag indicating that the RLINK needs to be recovered. This command recovers the RLINK if automatic recovery of RLINK does not happen.

Syntax for `vxrlink recover` command

```
vxrlink [-g <diskgroup>] [-r <rvg>] recover <rlink>
```

Example

```
vxrlink -g vvrdg -r rvg recover rlink
```

## Restoring the RLINK

Use the `vxrlink restore` command to restore the state of the RLINK from the FAIL state. Valid only for Secondary. This command is used when you restore data volumes at a Secondary host from online backup data that is maintained at the Secondary site (as opposed to restoring Secondary data volumes using the data that is copied from the Primary host).

**Note:** The restore keyword must be used with the `-c <checkpoint>` option to specify the checkpoint corresponding to the backup, that is used to perform the restore operation.

Syntax for `vxrlink restore` command

```
vxrlink -c <checkpoint> [-g <diskgroup>] [-r <rvg>] restore <rlink>
```

Example

```
vxrlink -g vvrdg -r rvg -c checkpoint restore rlink
```

## Resuming the RLINK

Use this command to resume replication to a Secondary that has been paused. After the replication is resumed, all writes that were logged while the RLINK was paused are sent to the Secondary.

Syntax for `vxrlink resume` command:

```
vxrlink [-g <diskgroup>] [-r <rvg>] resume <rlink>
```

Example

```
vxrlink -g vvrdg -r rvg resume rlink
```

## Removing the RLINK

Use the `vxrlink rm` command to remove the specified RLINK from the disk group. Use the `-f` option to delete the RLINK forcefully even if it is attached and associated to an RVG.

Syntax for `vxrlink rm` command

```
vxrlink [-g <diskgroup>] [-f] rm <rlink>
```

Example

```
vxrlink -g vvrdg -f rm rlink
```

## Setting the RLINK attributes

Use the `vxrlink set` command to set the specified attribute field to the RLINK. The attribute names specify the field that needs to be set within the specified RLINK.

Syntax for `vxrlink set` command

```
vxrlink [-g <diskgroup>] set attribute=value....<rlink>
```

Example

```
vxrlink -g vvrdg -r rvg set synchronous=off rlink
vxrlink -g vvrdg -r rvg set srlprot=dcm rlink
vxrlink -g vvrdg -r rvg set bandwidth_limit=2M rlink
vxrlink -g vvrdg -r rvg set synchronous=off srlprot=autodcm
latencyprot=fail packet_size=1400 protocol=TCP
bandwidth_limit=2M compression=true rlink
```

See “Setting replication attributes” on page 247.

The following table lists the `vxrlink set` command attributes.

**Table 7-12** Attributes for `vxrlink set` command

| Attribute | Description |
|---|---|
| `synchronous` | Specifies the mode of replication. |
| `srlprot` | Enables or disables log protection. |
| `latencyprot` | Enables or disables latency protection. |
| `latency_high_mark` | Specifies the maximum number of outstanding requests that are allowed when latency protection is enabled. |
| `latency_low_mark` | Specifies a value such that when the writes are stalled, the number of outstanding requests must drop to this value before latency protection can be disabled. |
| `local_host` | Specifies the name or IP address of the local host. |
| `remote_host` | Specifies the name or IP address of the remote host. |
| `packet_size` | Specifies the size of packets in which data can be sent through the network during replication.<br>See “Setting replication attributes” on page 247. |
| `bandwidth_limit` | Specifies a value that can be used to control the bandwidth that Volume Replicator needs to use for replication. If this attribute is not specified Volume Replicator uses the entire available bandwidth for replication. To disable bandwidth throttling, set this attribute to `none`. Note that the specified bandwidth value must be at least 56 Kbps. You can specify the value in units of `Kbps`, `Mbps`, `Gbps`, or `bps`. The default is Kbps. If no value is specified then bandwidth throttling is disabled. |

**Table 7-12** Attributes for `vxrlink set` command *(continued)*

| Attribute | Description |
|---|---|
| `protocol` | Specifies the protocol to be used for replication between the Primary and Secondary. Specify TCP or UDP.<br>If the setup includes a Bunker Secondary and replication is over IP, the protocol can be set to UDP or TCP. The default is UDP.<br>If the storage at the Bunker Secondary is directly accessible from the Primary, for example, DAS or SAN, use the STORAGE protocol, otherwise use TCP/IP or UDP/IP<br>**Note:** If the replication protocol for the Bunker Secondary has been set to STORAGE then you can change it only by deleting and recreating the Bunker with the required protocol; UDP/IP or TCP/IP. You cannot change the STORAGE protocol using the Change Replication Settings option. |
| Compression | Specifies whether compression is enabled or disabled and takes the value true or false respectively. |

# Displaying the network statistics for the RLINK

Use the `vxrlink stats` command to display the network statistics for a Secondary host to which the specified RLINK points.

The values that are displayed are cumulative except when used with the `-i` option. In this case the values indicate the change since the last time interval.

Syntax for `vxrlink stats` command

```
vxrlink [-g <diskgroup_name>] [-p] [-e] [-r <rvg_name>]
[-i    <interval>] [-t <timestamp>] stats <rlink_name>
```

**Note:** The `-e` option is a hidden option and is used for diagnostic purposes.

## Output values for vxrlink stats without the `-e` option

When `vxrlink stats` is used without the `-e <extended rlink stats>` option, it displays the values as mentioned in the table below.

The following table describes the values for the `vxrlink stats` command without the `-e` option.

**Table 7-13** vxrlink stats values without the -e option

| Values | Description |
|---|---|
| # | Number of messages transmitted |
| Blocks | Number of 512-byte blocks transmitted |
| RT | Average round-trip time per message |
| Timeout | Number of time-out errors |
| Stream | Number of stream errors |
| Memory | Number of errors due to insufficient buffer space on the Secondary |
| Flow Control | Displays the internal flow control parameters that indicate how fast the RLINK sends messages. These values are displayed in terms of Transmission Delays, Network Bytes, and Network Delays |

Example

```
vxrlink -g vvrdg -r rvg -i 5 stats rlink
```

The minimum value for `-i <interval>` is 1 second and the minimum value for `-t <timestamp>` is 10 seconds. If the `interval (-i option)` value is not specified, then the statistics are displayed only once.

```
vxrlink -g vvrdg -i 1 -t 25 stats rlk
```

## Output values for vxrlink stats with the -e option

When `vxrlink stats` is used with the `-e` option, some additional error values are also displayed.

The following figure lists the values for the `vxrlink stats` command with the `-e` option.

**Table 7-14** `vxrlink stats` values with the `-e` option

| Values | Description |
|---|---|
| SendRate | Specifies the rate of data (in Mbps) sent by Primary to the Secondary RVG, considering all data is sent in uncompressed form. |

**Table 7-14** `vxrlink stats` values with the `-e` option *(continued)*

| Values | Description |
|---|---|
| `#Msgs` | Specifies the messages that are sent in compressed form. |
| `OriginalSz` | Specifies the total size of the uncompressed form of the data that is compressed. |
| `CompressedSz` | Specifies the total size of the compressed form of data that is compressed. |
| `BWUsed` | Specifies the bandwidth that Volume Replicator uses (in Mbps)while sending the data to the Secondary RVG. |
| `BWSaved` | Specifies the bandwidth that Volume Replicator saves (in percentage)while sending the data in compressed form, as compared to sending data in an uncompressed form. |
| `NoSlot` | Errors due to non-availability of slots to hold incoming messages on Secondary. |
| `NoMemPool` | Number of memory pool errors due to insufficient amount of buffer space on Secondary for holding the incoming messages |
| `MissPkt` | Number of missing packet errors |
| `MissMsg` | Number of missing message errors |
| `Chksum` | Number of checksum errors |
| `Trans` | Transaction errors on the Secondary |
| `Compressed` | Data size after Compression is enabled. |
| `Uncompressed` | Specifies the original data size for the messages that are compressed. |

### Syntax for `vxrlink stats` command when used with the `-e` option

```
vxrlink [-g <diskgroup>] [-p] [-e] [-r <rvg>]
[-i <interval>]    [-t <timestamp>] stats <rlink>
```

### Example

```
vxrlink -i 5 -e stats rlink
```

The following table summarizes the options for the `vxrlink stats` command.

**Table 7-15** Available options for `vxrlink stats` command

| Options | Description |
|---|---|
| `-g <diskgroup>` | Specifies the disk group for the various operations. |
| `-p` | Shows the statistics for each connection of an RLINK. Useful for debugging performance problems. |
| `-e <extended stats>` | The `-e <extended stats>` option is used for diagnostic or analytical purposes. |
| `-i` | Displays the statistics at specified time interval. Note that the `-i <interval>` option should be specified in seconds and it represents the frequency at which the statistics of the RLINK are displayed. |
| `-t` | Specifies the number of times the stats are displayed before printing the next header |

# Displaying the RLINK status

Use the `vxrlink status` command to display the replication status of the Secondary, represented by the specified RLINK. This command can be run only on the Primary. During normal replication, if the Secondary is not up-to-date, the command displays the number of outstanding writes, and the percentage of the Replicator Log used by this RLINK. The command also displays the status of the autosynchronization process when it is in progress.

Use the `-i <interval>` option to display the replication status at the specified time intervals.

The `-t <timestamp>` option specifies the number of lines after which the current date and time is displayed.

Use the `-T` option to display the units of time by which the Secondary lags if it is not up-to-date.

When failback logging is enabled this command can be used on the new Primary and the original Primary. If the `vxrlink status` command is used on the new Primary, the command displays the status of the RLINK corresponding to the original Primary.

After takeover, if the `vxrlink status` command is used on the original Primary (acting Secondary) then the command appropriately displays the status of the Replicator Log or DCM replay being performed to the new Primary.

See “Performing disaster recovery operation” on page 199.

Syntax for `vxrlink status` command

```
vxrlink [-g <diskgroup>] [-r <rvg>] [-i <interval>]
\[-t    <timestamp>] [-T] status <rlink>
```

Example

```
vxrlink -g vvrdg -r rvg -i 5 status rlink_sec_host
```

**Note:** Interval must be specified in seconds. If the interval (`-i` option) value is not specified, then the statistics are displayed only once.

```
vxrlink -i1 -t 10 -T status rlink_sec_host
```

The output resembles:

```
4/6/2005 11:38:21 AM
RLINK is up to date. RLINK is up to date.
RLINK has 47 outstanding writes, occupying less than 1% (2994 KB)
of the Replicator Log.
RLINK rlink_sec_host is behind by 0 hrs 0 mins 0 secs with respect
to Primary.
RLINK has 56 outstanding writes, occupying less than 1% (3591 KB)
of the Replicator Log.
RLINK rlink_sec_host is behind by 0 hrs 0 mins 0 secs with respect
to Primary.
RLINK has 102 outstanding writes, occupying less than 1% (6371 KB)
of the Replicator Log.
RLINK rlink_sec_host is behind by 0 hrs 0 mins 0 secs with respect
to Primary.
4/6/2005 11:38:31 AM
RLINK has 101 outstanding writes, occupying less than 1% (6371 KB)
of the Replicator Log.
RLINK rlink_sec_host is behind by 0 hrs 0 mins 0 secs with respect
to Primary.   :   :   :   :   :   :   :
RLINK has 40 outstanding writes, occupying less than 1% (2600 KB)
of the Replicator Log.
RLINK rlink_sec_host is behind by 0 hrs 0 mins 15 secs with respect
to Primary.
RLINK is up to date. RLINK is up to date. RLINK is up to date.
4/6/2005 11:38:54 AM
RLINK is up to date.
```

# Identifying the most up-to-date Secondary

Use the `vxrlink updates` command to identify the most up-to-date Secondary in a Volume Replicator configuration. The `vxrlink updates` command can be issued only on a Secondary.

Syntax for `vxrlink updates` command

```
vxrlink [-g <diskgroup>] [-T] updates <rlink>
```

For multiple Secondaries, the `vxrlink` updates command enables you to determine the Secondary that contains the most up-to-date data and hence the most suitable replacement for the Primary in the case of a take over.

For a single Secondary, the `vxrlink updates` command can be used to determine the extent to which the Secondary is behind the Primary. You can decide whether or not to take over the Primary role by looking at the update ID of the Secondary and the number of updates by which the Primary is ahead of the Secondary.

If the Secondary is paused and is behind the Primary, the `vxrlink updates` command may show inaccurate values as long as the Replicator Log is being written to, because the status is the same as it was before the pause. However, if the Replicator Log overflows and the DCM is activated then the `vxrlink updates` command output displays the correct value by which the Secondary is behind. When the Primary switches to the DCM mode, it reconnects the Secondary RLINK and also sends updated information. This information also includes the last update sequence number on the Primary and the time that is associated with this update. Hence, the latest values are displayed.

To display the output only in terms of an update ID, use the `vxrlink updates` command without the `-T` option. The output displays the update ID as a sequence number. A sequence number is a 64-bit value that increases incrementally and hence is unique for each new update and is assigned for every new update that arrives at the Primary. The output of the `vxrlink updates` command displays the 64-bit number as two 32-bit sequence numbers that are separated by a dot. For example,

```
high_seq_num.low_seq_num
```

To display the exact time on the Primary at which the Secondary is up-to-date use the `vxrlink updates` command with the `-T` option. The `-T` option displays the exact time in hours by which the Secondary is behind.

The output of the `vxrlink -T updates` command is displayed in a three-column structure with two rows; ID and Time. The ID row displays the update IDs.

If the local time of the Primary node has been adjusted to accommodate the daylight savings or for any other reason, then the updates in the Replicator Log may still have the time stamp based on the earlier clock settings. This may appear incorrect. However, the new updates are time stamped based on the changed time settings.

The following table describes the values of the most up-to-date Secondary.

**Table 7-16** Most up-to-date secondary status

| Values | Last update<br>on Primary | Secondary<br>up-to-date as of | Secondary behind by |
|---|---|---|---|
| ID | 62010.0 | 62010.0 | 0 |
| Time | 12/24/2005 2:31:44 P.M. | 2/24/2005 2:31:44 P.M. | 0 hours 0 mins 0 secs |

The time stamp in the Time row indicates the time at which the update was written on the Primary. The first column displays the last update ID and the time at which it was written on the Primary.

The second column displays the last update ID that has been received on the Secondary and the time when it was written on the Primary. If the Secondary is up-to-date then the ID and the time in this column is the same as that in the first column. However, if the Secondary is behind, then the ID and the time is different from that in the first column.

The third column indicates the exact number of updates by which the Secondary is behind. This value is obtained as a difference between the second and first column.

**Note:** If the system time is reset to a value different from that of the current system time, then, the output of the `vxrlink -T updates` command appropriately shows a negative or an inaccurate value, until the updates that were done before resetting the system time get replicated.

## Verifying the RLINK

Use the `vxrlink verify` command to verify the configuration status for the specified RLINK or RVG. This information is useful in determining the reason why the Secondary is in the `config error` state.

**Note:** The Secondary may be in a paused state due to a configuration error. If a new configuration error is introduced when the Secondary is already in configuration error state, then the new configuration error is not reflected in the output of the `vxrlink verify` command until the Secondary is resumed.

The `vxrlink verify` command can be run either from the Primary or Secondary host. This command is displayed as:

```
    RLINK REMOTE HOST LOCAL HOST STATUS STATE
```

The information that is displayed consists of the name of the RLINK, the local host that it is connected to, and the remote host that it is connected to. STATUS displays whether the RLINK is verified (OK) or there is some configuration error (ERROR). If the STATUS is ERROR, then a detailed message describing the configuration error is displayed below this. STATE displays the RLINK state.

Syntax for `vxrlink verify` command

```
vxrlink [-g <diskgroup>] [-r <rvg>]
verify <rlink> | <rvg>
```

Example

```
vxrlink -g vvrdg -r rvg verify rlinkvxrlink -g vvrdg verify rvg
```

Providing an RVG name displays the configuration information for all the RLINKs in the RVG.

```
vxrlink -g vvrdg verify rlink_sec_host
```

When replication is active, the output resembles:

```
    RLINK REMOTE HOST LOCAL HOST STATUS STATE
rlink_sec_host sec_host pri_host ACTIVE
```

When replication is not active, the output resembles:

```
vxrlink -g vvrdg verify rlink_sec_host
RLINK REMOTE HOST LOCAL HOST STATUS STATE  rlink_sec_host
 sec_host pri_host STALE
```

## Starting the Historic Bandwidth Data Collection using the CLI

Use the `vxrlink startstats` command to start the historic bandwidth data collection for RLINKs in an RDS.

For historic bandwidth usage graphs, the user first has to start collecting the historic data for an RLINK in an RDS. To do this, you need to use the `vxrlink startstats` CLI option. After Historic Bandwidth Data Collection is started, it is possible to view the statistics through the right-click menu of a Secondary RVG node on which the collection was earlier enabled and select the View Historic Bandwidth Usage option.

**Note:** In a multinode Primary cluster, if historic data collection is enabled on an RLINK and the storage group is moved to another node, you may need to explicitly Start Historic Data Collection on the new node. The data that is collected on the old and the new node cannot be merged.

Syntax for `vxrlink startstats` command

```
vxrlink [-g <diskgroup>] startstats <rlink>
```

where `diskgroup` is the name of the dynamic disk group and `rlink`is the name of the RLINK in an RDS.

See “Starting or stopping the Historic Bandwidth Data Collection” on page 226.

## Stopping the Historic Bandwidth Data Collection using the CLI

Use the `vxrlink stopstats` command to stop the Historic Bandwidth Data Collection after bandwidth collection has been started on a secondary RVG. After starting the Historic Bandwidth Data Collection, bandwidth usage for RLINKs in an RDS is collected in the form of a graph file. The file collects the data as long as the Start Historic Bandwidth Data Collection option is enabled.

Syntax for `vxrlink stopstats` command:

```
vxrlink [-g <diskgroup>] stopstats <rlink>
```

where `diskgroup` is the name of the dynamic disk group and `rlink` is the name of the RLINK in an RDS.

See “Starting or stopping the Historic Bandwidth Data Collection” on page 226.

# Administering the RVGs using the `vxrvg` command

Use the `vxrvg` command to perform various operations on RVGs. A specific local disk group can be selected with `-g <diskgroup>` option. The `vxrvg` command has a number of keywords enabling it to perform various operations on RVG objects.

The following table lists the keywords that are available with the `vxrvg` command with their descriptions.

**Table 7-17** Keywords for `vxrvg` command

| Keyword | Description |
|---|---|
| `addlog` | Adds DCM log to the volume.<br>See “Adding DCM log” on page 276. |
| `aslog` | Associates the volume as Replicator Log Volume to the RVG.<br>See “Associating the Replicator Log volume to an RVG” on page 276. |
| `assoc` | Associates a volume as a data volume to the RVG.<br>See “Associating data volume with the RVG” on page 277. |
| `checkend` | Marks the end of the RVG checkpoint operation.<br>See “Ending checkpoint” on page 277. |
| `checkdelete` | Deletes the specified checkpoint.<br>See “Deleting the RVG checkpoint” on page 278. |
| `checkstart` | Marks the beginning of the RVG checkpoint operation.<br>See “Starting the checkpoint” on page 277. |
| `cplist` | Displays a list of RVG checkpoints.<br>See “Displaying RVG checkpoints” on page 278. |
| `dis` | Dissociates the volume from the RVG.<br>See “Dissociating volumes from RVG” on page 278. |
| `dismount` | Dismounts all the data volumes in an RVG.<br>See “Dismounting data volumes” on page 279. |
| `make` | Creates a new RVG based on the specified attributes.<br>See “Creating new RVG” on page 280. |
| `makeprimary` | Converts an existing Secondary to a Primary. This enables you to convert a Secondary RVG to a Primary using takeover (with or without failback logging) or migration.<br>See “Converting a Secondary RVG to Primary RVG” on page 280. |
| `makeSecondary` | Converts an existing Primary to a Secondary.<br>See “Converting a Primary RVG to Secondary RVG” on page 281. |
| `recover` | Recovers an RVG.<br>See “Recovering the RVG” on page 282. |

**Table 7-17** Keywords for `vxrvg` command *(continued)*

| Keyword | Description |
|---|---|
| `rm` | Deletes the specified RVG.<br>See "Removing an RVG" on page 282. |
| `resync` | Resynchronizes all the Secondary hosts that have the Replicator Log protection set to DCM and the DCM logs are activated due to Replicator Log overflow.<br>See "Resynchronizing the RVG" on page 283. |
| `set` | Sets the attributes for the specified RVG.<br>See "Setting RVG attributes" on page 283. |
| `snapshot` | Creates the snapshots for each data volume in the RVG. Each data volume must have a prepared plex associated with it. The Storage Foundation `Prepare` operation can be used to create and attach a prepared plex to the volume.<br>See "Creating snapshots for data volumes in an RVG" on page 283. |
| `snapback` | Reattaches the snapshots back to the original data volumes in an RVG.<br>See "Reattaching the snapshot volumes back to the data volumes in an RVG" on page 284. |
| `start` | Enables data access to an RVG.<br>See "Enabling data access (Starting the RVG)" on page 285. |
| `stats` | Displays the application statistics for the specified RVG. |
| `stop` | Disables data access to an RVG.<br>See "Disabling data access (stopping the RVG)" on page 286. |

The following table lists the options that are available with the `vxrvg` command:

**Table 7-18** `vxrvg` command options

| Option | Description |
|---|---|
| `-a` | This option is used with the *snapback* keyword. It is used to reattach all the snapshots of all the data volumes in an RVG at the same time. If there are some data volumes that do not have snapshot volumes, a warning message is displayed. |
| `-C <count>` | Specifies the number of times the statistics are displayed. This option must be used with the `-i` option. |

**Table 7-18** vxrvg command options *(continued)*

| Option | Description |
|---|---|
| -c <checkpoint> | This option is used with the checkstart and checkdelete keyword. The checkpoint string is associated with the checkstart and checkend marks in the Replicator Log volume.<br>See "Starting the checkpoint" on page 277. |
| -f | This option forces the specified operation to be performed and can be interpreted differently for different keywords. This option can be used with the keywords: aslog, assoc, dis, rm, set, snapshot, and snapback.<br>**Note:** This -f option can cause data corruption because the Secondary may miss the writes that may have been present on the Replicator Log but did not reach the Secondary. Because there is no way of knowing whether the Replicator Log had some pending writes that have not yet been sent to the Secondary, use this option only if you know that the Secondary is completely up-to-date. |
| -F | Enables Failback logging if you want to perform a takeover operation using the vxrvg command with the makeprimary keyword. To ensure successful failback logging, make sure that:<br>■ All the volumes of the new Primary have DCM logs<br>■ The original Primary is attached to new Primary |
| -g <diskgroup> | Specifies the local disk group for the RVG operation. |
| -i <interval> | Specifies the time interval in seconds after which the statistics are displayed. |
| -M | Allows the Secondary to become a Primary even when the original Primary host is reachable. This option is useful for planned migration.<br>To use this option effectively you are recommended to first use the vxrvg makesecondary command to convert an existing Primary to a Secondary and then use the vxrvg -M makeprimary. |
| -N | Disables Failback logging when taking over the role of the Primary. |
| -P <prefix> | This option is used with the snapshot and snapback keywords. It is used to specify a prefix for the snapshot volumes. This snapshot volume is named as follows:<br>*<prefix>-<volume name>*<br>The prefix option can be used to specify the exact volumes that need to be used to perform snapback. |

**Table 7-18** `vxrvg` command options *(continued)*

| Option | Description |
|---|---|
| `-r` | This option is used to resynchronize the original Primary with the new Primary after it becomes available again, after the takeover. |
| `-t <timestamp>` | Specifies the frequency at which the system time and date is displayed. For example, if you specify a value five then the time stamp is displayed after every five rows of information. This option must be used with the `-i` option. |
| `-z` | Resets the statistics for the specified RVG |

## Adding DCM log

Use the `vxrvg addlog` command to add the DCM log to a volume. The `vxrvg addlog` command cannot be used to add a log to a volume which is already a Replicator Log for an RVG.

By default, Volume Replicator calculates the DCM size based on the size of the volume. The default size of the DCM ranges from 1KB to 256KB depending on the size of the volume. However, you can use the `vxrvg addlog` command to set the size of the DCM to a maximum of 2 MB. You can specify the `logsize` parameter in units of megabytes (MB) or kilobytes (KB). If you do not specify any suffix such as K for KB or M for MB after the `logsize` parameter, then it is taken as KB by default.

Syntax for `vxrvg addlog` command

```
vxrvg [-g <diskgroup>] addlog <volume> [logsize=value]
```

Example

```
vxrvg -g vvrdg addlog rep_vol logsize=2M
```

## Associating the Replicator Log volume to an RVG

Use the `vxrvg aslog` command to associate the specified volume as a Replicator Log volume to the RVG.

Syntax for `vxrvg aslog` command

```
vxrvg [-g<diskgroup>] [-f] aslog <rvg><volume>
```

Example

```
vxrvg -g vvrdg aslog rvg rep_log
```

# Associating data volume with the RVG

Use the `vxrvg assoc` command to associate the specified volume as a data volume to the RVG.

Syntax for `vxrvg assoc` command:

```
vxrvg [-g<diskgroup>] [-f] assoc <rvg><volume>
```

Example:

```
vxrvg -g vvrdg assoc rvg datavol
```

# Ending checkpoint

Use the `vxrvg checkend` command to mark the Replicator Log volume that is associated with the specified RVG, to indicate the end of the checkpoint. You can use this command only after the `vxrvg checkstart` command.

Syntax for `vxrvg checkend` command

```
vxrvg [-g<diskgroup>] checkend <rvg>
```

Example

```
vxrvg -g vvrdg checkend rvg
```

# Starting the checkpoint

Use the `vxrvg checkstart` command to mark the Replicator Log volume that is associated with the specified RVG with a start mark to indicate the point from which the data needs to be replicated. Any updates to any of the data volumes subsequent to the checkstart are logged in the Replicator Log volume until you run the `vxrvg checkend` command. The `-c` option is mandatory and is used to specify a checkpoint string which is associated with the start and end marks in the Replicator Log volume. Use this command before starting the backup of the Primary data.

The `vxprint -l rvg` command displays the last checkpoint that was written to the Replicator Log. However, all the checkpoints still exist in the Replicator Log volume until the entries that are written to the Replicator Log volume wrap around or are overwritten or intentionally deleted.

The checkstart and checkend marks indicate the series of updates that the Secondary must receive for it to become consistent. The Secondary is inconsistent when it receives these updates and cannot be used for migration or takeover during this period.

Syntax for `vxrvg checkstart`

```
vxrvg [-c<checkpoint>] [-g<diskgroup>] checkstart <rvg>
```

Example

```
vxrvg -g vvrdg -c checkpoint checkstart rvg
```

## Deleting the RVG checkpoint

Use the `vxrvg checkdelete` command to delete a checkpoint that you have created. By default, the command only deletes a checkpoint that has checkended. However, you can choose to forcefully delete a checkpoint that has not ended using the `-f` option. This command can be executed only on the Primary.

Syntax for `vxrvg checkdelete` command

```
vxrvg [-g <diskgroup>] [-f] -c <checkpoint> checkdelete <rvg>
```

Example

```
vxrvg -g vvrdg -c checkpoint checkdelete rvgvxrvg
-g vvrdg -f -c checkpoint checkdelete rvg
```

## Displaying RVG checkpoints

Use the `vxrvg cplist` command to display a list of all the existing checkpoints that are associated with the specified RVG. If the Replicator Log overflows, the checkpoint is overwritten and becomes unusable. The `vxrvg cplist` command displays all the RVG checkpoints (that have been created using `vxrvg checkstart` command and `vxrvg checkend` command). This command can be run only on the Primary host.

Syntax for `vxrvg cplist` command:

```
vxrvg [-g<diskgroup>] cplist <rvg>
```

Example

```
vxrvg -g vvrdg cplist rvg
```

## Dissociating volumes from RVG

Use the `vxrvg dis` command to dissociate the specified volumes from the RVG. If the `-f` option is specified, the command forcefully dissociates the volumes even when the data access is enabled.

Syntax for `vxrvg dis` command:

```
vxrvg [-f] [-g<diskgroup>] [-r <rvg>] dis <volume>
```

Example

```
vxrvg -g vvrdg -r rvg dis volume
```

# Dismounting data volumes

Use the `vxrvg dismount` command to dismount all the data volumes in an RVG which have file systems. Dismounting is a process to ensure that the file system flushes the buffers of cached data that need to be written and disowns the volume until some application tries to open files from it. The command goes through each data volume one by one and tries to dismount it. The status for each volume is displayed in a tabular format.

**Note:** Volumes that do not have a drive letter or volumes that do not have a file system (raw volumes) are skipped.

To run `vxrvg dismount` command successfully and dismount the specified data volumes, ensure that no application or process has its file handles open on these volumes. If the file handles for some application or process are open on these volumes, you must identify them and stop any such processes.

Dismounting the volume doesn’t cause any data loss or does not limit the functionality. After the volume has been successfully dismounted, any process can still open any file on this volume, provided the volume itself is available for read or write. This command can also be used to check whether any application or process actively uses the data volumes of the RVG.

Syntax for `vxrvg dismount` command

```
vxrvg [-g <diskgroup>] dismount <rvg>
```

Output of `vxrvg dismount` command for an RVG with four data volumes that are associated to it is as follows:

```
Volume File System Status
J: NTFS Dismounted Successfully.
dv3 Skipped.(No Drive letter assigned!)
F: NTFS Dismounted Successfully.
N: RAW Skipped.(RAW Volume!)
```

## Creating new RVG

Use the vxrvg make command to create a new RVG based on the attributes specified.

Syntax for vxrvg make command

```
vxrvg -g <diskgroup> make <rvg> attribute=value
```

Example

```
vxrvg -g vvrdg make rvg datavol=dv1 srl=rep_log rlink=rlink1
\    Primary=true rds=rds
```

The following table lists attributes for vxrvg make command.

**Table 7-19** Attributes for vxrvg make command

| RVG Attributes | Description |
|---|---|
| datavol | Specifies the list of names of the data volumes to be associated to the RVG. The names of the volumes are comma-separated. |
| srl | Specifies the name of the volume to be associated as a Replicator Log to the RVG. |
| rlink | Specifies the list of names of the RLINKS separated by a comma, to be associated to the RVG. |
| Primary | Set to Primary=true for Primary RVG and Primary=false for Secondary RVG. |
| rds | RDS name to which this RVG is to be associated.<br>**Note:** By default, the RVG name is considered as the RDS name. |

## Converting a Secondary RVG to Primary RVG

Use the vxrvg makeprimary command to convert a Secondary RVG to a Primary, that is to take over the Primary role. If the RVG has multiple RLINKs, with none of them attached, then it is necessary to explicitly specify the name of the RLINK to the original Primary. If you do not want to enable failback logging use the command with the -N option.

**Note:** The vxrvg makeprimary command with the -F -r option performs the same task as the vxrds takeover command with the -autofb option. Hence, to perform a takeover operation you can use either of these commands.

The command when used with the `-M` option, enables you to migrate the Primary role even when the Primary is still available.

---

**Note:** The `vxrvg makeprimary` and `vxrvg makesecondary` commands can be used to perform planned migration to interchange the roles when the Primary and Secondary are connected. The outcome of this is similar to what the `vxrds migrate` command does. Veritas also recommends that you first use the `vxrvg makesecondary` command on the current Primary before using the `vxrvg makeprimary` command on the Secondary. Doing it in reverse order makes the volumes writable on both hosts for a short while in between, and can result in data corruption.

---

Syntax for `vxrvg makeprimary` command

```
vxrvg [-g <diskgroup>] {-F [-r]| -N | -M} makeprimary <rvg>
\[<rlink>]
```

Example

```
vxrvg -g vvrdg -F -r makeprimary rvg rlink_sechost
```

## Converting a Primary RVG to Secondary RVG

Use the `vxrvg makesecondary` command to convert a Primary RVG to Secondary. For an RVG with multiple Secondaries attached, it is necessary to specify the name of the RLINK that represents the new Primary. This command keeps the specified RLINK attached and detaches the remaining RLINKs.

However, if the Primary RVG is part of a VCS cluster and the RVGPrimary resource for this RVG exists, then Volume Replicator does not execute the `vxrvg makesecondary` command on this RVG as this can cause the resource to get into a faulted state.

---

**Note:** The `vxrvg makeprimary` and `vxrvg makesecondary` commands can be used to perform planned migration to interchange the roles even when the Primary and Secondary are connected. The outcome of this is similar to what the `vxrds migrate` command does. Veritas recommends that you first use the `vxrvg makesecondary` command on the current Primary before using the `vxrvg makeprimary` command on the Secondary. Doing it in reverse order makes the volumes writable on both hosts for a short while in between, and can result in data corruption.

---

Syntax for `vxrvg makesecondary` command

```
vxrvg [-g <diskgroup>] makeSecondary <rvg> [<rlink>]
```

## Recovering the RVG

Use the `vxrvg recover` command if the output of the `vxprint -l <rvg>` command displays the needs_recovery flag indicating that the RVG needs to be recovered. This command recovers the specified RVG, if automatic recovery does not happen.

Syntax for `vxrvg recover` command

```
vxrvg [-g<diskgroup>] recover <rvg>
```

Example

```
vxrvg -g vvrdg recover rvg
```

## Removing an RVG

Use `vxrvg rm` command to remove the specified RVG from the disk group. Before deleting, make sure that the data access to the RVG is disabled, and the Secondary is detached. You can run this command either on the Primary or Secondary host to delete either the Primary RVG or the Secondary RVG.

**Note:** If Volume Replicator is configured in a VCS or Microsoft Cluster and the cluster resource for this RVG exists, then Volume Replicator fails the delete RVG operation.

To forcefully delete the RVG even when data access is enabled for the RVG and the Secondaries are attached, use the `vxrvg rm` command with the `-f` option. However, if the RVG is part of a VCS or Microsoft Cluster and the cluster resource for this RVG exists, then Volume Replicator does not let you delete the RVG even with the `-f` option as this can cause the resource to fail.

**Note:** Volumes that are associated to the RVG are not deleted from the disk group. They are only dissociated from the RVG.

Syntax for `vxrvg rm` command

```
vxrvg [-g<diskgroup>] [-f] rm <rvg>
```

Example:

```
vxrvg -g vvrdg -f rm rvg
```

# Resynchronizing the RVG

Use the `vxrvg resync` command to replay the DCM logs that have been activated due to Replicator Log overflow. Replay occurs for all Secondary hosts on which the DCM logging has been enabled. If any of these Secondary hosts have been disconnected or paused, resynchronization is paused until the Secondary host has recovered from this condition. Detaching a disconnected or paused Secondary disables DCM logging for that Secondary, but allows resynchronization to proceed on any remaining Secondary hosts.

Syntax for `vxrvg resync` command

```
vxrvg [-g<diskgroup>] resync <rvg>
```

Example

```
vxrvg -g vvrdg resync rvg
```

# Setting RVG attributes

The `vxrvg set` command can be used to change the Primary attribute by setting it to a boolean value, `true` or `false`. Before setting this attribute, ensure that the RVG is stopped, that is, data access has been disabled. If the value of the attribute is set as `Primary=true`, then this RVG is considered the Primary RVG and writes to this RVG are replicated to any Secondary hosts with which it is associated. If the value of the attribute is set as `Primary=false`, then the RVG is considered as a Secondary RVG that receives writes from the Primary RVG.

This operation succeeds only if the Replicator Log volume is dissociated from the RVG and the RVG is in passthru mode.

Syntax for `vxrvg set` command:

```
vxrvg [-f] [-g <diskgroup>] set attribute=value....<rvg>
```

Example:

```
vxrvg -g vvrdg set Primary=false rvg
```

# Creating snapshots for data volumes in an RVG

Use the `vxrvg snapshot` command to create a snapshot for each data volume that is associated with an RVG. This command can be used on the Primary as well as the Secondary RVG.

Before creating snapshots using this command, the appropriate volumes must be prepared using the Storage Foundation Prepare operation. This operation creates mirrors (prepared plexes) for the data volumes.

For further details on preparing volumes, refer to the *Storage Foundation Administrator's Guide*.

Using the `-P <prefix>` option with the `vxrvg snapshot` command lets you specify a prefix for the name of the snapshot volumes that is created. The snapshot volume name follows the naming convention: `<prefix>-<vol-name>`.

**Note:** The snapshot volume name can consist of a maximum of 18 characters including the prefix and the dash (-).

Prefixes are useful if you have multiple snapshots for the same volume and you need to reattach specific snapshots back to the original volumes. You can specify the appropriate prefix to identify the snapshot volume that needs to be reattached.

To enable disk group split friendly snapshot operations, the prepared plexes must satisfy the conditions.

See "Conditions for creating disk group split friendly snapshots" on page 63.

The `-f` option can be used to force the snapshot operation even if disk group split friendly snapshot operation is not possible. Although the snapshot operation with `-f` succeeds, performing a subsequent disk group split operation may fail since the snapshot that was taken using the `-f` option may conflict with replication volumes.

See "Enabling data access (Starting the RVG)" on page 285.

Syntax for `vxrvg snapshot` command

```
vxrvg [-g <diskgroup>] [-f] [-P <prefix>] snapshot <rvg>
```

Example

```
vxrvg -g vvrdg -P snap snapshot rvg
```

# Reattaching the snapshot volumes back to the data volumes in an RVG

Use the `vxrvg snapback` command to reattach the snapshots back to the data volumes under an RVG. This command can be used from the Primary as well as the Secondary RVG. You can either choose to snapback specific snapshot volumes by using the `-P` *`<prefix>`* option or you can reattach all the snapshots of all the data volumes in the RVG using the `-a` option. The `-P` and `-a` options are mutually exclusive.

**Note:** A valid license for Storage Foundation FlashSnap feature must be present on all the systems on which you want to use the snapshot operations. For more information about the FlashSnap feature refer to the *Storage Foundation Administrator's Guide*.

The `-o` option can be used if you want the original volume to be synchronized with the contents of the snapshot volume. In this case after synchronization the original volume has the contents of the snapshot volume. By default, the contents of the original volumes are retained after snapback.

The `-f` option can be used to forcefully snapback the snapshot volumes even if the original volumes are in use.

**Note:** After the snapback operation is performed the data volumes contain its original contents or the contents of the snapshot volumes depending on whether the Resync from replica option is selected.

See "Understanding Volume Replicator support for FlashSnap" on page 60.

Syntax for `vxrvg snapback` command

```
vxrvg [-g <diskgroup>] [-o resyncfromreplica] [-f]
[-P <prefix>    | -a] snapback <rvg>
```

Example

```
vxrvg -g vvrdg -a snapback rvg
```

or

```
vxrvg -g vvrdg -P snap snapback rvg
```

# Enabling data access (Starting the RVG)

Use the `vxrvg start` command to start the specified RVG. This enables write access to the associated data volumes.

Syntax for `vxrvg start` command

```
vxrvg [-g<diskgroup>] start <rvg>
```

Example

```
vxrvg -g vvrdg start rvg
```

# Generating application statistics

Use the `vxrvg stats` command to display detailed application statistics for the specified RVG.

Syntax for `vxrvg stats` command

```
vxrvg [-g <diskgroup>] [[-i <interval> [-t <timestamp>]
\   [-C <count>]] | [-z]] stats <rvg>
```

The following describes the information that is displayed in the output:

`Read/Write Conflicts`

The number of times that the application attempted to read from a data block that is currently being written to.

`Concurrency`

Displays two values:

`Maximum Concurrency`—is the maximum number of threads performing writes at any point-in-time.

`Average Concurrency`—is the average number of threads performing writes at any point-in-time.

`Write-size`

Displays two values:

`Maximum Write-size`—is the maximum write-size in 512-byte blocks occurring on any data volume in the RVG.

`Average Write-size`—is the average write-size in 512-byte blocks occurring on any data volume in the RVG.

# Disabling data access (stopping the RVG)

Use the `vxrvg stop` command to stop the specified RVG. This command disables write access to the associated data volumes. If Volume Replicator is configured in a VCS or Microsoft Cluster, and the cluster resource for this RVG exists, then Volume Replicator does not stop the specified RVG as this can cause the resource to fail.

Syntax for `vxrvg stop` command

```
vxrvg [-g<diskgroup>] stop <rvg>
```

Example

```
vxrvg -g vvrdg stop rvg
```

# Displaying information using the `vxprint` command

The `vxprint` command keyword with its various options displays the complete or partial information of the Volume Replicator objects. To display the information for a specific object specify the name of the Volume Replicator object.

The hierarchies within the record associations can be displayed in an orderly fashion so that the structure of records can be understood.

Dashes (-) are displayed in the output wherever there is no available output value. If no option is specified, the default output uses the `-h` option. Specifying other options overrides this default.

The default output format consists of single-line records, each of which includes information such as record type, name, object association, object state, length, and other fields. A header line is also written before the record information.

When no disk group is specified with the command, objects in all the disk group are displayed.

Syntax for `vxprint` command

```
vxprint [-hnqlPV] [-g <diskgroup>] [name]
```

Example

```
vxprint rvgvxprint -l rvg
```

The `vxprint -l rvg` command displays list of RVG records in a verbose format.

The following table summarizes options available with the `vxprint` command.

**Table 7-20** `vxprint` command options

| Options | Description |
|---|---|
| `-h` | Lists record hierarchies |
| `-n` | Restricts output to record names |
| `-q` | Suppresses the output field header |
| `-l` | Lists all record information in a verbose format |
| `-g` <diskgroup> | Specifies a dynamic group to print |
| `-G` <diskgroup> | List the disk groups |
| `-P` | Lists the RLINK records |

**Table 7-20** `vxprint` command options *(continued)*

| Options | Description |
|---|---|
| `-p` | List the plex records |
| `-V` | Lists the RVG records |
| `-v` | List the volume records. |
| `-d` | Lists the disk records |
| `-s` | Lists the subdisk records |
| `-A` | Displays all the disk groups |
| `-Q` | Suppresses the disk group header |
| `-E` | Lists the enclosures |

## Displaying a specific RLINK

Use the `vxprint -Pl` command to display detailed information about the status of an RLINK. This command prints one record per RLINK. The following table lists the information that is displayed in the output.

To view a specific RLINK, run the following command format:

```
vxprint -Pl [-g <diskgroup_name>]rlink_name
```

The options and related descriptions for this command is as follows:

| | |
|---|---|
| `Disk Group` | Name of the disk group. |
| `RLINK Name` | Name of the RLINK. |
| `Info` | time-out, packet_size, bandwidth_limit, latency high, and low marks. |
| `State` | Displays the state of the RLINK - ACTIVE, STALE, etc. |
| `synchronous, latencyprot, and srlprot` | The current configuration settings for the replication mode, the latency protection, and Replicator Log protection. |
| `assoc` | The name of the RVG to which the RLINK is associated. |
| `protocol` | Displays the protocol that is used for replication between the Primary and Secondary. |
| `flags` | Displays the information about the object state and replication status. |

## Interpreting RLINK flag settings

The following table lists the various flags that can appear in the flags field of the `vxprint -Pl` output.

The Primary and Secondary RLINKs communicate only when the `connected` flag is on. However, replication takes place only if the following set of flags is displayed

```
write enabled attached consistent connected
```

In all other cases, corrective action may be needed. The following table explains the flags settings available for this command:

| | |
|---|---|
| `autosync` | The RDS is in the process of Automatic Synchronization. |
| `attached` | The RLINK is attached to the RVG. |
| `cant_sync` | The RLINK is inconsistent, and this Secondary needs a complete resynchronization before it can take over or replicate. |
| `connected` | The RLINK is connected to the corresponding RLINK on the remote host and replication can take place. |
| `consistent` | The state of the data volumes on the Secondary is suitable for takeover. |
| `dcm_logging` | DCM is in use, due to either autosync, failback sync, or a Replicator Log overflow. |
| `detached` | The RLINK is STALE and not taking part in replication. |
| `disabled` | The RLINK is not attached and is not replicating. |
| `disconnected` | The two RLINKs are not connected and are not replicating. |
| `enabled` | The RLINK is attached. If the `connected` flag is displayed, replication can take place. If the `disconnected` flag is displayed, replication is not taking place. |
| `inconsistent` | The data in the Secondary volumes is not consistent and the Secondary cannot take over. |
| `needs_recovery` | State after an import or system restart. The `vxrecover` command clears this state. |
| `Primary_paused` | The Primary RLINK has been paused and the RLINKs are not replicating. |
| `resync_started` | The resynchronization of the Secondary has been started. |

resync_paused The resynchronization has been started but is not currently active because of some problem.

Secondary_config_err There is a mismatch between the configuration of the volumes on the Primary and the Secondary, either a volume is missing on the Secondary or its length is not the same as that of the corresponding volume on the Primary.

Secondary_log_err An I/O error has occurred on the Secondary Replicator Log; replication cannot continue until the Replicator Log has been dissociated and a new one associated.

Secondary_paused The Secondary RLINK has been paused and the RLINKs are not replicating.

Bunker_sync Indicates that the RVG to which the RLINK is associated can be synchronized from a Bunker host that the RLINK points to.

Bunker Indicates that the RVG to which the RLINK is associated is a Bunker RVG, or the RLINK points from a normal Primary to a Bunker Secondary.

## Displaying an individual RVG

The vxprint -Vl command displays detailed information about the status of an individual RVG. This command is useful to determine the role of the Primary or Secondary RVG and the state of the RVG as seen by the operating system.

To display an individual RVG, run vxprint -Vl

```
vxprint -Vl rvg_name
```

The following table lists the output of the vxprint -Vl command:

Disk Group Name of the disk group in which this RVG resides.

RVG Name of the RVG.

state Displays the state of the RVG, ACTIVE, or FAIL.

assoc Data volumes, Replicator Log, and RLINKs associated with the RVG.

att The RLINKs that are attached. A Primary can have multiple associated and attached RLINKs. A Secondary can have multiple associated RLINKs, but only one attached RLINK.

checkpoint If a checkpoint name appears in the output, then this is the last created RVG checkpoint that is still active.

| | |
|---|---|
| `flags` | Displays the information about the RVG state and role. |

### Interpreting RVG flag settings

The status of an RVG can be interpreted on the basis of its flag setting.

The following table lists the various flag settings that an RVG displays:

| | |
|---|---|
| `Primary/Secondary` | Indicates the role of the RVG. |
| `enabled/attached` | I/O and IOCTLs can be performed. |
| `disabled/detached` | I/O and IOCTLs cannot be performed. |
| `clustered` | Indicates that the RVG is created on a clustered disk group. |
| `Bunker` | Indicates that the RVG is a Bunker RVG. |

## Displaying an individual data volume or Replicator Log

Use the `vxprint -l` *`volume_name`* command to display information about a specific volume.

For more details on the volume-specific output fields, see *Storage Foundation Administrator's Guide*.

The output fields of special interest for Volume Replicator are shown in the following table:

| | |
|---|---|
| Volume | Displays the name of the volume |
| info | Displays the length of the volume in bytes |
| assoc | Shows the RVG to which this data volume is associated |
| DriveLetter | Displays the drive letter of the specified volume |
| DeviceName | Displays the device name |
| VSS Snapshot | Displays whether the specified volume is a snapshot volume |
| state | Displays the state of the volume. For example, if the volume is accessible for Input/Output operations, the output field displays started. |
| Type | Displays the type of the volume, for example, Mirrored Concatenated. |

# Creating snapshots using the `vxsnap` command

The `vxsnap` command can be used to create synchronized snapshots on the Primary and Secondary. These snapshots can be very useful in recovering data to a consistent data point on the Secondary if the data is corrupt and the Primary had a disaster. This section focuses on how you can use the `vxsnap` command options for creating synchronized snapshots.

For any additional information about the `vxsnap` command and the other options available with this command, see *Storage Foundation Administrator’s Guide* Chapter "Command Line Interface".

The following table lists the `vxsnap` command keywords and related descriptions.

**Table 7-21** Keywords for the `vxsnap` command

| Keywords | Description |
|---|---|
| `prepare` | Creates the snapshot mirrors of the volumes in the specified component. The component in consideration is the Exchange storage group. The snapshot mirrors remain attached to and synchronized with the original volumes.<br>**Note:** Either the prepare or snapstart keyword may be used in the CLI, however prepare is recommended. |
| `create` | Creates simultaneous snapshots of all the volumes in the specified Exchange storage group component on the Primary, with simultaneous synchronized snapshots on the Secondary providing a point-in-time and up-to-date snapshot set. This parameter must be used with the `sechosts` parameter for creating synchronized snapshots. |
| `reattach` | Reattaches and resynchronizes an existing snapshot set to the original database volumes. |

The following table lists the `vxsnap` command attributes.

**Table 7-22** Attributes for the `vxsnap` command

| Attributes | Description |
|---|---|
| `component=<ComponentName>` | Name of the component; for Exchange, this is the storage group name found in the Exchange System Manager, for example, "First Storage Group". |
| `writer=<WriterName>` | Unique ID of the VSS writer, for example, in Exchange this is, "Microsoft Exchange Writer". |

**Table 7-22** Attributes for the `vxsnap` command *(continued)*

| Attributes | Description |
|---|---|
| `source=<Volume>` | Indicates the source volume for the snapshot mirror that is specified by a drive letter, drive path (mount point), or volume name of the form "device\harddiskDMVolumes\DynamicGroup\volume1". Repeat this parameter for each volume that is associated with the specified component (for example, Exchange storage group). |
| `sechost=<sec host list>` | Specifies a comma-separated list of Secondary host names on which you want to create synchronized snapshots. |
| `harddisk=<Harddisk>` | Name of the disk where the mirror is to be created, for example, harddisk2. |
| `[/plex=<PlexName>]` | Specifies the name of the mirror or plex that is to be detached. Use this parameter if there are multiple snap plexes for which you need to create snapshots. |
| `[/DriveLetter=<DriveLetter>]` | Specifies the drive letter to be assigned to the new snapshot volume. |
| `[/DrivePath=<DrivePath>]` | Specifies the drive path to be assigned to the new snapshot volume. The drive path must reference an empty local NTFS folder, which was created beforehand. The path must include the drive letter and folder to be mounted, for example, C:\DB1VOL. |
| `[/Label=<VolLabel>]` | Volume label that can be assigned to new snapshot volume. |
| `[/Newvol=<NewVolName>]` | Specifies the name for the new snapshot volume that is to be created. If no name is specified using this option, then a snapshot with the default naming format "SnapVolume01" is created. The full device path then becomes:<br>`\Device\HarddiskDmVolumes\<DiskGroupName>\<NewVolName>` |
| `backuptype=<Backuptype>` | Specifies the type of backup, either a Full or Copy. If no option is specified then Copy is the default. Copy backup creates a copy of the database and transaction logs volumes. Full backup creates a copy of the database and transaction logs volumes, runs `Eseutil` to check for consistency. If the copy is consistent, Full backup truncates the transaction logs. |

## Preparing volumes for snapshots

The `vxsnap prepare` command creates snapshot mirrors of all the volumes in the specified storage group component. You can also specify the volumes for which you want the command to create the snapshots. The snapshot mirrors remain attached to and synchronized with the original volumes.

Syntax for `vxsnap prepare` command

```
vxsnap prepare component=<ComponentName>/writer=<WriterName>
[-b] [source=<Volume>/harddisk=<Hardisk>...]
```

Example

```
vxsnap prepare component=exchg_sg/writer="Microsoft Exchange
Writer" -b source=exchg_dv1/harddisk=disk1
```

The following table summarizes the `vxsnap prepare` command option.

**Table 7-23** Option for `vxsnap prepare` command

| Parameter | Description |
|---|---|
| `-b` | Run the process as a background process. |

## Creating Synchronized Snapshots

The `vxsnap create` command creates snapshots of all volumes in the Exchange storage group or the SQL database components on the Primary and Secondary hosts, at the same point of data consistency. You can specify a name of your choice for the `xml` file that stores the snapshot metadata. If nothing is specified, then the snapshot is named according to a default naming convention.

See "About Synchronized Snapshots" on page 65.

See "Creating synchronized snapshots using the VSS Snapshot wizard " on page 184.

**Warning:** If you have created the RVG on the Primary and Secondary using the `vxrvg` command and created the RLINKs using the `vxrlink` command, then you must ensure that the RVG, disk group names, and volume names are the same before creating the RLINK. Having different component names can cause the `snapshot` command to fail.

**Note:** Separate the source volumes and attributes with forward slashes, not spaces. Source and snapshot volume attributes are paired. You must specify the source volume if you choose to specify the snapshot volume plex, drive letter, drive path, label, or volume name.

Syntax for `vxsnap create` command

```
vxsnap -x <filename> create source=<volume>
    [/DriveLetter=<driveLetter>][/DrivePath=<drivePath>
```

```
[/Label=<volLabel>][/Newvol=<newVolName>][/Plex=<plexName>]
                                ...[   writer=<writerName>]
[component=<componentName>][backuptype=<backuptype>] [-E] [-O]
[secHosts=<Secondary hosts>]
```

The following table lists the output parameters of the `vxsnap create` command.

**Table 7-24** Output parameters for the `vxsnap` command

| Parameter | Description |
|---|---|
| `-x <Filename>` | Indicates the name to be assigned to the XML metadata file that the `vxsnap create` command creates. The file name must include the ".xml" extension. By default, the file is stored at: `C:\Documents and Settings\All Users\Application Data\Veritas\VxSnapExchangeBackup`<br>If you want to place the file in another directory, specify a full path before the file name, for example `J:\XML\Image1.xml`. |
| `-E` | Runs the `Eseutil` consistency check for the Exchange database and log files. `Eseutil` is run automatically with a full backup, but must be optionally specified for a copy backup. |
| `-o` | Allows an existing XML file of the same name to be overwritten. If `-O` is not specified the `vxsnap create` command does not overwrite an existing XML file of the same name and the operation fails. |

## About snapshot naming convention on the Secondary

The volume name by convention can have a max of 18 characters, one is an underscore (_) that leaves 17 characters. On the Secondary, the snapshots are named uniquely according to a specific naming convention so that it can be easily associated to the specific volumes that we may want to reattach later. The volume name uses the last seven characters of the original volume name and last 10 characters of the data volume name, separated by an underscore. This name is unique to every snapshot.

See “Creating synchronized snapshots using the VSS Snapshot wizard ” on page 184.

**Note:** Because the XML file name is used for creating a unique snapshot name identifier, Veritas recommends that you have a unique string in the last 10 characters of the file name.

## Reattaching the Snapshots

Use the vxsnap reattach command reattaches and resynchronizes the snapshot volumes in the snapshot set to the original volumes.

**Note:** After reattaching the snapshot, the contents of the original volume and not that of the snapshot are retained.

See "Reattaching synchronized snapshots" on page 193.

Syntax for vxsnap reattach command

```
vxsnap -x <filename> [-f] [-b] reattach [writer=<writername>]
   [secHosts=<Secondary hosts>]
```

The following table lists the options that can be used with the vxsnap reattach command.

**Table 7-25** Options used with vxsnap reattach command

| Parameter | Description |
|---|---|
| -x <Filename> | Indicates the name to be assigned to the XML metadata file that is created with the command. The file name must include the ".xml" extension. The default path to the file is in the VSSXML folder under the SFW program files directory (normally C:\Documents and Settings\All Users\Application Data\Veritas\VxSnapExchangeBackup). If you want to place the file in another directory, specify a full path before the file name, for example J:\XML\Image1.xml. |
| -b | Resynchronizes the volume in the background. A new snapshot cannot be made until the resynchronization is complete. |
| -f | Forces the reattach. Make sure that the volume is not in use by another application before using this command. Use this option with care. |

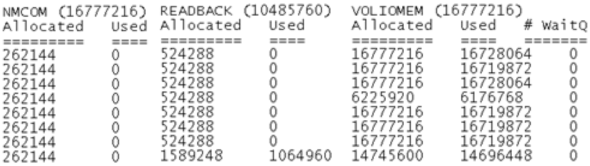
# Displaying memory statistics using the vxmemstat command

The vxmemstat command with its options displays the memory usage information for the Volume Replicator memory pools. Volume Replicator uses different memory pools during replication. The output of the vxmemstat command can be used to obtain the memory usage information that can help to diagnose memory-related problems. When the command is used without any options then the command displays the usage information for all the memory pools. You may want to use the command with the appropriate options to display the outputs at the required time intervals. This command can be used on the Primary and on the Secondary.

Syntax for `vxmemstat` command:

```
vxmemstat [-i <interval>] [-d] [-u]
```

The output for vxmemstat resembles:



```
NMCOM (16777216) READBACK (10485760)  VOLIOMEM (16777216)
Allocated   Used Allocated   Used     Allocated   Used   # WaitQ
=========   ==== =========   ====     =========   ====   =======
262144      0    524288      0        16777216    16728064    0
262144      0    524288      0        16777216    16719872    0
262144      0    524288      0        16777216    16728064    0
262144      0    524288      0        6225920     6176768     0
262144      0    524288      0        16777216    16719872    0
262144      0    524288      0        16777216    16719872    0
262144      0    524288      0        16777216    16719872    0
262144      0    1589248     1064960  14745600    14696448    0
```

See “Tuning Volume Replicator” on page 312.

The following table describes the output parameters of the `vxmemstat` command.

**Table 7-26** Output parameters of the `vxmemstat` command

| Output Parameters | Description |
|---|---|
| `Pool` | Displays the name of the memory pool and the maximum amount of memory that can be allocated to this pool. |
| `Used` | Displays the amount of memory out of the allocated memory that the consumer of the memory pool uses. |
| `Allocated` | Displays the amount of memory currently allocated to the memory pool, which ranges between the minimum and the maximum pool size. |
| `WaitQ` | Displays the number of I/Os waiting to allocate memory from the `VOLIOMEM` pool. |

The following table describes the options that can be used with the `vxmemstat` command.

**Table 7-27** `vxmemstat` command options

| Options | Description |
|---|---|
| `-u` | Displays the output record for each memory pool in a row format rather than the default tabular format. |

**Table 7-27** vxmemstat command options *(continued)*

| Options | Description |
|---|---|
| -i | Displays the statistics at the specified time intervals. |
| -d | Displays the date and time after every pageful of information. |

# Administering replicated volumes using the vxvol command

The vxvol command provides keywords for administering volumes. This section specifically describes the keywords of this command that are applicable to Volume Replicator.

For detailed information about the other keywords, refer to the *Storage Foundation Administrator's Guide Chapter* "Command Line Interface" under Section 2 Managing.

The following table lists the keywords that can be specified for vxvol command.

**Table 7-28** Keywords for the vxvol command

| Keyword | Description |
|---|---|
| assoc | Associates the specified volume to the indicated RVG as a data volume.<br>See "Associating a data volume with an RVG" on page 299. |
| aslog | Associates the specified volume to the indicated RVG as the Replicator Log.<br>See "Associating a volume to an RVG as a Replicator Log" on page 300. |
| dis | Dissociates the specified volume from the RVG.<br>See "Dissociating a volume from an RVG" on page 301. |

The following table lists the Volume Replicator-specific options that are available with the vxvol command.

**Table 7-29** vxvol command options

| Option | Description |
|---|---|
| -g <DynamicDiskGroupName> | Specifies the disk group name for the required operations. |

**Table 7-29** `vxvol` command options *(continued)*

| Option | Description |
|---|---|
| `-f force` | Forcefully dissociates the:<br>■ Data volume from the Primary RVG even when the data access is enabled.<br>■ Replicator Log volume when data access to the volume is enabled and the log may have pending updates.<br>**Note:** The `-f` option can cause data corruption because the Secondary may miss the writes that may have been present on the Replicator Log but did not reach the Secondary. Because there is no way of knowing whether the Replicator Log had some pending writes that have not yet been sent to the Secondary, use this option only if you know that the Secondary is completely up-to-date. |

# Associating a data volume with an RVG

The `vxvol assoc` command enables you to associate the specified data volume to the required RVG.

Volume Replicator does not support the following volumes for replication:

- Storage Foundation (software) RAID 5 volumes
- Volume with a Dirty Region Log (DRL)
- Volume with a comma in the name
  Using this command, you can add only one volume at one time.

Syntax for `vxvol assoc` command

```
vxvol -g <DynamicDiskGroupName> assoc <rvg>
<Volume:DriveLetter:VmName>
```

Example

```
vxvol -g vvrdg assoc rvg vol1
```

The following table describes the attributes that you can specify with `vxvol` command.

**Table 7-30** Attributes for `vxvol assoc` command

| Attributes | Description |
|---|---|
| `VolumeName` | Specifies the DeviceName of the volume. For example, `\HarddiskDmVolumes\<diskgroup>\<volume name>` |

**Table 7-30** Attributes for `vxvol assoc` command *(continued)*

| Attributes | Description |
|---|---|
| `DriveLetter` | The drive letter of the existing volume. |
| `VmName` | Specifies the internal name of the volume. You can obtain this by using the `vxvol volinfo` *`<volume>`* command. |

# Associating a volume to an RVG as a Replicator Log

The `vxvol aslog` command enables you to associate a specified volume to the required RVG as a Replicator Log. Before proceeding with adding the Replicator Log to an RVG make sure that the replication has been stopped and the data access to the RVG is disabled.

Volume Replicator does not support the following volumes for Replicator Log:

- Storage Foundation (software) RAID 5 volumes
- Volume with a Dirty Region Log (DRL)
- Volume with a comma in the name
- Volume with a DCM log

Syntax for `vxvol aslog` command

```
vxvol -g<DynamicDiskGroupName> aslog <rvg> \
<Volume:DriveLetter:VmName>
```

Example

```
vxvol -g vvrdg aslog rvg rep_log
```

The following table describes the `vxvol aslog` command attributes.

**Table 7-31** Attributes for `vxvol aslog` command

| Attributes | Description |
|---|---|
| `VolumeName` | Specifies the DeviceName of the volume. For example, `\HarddiskDmVolumes\<diskgroup>\<volume name>` |
| `DriveLetter` | The drive letter of the existing volume. |
| `VmName` | Specifies the internal name of the volume. You can obtain this by using the `vxvol volinfo` *`<volume>`* command. |

## Dissociating a volume from an RVG

The vxvol dis command enables you to dissociate the specified volume from an RVG. If the volume that you plan to dissociate is a data volume then make sure that you have disabled data access to the RVG. If the volume is a Replicator Log then ensure that the Secondary is up-to-date before you dissociate it.

You can forcefully dissociate the data or Replicator Log volume using the -f option even when data access to the data volumes is enabled. However, this operation can result in data loss. Using this command, you can dissociate only one volume at one time.

Syntax for vxvol dis command

```
vxvol -g<DynamicDiskGroupName> [-f] dis <Volume:DriveLetter:VmName>
```

Example

```
vxvol -g vvrdg dis rvg rep_log
vxvol -g vvrdg -f dis rvg volume
```

The following table describes the attributes that you can specify with the vxvol dis command.

**Table 7-32** Attributes for vxvol dis command

| Attributes | Description |
|---|---|
| VolumeName | Specifies the DeviceName of the volume. For example, \HarddiskDmVolumes\<diskgroup>\<volume name> |
| DriveLetter | The drive letter of the existing volume. |
| VmName | Specifies the internal name of the volume. You can obtain this by using the vxvol volinfo *<volume>* command. |

# Displaying and changing replication ports using the vrport command

Use the vrport command to display, change, or set the port numbers that Volume Replicator uses.

You may need to change the port numbers in the following cases:

- To resolve a port number conflict with other applications.
- To configure Volume Replicator to work in your firewall environment.

- To configure Volume Replicator to work in your firewall environment when using UDP; to specify a restricted number of ports to replicate data between the Primary and the Secondary.

The following table lists the keywords that can be used with the `vrport` command.

**Table 7-33** Keywords for `vrport` command

| Keyword | Description |
|---|---|
| `data` | Specifies the ports to be used for replicating data between the Primary and Secondary hosts. The `portlow` and `porthigh` arguments specify a range of ports to that Volume Replicator uses for replicating over TCP and UDP.<br>■ `portlow` specifies the low end port value of the range of values to be used.<br>■ `porthigh` specifies the high end port value of the range of values to be used.<br>See "Displaying or setting ports for replicating data" on page 302. |
| `heartbeat` | Displays the UDP port number that Volume Replicator uses for exchanging heartbeat messages between the Primary and Secondary.<br>See "Displaying or setting ports for heartbeats" on page 303. |
| `vradmind` | Displays the TCP port number that the `VRAS` engine uses for exchanging information between the Primary and Secondary and for performing distributed operations.<br>See "Displaying or setting ports for `vradmind`" on page 304. |
| `vxrsyncd` | Displays the TCP port number that the `vxrsync` utility uses.<br>See "Displaying or setting ports for `vxrsyncd`" on page 305. |

# Displaying or setting ports for replicating data

Use the `vrport data` command to display the ports that are used to replicate data from Primary to Secondary. This command displays both the TCP or UDP ports depending on what has been specified. To change the ports that are used to replicate data, specify the list of port numbers to use with the `vrport data` command.

Each RLINK requires one UDP port for UDP communication and a TCP+UDP port for TCP replication. Make sure that you specify an unused, reserved port number so that there is no port conflict with other applications. The number of ports that are specified must be equal to or greater than the number of RLINKs on the system.

To display the ports that are used to replicate data:

```
vrport data
```

To change the ports that are used to replicate data:

```
vrport data <portlow>-<porthigh>
```

To change the port numbers you need to specify a range of values. After you have changed the data port the new value is immediately reflected in the output of the `vrport` command. Run the `vrport data` command after changing the value to verify that the port number has changed. RLINKs must be disconnected and connected for these port numbers to get reflected. To disconnect and connect the RLINKs, use the Pause and Resume replication feature of Volume Replicator.

See “Pausing replication using Volume Replicator” on page 178.

In case of multiple TCP connection, if `tcp_src_port_restrict` tunable is set to `False`, these data ports do not work. This tunable must be set to `True` for the data port values to get reflected.

## Displaying or setting ports for heartbeats

Use the `vrport heartbeat` command to display or change the port number that Volume Replicator uses for heartbeats. Heartbeat messages use the UDP protocol.

**Note:** When changing the port numbers, you must change it on all the hosts that are part of the RDS.

To display the port number thatVolume Replicator uses for heartbeats, use the command

```
vrport heartbeat
```

To change the port number for heartbeats, use the command

```
vrport heartbeat port
```

**Note:** After changing the port number, the command displays a message asking you to restart the system. The changes take effect after the restart.

**To change the replication heartbeat port on a host from 4145 to 5000**

1. Use the `vrport` command to change the heartbeat port to 5000 on the required host.

   ```
   vrport heartbeat 5000
   ```

2. The changes are displayed immediately by the `vrport heartbeat` command, however, you must restart the system on which you have changed the heartbeat port for the changes to take effect:

   Follow the above steps to change the heartbeat port on Secondary host.

## Displaying or setting ports for `vradmind`

The `vrport vradmind` command enables you to display or change the port numbers depending on whether you use it with the port parameter. For `vradmind` this command sets only the TCP port, as the `vradmind` uses the TCP port for replicating between the Primary and Secondary.

To display the current TCP port number that `vradmind` uses, use the command

```
vrport vradmind
```

To change the port number that `vradmind` uses for communication between the Primary and Secondary, use the command

```
vrport vradmind port
```

**Note:** After changing the `vradmind` port restart the Veritas Storage Agent Service (`vxsvc`).

**To change the current TCP port number used by `vradmind` from the default value 4545 to 4646:**

1 Use the `vrport` command to change the `vradmind` port to 4646 on the required host.

```
vrport vradmind 4646
```

2 Restart the `vxsvc` service using the command

```
net stop vxsvc
fnet start vxsvc
```

3 Run the `vrport vradmin` command. The command displays the new port value for `vradmind`.

Make sure that you perform these steps on the corresponding Secondary host to ensure that both hosts use the same port.

## Displaying or setting ports for `vxrsyncd`

The `vrport vxrsyncd` command enables you to display or change the port numbers depending on whether you use it with the port parameter. For `vxrsyncd`, this command sets the default TCP port that the `vxrsync` server uses.

To display the default TCP port number that `vxrsyncd` uses, use the command

```
vrport vxrsyncd
```

To change the default TCP port number that `vxrsyncd` uses for replicating between the Primary and Secondary, use the command

```
vrport vxrsyncd port
```

**To change the current TCP port number used by `vxrsyncd` from 4545 to 4646:**

1 Use the `vrport` command to change the `vxrsyncd` port to 4646 on the required host.

```
vrport vxrsyncd 4646
```

2 The changes are displayed immediately by the `vrport vxrsyncd` command, however, you must restart the system on which you have changed the heartbeat port for the changes to take effect.

# Administering the RVG using the vxedit

The vxedit command associates a comment with the specified SFW objects. These include the volume, plex, subdisk, disk media, and disk group. You can also set properties for the Volume Replicator objects using this command.

The vxedit command also provides keywords for editing the comments that are associated with the volumes. This section specifically describes the keywords that are applicable to Volume Replicator.

For detailed information about all "the keywords refer to the *Storage Foundation Administrator's* Guide Chapter "Command Line Interface under Section 2 Managing.

The following table describes the keywords that can be set for the vxedit command.

**Table 7-34** Keywords for vxedit command

| Keyword | Description |
|---|---|
| rm | Deletes the specified Volume Replicator object; RVG or RLINK. |
| set | Sets the replication attributes on the Secondary and Primary. |

The following table describes the options that can be used with the vxedit command.

**Table 7-35** vxedit command options

| Keyword | Description |
|---|---|
| -V | Indicates that the vxedit command needs to perform the specified operation for the RVG. |
| -P | Indicates that the vxedit command needs to perform the specified operation for an RLINK. |
| -f | Forcefully removes the specified Volume Replicator object; RVG or RLINK.<br>The delete operation is performed even if the RLINK is attached and associated to an RVG or the data access is enabled for the RVG. Some operations may be disallowed even with this flag.<br>**Note:** This -f option can cause data corruption because the Secondary may miss the writes that may have been present on the Replicator Log but did not reach the Secondary. Because there is no way of knowing whether the Replicator Log had some pending writes that have not yet been sent to the Secondary, use this option only if you know that the Secondary is completely up-to-date. |

**Table 7-35** `vxedit` command options *(continued)*

| Keyword | Description |
|---|---|
| `-r` | Performs the specified operations recursively on the objects that are associated with the selected object. For example, when used with the `rm` keyword, for an RVG, all the associated objects such as the data volumes, RLINKs and Replicator Log are also removed. |

## Deleting the Volume Replicator objects

The `vxedit rm` command deletes the specified RVG or RLINK. The command when used with the `-f` option ensures that the RVG is deleted even when the data access is enabled or if the RLINK is attached and associated to an RVG. The `-r` option performs the delete operation recursively, that is, for all the data volumes, Replicator Log volume, and the associated RLINKs.

**Note:** If Volume Replicator is configured in a VCS or Microsoft Cluster and the cluster resource for this RVG exists, then Volume Replicator fails the `vxedit rm` operation.

Syntax for `vxedit rm` command

```
vxedit [-g <DynamicDiskGroupName>] [-fr] rm <rvg> | <rlink>
```

## Setting the attributes

Use the `vxedit set` command to set the attributes for the local RLINK, RVG, and the SFW objects. The attribute names specify the field that needs to be set within the specified RLINK or RDS.

The attributes that the `vxedit set` command sets for the RLINK are similar to the attributes that the `vxrlink set` command sets.

Syntax for `vxedit set` command

```
vxedit [-PV] [-g<DynamicDiskGroupName>] set attribute=value<Object>
```

See “Setting the RLINK attributes” on page 262.

The following table lists the attributes for the `vxedit set` command.

**Table 7-36** Attributes for `vxedit set` command

| Attribute | Description |
|---|---|
| `comment` | Specifies a comment that is displayed against the SFW objects such as a volume, plex, subdisk, disk media, and disk group. These comments are useful if you want to display some additional information for these objects. The comment size cannot exceed 40 bytes. |
| `Primary` | Specifies a boolean value `true` or `false`.<br>If set to `true`, then the RVG is considered the Primary RVG and writes to this RVG are replicated to any RLINK with which it is associated and attached. If set to `false` (default), then the RVG is a Secondary RVG and receives writes from the Primary RVG.<br>**Note:** Before setting this attribute, ensure that the RVG is stopped, that is, data access has been disabled. |
| `synchronous` | Specifies the mode of replication. |
| `Rsrlprot` | Enables or disables log protection. |
| `latencyprot` | Enables or disables latency protection. |
| `latency_high_mark` | Specifies the maximum number of outstanding requests that are allowed when latency protection is enabled. |
| `latency_low_mark` | Specifies a value such that when the writes are stalled, the number of outstanding requests must drop to this value before latency protection can be disabled. |
| `local_host` | Specifies the name or IP address of the local host. |
| `remote_host` | Specifies the name or IP address of the remote host. |
| `packet_size` | Specifies the size of packets in which data can be sent through the network during replication. |
| `bandwidth_limit` | Specifies a value that can be used to control the bandwidth that Volume Replicator needs to use for replication. If this attribute is not specified, then by default, Volume Replicator uses the entire available bandwidth for replication. To disable bandwidth throttling, set this attribute to `none`. Note that the specified bandwidth value must be at least 1 `Mbps` (Megabits per second). You can specify the value in units of `Kbps`, `Mbps`, `Gbps`, or `bps`. The default is Kbps. If no value is specified then bandwidth throttling is disabled. |

**Table 7-36** Attributes for `vxedit set` command *(continued)*

| Attribute | Description |
|---|---|
| `protocol` | Specifies the protocol to be used for replication between the Primary and Secondary. Specify TCP or UDP.<br>If the setup includes a Bunker Secondary and replication is over IP, the protocol can be set to UDP or TCP. The default is UDP.<br>If the storage at the Bunker Secondary is directly accessible from the Primary, for example, DAS or NAS, use the STORAGE protocol, otherwise use TCP/IP or UDP/IP<br>**Note:** If the replication protocol for the Bunker Secondary has been set to STORAGE then you can change it only by deleting and recreating the Bunker with the required protocol; UDP/IP or TCP/IP. You cannot change the STORAGE protocol using the Change Replication Settings option. |
| `remote_rlink` | Specifies the name of the remote RLINK. |
| `remote_dg` | Specifies the disk group name of the remote RLINK. |

# Administering the RVG using the `vxassist` command

The `vxassist` command along with its keywords enables you to create volumes and perform volume-related operations. This section specifically describes the `vxassist` keywords that are applicable to Volume Replicator.

For detailed information about all the keywords refer to the *Storage Foundation Administrator's* Guide Chapter "Command Line Interface".

The following table describes Volume Replicator-specific keywords for the `vxassist` command.

**Table 7-37** Keywords for `vxassist` command

| Keyword | Description |
|---|---|
| `addlog` | Adds a DCM log to the volume. |
| `remove` | Removes a volume, a mirror, or a log. |
| `growby` | Grows the volumes by the specified value. |

# Adding a DCM log

The `vxassist addlog` command with its parameters enables you to add a log (DRL or DCM) or DCO to a volume. For Volume Replicator purposes, the DCM is the only log that we need to add. The DCM log is used for fast resynchronization of a Secondary RVG with the Primary RVG when the Replicator Log overflows. It is also used for failback logging in case of a takeover with fast-failback.

Syntax for `vxassist addlog` command

```
vxassist [-g<DynamicDiskGroupName>] \
addlog <VolumeName|DriveLetter|VmName|DrivePath> \
[LogType=<DRL | DCM | DCO>] [nlog=<#>] [<diskname | p#c#t#l#>...]
```

The following table describes the attributes that you can specify with the `vxassist addlog` command.

**Table 7-38** Attributes for `vxassist addlog` command

| Attributes | Description |
|---|---|
| `VolumeName` | Specifies the DeviceName of the volume. For example, `\HarddiskDmVolumes\<diskgroup>\<volume name>` |
| `DriveLetter` | The drive letter of the volume. |
| `VmName` | Specifies the internal name for the volume, which you see when you use the `vxvol volinfo` *`<volume>`* command. |
| `DrivePath` | Specifies the drive path to volumes that are NTFS mounted. |
| `Logtype` | Specifies the type of log you want to add. This includes:<br>■ DCM<br>Adds a Data Change Map log.<br>■ DRL<br>Adds a Dirty Region Log to volumes. This is the default log type for mirrored volumes.<br>■ DCO<br>Adds a Data Change Object. This is used to implement Persistent FastResync. |
| `nlog <>` | Specifies the number of logs that need to be created for the specified volume. |
| `Diskname` | Indicates the designated hard disk, which can be specified by the device name (such as Harddisk2) or the internal disk name (such as Disk2). |

## Growing the volumes

Use the `vxassist growby` command to grow the size of the specified data volume or Replicator Log volume. Note that if you grow the size of the data volume using this command it is not applicable across the RDS but specific to the RVG only. Use the `length` parameter to specify the size you want to grow the volume by. This command does not require you to stop replication before growing the volumes, however, if replication is active, pause replication to all the Secondaries before growing the Primary and Secondary volumes. When you grow the size of the data volumes using the `vxassist growby` command, Veritas recommends that you do it for the selected volume on each host in the RDS. Not doing this can cause the replication to pause with a configuration error due to mismatch in volume sizes.

Syntax for `vxassist growby` command

```
vxassist [-b] [-g<DynamicDiskGroupName>] growby \
<VolumeName|DriveLetter|VmName|DrivePath> <length> [<diskname
|p#c#t#l#> ...]
```

## Removing a DCM log

The `vxassist remove` command removes (deletes) a volume or the DCM log from an existing volume. When you specify removing a volume, the command works the same as `vxassist delete`.

Syntax for `vxassist remove` command

```
vxassist [-g<DynamicDiskGroupName>] remove <volume|mirror|log>
<VolumeName|DriveLetter|VmName|DrivePath>
[LogType=<DRL|DCM|DCO>] [nlog=<#>] [plex=<PlexName>
```

The following table describes the attributes that you can specify with the `vxassist remove` command.

**Table 7-39** Attributes for `vxassist` command

| Attributes | Description |
|---|---|
| `volume\|mirror\|log` | Specifies whether a volume, mirror, or log needs to be removed. |
| `VolumeName` | Specifies the DeviceName of the volume. For example, `\HarddiskDmVolumes\<diskgroup>\<volume name>` |
| `DriveLetter` | Specifies the drive letter of an existing volume. |
| `VmName` | Specifies the internal name for the volume, which you see when you use the `vxvol volinfo` *`<volume>`* command. |

**Table 7-39** Attributes for `vxassist` command *(continued)*

| Attributes | Description |
|---|---|
| `DrivePath` | Specifies the complete drive path to volumes that are NTFS mounted. |
| `Logtype` | Specifies the type of log you want to remove. This includes:<br>■ DCM<br>Data Change Map log for the volumes that are part of an RVG. This is the default for replicated volumes.<br>■ DRL<br>Dirty Region Log to volumes. This is the default log type for mirrored volumes.<br>■ DCO<br>Data Change Object. This is used to implement Persistent FastResync. |
| `nlog <>` | Specifies the number of logs that need to be removed from the specified volume. |
| `Plex=<Plexname>` | Specifies the mirror or plex that needs to be removed. |

# Tuning Volume Replicator

Volume Replicator provides the `vxtune` command that enables you to tune memory tunables to best suit your environment. This command is especially useful if you want to experiment with different values to arrive at an optimum value that suits your requirements.

See “About Volume Replicator memory monitoring and control support” on page 72.

Syntax for `vxtune` command

```
vxtune [-r] [ <tunable> [<value>] ]
```

The following table describes the parameters that you can specify with the `vxtune` command.

**Table 7-40** Parameters for vxtune command

| Parameter | Description |
|---|---|
| `tunable` | Specifies the tunable name whose value you want to display or change. |
| `value` | Specifies the value that you want to set for the tunable. |

**Note:** The iopath_logging tunable should be enabled only after consulting the Support team. If enabled without caution, it can adversely affect the performance of Volume Replicator.

The following table describes various Volume Replicator memory tunables.

**Table 7-41** Volume Replicator tunables

| Tunable | Value |
|---|---|
| `NMCOM_POOL_SIZE`<br>`(vol_max_nmpool_sz)` | Specifies the maximum memory that Volume Replicator uses on a Secondary, to hold the write requests coming from the Primary. The default value for this tunable is 131072 K, however you can specify a value from 4096K to 524288K. However, if you have assigned a lower value than the specified one, then you may need to restart the system for the changed values to get into effect. |
| `READBACK_POOL_SIZE`<br>`(vol_max_rdback_sz)` | Specifies the maximum memory that Volume Replicatoruses, when write requests are read back from the Replicator Log. The default value for this tunable is 262144 K. However, if you have assigned a lower value than the specified one, then you may need to restart the system for the changed values to get into effect |
| `BASE_MEMORY`<br>`(vol_min_lowmem_sz)` | Specifies the minimum threshold of available Volume Replicator memory that is needed to keep the write requests in memory on the Primary RVG before sending it to Secondary. The default value for this tunable is 1024K, however you can specify a value from 512K to 10240K. However, if you have assigned a lower value than the specified one, then you may need to restart the system for the changed values to get into effect. |
| `MAX_MEMORY`<br>`(vol_rvio_maxpool_sz)` | Specifies the maximum memory that is requested from the system by Volume Replicator for its use. The default value for this tunable is 32768K, however you can specify a value from 4096K to 1048576K. However, if you have assigned a lower value than the specified one, then you may need to restart the system for the changed values to get into effect. |

**Table 7-41** Volume Replicator tunables *(continued)*

| Tunable | Value |
|---|---|
| MAX_TCP_COUNT<br>(max_tcp_conn_count) | Specifies the maximum number of TCP connections per RLINK. The default value for maximum TCP connections per RLINK is 64. This value is used as upper bound while calculating the number of connections required per RLINK. When the value is changed, the following message is displayed:<br>`Command executed successfully. Note: The changed value will get reflected only in the next connect cycle with the Secondary. To force reconnect, please pause and resume replication.` |
| NMCOM_MAX_MESSAGES<br>(nmcom_max_msgs) | Specifies the number of outstanding messages waiting to be processed on the Secondary array. It takes the value from 128 K to 2048 K. The default value is 2048 K. It is recommended that the values for this tunable should be changed only after consulting the support team. |
| MAX_RECEIVE_GAP<br>(max_rcvgap) | Specifies the tolerable gap between the expected message ID and the actual out of order received message ID on the Secondary. It takes the value from 5 to 25, however, the default value is 5. Values for this tunable should be altered only after consulting the support team. |
| RLINK_READBACK_LIMIT<br>(rlink_rdbklimit) | Specifies the upper limit that is allocated for per RLINK readback memory. You can specify a value from 4096K to 65536K. The default value, however, is 262144K. Values for this tunable should be changed with the assistance of support team |

**Table 7-41** Volume Replicator tunables *(continued)*

| Tunable | Value |
|---|---|
| `COMPRESSION_SPEED`<br>`(compression_speed)` | Specifies the current speed limit of the compression that Volume Replicator performs. The default value for this tunable is 7. You can specify a value ranging from 1 to 9.<br>Examples:<br>■ To display value for this tunable, run the following command: `vxtune compression_speed`<br>■ To set the value to 5: `vxtune compression_speed 5`<br>The compression speed is inversely proportionate to the compression that Volume Replicator performs. If `compression_speed` value is smaller, then the amount of data that gets compressed is larger. If the value is bigger, then the amount of compressed data is smaller. |
| `COMPRESSION_THREAD`<br>`(compression_threads)` | Specifies the number of threads that are dedicated for compression and decompression of data. It can take values in the range 1 to 63. The default value is 10. Its value can be displayed or set through the `vxtune` command.<br>Examples:<br>■ To display value for this tunable, run the following command: `vxtune compression_threads`<br>■ To set the value to 5: `vxtune compression_threads 5`<br>The maximum value to which `compression_threads` value can be set depends on the number of CPUs the system has. If CPU usage is very high during compression, then you can change the `compression_threads` value to lower the CPU usage. However, setting the tunable value to a very low value can considerably increase the data compression time. |

**Table 7-41** Volume Replicator tunables *(continued)*

| Tunable | Value |
|---|---|
| `COMPRESSION_WINDOW`<br>`(compression_window)` | Specifies the data window size in kilobytes (KB) for compression. The default value for this tunable is 0, which means a window of unlimited size. If `compression_window` tunable is set to a default value of 0, then almost all of the data that is sent to the DR site is sent in a compressed state when the RLINK is set with `COMPRESSION_ENABLED` flag. Data compression can sometimes cause high amount of CPU or memory consumption. The `compression_window` tunable can be set to reduce the resource usage. If `compression_window` size is set to a smaller value, then the amount of data that gets compressed on the primary is less and the remaining data is sent in an uncompressed form to the secondary. However, if this tunable is set with a high value, then large amount of data is sent to the secondary in a compressed form.<br>Examples:<br>■ To display the value of this tunable through `vxtune` command:<br>`vxtune compression_window`<br>■ To set the value of `compression_window` to 256 KB:<br>`vxtune compression_window 256` |

**Table 7-41** Volume Replicator tunables *(continued)*

| Tunable | Value |
|---|---|
| `COMPRESSION_THRESHOLD`<br>`(compression_threshold)` | Specifies the CPU usage threshold after which Volume Replicator starts reducing the compression_thread to reduce the CPU utilization and, if required, disable Volume Replicator compression. This value is node-specific and is applicable on both Primary and Secondary.<br>You can specify a value from 0 to 100. The default value is 0. You can disable Adaptive Compression by specifying this value as 0.<br>When the RLINK is set with `COMPRESSION_ENABLED` flag, data compression can sometimes cause high amount of CPU consumption. The `compression_threshold` tunable can be set to reduce the CPU usage. If `compression_threshold` size is set to a value other than zero (for example, 30) and the CPU consumption goes beyond the mentioned threshold, then Volume Replicator starts decreasing the rate of compression on the Primary or decompression on the Secondary depending on the node on which the value is set.<br>Examples:<br>■ To display the value of this tunable through the `vxtune` command:<br>`vxtune compression_threshold`<br>■ To set the value of `compression_threshold` to 50:<br>`vxtune compression_threshold 50` |
| `FORCE_MAX_CONNECTION`<br>`(force_max_conn)` | Specifies whether to force Volume Replicator to use the maximum number of TCP connections. It takes the value of either True or False. If this tunable is set to True, then the automatic way of estimating the number of connections that are required for RLINK based on latency of network can be overridden and the value specified for `max_tcp_conn_count` can be used instead |
| `NETWORK_LOSS_TOLERANC`<br>`(rp_incr_decr)` | Specifies the degree of network loss-tolerance during replication. Increasing the value of `rp_incr_decr increases` the degree of network error tolerance. The default value of rp_incr_decr is 8, which works fine for good networks. However, it takes value between 1 to 100. This tunable should be changed only after consulting the support team. |

**Table 7-41** Volume Replicator tunables *(continued)*

| Tunable | Value |
|---|---|
| TCP_SOURCE_RESTRICT<br>(tcp_src_port_restrict) | Specifies whether to restrict the TCP source port usage of Volume Replicator data connections. This tunable takes the value of either True or False. The value is shown as True when data port range is used for source ports. These range of ports should be opened as source ports in firewall. To get multi-connection working across a firewall or NAT environment, you are required to open port 4145 for both UDP and TCP. |
| IOPATH_LOGGING<br>(iopath_logging) | Specifies whether to enable IO path logging and takes the value of either True or False. By setting iopath_logging to True, extra log messages get logged to the log files which help in debugging certain issues in I/O. However, the extra log messages can affect the I/O performance adversely. |
| NAT_SUPPORT<br>(nat_support) | This tunable specifies the presence of Network Address Translation (NAT) in the network. Use the vxtune command to see whether NAT is enabled or not. It takes the value of either True or False. When NAT support is enabled, the value is shown as True and when disabled it is shown as False.<br>See "Displaying the tunable values " on page 319.<br>See "Tuning the Volume Replicator memory parameters" on page 373. |
| HB_TIMEOUT<br>(hb_timeout) | Specifies the number of heartbeat messages that can be missed before the RLINK gets disconnected. The default value is 15 and it can take values from 1 to 60. It should only be tuned for lossy networks where the Volume Replicator frequently disconnects the RLINKs because it doesn't receive heartbeat messages. |
| TCP_ROUND_ROBIN<br>(tcp_round_robin) | Specifies whether Volume Replicator should use round-robin method of sending data over multiple TCP connections. The value can be either True or False. |

**Table 7-41** Volume Replicator tunables *(continued)*

| Tunable | Value |
|---|---|
| `DHCP_IP_Support`<br>`(dhcp_ip_support)` | Determines whether or not DHCP IPs can be used while configuring Volume Replicator.<br>The value can either be `True` or `False`.<br>This tunable must be enabled if you plan to set up replication in an Azure environment. DHCP IP addressess are assigned to the virtual machines that are created in an Azure environment. To identify the DHCP IP addresses, DHCP_IP tunable must be set to True. |

# Displaying the tunable values

Use the `vxtune` command without any parameters to display the values that are currently assigned to the Volume Replicator tunables. Use the `-r` option with any of the command parameters to display the value in bytes. By default, the tunable value is displayed in kilobytes.

Syntax for `vxtune` command:

To display the default values for all the tunables:

```
vxtune
```

To display the default value for a specific tunable:

```
vxtune <tunable>
```

The output for `vxtune` command resembles the following:

```
C:\Documents and Settings\administrator.INDSSMG>vxtune
vol_max_nmpool_sz = 16384 kilobytes
vol_max_rdback_sz = 8192 kilobytes
vol_min_lowmem_sz = 1024 kilobytes
vol_rvio_maxpool_sz = 32768 kilobytes
compression_window = 0 kilobytes
max_tcp_conn_count = 64
nmcom_max_msgs = 512
max_rcvgap = 5
rlink_rdbklimit = 16384 kilobytes
compression_speed = 7
compression_threads = 10
msgq_sequence = 1
vol_maxkiocount = 1048576
```

```
force_max_conn = False
tcp_src_port_restrict = False
nat_support = False
```

## Setting the tunable values

Use the `vxtune tunable` command with the `value` argument to set the tunable to a new value. You can specify the value in Bytes (B), kilobytes (KB), megabytes (MB), or gigabytes (GB).

After modifying the tunable, the new tunable values are displayed immediately. However for some tunables like `NMCOM_POOL_SIZE (vol_max_nmpool_sz)`, `READBACK_POOL_SIZE (vol_max_rdback_sz)`,`BASE_MEMORY (vol_min_lowmem_sz)`, and `MAX_MEMORY (vol_rvio_maxpool_sz)` you need to restart the system if you have assigned lower values than the specified one for these tunables. The changed value is in effect after the system has been restarted.

Syntax for `vxtune tunable <value>` command

```
vxtune [ <tunable> [<value>] ]
```

See “Tuning Volume Replicator” on page 312.

# Examples: Using the command line

This section details examples for some Volume Replicator tasks. The following configuration is used throughout all the examples.

## Sample setup using the command line

Primary host name: VVRPRI

The sample configuration is as follows:

| | |
|---|---|
| `vvrdg` | Disk Group |
| `rvg` | Primary RVG |
| `rlk_vvrsec_vvr_rvg` | RLINK to Secondary `VVRSEC` |
| `rlk_vvrbnkrsec_vvr_rvg` | RLINK to Bunker Secondary `VVRBunkerSEC` |
| `host ip` | 10.212.80.251 |
| `datavol` | Primary data volume #1 |
| `exchg_datavol` | Primary Exchange database volume |

| | |
|---|---|
| `rep_log_vol` | Primary Replicator Log volume |

Bunker Secondary: VVRBunkerSEC

The sample configuration is as follows:

| | |
|---|---|
| `vvrdg` | Disk Group |
| `rvg` | Bunker RVG |
| `rlk_vvrpri_vvr_rvg` | RLINK to Primary `VVRPRI` |
| `rlk_vvrsec_vvr_rvg` | RLINK to Secondary `VVRSEC` |
| `host ip` | 10.212.82.251 |
| `rep_log_vol` | Primary Replicator Log volume |

Secondary host name: VVRSEC

The sample configuration is as follows:

| | |
|---|---|
| `vvrdg` | Disk Group |
| `rvg` | Secondary RVG |
| `rlk_vvrpri_vvr_rvg` | Secondary RLINK to Primary `london` |
| `rlk_vvrbnkrsec_vvr_rvg` | Secondary RLINK to Bunker node `london`. This links gets activated only when the Primary fails |
| `host ip` | 10.256.88.126 |
| `datavol` | Secondary data volume #1 |
| `exchg_datavol` | Primary Exchange database volume |
| `rep_log_vol` | Secondary Replicator Log volume |

## Example 1: Setting up replication using the command line interface

This examples details the procedure to set up replication between the hosts `VVRPRI` as the Primary and `VVRSEC` as the Secondary host.

Both the systems have the same disk group already created on them called `vvrdg`. This disk group has two volumes `datavol` and `rep_log_vol`.

Perform the following steps on the Primary system VVRPRI

## Creating the RLINK on the Primary system VVRPRI

The RLINK can be created by running the command:

```
vxrlink -g vvrdg make rlk_vvrsec synchronous=off \local_host=VVRPRI
remote_host=VVRSEC remote_dg=vvrdg \remote_rlink=rlk_vvrpri srlprot=off
latencyprot=off protocol=TCP
```

## Creating the Primary RVG on the Primary system VVRPRI

The Primary RVG can be created by running the following command:

```
vxrvg -g vvrdg make rvg datavol=datavol srl=rep_log_vol
\rlink=rlk_vvrsec Primary=true rds=rds
```

Repeat the same on the Secondary system VVRSEC

## Creating the RLINK on the Secondary system VVRSEC

To create RLINK on the Secondary system VVRSEC, run the following command:

```
vxrlink -g vvrdg make rlk_vvrpri synchronous=off
\local_host=VVRSEC remote_host=VVRPRI remote_dg=vvrdg \
remote_rlink=rlk_vvrsec srlprot=off latencyprot=off protocol=TCP
```

## Creating the Secondary RVG on Secondary system VVRSEC

To create the Secondary RVG on Secondary system VVRSEC, run the following command:

```
vxrvg -g vvrdg make rvg datavol=datavol srl=rep_log_vol
\rlink=rlk_vvrpri Primary=false rds=rds
```

## Attaching the RLINKs and starting replication on the Secondary

Attach the RLINK by running the following command:

```
vxrlink -a -g vvrdg -r rvg att rlk_vvrpri
```

## Attaching the RLINKs and starting replication on the Primary

Attach the RLINK using the following command:

```
vxrlink -g vvrdg -a -r rvg att rlk_vvrsec
```

After executing all the above mentioned steps on both the Primary and Secondary systems, they are now ready for replication.

**Note:** For UDP mode, the `vxprint -lPV` command shows the packet size and not the number of TCP connections.

Run the `vxprint -lPV` command on the Primary system.

Following is the output of the command if replication is done in the TCP/IP mode:

```
Diskgroup = vvrdg
Rvg         : rvg
state       : state=ACTIVE kernel=ENABLED
assoc       : datavols=\Device\HarddiskDmVolumes\vvrdg\datavol
              srl=\Device\HarddiskDmVolumes\vvrdg\rep_log_vol
              rlinks=rlk_vvrsec
att         : rlinks=rlk_vvrsec
checkpoint :
flags       : Primary enabled attached
Rlink       : rlk_vvrsec
info        : timeout=500 connections=11
latency_high_mark=10000 latency_low_mark=9950
              bandwidth_limit=none
state       : state=ACTIVE
              synchronous=off latencyprot=off srlprot=off
assoc       : rvg=rvg             remote_host=VVRSEC
              remote_dg=vvrdg             remote_rlink=rlk_vvrpri
              local_host=VVRPRI
protocol    : TCP/IP
flags       : write attached consistent connected
```

## Example 2: Setting up Bunker replication

You can choose to add the Bunker node either after the Primary RVG has been created or after the RDS with the Primary and Secondary RVG is created. You can add a Bunker to an existing RDS without interrupting replication from the Primary to the Secondary.

### To create and add the Bunker RVG to an RDS with Primary and Secondary RVG

1. On the Bunker node `VVRBunker`, create a new disk group, `vvrdg`, containing the volume that is intended to be used as the Replicator Log.
2. On the Primary node VVRPRI, create and add the Bunker using the command:

   ```
   vxrds -g vvrdg addBunker vvrrvg VVRPRI VVRBunker
   ```

   where, `vvrrvg` is the RVG name; `VVRPRI` is the Primary; `VVRBunker` is the Bunker node.

   This command creates RLINKs between the Bunker and the Primary, and also between the Bunker and each Secondary in the RDS.

### To create and add the Bunker RVG to an RDS that consists of only the Primary RVG

1. On the Bunker node `VVRBunker`, create a new disk group, `vvrdg`, containing the volume, `rep_log_vol` intended to be used as the Replicator Log.
2. On the Primary node VVRPRI, create and add the Bunker using the command:

   ```
   vxrds -g vvrdg addBunker vvrrvg VVRPRI VVRBunker
   ```

   where, `vvrrvg` is the RVG name; `VVRPRI` is the Primary name; `VVRBunker` is the Bunker node name.
3. Create a Secondary RVG with the same name as the Primary RVG and add it to the RDS.

   ```
   vxrds -g vvrdg addsec vvrrvg VVRPRI VVRSEC
   ```

4. Attach the Secondary RLINKs and start replication on the Primary using the command:

   ```
   vxrlink -g vvrdg -a startrep vvr_rvg rlk_vvrsec
   ```

   To add multiple Bunker hosts refer to the following sections:

   See “To create and add the Bunker RVG to an RDS with Primary and Secondary RVG” on page 324.

   See “To create and add the Bunker RVG to an RDS that consists of only the Primary RVG” on page 324.

## Example 3: Using Bunker node for disaster recovery

If the Primary site VVRPRI fails, update the Secondary from the Bunker.

See “Example 2: Setting up Bunker replication” on page 323.

After the Secondary is up-to-date, the Secondary can take over the role of Primary.

See “Updating the Secondary from the Bunker” on page 325.

When the Primary recovers, failback to the original Primary.

See “Transferring the Primary role” on page 326.

See “Restoring the original Primary in a Bunker setup” on page 327.

## Updating the Secondary from the Bunker

When disaster strikes and the Primary host becomes unavailable, update the Secondary from the Bunker using the following steps.

**Note:** If the Primary Replicator Log has overflowed for a Secondary, or if the Secondary is inconsistent because it is resynchronizing, you cannot use the corresponding Bunker Replicator Log to recover the Secondary. Because the Bunker node does not have data volumes, it cannot use DCM to track overflows.

**Note:** As the Bunker Replicator Log does not store Primary checkpoints, it does not support attaching the Secondary from a checkpoint.

### To update the Secondary from the Bunker

1. Activate the Bunker by using the following command from the Bunker host:

   ```
   vxrds -g vvrdg activatebunker vvrrvg
   ```

   This converts the Bunker RVG to a Primary, that is from receiving mode (Secondary) to replicating mode (Primary).

   The `activatebunker` command needs to be run only once, even if you update multiple Secondaries.

2. Start replication to the Secondary from the Bunker host.

   ```
   vxrds -g vvrdg -b startrep vvrrvg VVRSEC
   ```

   This command switches the RLINK on the Secondary that points to the original Primary to point to the Bunker node which is now the Primary and begins replaying the Bunker Replicator Log.

   If you have more than one Secondary that uses this Bunker, repeat the `vxrds startrep` command for each Secondary.

3. Monitor the status of the replication from Bunker to Secondary using the Monitor view.

   See “About monitoring replication” on page 114.

4. When the replay is complete, verify that the Secondary is up-to-date using the `vxrlink status` command.

5. Stop replication to the Secondary. You can also stop the replication before the replay is finished, for example, if the Primary is restored or depending on your RTO.

   ```
   vxrds -g vvrdg stoprep vvr_rvg seattle
   ```

   You can also choose not to replay the Bunker Replicator Log, after a disaster at the Primary, if you want zero RTO. However, in this case the pending updates that were present on the Bunker Replicator Log are lost.

6. After using the Bunker for replay, if it is no longer needed for any more replays, the Bunker should be deactivated. Deactivate the Bunker only after all the replays from the Bunker have been stopped.

   To deactivate the Bunker, issue the following command from the Bunker node:

   ```
   vxrds -g vvrdg deactivatebunker vvr_rvg
   ```

   The command needs to be run only once.

7. The Secondary is now ready for take over.

## Transferring the Primary role

For zero RPO you must ensure that the Secondary is up-to-date and then perform the takeover:

```
vxrds -g vvrdg -autofb takeover <local_rvg>
```

Use this command to enable the Secondary host to take over the Primary role with fast-failback. When the automatic failback feature is enabled using the `-autofb` option, the original Primary automatically becomes the Secondary, after it is available again.

After takeover the Secondary RVG is converted to a Primary RVG. However, the original Primary must become available again for the fast-failback to work successfully.

See “Taking over the Primary role using the fast-failback option” on page 201.

**Note:** If minimal or zero RTO is important for your requirements, then you can stop the replay after your required RTO time. If the Primary becomes available you can immediately start replication from the Primary.

## Restoring the original Primary in a Bunker setup

In most cases, when the original Primary recovers after a failure, you will restore the RDS to the original configuration.

See "Migrating the Primary role back to the original Primary" on page 327.

See "Restoring the Bunker setup after failback to original Primary" on page 328.

## Migrating the Primary role back to the original Primary

In a Bunker setup, how you restore the Primary role to the original Primary depends on the status of the Bunker replay.

## Recovering the original Primary during Bunker replay

If the original Primary recovers when the Bunker replay is still in progress, the original Secondary has not yet taken over the Primary role. You can therefore restore replication from the original Primary without completing the replay and the Secondary does not need to takeover the Primary role.

**To restore the original Primary during Bunker replay**

1 Stop the replication from the Bunker to the Secondary

```
vxrds -g vvrdg stoprep vvr_rvg seattle
```

2 Deactivate the Bunker by running the following command

```
vxrds -g vvrdg deactivatebunker vvr_rvg
```

Replication from the Primary to the Secondary resumes from that point in the Replicator Log which indicates the last write received by the Secondary. For example, suppose the Bunker Replicator Log contained 10 GB when the Primary failed. After 7GB of the writes were replayed to the Secondary, the Primary recovered. The Primary only needs to synchronize the 3GB of pending data.

After the original Primary has recovered, restart replication to the Bunker. After the original Primary again becomes the Primary, you must re-establish the RLINK between the Bunker and the Primary.

See "Restoring the Bunker setup after failback to original Primary" on page 328.

### Failing back to the original Primary

After the original Secondary has taken over the Primary role, and the original Primary has become available again, resynchronize the original Primary with the writes on the new Primary. After the resynchronization completes, failback the Primary role to the original Primary.

**Note:** You can use the Bunker node only as a Bunker to the original Primary and not the new Primary as the Bunker node needs to be physically closer to the Primary.

After the failback has completed, and the Primary role has been restored to the original Primary, you must restart replication to the Bunker. The Primary RLINK to the Bunker host is detached when the original Primary becomes the Secondary of the new Primary as part of the failback process. Therefore, after migrating the Primary role back to the original Primary, you must reestablish the RLINK between the Bunker and the Primary.

See “Restoring the Bunker setup after failback to original Primary” on page 328.

### Restoring the Bunker setup after failback to original Primary

After the original Primary is restored and the failback is complete, restore the Bunker setup so that the original Primary can again start replicating to the Bunker Secondary.

**To restore the Bunker setup**

1 Deactivate the Bunker, if it is not already deactivated.

```
vxrds -g vvrdg deactivatebunker vvr_rvg
```

2 Restart replication from the Primary host to the Bunker host.

```
vxrds -g vvrdg startrep vvrrvg VVRBunker
```

## Example 4: Using synchronized snapshots to restore data

This example provides steps to restore the data on the Secondary volumes using the synchronized snapshots.

## Sample setup showing how to restore data using synchronized snapshots

The Primary and Secondary sites have InfoScale Storage or InfoScale Enterprise installed on the nodes. Exchange has been installed and configured on all the nodes and the required Exchange database and mailboxes have been moved under the RVG volumes

Primary host names: VVRPRI1 and VVRPRI2

The sample configuration is as follows:

| | |
|---|---|
| `vvrdg` | Disk Group |
| `rvg` | Primary RVG |
| `rlk_vvrsec_vvr_rvg` | RLINK to Secondary `VVRSEC` |
| `host ip` | 10.212.80.251 |
| `exchg_datavol1 (E:)` | Primary Exchange database volume |
| `exchg_datavol2 (F:)` | Primary Exchange mailbox volume |
| `exchg_datavol3 (G:)` | Primary Exchange mailbox volume |
| `rep_log_vol` | Primary Replicator Log volume |

Secondary host name: VVRSEC1 and VVRSEC2

The sample configuration is as follows:

| | |
|---|---|
| `vvrdg` | Disk Group |
| `rvg` | Secondary RVG |
| `rlk_vvrpri_vvr_rvg` | Secondary RLINK to Primary `london` |
| `host ip` | 10.256.88.126 |
| `exchg_datavol1` | Secondary Exchange database volume |
| `exchg_datavol2` | Secondary Exchange mailbox volume |
| `exchg_datavol3` | Secondary Exchange mailbox volume |
| `rep_log_vol` | Secondary Replicator Log volume |

## Configuration Details

This example consists of a complete disaster recovery configuration that has been set up using the instructions that are provided in the *Storage Foundation and High Availability Solutions HA and Disaster Recovery Solutions Guide for Microsoft Exchange*.

After completing the configuration a user *test_user* is created in the domain and a mailbox is associated with the user. The user mailbox has some mails in the user *test_user* mailbox. After this is done, replication is started.

The following steps display the procedure to recover the Secondary Exchange database and mailboxes using the synchronized snapshots, when a disaster has occurred at the Primary, and the Secondary volumes with the Exchange database and mailbox data has got corrupt.

## Preparing the Volumes

You must prepare the volumes before you plan to create snapshots. The prepare operation creates mirrored plexes that are attached to the source volumes. Note that the snapshots must be created before the disaster to be able to recover the data from the snapshots.

Create snapshot mirrors of the volumes in the Exchange storage group component by using the following command from the Primary and Secondary:

```
vxsnap prepare component=exchg_storage_group/writer="
Microsoft Exchange Writer" -b source=exchg_datavol/harddisk=disk1
source=exchg_datavol2/harddisk=disk2source=exchg_datavo3/harddisk=disk3
```

This command creates the snapshot mirrors that are still attached to the original volume.

## Creating snapshot volumes

To create snapshot volumes, you need to perform the following.

**To create the snapshot volumes**

1 Create synchronized snapshots by using the following command from the Primary host:

```
vxsnap -x snapshot1data.xml create
source=exchg_datavol1/DriveLetter=P/Newvol=exchg_snap_datavol1
source=exchg_datavol2/DriveLetter=Q/Newvol=exchg_snap_datavol2
source=exchg_datavol3/DriveLetter=R/Newvol=exchg_snap_datavol3
writer="Microsoft Exchange Writer"component=exchg_storage_group
backuptype=Full -Osechosts=VVRSEC1
```

When specifying the name of the Secondary host that is part of a cluster, specify the host on which the resources are online.

This command quiesces the application on the Primary, then creates a snapshot of name `exchg_snap_datavol`. It then runs the `vxibc regsend` command before thawing the application. After the Secondary receives the IBC message the Secondary RVG snapshots are created.

Based on the default naming convention the snapshot names are:

`pshot1data_exchg_datavol1`,

`pshot1data_exchg_datavol2` and

`pshot1data_exchg_datavol3`

2 Verify that the RLINK is up-to-date using the `vxrlink status` command.

## Using the snapshots to recover the corrupted volumes

Consider that all the mails in the *test_user* mailbox have got accidentally deleted or there has been a virus attack on the Primary and the corrupted data has got replicated to the Secondary. At this time a disaster occurs on the Primary leaving neither the Primary or Secondary data in good shape. Following are details of the steps that can be used to recover the data using the synchronized snapshots on the Secondary.

Because the Primary is no longer available you must perform a take over on the Secondary by running the command:

```
vxrds -g vvrdg -autofb takeover rvg
```

Use this command to enable the Secondary host to take over the Primary role with fast-failback. When the automatic failback feature is enabled using the `-autofb` option, the original Primary automatically becomes the Secondary, after it is available again. After takeover the Secondary RVG is converted to a Primary RVG.

**To restore the data using the snapshot volumes**

1. To restore the data on the Secondary data volumes from the snapshot volumes run the following command:

   ```
   vxassist -g vvrdg -o resynchfromreplica
   snapbackpshot1data_exchg_datavol1

   vxassist -g vvrdg -o resynchfromreplica
    snapbackpshot1data_exchg_datavol1

   vxassist -g vvrdg -o resynchfromreplica
    snapbackpshot1data_exchg_datavol1
   ```

2. Mount the volumes with the same drive letter as on the Primary from the VEA. Select **File System > Change Drive Letter and Path** from the volume right-click menu. Assign the volumes with the same drive letter as the volumes on the Primary had before the disaster.

   or

   Through the command line, execute the following command to mount the volumes:

   ```
   vxassist -g vvrdg assign exchg_datavol1:E
   vxassist -g vvrdg assign exchg_datavol2:F
   vxassist -g vvrdg assign exchg_datavol3:F
   ```

3. Bring all the resources online on the new Primary.

## Restoring the original Primary

In most cases, when the original Primary recovers after a failure, you will restore the RDS to the original configuration.

## Migrating the Primary role back to the original Primary (failing back to the original Primary)

Because the takeover command was specified with the `-autofb` option the resynchronization of the Primary from the new Primary and the failback are done automatically after the Primary becomes available again.

If you had not specified the `-autofb` option, then after the original Primary becomes available again, resynchronize the original Primary with the new Primary. After the resynchronization completes, failback the Primary role to the original Primary.

After the failback has completed, and the Primary role has been restored to the original Primary, you must restart replication to the Secondary. The data on the Primary and Secondary is at the same consistent point as was in the snapshot.

Chapter 8

# Configuring Volume Replicator in a VCS environment

This chapter includes the following topics:

## About configuring Volume Replicator in a VCS environment

This chapter discusses the procedure to configure Volume Replicator in a VCS environment. Cluster Server (VCS) connects or clusters, multiple, independent, systems into a management framework for increased availability. Each system, or node, runs its own operating system and cooperates at the software level to form a cluster. VCS links commodity hardware with intelligent software to provide application failover and control. When a node or a monitored application fails, other nodes can take predefined steps to take over and bring up services elsewhere in the cluster, thus providing an image of a single system to the client.

Volume Replicator can enable applications to be highly available at geographically-separated sites, by configuring them in a VCS cluster. A VCS cluster is configured at a Primary site as well as the Secondary site. In this case, the servers where the application is running can be referred to as the source or Primary cluster and the servers to which the data is replicated, can be referred to as the destination or Secondary cluster. The failover between these sites is enabled with the help of agents that are explained in this chapter. The VCS Agent for Volume Replicator is installed and configured on each VCS node to enable Volume Replicator RVGs to failover between nodes, in a VCS cluster. Different service groups are created to represent the application and replication-related resources. A dependency is set between the application and replication service groups.

Local clustering provides local failover for each site or building. Campus and replicated data cluster configurations offer some degree of protection against disasters affecting limited geographic regions. But, these configurations do not provide protection against large-scale disasters such as major floods, hurricanes, and earthquakes that cause outages for an entire city or region. Such an outage can affect the entire cluster. In such situations, you can ensure data availability by migrating applications to remote clusters located considerable distances apart using global clustering.

In such a global cluster, if an application or a system fails, the application is migrated to another system within the same cluster. If the entire cluster fails, the application is migrated to a system in another cluster. This is known as a wide-area failover. If the configuration consists of a Bunker setup, and the entire Primary site fails, then during failover to another system global clustering ensures that the new system is synchronized with pending updates from the Bunker node.

Clustering on a global level requires replicating application data to the remote site. Thus, Volume Replicator along with VCS can be used to provide an effective disaster recovery solution.

# Components of a VCS cluster

Resources, attributes, and service groups are components integral to cluster functionality. This section provides a brief overview on each of these components.

For more information, see the *Cluster Server Administrator's Guide*.

The following topics describe the components of a VCS cluster are as follows:

- See “Resources” on page 336.
- See “Attributes” on page 336.
- See “Service groups” on page 336.

# Resources

Resources are hardware or software entities such as disks, volumes, file system mount points, network interface cards (NICs), IP addresses, applications, and databases. Resources work together to provide a service to clients in a client or server environment.

Resource types are defined in the `types.cf` file by a collection of attributes. The VCS configuration file, `main.cf`, contains the values for the attributes of the resources.

# Attributes

Attributes contain data regarding the cluster, nodes, service groups, resources, resource types, and agents. A specified value for a given attribute configures the resource to function in a specific way. By modifying the value of an attribute of a resource, you change the way the VCS agent manages the resource. Each attribute has a definition and a value. You define an attribute by specifying its data type and dimension. Attributes also have default values that are assigned when a value is not specified.

# Service groups

A service group is a logical grouping of dependent resources. It is a management unit that controls resource sets. When a service group is brought online, all the resources within the group are brought online.

For setting up Volume Replicator in a VCS environment, you require two service groups as the following:

- Application Service Group
- Replication Service Group

## Application Service group

An application service group is a group that comprises the resources that the application requires.

## Replication Service group

A replication service group can be defined as a group that is comprised of certain types of resources. The replication service group can have one or more of the types of resources that are explained below, however, it cannot include any additional components. This service group cannot be a part of the application service group, as the replication service group represents the replicated volumes and other resources that must be available and online on the Primary or Secondary nodes at the same time, unlike the application service group.

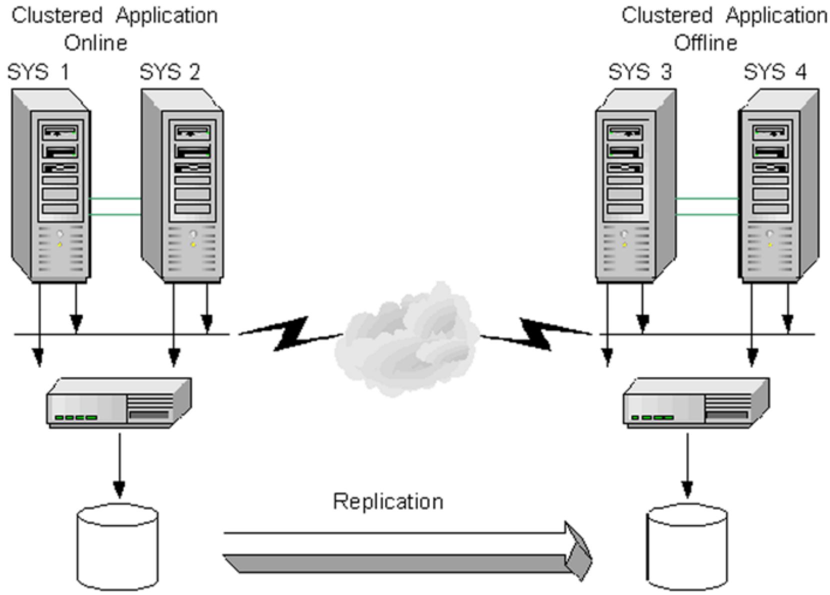
Types of resources in a replication service group are as follows:

- IP
- NIC
- VMDg
- VvrRvg

# Illustrating a highly available Volume Replicator setup

The illustration below shows a configuration where application data is replicated from a Primary site to the Secondary site. This provides disaster recovery; in the event that the Primary site is destroyed, application data is immediately available at the Secondary site, and the application can be restarted at the Secondary site.

**Figure 8-1** Typical Volume Replicator Disaster Recovery setup



On each site, a VCS cluster provides high availability to both the application and the replication. A VCS cluster is configured at the Primary site, and another cluster at the Secondary site. If a single clustered node fails at either site, any online group of resources can failover to other nodes in the VCS cluster. The `VvrRvg` agent is installed and configured on each VCS node to enable the Volume Replicator RVGs to failover between nodes in a VCS cluster. Note that while a replication group is online on both sites to handle both sides of the replication (source and destination), the clustered application is online only on the Primary site. The application data on the Secondary site is accessible after a takeover or migrate operation.

The `RVGPrimary` agent can be configured on all the nodes. This agent can also be used to automate the process of takeover in case of a failure of the Primary cluster. In a setup with multiple Secondary hosts where the RLINKs between the Secondaries are already created, this agent also automates the process of adding the orphan Secondaries back into the RDS after failover and also synchronizes these Secondaries with the new Primary.

To modify the configuration of the resources managed by an agent you can use use the following:

- VCS Cluster Manager GUI (Java console)
- VCS command line interface (CLI)

## List of agents for Volume Replicator

Agents for Volume Replicator includes VCS-type declarations and agent executables, which represent a resource type.

The VCS Agent for Volume Replicator is installed automatically when InfoScale Enterprise is installed.

See "How the agents work" on page 338.

The Volume Replicator Agent Configuration Wizard for VCS helps in configuring and managing of replication service groups with the help of these agents.

The Volume Replicator agents include the following:

- VvrRvg Agent
- RVGPrimary Agent

# How the agents work

This section explains how each VvrRvg and RVGPrimary agents work, their functions (entry points), attributes and dependency graphs for agents.

This can be summarized as follows:

- How each agent works

- Agent functions (entry points)
- State definitions and attributes for each agent
- Dependency graphs for each agent.

See "VvrRvg agent" on page 339.

See "RVGPrimary agent" on page 343.

# VvrRvg agent

The VvrRvg agent represents the RVG and enables failover of the RVG between nodes in a cluster, thus making it highly available. The VvrRvg agent is installed and configured separately on the Primary and Secondary cluster. The agent enables replication between clusters by managing the Volume Replicator Primary node in one cluster and the Volume Replicator Secondary node in another cluster, each of which can be failed over in its respective cluster. In this way, the replication role between the Primary and Secondary is made highly available.

If your configuration uses a storage Bunker, then the action that the VvrRvg agent takes during failover depends on the replication status. If a storage Bunker is initially associated with SYS1 of the Primary cluster, then whenever the replication service group fails over to the other node SYS2 in the cluster, the storage Bunker disk group is also moved to the other node.

The following table provides some conditions that govern the failover to the other node.

**Table 8-1** Conditions that govern the failover to other node

| Mode of replication | Replication Status (RLINK STATE) | Action that the VvrRvg agent performs during online operation |
|---|---|---|
| synchronous | STATE=Active | During failover, the Bunker disk group gets deported on the first node and is automatically imported on the second node before the RVG resource is brought online. |
| synchronous override | STATE=Active | The RVG resource comes online.<br>The Bunker disk group then gets imported asynchronously on the new node in a separate thread. |

## VvrRvg agent-specific functions, state definitions, and attributes

The following table provides information about Agent Functions, State Definitions, and the Attribute definitions for the VvrRvg agent.

The following table lists the VvrRvg agent-specific functions, definitions, and attributes.

**Table 8-2** VvrRvg agent functions, state definitions, and attributes.

| Description | Agent Functions (Entry Points) | State Definitions |
|---|---|---|
| Brings the RVG online, monitors read or write access to the RVG, and takes the RVG offline; this is a failover resource that enables Volume Replicator to failover between nodes in a cluster | ■ `online`—Enables data access to the RVG data volumes.<br>■ `offline`—Disables the data access to the volumes in the RVG.<br>■ `monitor`—Monitors the state of the RVG.<br>The VvrRvg agent monitors an RVG for local data access only; it does not monitor replication. | ■ `ONLINE`—Indicates that the data access is enabled to all the volumes in the RVG.<br>■ `OFFLINE`—Indicates that the RVG is not in active state and data access is disabled.<br>■ `UNKNOWN`—Indicates that the state of the RVG cannot be determined. |

The following table summarizes the required agent attributes for an RVG resource.

Review the following information to familiarize yourself with the required agent attributes for an RVG resource type. This information assists you during the agent configuration.

**Table 8-3** Required agent attributes for the RVG resource

| Attribute | Type and Dimension | Definition |
|---|---|---|
| `RVG` | string-scalar | The name of the Replicated Volume Group (RVG) that is managed. |
| `VMDgResName` | string-scalar | The name of the Storage Foundation cluster disk group resource containing the RVG. |
| `IPResName` | string-scalar | The name of the IP resource managing the IP address that the RVG uses for replication. |

## Sample replicated Service group definition

The following is a sample of the replicated service group definition as specified in the main.cf file file. You can directly edit the contents of this file to create a replicated service group for your setup.

```
group RepSrvcGrp    SystemList = { SYS1 = 1, SYS2 = 2 }
   IP RepSrvcGrp_ip    Address = "10.216.136.226"
   SubNetMask = "255.255.248.0"
   MACAddress @SYS1 = "00-0B-DB-90-B9-07"   NIC RepSrvcGrp_nic
   MACAddress @SYS1 = "00-0B-DB-90-B9-07"   VMDg AppDg
   DiskGroupName = cdg1   DGGuid = bff57c70-666a-4cbe-bdb7-75090be7c0b0
   VvrRvg RepSrvcGrp-VvrRvg    RVG = RVG   VMDgResName = AppDg
   IPResName = RepSrvcGrp_ip   RepSrvcGrp-VvrRvg requires RepSrvcGrp_ip
   RepSrvcGrp-VvrRvg requires AppDg
   RepSrvcGrp-AppSrvcGrp_ip requires RepSrvcGrp_nic
```

## Dependency graph

This section describes a configuration with typical service groups configured to monitor the state of Volume Replicator in a VCS cluster. A typical configuration includes an application service group (parent) and a replication service group (child).

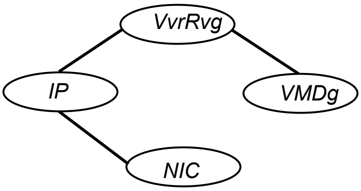
## Resource dependencies within a replication Service group

The resources in a service group must come online and go offline in a particular order; the dependencies represent this order. In the following dependency graphs, resources must come online starting at the "bottom" and moving up the dependency lines.

In the sample configuration of a replication service group that is shown in the dependency graph below, the shared disk group is configured using the SFW (VMDg) resource type. The service group IP address for Volume Replicator (which manages the replication IP) is configured using the IP resource type, which in turn has a dependency on a NIC resource. The VvrRvg resource represents the RVG and monitors the RVG failover within a cluster.

The replication service group comes online after each of these resources is brought online.

**Figure 8-2** Resource dependencies within a replication service group



## Service group dependencies

The dependency graph below depicts a typical replication configuration with two groups, a sample application service group (SQL Server 2000) and a replication service group. The application service group is created before creating the replication service group and has the VMDg resource on which the RVG resides.

**Note:** No Lanman resource is created for the replication service group. The IP resource must be different, that is, the IP attribute must have different values for the application and replication service group although they can share the same NIC resource.

The Volume Replicator agent configuration wizard for VCS performs certain tasks automatically when creating the replication service group.

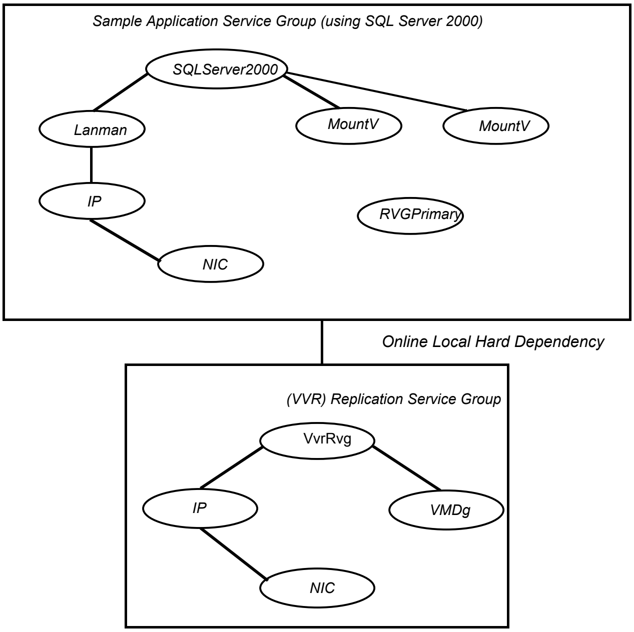
The tasks are follows:

- Creates the replication service group during which the `VMDg` resource is automatically moved from the application service group into the replication service group.
- Creates the RVGPrimary resource in the application service group if the option to create the RVGPrimary is selected.
- Establishes an online local hard service group dependency between the application service group and the replication service group, after the service groups are configured.

  This online local hard dependency indicates that the replication service group (child) must first come online, before the application service group (parent) comes online. Conversely, the application service group must go offline first before the replication service group goes offline.

  If the replication service group faults, the parent application service group is taken offline before the child replication service group.

**Figure 8-3** Service group dependencies


Sample Application Service Group (using SQL Server 2000)
SQLServer2000
Lanman
MountV
MountV
IP
RVGPrimary
NIC
Online Local Hard Dependency
(VVR) Replication Service Group
VvrRvg
IP
VMDg
NIC

# RVGPrimary agent

To make the application highly available across clusters, the RVGPrimary agent enables the migrate or takeover operation for Volume Replicator.

If the RVGPrimary resource is online, it indicates that the corresponding RVG is a Primary. However, if the RVG is a Secondary and the RVGPrimary resource is made online, then depending on the state of replication, the RVGPrimary agent performs a Migrate or Takeover. Thus, the agent monitors the role of the RVG and ensures that the RVG is Primary as long as the resource is online.

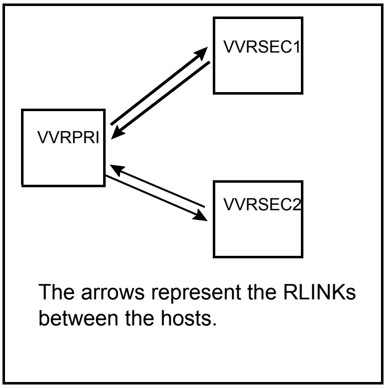
Migrate or takeover operation depends on the following:

- The state of the RVG
- The status of replication
- The state of the cluster

## Typical multiple Secondary setup

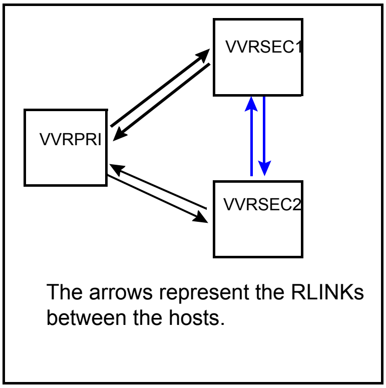
The figure below illustrates a multiple Secondary setup, after the RDS has been created using the Setup Replicated Data Set Wizard.

**Figure 8-4** Typical multiple Secondary setup



The following figure illustrates a setup after additional RLINKs have been created between the Secondaries. This is required to enable `RVGPrimary` to add the orphaned Secondaries back into the RDS, after failover.

**Figure 8-5** Typical multiple Secondary setup with RLINKs between the Secondaries



## How the agent works in a multiple Secondary setup

In the first example, if the Primary VVRPRI crashes and you perform a takeover on vvrsec1, then after takeover one of the Secondaries becomes a Primary and the other Secondary is orphaned. The same thing holds true for a Migrate operation. This orphaned Secondary needs to be manually added to the RDS and synchronized with the new Primary.

In the second example, if the RVGPrimary agent is brought online on one of the Secondaries, it ensures that the additional Secondaries are added to the new

Primary. The RVGPrimary agent also creates the RLINKs between every pair of Secondary hosts. Each Secondary must have an RLINK pointing to the Primary and an RLINK to every other Secondary. After failover, the RVGPrimary agent detects additional RLINKs present on the Secondary. On each such Secondary, the RVGPrimary agent detaches the RLINK pointing to the original Primary and start the vxrsync server.

The RVGPrimary agent then checkstarts the new Primary RVG and enables difference-based synchronization, to synchronize the additional Secondaries with the new Primary. The RVGPrimary resource does not wait for the synchronization to complete. After starting the process the RVGPrimary resource comes online. The applications dependent on the RVGPrimary resource can also come online while the additional Secondaries are synchronized asynchronously through the spawned process.

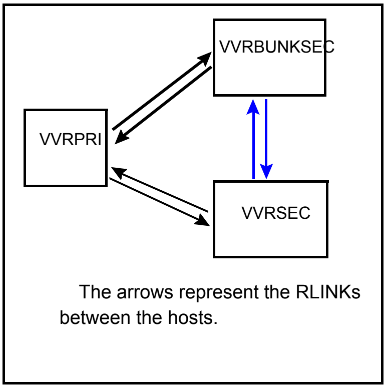
The spawned process performs difference-based synchronization, checkend and then attaches the additional Secondaries with checkpoint.

**Note:** After migration or takeover is performed successfully, RVGPrimary does not fault even if any of the intermediary steps to include the additional Secondaries to the RDS fails. The failure is logged in the VCS engine log.

## How the agent works in a Bunker set up

Under normal operating conditions the VVRPRI site replicates data to the Bunker Secondary in the synchronous override mode to ensure that it is up-to-date.

**Figure 8-6** Agent functioning in a Bunker set up



The arrows represent the RLINKs between the hosts.

If a disaster occurs at the Primary cluster site VVRPRI, the RVGPrimary agent on VVRSEC activates the Bunker node and starts replay from the Bunker Replicator

Log to VVRSEC. During this replay the Bunker node is converted to a Primary and the data in its Replicator Log is used to bring the Secondary up-to-date. When the replay completes or the time-out limit that is specified in the BunkerSyncTimeout has elapsed, the Secondary takes over the Primary role and the Bunker node is deactivated.

For a storage Bunker configuration, if a disaster occurs at the Primary, then the RVGPrimary agent comes online on the Secondary node VVRSEC, and first imports the disk group on the Bunker node. Then the agent activates the Bunker node to start replay to the Secondary.

When the original Primary becomes available again, you may want to migrate the Primary role back to the original site. If you had performed takeover with auto failback then failback logging is enabled when takeover is performed. If the original Primary becomes available again it is automatically converted to a Secondary and the writes from the new Primary are written to the original Primary to bring it up-to-date. The RVGPrimary agent then starts replication to the original Secondary using difference-based synchronization and to the Bunker using Automatic Synchronization to bring it up-to-date with the original Primary.

In a multiple Bunker configuration, if the most up-to-date Bunker fails, then the RVGPrimary agent activates each of the other Bunkers and try to replay data from them to the Secondary, one after the other.

## RVGPrimary agent-specific functions, state definitions, and attributes

This section provides information about RVGPrimary agent and state definitions.

The following table provides the RVGPrimary agent-specific information with agent and state definitions.

**Table 8-4** RVG Primary agent-specific information

| Description | Agent Functions(Entry Points) | State Definitions |
|---|---|---|
| Enables taking over of the Primary role by the Secondary if the Primary becomes unavailable.<br>Enables the migration of the Primary role to the Secondary. | ■ online<br>Depending on network availability either migrate or takeover is performed to convert the Secondary to a Primary.<br>■ offline<br>Takes the resource offline.<br>■ monitor<br>Monitors the role of the RVG based on whether it is the Primary or Secondary.<br>■ fbsync<br>Resynchronizes the original Primary with the new Primary that has taken over with fast-failback, after the original Primary had become unavailable. This is an action entry point and is available from the Actions dialog box, which appears when you click the RVGPrimary resource and select **Actions** from the menu that appears. | ■ ONLINE<br>Indicates that the RVG managed by the resource is Primary.<br>■ OFFLINE<br>Indicates that the RVG managed by the resource is not Primary. |

Review the following information to become familiar with the agent attributes required for an RVGPrimary resource type. This information assists you during the agent configuration.

The following table describes the agent attributes for the RVGPrimary resource.

**Table 8-5** Agent attributes for RVGPrimary resource type

| Attribute | Type and Dimension | Definition |
|---|---|---|
| RvgResourceName | string-scalar | The name of the `VvrRvg` resource in the replication group on which the application group depends. |

**Table 8-5** Agent attributes for RVGPrimary resource type *(continued)*

| Attribute | Type and Dimension | Definition |
|---|---|---|
| AutoTakeover | int | If set to 1, the agent automatically enables the Secondary to take over the Primary role when it detects that the Primary has become unavailable.<br>If set to 0, no automatic takeover is performed. In that case you must manually perform the takeover operation on the Secondary |
| AutoResync | int | If set to 1, the agent automatically performs a resynchronization operation to synchronize the failed Primary with the new Primary when it becomes available after a takeover operation with fast-failback.<br>If set to 0, manually resynchronize the original Primary with the new Primary, after it becomes available again. |

**Table 8-5** Agent attributes for RVGPrimary resource type *(continued)*

| Attribute | Type and Dimension | Definition |
|---|---|---|
| BunkerSyncTimeout | int | If set to `Null` (no value), the `RVGPrimary` agent considers this as infinite time-out value. It replays all the writes on the Bunker Replicator Log to the Secondary and only after all the writes are sent the takeover is performed on the Secondary.<br>If set to 0 indicating a zero RTO, the `RVGPrimary` agent immediately performs a take over on the Secondary and no pending writes from the Bunker are sent to the Secondary.<br>If the value is set to a specific integer, <*T*> seconds, then the `RVGPrimary` agent makes sure that writes for <*T*> seconds are sent to the Secondary before performing a takeover on the Secondary. Thus, the RTO in this case is equal to <*T*> seconds. |

The following table describes the factors affecting the RVGPrimary resource on the Primary and Secondary nodes.

**Table 8-6** Factors affecting the RVGPrimary resource on Primary and Secondary nodes

| Node on which RVGPrimary is online | Factors affecting the RVGPrimary resource actions | Action that the RVGPrimary agent performs during online operation |
|---|---|---|
| Primary | None | The resource is online. |

**Table 8-6** Factors affecting the RVGPrimary resource on Primary and Secondary nodes *(continued)*

| Node on which RVGPrimary is online | Factors affecting the RVGPrimary resource actions | Action that the RVGPrimary agent performs during online operation |
|---|---|---|
| Secondary | The Secondary is connected and is up-to-date. | The resource performs a migrate operation and the `RVGPrimary` resource becomes online.<br>If there are multiple Secondaries in the RDS, and RLINKs between the Secondaries have been created, then, the `RVGPrimary` agent adds these Secondaries back into the RDS and synchronizes them with the new Primary. This happens in the background once the resource has come online. |
| Secondary | The Secondary is connected but is not up-to-date. | The resource waits until the online time-out period is reached, for the Secondary to become up-to-date. If the Secondary becomes up-to-date then the resource performs a migrate operation and the `RVGPrimary` resource is brought ONLINE, else it will fault.<br>If there are multiple secondaries in the RDS, and RLINKs between the secondaries are created, then, the `RVGPrimary` agent adds these secondaries back into the RDS and synchronizes them with the new Primary. |

**Table 8-6** Factors affecting the RVGPrimary resource on Primary and Secondary nodes *(continued)*

| Node on which RVGPrimary is online | Factors affecting the RVGPrimary resource actions | Action that the RVGPrimary agent performs during online operation |
|---|---|---|
| Secondary | The Secondary is not connected, and the following attributes are set:<br>`AutoTakeover=1`<br>`AutoResync=1` | If the original primary node has a bunker RVG associated with it, then the resource first synchronizes the secondary node from the Bunker before performing a takeover with fast-failback logging. When the original Primary becomes accessible, it is converted to a secondary and is automatically synchronized with the new Primary.<br>If there are multiple secondaries in the RDS, and RLINKs between the secondaries are created, then, the `RVGPrimary` agent adds these secondaries back into the RDS and synchronizes them with the new Primary. |

**Table 8-6** Factors affecting the RVGPrimary resource on Primary and Secondary nodes *(continued)*

| Node on which RVGPrimary is online | Factors affecting the RVGPrimary resource actions | Action that the RVGPrimary agent performs during online operation |
|---|---|---|
| Secondary | The Secondary is not connected, and the following attributes are set:<br>`AutoTakeover=1`<br>`AutoResync=0` | The resource performs a takeover with fast-failback, but without performing the automatic synchronization. In the case of a Bunker setup, the resource first synchronizes the secondary node from the Bunker node before performing a takeover with fast-failback logging.<br>You need to manually resynchronize the original Primary when it becomes available again using:<br>■ **Resynchronize Secondaries** option from the GUI<br>■ `fbsync` action from the Actions dialog that appears when you right-click and select `RVGPrimary` resource > **Actions** from the Cluster Manager (Java Console)<br>The `fbsync` action is very useful as it enables you to perform synchronization from the VCS console itself without having to switch to the VEA console.<br>If there are multiple secondaries in the RDS, and RLINKs between the secondaries are created, then, the `RVGPrimary` agent adds these secondaries back into the RDS and synchronizes them with the new Primary. |

**Table 8-6** Factors affecting the RVGPrimary resource on Primary and Secondary nodes *(continued)*

| Node on which RVGPrimary is online | Factors affecting the RVGPrimary resource actions | Action that the RVGPrimary agent performs during online operation |
|---|---|---|
| Secondary | The Secondary is not connected and the Primary cluster state has been declared as disaster or replica. For more information, see *Cluster Server Administrator's Guide*.<br>In this you have set the following attributes:<br>`AutoTakeover=1`<br>`AutoResync=0` | The resource performs a takeover without fast-failback.<br>If there are multiple secondaries in the RDS, and RLINKs between the secondaries are created, then, the `RVGPrimary` agent adds these secondaries back into the RDS and synchronizes them with the new Primary |
| Secondary | The Secondary is inconsistent. | The resource fails to come online. |
| Acting Secondary | | The resource fails to come online. |

## Sample RVGPrimary resource definition

The following is a sample of the RVGPrimary resource definition as specified in the main.cf file.

```
RVGPrimary SQL_CLUSTER_GRP-RVGPrimary
     RvgResourceName = VVR_Rep_Grp-VvrRvg
```

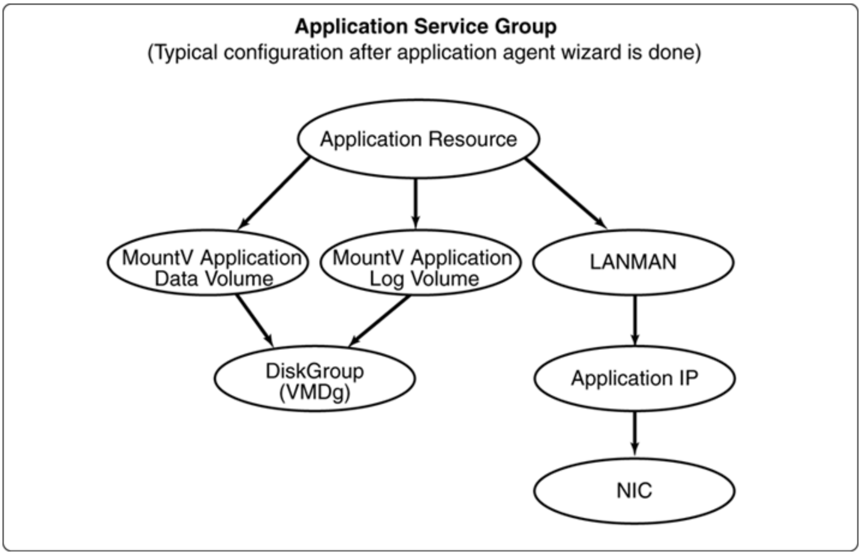
# Configuring the agents

This section explains how you can cover replication under a cluster.

Before that, you must first create the application service group for the application whose data Volume Replicator is replicating. The following figure illustrates a typical configuration after the application service group has been created using the application-specific wizard.

For more information about creating the application service group, see *Cluster Server Administrator's Guide*.

**Figure 8-7** Typical service group configuration in a clustered environment


Application Service Group
(Typical configuration after application agent wizard is done)
Application Resource
MountV Application Data Volume
MountV Application Log Volume
LANMAN
DiskGroup (VMDg)
Application IP
NIC

You can create the replication service group using the Volume Replicator agent configuration wizard to cover replication under a cluster. However, the replication setup must exist before it can be covered under a cluster. Create a Replicated Data Set (RDS) using an IP address that is available in the setup.

For a cluster setup you need to set the Primary and Secondary to a virtual IP address. The Volume Replicator agent configuration wizard creates the resource for this IP address. Currently however, the resource for this is not created. Therefore you need to use the Change Replication Settings wizard to set the replication IP to a non-existent virtual IP. When using the Volume Replicator Agent Configuration Wizard you can specify this IP for which the resource is then created.

Use the Volume Replicator agent configuration wizard to create the replication service group along with the IP resource for replication. Using this wizard you can specify the IP address used in the existing replication setup to create the corresponding IP resource in the replication service group, when necessary. If you already have an IP resource created then you can choose to use this resource by either copying or linking to the resource.

After the replication service group has been created, the RVG resource must be dependent on the IP address of the local host that is used for replication. In some cases, it is possible that an RVG uses more than one IP on the local host, for replication. This is especially true if the RVG is a Primary with more than one Secondary and different IPs on the Primary are used to create RLINKs to each Secondary. In this case, a resource should be present for each of these IPs in the replication service group. The resource for this Primary RVG should then depend on each of the IP resources.

The following table lists the procedures and description related to agent configuration.

**Table 8-7** Procedures and related description for agent configuration.

| Procedure | Description |
|---|---|
| Creating the application service group | You must first create the application service group. After creating it on the Primary, take the application service group offline before creating it on the Secondary.<br>For more information, see *Cluster Server Administrator's Guide*. |
| Taking the application group offline | Before creating the replication service group, take the application service group offline, but make sure that the disk group is imported.<br>See "Taking the application group offline on Secondary" on page 356. |
| Setting up Replication | Use the setup RDS wizard.<br>See "Setting up replication using a virtual IP address" on page 356. |
| Changing the Primary and Secondary IP | Use the Change Replication Settings option from the Secondary to change the Primary and Secondary IP to one that is intended for replication and currently does not have any resources created.<br>See "Changing the Primary and Secondary IP" on page 357. |
| Creating RLINKs between each pair of Secondaries | If your setup has multiple Secondary hosts, the RLINKs are automatically created when a Secondary is added to the RDS.<br>See "Creating RLINKs between each pair of Secondary hosts" on page 357. |
| Running the Volume Replicator agent configuration wizard | Run the Volume Replicator agent configuration wizard.<br>See "Creating the replication service group" on page 357.<br>For a setup using multiple IP addresses for replication, run the wizard in the modify mode to create the IP resources for each of these IPs and make the VvrRvg resource dependent on each of them.<br>See "Modifying an existing resource in the replication service group" on page 365. |

**Note:** The VCS NIC resource can be duplicated because it is possible that other IP resources excluding the replication IP addresses share the same NIC resource.

## About configuring the Disaster Recovery Solutions using the DR Wizard

The section describes the process of setting up a DR configuration using the Volume Replicator Agent Configuration Wizard. You can however perform the same tasks automatically using the DR wizard. The Disaster Recovery (DR) wizard clones the storage configuration and service group configuration from the Primary site to the Secondary site. It also configures replication settings and connects the clusters into a global cluster. Although all the tasks can be performed using this single wizard, you need to exit the wizard after cloning the storage to install the required application. You can exit the wizard, after the logical completion of each task.

For detailed information about configuring DR solutions using the DR wizard, refer to the Storage Foundation and High Availability Solutions HA and Disaster Recovery solutions guides.

## Taking the application group offline on Secondary

Before proceeding with configuring the replication service group, make sure that you take the application service group offline, if the RVG is a Secondary. However, make sure that you import the disk group.

**To take the application group offline**

1. Open the Java Console from a system where you have installed it, double-click on the Veritas Cluster Manager (Java Console) icon on the desktop.

   You can also launch Veritas Cluster Manager (Java Console) from **Start > All Programs > Veritas > Veritas Cluster Server > Veritas Cluster Manager (Java Console)** or, on Windows 2012 operating systems, from the **Apps** menu in the **Start** screen.

2. From the left-pane, right-click the service group you want to take offline and select the node you want to take offline. Click **Yes** to take the application group offline.

   Manually online the disk group resource.

## Setting up replication using a virtual IP address

Use the Setup Replicated Data Set wizard to set up the RVG and RDS.

See “About setting up replication” on page 74.

Depending on whether you have a Bunker or a non-Bunker set up, follow the appropriate set of instructions. To configure replication, use the IP address that is available with the setup. For a cluster, the IP address must be a virtual IP address which can failover along with the other resources in the replication service group.

## Changing the Primary and Secondary IP

Use the Change Replication Settings option to set the Primary and Secondary IP to one for which no resource has been created and is intended for replication.

**To change the replication IP**

1. Click the **Change Replication Settings** option from the Secondary RVG to display the Change Replication Settings dialog.
2. Modify the replication IPs for the Primary and Secondary.

   If the required IP is not available because the resource for this IP does not exist, then add the IP manually.

   See “Changing replication settings for an RDS” on page 173.

## Creating RLINKs between each pair of Secondary hosts

If your configuration has more than one Secondary host then Volume Replicator automatically creates the RLINKs between each pair of Secondary hosts. These RLINKs enable the RVGPrimary agent to automatically manage the process of attaching these Secondaries to the new Primary, after a migrate or takeover operation.

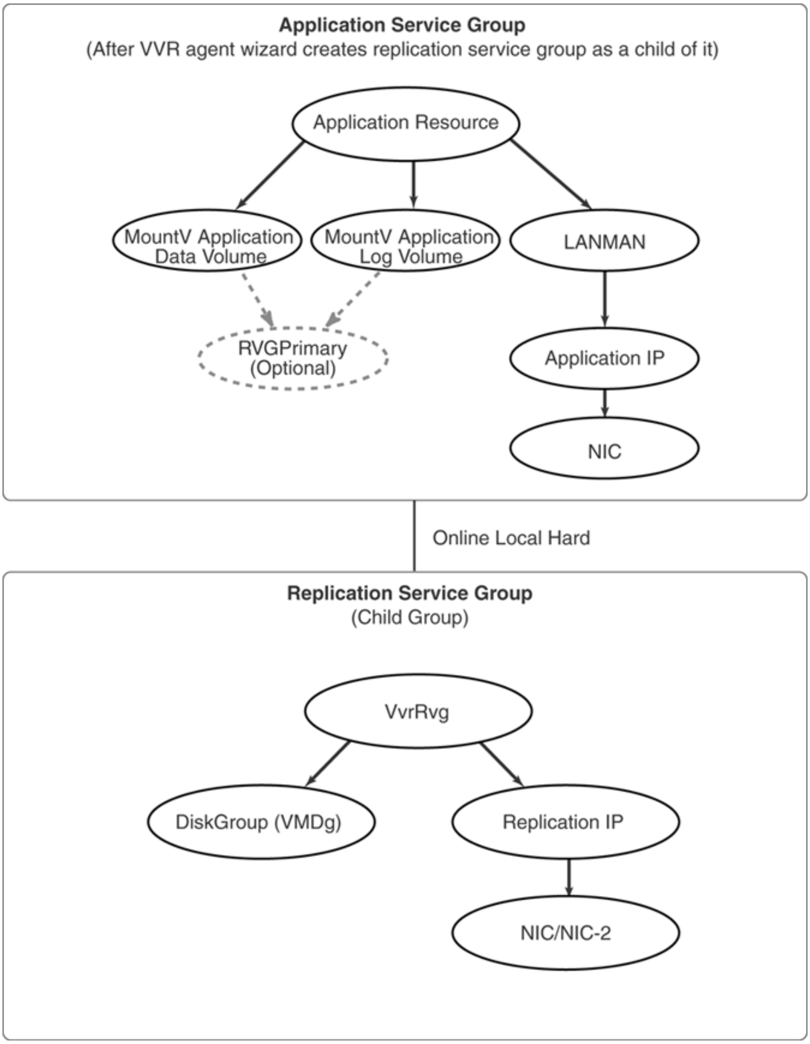
## Creating the replication service group

Use the Volume Replicator Agent Configuration wizard to create the replication service group. Perform the following steps on each node of the clustered Primary and repeat the same on the nodes of a clustered Secondary. Before proceeding, make sure that the disk group has been imported on the node on which you create the replication service group.

Note that after running the wizard, a replication service group is created. The wizard also sets the dependency between the replication service group and the application service group.

The following figure illustrates what your setup looks like after the replication service group has been created.

**Figure 8-8** Typical replication service group configuration


Application Service Group
(After VVR agent wizard creates replication service group as a child of it)
Application Resource
MountV Application Data Volume
MountV Application Log Volume
LANMAN
RVGPrimary (Optional)
Application IP
NIC
Online Local Hard
Replication Service Group
(Child Group)
VvrRvg
DiskGroup (VMDg)
Replication IP
NIC/NIC-2

## Prerequisites for creating the replication service group

Before creating a replication service group, certain considerations should be taken into account.

Check for the following prerequisites:

- Verify that the disk group is imported on the node on which you want to create the Replication Service Group.

- Verify VCS is running, by running the following command on the host on which you intend to run the Volume Replicator Agent Configuration Wizard.

  ```
  > hasys -state
  ```

**To create a replication service group**

1. Launch the configuration wizard from the active node of the cluster at the Primary site from **Start > All Programs > Veritas > Veritas Cluster Server > Configuration Tools > Volume Replicator Agent Configuration Wizard**, or on Windows 2012 operating systems, from the **Apps** menu in the **Start** screen.
2. Review the requirements on the **Welcome** panel and click **Next**.
3. In the **Wizard Options** panel, click **Create a new replication service group** and click **Next**.
4. Specify the service group name and system priority list, and then click **Next**:
   - To remove a node from the service group's system list, click the node in the **Systems in Priority Order** box, and click the left arrow icon.
   - To change the priority of a node in the system list, click the node in the **Systems in Priority Order** box, then click the up and down arrow icons. The node at the top of the list has the highest priority.
   - To enable the service group to automatically come online on one of the systems, select the **Include selected systems in the service group's AutoStartList attribute** check box.
     For information about the AutoStartList attribute, see the *Cluster Server Administrator's Guide*.
5. A message appears, indicating that the configuration will change from Read Only to Read/Write. Click **Yes** to continue.
   - Select **Configure RVGPrimary resource for selected RVG**.
     This resource is required when you want to configure your setup to automatically enable takeover in case of a failure of the Primary cluster. The `RVGPrimary` resource is created in the application service group and replaces the `VMDg` resource.
   - Select the replicated volume group for which you want to configure the RVG Primary resource.
   - Click **Next**.

   You can create the `RVGPrimary` resource only while creating a new RVG resource and not when modifying an existing RVG resource. For an existing

RVG resource, you can use VCS Java Console to create the `RVGPrimary` resource in the appropriate application service group and then set the dependencies for all the resources in the application service group that depend on `VMDg` to `RVGPrimary`.

For more information about using the VCS Java Console, see *Cluster Server Administrator's Guide*.

6 In the **IP Resource Options** panel, select **Create a new IP resource** and click **Next**.

If you want to create a copy of an IP resource that already exists in another service group, select **Create a copy of an IP resource existing in a different service group**. When you select this option, the list of available IP resources are displayed in the **Available IP Resources** pane. Choose the required IP resource.

- Verify or enter the virtual IP address; use the IP address that is specified as the Primary IP address when you configured the RDS.
- Specify the subnet mask.
- Specify the adapters for each system in the configuration.
- Click **Next**.
- If you had chosen the option to create a copy of an existing IP resource then the panel is filled up as described in the following table

The **Resources** box lists the configured resources. Click a resource to view its attributes and their configured values in the Attributes box.

- If necessary, change the resource names; the wizard assigns unique names to resources based on their respective name rules.
- To edit a resource name, click the resource name and modify it. Press Enter after editing each resource name. To cancel editing a resource name, press **Esc**.
- Click **Next** to create the replication service group.

7 A warning informing you that the service group is created is displayed. When prompted, click **Yes** to create the service group.

8 Click **Finish** to bring the replication service group online.

9 Check the prerequisites, then repeat the wizard at the Secondary site, specifying the appropriate values.

The name for the application service group must be the same on both sites.

Repeat the steps on one node of the Secondary cluster.

# Working with existing replication service groups

This section details some tasks that you can perform on an existing replication service group.

See "Adding a new RVG resource to an existing replication Service group " on page 361.

See "Modifying an existing resource in the replication service group" on page 365.

## Adding a new RVG resource to an existing replication Service group

This option is required when a disk group has multiple RVGs. Using this option you can add the resource for additional RVGs to an existing replication service group.

**Note:** The systems that are selected for the replication service group must be a superset, and must have the same order, as those you had selected for the application service group.

**To add a resource into an existing service group**

1. Verify that VCS is running. From the Java Console, logon to the Primary site.
2. On a clustered node on the Primary site, launch the configuration wizard from **Start > All Programs > Veritas > Veritas Cluster Server > Configuration Tools > Volume Replicator Agent Configuration Wizard** or, on Windows 2012 operating systems, from the **Apps** menu in the **Start** screen.
3. Read the information on the **Welcome** panel. Click **Next**.
4. In the **Wizard Options** panel, click **Add RVG resource to an existing replication service group**. Select the replication service group to which you want to add an RVG resource, and click **Next**.

5 In the **Service Group Configuration** panel, the appropriate replication service group name is selected.

The current system list is displayed. Select the nodes from the Available Cluster Systems list and click the appropriate arrow button to add them to the **Systems in Priority Order** list. Make sure that all the listed nodes on which the disk group can be imported must be selected. The nodes to be added to the service group system list are listed in priority order.

Use the up and down arrows to change the priority of the clustered nodes on which the service group needs to be brought online.

To enable the service group to automatically come online on one of the systems, select the **Include selected systems in the service group's AutoStartList attribute** check box. For information about the AutoStartList attribute, see the *Cluster Server Administrator's Guide*.

Click **Next**.

6 Complete the Disk Group and Replicated Volume Group Configuration panel.

This can be done as follows:

| | |
|---|---|
| Configure RVGPrimary resource for the selected RVG | Select this option, if you want to create an `RVGPrimary` resource for the selected RVG. This resource is required when you want to configure your setup to automatically enable takeover in case of a failure of the Primary cluster. The `RVGPrimary` resource is created in the application service group and replaces the `VMDg` resource. |
| Available Replicated Volume Groups | Select the RVG to which you want the new replication service group to be added. The Next option is enabled. |

Click **Next** to display the IP Resource Options panel.

7 When adding a new resource to an existing replication service group, the following options are enabled.

Complete the IP Resource panel as follows:

| | |
|---|---|
| Create New IP resource | Select this option to create a new IP resource for the resource that you create. |
| Create a copy of an IP resource existing in another service group | Select this option to create a copy of an IP resource that already exists in another service group. When you select this option, the list of available IP resources are displayed in the **Available IP Resources** pane. Choose the required IP resource. |
| Link to an IP resource existing in the current service group | Select this option to use an IP resource that exists in the current service group.<br>This option cannot be used to choose an IP resource that lies outside the current service group. |

Click **Next** to display the Network Configuration panel.

8 Complete the Network Configuration panel as follows:

If you had chosen the Create New IP resource on the preceding panel, then complete the panel as follows:

| | |
|---|---|
| Virtual IP address | Specify the virtual IP address in this field. Volume Replicator uses this address for replication. |
| Subnet Mask | Enter the subnet mask. |
| Adapter Display Name (Mac address) | Specify the correct adapter name (Mac address) of each system to which you want to assign the IP resource and the corresponding NIC resource, in the Adapter Display Name column. |

If you had chosen the option to create a copy of an existing IP resource or to link to an existing IP resource, then the panel is filled up as described in the following table:

| | |
|---|---|
| Virtual IP address | If the IP specified for replication has a resource that is created in the cluster, the wizard copies that IP and the corresponding NIC resource to the replication service group. |
| Subnet Mask | If the resource for the IP already exists then the wizard disables the Subnet Mask field and other inputs as these values are taken from the existing IP resource.<br>If no resource has been created for the specified IP then you can enter a subnet mask value and choose the proper adapter on each system. |
| Adapter Display Name (Mac address) | The appropriate adapter name (Mac address) is displayed for each system. |

Verify that you have specified the correct IP and Subnet Mask information.

If you need to change this information later then you can do it by running the wizard in the modify mode.

Click **Next**.

9 In the **Service Group Summary** panel you can modify the resource name for the new resource that you add. Click on the resource name in the left pane to modify the name. After you are done, click **Next** to proceed with creating the resources.

A warning informing you that the service group will be created is displayed. Click **Yes** to proceed.

10 After the resource has been successfully created, the completion panel appears. Click **Finish** to complete the procedure and exit the wizard.

11 Repeat the steps on one node of the Secondary cluster.

## Modifying an existing resource in the replication service group

To modify an existing resource in the replication service group, you need to perform the following.

**To modify a resource in an existing replication service group**

1 Verify VCS is running. From the Java Console, log on to the Primary site.

2 On a clustered node on the Primary site, launch the configuration wizard from **Start > All Programs > Veritas > Veritas Cluster Server > Configuration Tools > Volume Replicator Agent Configuration Wizard** or, on Windows 2012 operating systems, from the **Apps** menu in the **Start** screen.

3 Read the information on the **Welcome** panel. Click **Next.**

4 In the **Wizard Options** panel, choose the Modify an existing replication service group option and then select the replication service group whose resources you want to modify from the list that is displayed in the pane. Click **Next**.

5 In the **Service Group Configuration** panel view the information that is selected for the fields. Click **Next.**

6 In the **Disk Group and Replicated Volume Group Configuration** panel, select the RVG whose resources you want to modify from the list that is displayed in the **Available Replicated Volume** pane. This enables the Next option.

When you modify an existing replication service group, the Configure RVGPrimary resource for the selected RVG option is unavailable for selection.

Click **Next**.

7 When modifying an existing replication service group, the following options are enabled.

Complete the IP Resource panel as follows:

| | |
|---|---|
| Create New IP resource | Select this option to create a new IP resource for the resource that you create. |
| Create a copy of an IP resource existing in another service group | Select this option to create a copy of an IP resource that already exists in another service group. When you select this option the list of available IP resources are displayed in the **Available IP Resources** pane. Choose the required IP resource. |
| Link to an IP resource existing in the current service group | Select this option to establish a link to an IP resource within the current service group.<br>This option cannot be used to link to an IP resource that lies outside the current service group.<br>**Note:** All the resources in the replication service group, except `VvrRvg` and `VMDg` should be offline so that the IP address and subnet mask values of the IP resource can be modified. |
| Modify IP resource | Select this option to modify attributes of an existing IP resource. When you select this option, the list of available IP resources in the current service group are displayed in the **Available IP Resources** pane. Choose the required IP resource. |

Click **Next** to display the **Network Configuration** panel.

8 Complete the **Network Configuration** panel as follows depending on the option that you had chosen in the preceding panel:

- If you had chosen the Create New IP resource on the preceding panel, then complete the panel as follows:

| | |
|---|---|
| Virtual IP address | Specify the virtual IP address in this field. Volume Replicatoruses this address for replication. |
| Subnet Mask | Enter the subnet mask that is provided in field provided. |
| Adapter Display Name (Mac address) | Specify the correct adapter name (Mac address) of each system to which you want to assign the IP resource and the corresponding NIC resource, in the Adapter Display Name column. |

- If you had chosen any one of the options; to create a copy of an existing IP resource, link to an existing IP resource or modify an existing resource, then the panel is filled up as described in the following table. However, you can edit the information on this panel to change it to values that you require:

| | |
|---|---|
| Virtual IP address | Because the IP specified for replication has a resource that is created in the cluster, if copy option is selected, the wizard copies that IP and the corresponding NIC resource to the replication service group.<br>If the link option is selected, then, the Network Configuration panel displays the value of the IP resource to which the selected RVG resource is linked. |
| Subnet Mask | If the resource for the IP already exists then the wizard disables the Subnet Mask field and other inputs as these values are taken from the existing IP resource.<br>If no resource has been created for the specified IP then you can enter a subnet mask value and choose the proper adapter on each system. |
| Adapter Display Name (Mac address) | The appropriate adapter name (Mac address) is displayed for each system. |

Click **Next**.

9 In the **Service Group Summary** panel modify the resource name. Click on the resource name in the left pane to modify the name. After you are done, click **Next** to proceed with creating the resources.

A warning informing you that the service group will be created is displayed. Click **Yes** to proceed.

10 After the resources have been successfully modified, the completion panel appears. Click **Finish** to complete the procedure and exit the wizard.

Chapter 9

# Configuring Volume Replicator with Hyper-V

This chapter includes the following topics:

## Implementing Volume Replicator replication on Hyper-V with Microsoft failover cluster

Volume Replicator provides support for replicating virtual machine images on Hyper-V. When combined with Microsoft failover clustering services, this setup can ensure complete disaster recovery as well as high availability for the virtual machines.

## Prerequisites for setting up Volume Replicator with Hyper-V

Before proceeding with configuring Volume Replicator on Hyper-V, ensure that your setup meets the following requirements:

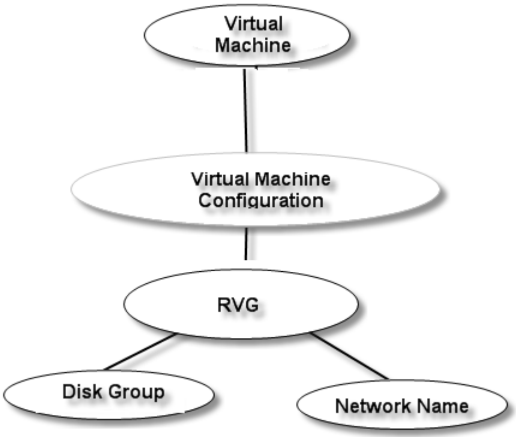
**Note:** To implement replication on the virtual machine, you must ensure that all disks that are given to the virtual machine belong to the same disk group.

- To failover a virtual machine, you need to failover all the disks that are associated with a virtual machine. This way, the virtual machine has a resource dependency on its disk groups (DG).
- For a successful failover, the virtual machine must have a dependency on its Replicated Volume Group (RVG) resource. A dependency on the RVG resource implies a dependency on the disk group to which an RVG belongs.
- If all disks associated with a virtual machine belong to the same disk group as an RVG, then the Virtual machine needs to have a resource dependency only on its RVG.

# Configuring a virtual machine group and resource dependencies

This section deals with configuring a virtual machine group and its related resource dependencies.

The following dependency graph illustrates a typical configuration for Volume Replicator on Hyper-V with MSCS:



The resources in a service group must come online and go offline in a particular order; the dependencies represent this order. In the above graph, resources must come online starting at the "bottom" and moving up the dependency lines.

In the sample configuration that is shown in the dependency graph above, the resource group is configured using the virtual machine configuration which in turn has a dependency on the RVG. RVG has a dependency on disk group (DG) and Network Name (Client Access Point) resource.

# Configuring replication for the virtual machine

The disks that are assigned to virtual machines can be categorized into two types.

They are as follows:

- System disk
  The disk on which the system boot volume resides.
- Data disk
  Disk where the application data resides.

On the basis of the above, two types of setup or configuration are possible for Volume Replicator on Hyper-V with failover clusters.

They are as follows:

- Replicating System as well as Data disk
  See "Setup 1: Replicating the System as well as Data disks" on page 370.
- Replicating only the Data disk
  See "Setup 2: Replicating the Data disks" on page 371.

## Setup 1: Replicating the System as well as Data disks

For Setup 1, both the System as well as the Data disks are replicated. This type of configuration has its advantages as well as disadvantages as shown in the table below:

| Type of setup | Advantages | Disadvantages |
|---|---|---|

| | | |
|---|---|---|
| Setup 1: Replicating the System as well as Data disks | Since system drive is replicated, all system settings and changes are replicated to the Secondary site. Thus, the Secondary site is an exact replica of the Primary site. | ■ If the guest image on the Primary is corrupted due to a crash, then the guest image which is replicated to the Secondary may not boot or start.<br>■ The application on the system drive generates huge I/O on the system drive, which when replicated adds to network traffic.<br>■ During migrate and snapback operations (resync from replica), users may encounter "fail to acquire lock on volumes” error. This happens due to open handles on the volume. |

## Setup 2: Replicating the Data disks

For Setup 2, only the Data disk is replicated without the System disk. The possible advantages as well as disadvantages are explained in the table below for such type of configuration:

| Type of setup | Advantage | Disadvantages |
|---|---|---|
| Setup 2: Replicating the Data disks | If guest image is corrupted due to a crash, the image on the Secondary remains unaffected. | Users have to manually ensure that the state of Primary and Secondary system is similar. |

# Recommendations and workarounds

This section deals with certain recommendations and workarounds for resolving the errors that are encountered on a Volume Replicator and Hyper-V configuration.

**Note:** All Volume Replicator operations can be performed on a Hyper-V MSCS configuration except the volume shrink and restore operation. This is by design and expected.

**Note:** It is recommended to have one virtual machine per disk group.

Some recommendations are as follows:

- Before performing a restore operation, ensure that virtual machines are shut down and original volumes are not in use. If virtual machines are not shut down and a restore operation is performed forcefully, then I/O device error is seen on the data disk which is under replication. Due to this no I/Os occur on the volume on the guest (Hyper-V) which is under replication. Data volume state becomes unpredictable at this stage. System Eventviewer on the guest shows "Failed to flush data to the transaction log. Corruption may occur." This error gets resolved after restarting the virtual machine.
- During migrate operation, the virtual machine must be shut down. If after shutting down the virtual machine and performing a migrate operation returns error, check for open handles on the volume that the virtual machines use. The vmms.exe and System processes may have open handles on volumes. Ensure to close all open handles before performing a migrate or restore operation. The volumes can also be forcefully dismounted before a migrate or restore operation to close all open handles. However, this may cause I/O errors.

Chapter 10

# Advanced settings in Volume Replicator

This chapter includes the following topics:

## About using the advanced settings in Volume Replicator

This chapter describes the advanced features that help you use Volume Replicator more effectively and efficiently.

- See "Tuning the Volume Replicator memory parameters" on page 373.
- See "Understanding IBC messaging" on page 376.

## Tuning the Volume Replicator memory parameters

This section describes how you can modify the tunable parameters which control the system resources that Volume Replicator uses. Depending on the system resources that are available, adjustments may be required to the values of some tunable parameters to optimize performance. Note that all the tunable values must be in multiples of kilobytes (KB).

## Understanding the concept of a buffer space

When a write is replicated, Volume Replicator allocates data buffers for it. These data buffers are allocated some memory. The amount of memory (buffer space) available to Volume Replicator affects its performance, which can affect the write performance to the underlying volumes.

To manage buffer space on the Primary and Secondary according to your requirements, use the following tunables:

- `MAX_MEMORY`—Use the `vxtune vol_rvio_maxpool_sz` command to set a value for the `MAX_MEMORY` tunable or to view the value that is currently assigned to it.
- `BASE_MEMORY`—Use the `vxtune vol_min_lowmem_sz` command to set a value for the `BASE_MEMORY` tunable or to view the value that is currently assigned to it.
- `NMCOM_POOL_SIZE`—Use the `vxtune vol_max_nmpool_sz` command to set a value for the `NMCOM_POOL_SIZE` tunable or to view the value that is currently assigned to it.
- `READBACK_POOL_SIZE`—Use the `vxtune vol_max_rdback_sz` command to set a value for the `READBACK_POOL_SIZE` tunable or to view the value that is currently assigned to it.
- `FORCE_MAX_CONNECTION`—Use the `force_max_conn` command to set a value for the `FORCE_MAX_CONN` tunable or to view the value that is currently assigned to it.
- `MAX_TCP_COUNT`— Use the `max_tcp_conn_count` command to set a value for the `MAX_TCP_COUNT` tunable or to view the value that is currently assigned to it.
- `NMCOM_MAX_MESSAGES`—Use the `nmcom_max_msgs` command to set a value for the `NMCOM_MAX_MESSAGES` or to view the value that is currently assigned to it.
- `MAX_RECEIVE_GAP`—Use the `max_rcvgap` command to set a value for the `MAX_RECEIVE_GAP` tunable or to view the value that is currently assigned to it.
- `RLINK_READBACK_LIMIT`—Use the `rlink_rdbklimit` command to set a value for the `RLINK_READBACK_LIMIT` tunable or to view the value that is currently assigned to it.
- `NETWORK_PACKET_LOSS_TOLERANCE`—Use the `rp_incr_decr` command to set a value for the `NETWORK_PACKET_LOSS_TOLERANCE` tunable or to view the value that is currently assigned to it.
- `TCP_SOURCE_RESTRICT`—Use the `tcp_src_port_restrict` command to set a value for the `TCP_SOURCE_RESTRICT` tunable or to view the value that is currently assigned to it.

- IOPATH_LOGGING—Use the iopath_logging command to set a value for the IOPATH_LOGGING tunable or to view the value that is currently assigned to it.
- NAT_SUPPORT—Use the nat_support command to set a value for the NAT_SUPPORT tunable or to view the value that is currently assigned to it.

## Shared memory between Volume Replicator and SFW

You can use the following tunable to specify the amount of memory that needs to be shared between Volume Replicator and SFW:

```
vol_rvio_maxpool_sz (MAX_MEMORY)
```

You can use the following tunable to specify the amount of buffer space that is shared between Volume Replicator and SFW for processing the incoming Input/Output. The default value for this parameter is 32 MB. However, you can specify any value that lies in the range 4MB to 1GB.

## Minimum memory required by SFW and Volume Replicator

You can use the following tunable to specify the minimum memory that needs to be shared between Volume Replicator and SFW:

```
vol_min_lowmem_sz (BASE_MEMORY)
```

The value that this tunable specifies indicates the minimum amount of memory that SFW and Volume Replicator always keep, that is, this memory is not freed even if it is not used. The default value is 1MB. However, you can specify any value that lies in the range 512KB to 10MB.

## Size of the memory available on the Secondary

Defines the amount of memory (buffer space) to that the Secondary uses to store the received updates.

```
vol_max_nmpool_sz (NMCOM_POOL_SIZE)
```

The Volume Replicator tunable, NMCOM_POOL_SIZE, determines the amount of buffer space available for requests coming in to the Secondary over the network, which by default is 16MB. However, you can specify any value that lies in the range from 4MB to 512MB. The NMCOM_POOL_SIZE tunable is used only on the Secondary.

**Note:** Since this value is global, and is restricted to all the Secondary RVGs on a node, it may also be useful to increase the value of the NMCOM_POOL_SIZE tunable if multiple Secondary RVGs are present on the Secondary node.

### Readback buffer space on the Primary

Defines the amount of memory that the Primary can use to read data from the Replicator Log. The default value is 16MB. However, you can also specify any value that lies in the range from 4MB to 512MB.

```
vol_max_rdback_sz (READBACK_POOL_SIZE)
```

When a write request is made, a Volume Replicator data buffer is allocated to it. The data buffer is not released until the data has been written to the Primary Replicator Log and sent to all the Secondaries that are connected by synchronous RLINKs. When the buffer space becomes low, several effects are possible, depending on the configuration. Volume Replicator begins to free some buffers before sending the data across the asynchronous RLINKs. This frees up more space for incoming write requests so that they are not delayed. The cost is that it forces the freed requests to be read back from the Replicator Log later, when an RLINK is ready to send them. The need to perform readbacks have an effect on write latency because it makes the Replicator Log perform more non-sequential I/O. It also increases the load on the system and slows the rate at which data is sent to the Secondaries.

## Modifying the tunable values

You can modify the tunable values using the `vxtune` command.

See “Tuning Volume Replicator” on page 312.

# Understanding IBC messaging

Volume Replicator maintains a block-level consistency between a Primary volume and the corresponding Secondary volumes. Applications that are built on Storage Foundation volumes, such as a file system, require a higher level of consistency. To support this higher-level consistency model, the Volume Replicator provides the IBC messaging facility.

The IBC messaging facility allows applications to insert control messages into a Replicated Volume Group’s (RVG) update stream. This control message is application-defined and is completely transparent to the replication process. The IBC messages follow the same consistency rules as updates to a volume. When you send the IBC messages, if you ensure that it is sent when there is no major concurrent activity then it is sent in the same sequence as it was issued. If it is sent while there is concurrent activity, the message is delivered arbitrarily in relation to the activity.

The administrators at the Primary and Secondary hosts decide the protocol that the Primary uses to specify the message and the Secondary to understand that message.

## Features of the IBC messaging

IBC messaging facilitates applications to insert control messages into an RVG's update stream.

The features of the IBC messaging are as follows:

- An IBC always causes any previous update activity to be flushed before delivery.
- As an administrator, you can decide the sequence of activities at the Secondary after receiving the message. For example you may decide to continue with the updates that have been received from the Primary or you may decide to freeze the replication and perform the activity mentioned in the IBC message.
- The IBC messaging functionality ensures that the messages are delivered correctly at least once.
- In the case of a network failure or system crash during the delivery of an IBC message, the IBC may be delivered more than once. The applications using the facility must be able to handle multiple delivery of the same IBC.
- Some IBC messages may freeze replication activity, until the application releases it. The delivery definition must therefore include the complete instruction to freeze and unfreeze.
- All the IBC messages are also logged in the Primary Replicator Log Volume.

These features of IBC messaging facility ensure that the message is successfully delivered and processed at least once.

## Application of IBC messaging

A typical use of IBC messages is to checkpoint application-level data consistency within a replicated volume group. An application running on the Primary node can insert an IBC message into the update stream at a point at which the application considers its data on replicated volumes to be consistent. An instance of the same application running on the Secondary host is then assured that the data on the Secondary is consistent at the application-level when it retrieves the IBC message. The IBC functionality has an option to freeze the replication on the Secondary host on receipt of an IBC message. The data on Secondary volumes would not change till the freeze is in effect. During this time the application on the Secondary node can perform a backup of the data volumes or take a snapshot or carry out any such activity.

# IBC messaging commands

Use the vxibc command to perform IBC messaging operations in the Volume Replicator environment. It allows applications to insert user-defined control messages into a Replicated Volume Group's (RVG) update stream. An IBC message is delivered on the Secondary node in the same order that it was sent from the Primary. Before delivery of the message on the Secondary node, any previous update activity is flushed. You have the option to allow subsequent updates to be applied immediately to the Secondary data volumes or freeze replication until released by the application.

Each application must be registered under an identical application name before beginning with the IBC messaging operations.

**Note:** If the Secondary host crashes, the registration is not applicable anymore, whereas IBC messages once sent are still available for sending even after the host crashes as they are logged in the Replicator Log. Veritas therefore recommends that you start and register the application on the Secondary host as a part of system startup.

The first operand to the vxibc command is a keyword that determines the specific operation to perform. The vxibc command has various keywords to perform the different IBC messaging functions. Each operation can be applied to only one dynamic disk group at a time. You must specify the name of the dynamic disk group using the -g option.

The following table describes the keywords that can be specified for vxibc command.

**Table 10-1** vxibc command options

| Option | Description |
|---|---|
| -D <deliver_timeout> | The deliver_timeout argument to the -D option specifies the time-out value in seconds for delivery of an IBC message after it has arrived at the Secondary RVG. When the time-out expires, the Secondary RVG discards the IBC message and continues replication. Default value for deliver_timeout is 10 minutes. A deliver_timeout value of 0 means infinite time-out. The deliver_timeout value should be specified only on the Primary. |

**Table 10-1** vxibc command options *(continued)*

| Option | Description |
|---|---|
| -F <freeze_timeout> | The freeze_timeout argument to the -F option specifies the time-out value in seconds between delivery of an IBC message on the Secondary node and execution of an unfreeze operation on the Secondary RVG. When the time-out expires, replication continues at the Secondary RVG. Default value for freeze_timeout is 10 minutes. A freeze_timeout value of 0 means infinite time-out. |
| -N | This option is used with a send or a regsend operation and specifies that replication is not to be frozen when the IBC message arrives on the Secondary RVG. |
| -R <receive_timeout> | The receive_timeout argument with the -R option specifies the time-out value in seconds to block the waiting for an IBC message if the receive or the regrecv operation is run in blocking mode, that is, without the -n option. Default value for receive_timeout is 10 minutes. A receive_timeout value of 0 means infinite time-out. |
| -f <filename> | Used with the send or the regsend operation, to read the message from the specified file name. When this option is used with the receive or the regrecv operation, the received message is saved to a file with the specified file name. The maximum size of the message file can be 128KB. If the message data is more that 128 KB the rest is ignored. |
| -g <diskgroup> | Specifies the name of disk group containing the RVG on which the IBC operation is to be performed. This option must be used with every vxibc command keyword. |
| -l <buf_length> | The buf_length argument to the -l option specifies the maximum length in bytes of the IBC message the user is willing to receive. If the length of the received message is greater than the value that buf_length specifies, then the message is truncated to buf_length bytes. |
| -m <message> | The message argument with the -m option is a user supplied string that is sent with the IBC message from the Primary node and received by the application performing the receive or the regrecv operation on the Secondary RVG. If the send or the regsend operation is executed without this option, a blank message is sent to the Secondary RVG. If a message consists of more than one word, it must be enclosed within double quotes. The format of the message is user-defined and may be used by the application performing IBC operations to exchange values or coordinate what tasks are to be performed. To send a large message that cannot be accommodated on the command line, use the -f option. |

**Table 10-1** `vxibc` command options *(continued)*

| Option | Description |
|---|---|
| `-n` | This option is used with the receive or the regrecv operations and indicates that the operation is non-blocking. Default is to block for receiving the IBC message. |

## Command arguments

The following table lists some of the arguments that need to be specified with the `vxibc` command keywords:

**Table 10-2** Arguments for `vxibc` command

| Arguments | Description |
|---|---|
| `application_name` | A unique identifying string that is used to match the IBC message sending application on the Primary host with the IBC message receiving application on the Secondary host. The application_name argument can accept an application name string of a maximum 31 bytes. If an application name is longer than 31 bytes, it is truncated to 31 bytes. |
| `command argument` | This command must be run when the Secondary host receives an IBC message through the `regrecv` command. If the command requires the arguments to be specified with space as delimiter, then the whole command and its arguments must be enclosed within double quotes. |
| `rlink` | Name of the RLINK on which the operation needs to be performed. You can get the name of the RLINK from the display of the `vxprint` command. |
| `rvg` | Name of the RVG on which the operation needs to be performed. |

## Registering an application

To register an application use the following IBC command:

```
vxibc -g <diskgroup> [-D <deliver_timeout>] register \
<application_name> <rvg>
```

This command registers the application name for the RVG. You must first register an application name for the required RVG before proceeding with any other IBC messaging operations.

You can perform all the further IBC operations on the specified RVG by using the application's registered name. The sender and the receivers of the IBC message must register the application with the same name. You can register a maximum of

32 applications for an RVG. Registration does not maintain persistency across node crashes. Applications on restarted nodes must be registered again.

For example, to register an application name for IBC you can use the command in the following way:

```
vxibc -g vvrdg1 -D 120 register app1 vvrRvg1
```

This command registers the application under the name app1 for IBC.

## Unregistering the application

To unregister an application, use the following IBC command:

```
vxibc -g <diskgroup> unregister <application_name> <rvg>
```

This command unregisters the application that had been registered earlier for the RVG. After unregistering, you cannot use the send operations against the `application_name` on the Primary RVG. IBC messages that were already inserted into the update stream before unregistering are delivered to the Secondary RVG. Unregistering the application on the Secondary causes the receive and the unfreeze operations for the registered application name to fail and any further IBC messages that are received for the application are discarded.

For example, to unregister an application run the command:

```
vxibc -g vvrdg1 unregister app1 vvrRvg1
```

## Sending a message

To send an IBC message for a Primary RVG, use the command:

```
vxibc -g <diskgroup> [-N | -F <freeze_timeout>] \[-f <filename> | -m
 <message>] send <application_name> \<rvg> [<rlink>  ...]
```

This command sends the IBC message from a Primary RVG for which the application_name has been previously registered. The IBC message is inserted into the update stream of the specified Secondary host. If the RLINK name to the Secondary host is not specified, the message is sent to all the RLINKS currently attached to the Primary RVG. Replication to the Secondary host is frozen at the point in time at which the IBC message is inserted at the Primary RVG. If the IBC message has been specified with the `-N` option then the replication is not frozen. Replication at the Secondary host remains frozen until an unfreeze operation is performed or the specified freeze_timeout expires.

For example, to send an IBC message run the command:

```
vxibc -g vvrdg1 -F 120 -f  msg.txt  send app1 vvrRvg1
```

This command reads the message from the file `msg.txt` and sends it to all Secondary hosts.

## Receiving a message

To receive an IBC message, run the command:

```
vxibc -g <diskgroup> [-n | -R <receive_timeout>] \[-l <buf_length>]
 [-f <filename>] receive <application_name> <rvg>
```

This command receives the IBC message that was sent from the Primary RVG to the Secondary host. The application_name must be previously registered for the Secondary RVG. Secondary replication is frozen at the point-in-time on the Secondary's update stream at which the IBC message was inserted at the Primary RVG. Secondary replication remains frozen until an unfreeze operation is performed or the freeze_timeout specified when the IBC message was sent expires. The default behavior for the receive operation is that until the IBC message is received the operation is not complete. For example, when you use the command from the command prompt, and you are running the `receive` command then you do not get the next command prompt until the IBC message is received. If the `receive` command is used with the `-n` option then the command is non-blocking, that is, the command is completed immediately even if the message has not been received.

If the operation succeeds, the received message is displayed. An unsuccessful exit code indicates that messages were dropped due to delivery time-outs and the drop count is displayed to standard error. If an error occurs while receiving a message, the error message is displayed with the drop count of messages.

For example, to receive an IBC message run the command:

```
vxibc -g vvrdg1 -R 120  -f  msg.txt receive app1 vvrRvg1
```

This command receives the message and stores it in a file named `msg.txt`.

## Unfreezing the Secondary RVG

To unfreeze the Secondary RVG, use the following command:

```
vxibc -g <diskgroup> unfreeze <application_name> <rvg>
```

The above command unfreezes the Secondary RVG. This operation must be performed after receiving the IBC message. The unfreeze operation allows the replication to continue by allowing the updates that were performed on data volumes after the send operation, to be applied to the Secondary RVG.

For example, to unfreeze the Secondary RVG:

```
vxibc -g vvrdg1 unfreeze app1 vvrRvg1
```

## Displaying registered application names

To display registered application names, use the following command:

```
vxibc -g <diskgroup> status <rvg>
```

This command displays the currently registered application names for the RVG. If the Secondary RVG is frozen then the `vxibc status` command output displays a message that the Secondary RVG is frozen.

Example:

```
vxibc -g vvrdg1 status vvrRvg1
```

## Registering and sending messages

To register and send messages, run the following command:

```
vxibc -g <diskgroup> [-D <deliver_timeout>] \[-N | -F
 <freeze_timeout>] [-f <filename> | -m <message>] \regsend
 <application_name> <rvg> [<rlink>  ...]
```

This operation registers an application, sends an IBC message, and unregisters the application in one command. You must start the `regrecv` operation on the Secondary node before performing regsend on the Primary node. Otherwise, the Secondary RVG does not have the corresponding registered application name as on the Primary RVG and the IBC message is discarded.

For example, to send an IBC message:

```
vxibc -g vvrdg1 -D 120 -F 120 -f msg.txt regsend app1 vvrRvg1
 rlink1 rlink2
```

This command reads the message from the file and send it to the specified Secondary host.

## Registering and receiving messages

To register and receive messages, run the following command:

```
vxibc -g <diskgroup>  [-R <receive_timeout>] [-f
 <filename>] \regrecv <application_name>  <rvg> <command>
 [<argument> ...]
```

Use this command at the Secondary host to register an application, receive the IBC message, run the command with the specified parameters, unfreeze the Secondary RVG and unregister the application in one single operation.

For example, to receive an IBC message run the command in this way:

```
vxibc -g vvrdg1 -R 120 -f msg.txt regrecv app1 vvrRvg1 backup.exe
```

or

```
vxibc -g vvrdg1 -R 120 -f msg.txt regrecv app1 vvrRvg1
 "backup.exe param1 param2"
```

**Note:** If you want to specify certain parameters then the command and its parameters must be specified within double quotes as shown above.

This command registers, receives, and then run the command, `backup.exe param1 param2` and then unfreeze and unregister the application.

# Example: Using the IBC messaging facility to take snapshots

The following example demonstrates the use of IBC messaging to ensure that the snapshots of the Secondary host volumes are taken at an application-defined consistency point.

In this example, a sample application `APP1` writes some sample files to the Primary data volumes. When the writes are completed, the application sends an IBC message to the Secondary which on receiving the message, executes the `vxrvg snapshot` command to take the snapshot of the Secondary data volumes. The methods given below are described for the following sample Volume Replicator setup.

## Sample setup showing how to take snapshots using the IBC messaging facility

Primary host name: `VVRPRI`

| | |
|---|---|
| `vvr_dg` | Disk Group |
| `vvr_rvg` | Primary RVG |
| `vvr_dv01` | Primary data volume `#1`<br>(Assigned drive letter E: NTFS Formatted) |

| | |
|---|---|
| `vvr_dv02` | Primary data volume `#2`<br>Assigned drive letter F: NTFS Formatted) |
| `vvr_rep_log` | Primary Replicator Log volume |

Secondary host name: `VVRSEC`

| | |
|---|---|
| vvr_dg | Disk Group |
| vvr_rvg | Secondary RVG |
| `vvr_dv01` | Secondary data volume `#1` |
| `vvr_dv02` | Secondary data volume `#2` |
| `vvr_rep_log` | Secondary Replicator Log volume |

The above example makes the following assumptions that the Secondary is attached and connected. The time-out values for various `vxibc` command options are arbitrarily chosen.

**To take a snapshot of the Secondary at an application-defined consistency interval**

**1** Prepare the volumes on Secondary using the following command:

```
vxassist -g vvr_dg prepare vvr_dv01
vxassist -g vvr_dg prepare vvr_dv02
```

To be able to create disk group split friendly snapshots, make sure that the snapshots are created on separate disks that do not contain the RVG objects.

**2** On the Secondary, wait for the IBC message from the Primary whose application is registered by the name `APP1`. Indicate that the `vxrvg snapshot` command should be executed on receiving this message using the command:

```
vxibc -g vvr_dg -R 300 regrecv APP1 vvr_rvg "vxrvg
 -g vvr_dg -f-P snap snapshot vv_rvg"
```

The command prompt is not available, unless the IBC message is received from the Primary or receive time-out (after 300 seconds) has occurred.

**3** On Primary, put the application into a consistent state after making sure that data is flushed from the cache to volumes using the command:

```
vxrvg dismount vvr_rvg
```

4 Send an IBC message to the Secondary, informing it that the application level consistency is achieved at the Primary and that the Secondary can now take a snapshot:

```
vxibc -g vvr_dg regsend APP1 vvr_rvg
```

5 On receiving this message, the Secondary side `vxibc regrecv` command that is awaiting this message in Step 2 comes out after creating the snapshots using the `snapshot` command.

6 You can now use the snapshot volumes on the Secondary for performing any tasks.

Chapter 11

# Setting up replication in cloud environments

This chapter includes the following topics:

## About data replication in cloud environments

Veritas replication technology in tandem with cloud services offer scalable, cost-effective disaster recovery options for your business. With Veritas volume replication, you can leverage the cloud as a DR site and replicate application data to or within the cloud without incurring the infrastructural costs needed to maintain a second physical site.

You can replicate data across Availability Zones/User Defined Sites (terminology may vary with the cloud vendor) and regions.

In the event of failures—node, storage, or Availability Zone— data will always be available.

## About the supported replication scenarios

In a cloud environment, you can use the Veritas replication technology to replicate data in any of the following scenarios:

### Replication from on-premise to cloud

In this scenario, replication is set between an on-premises data center to an on-cloud data center. The on-premise data center can be configured as a primary site and the on-cloud data center as a secondary site.

### Replication within a region

In this scenario, replication is set between two on-cloud data centers that are located in different or same Availability Zones/VNETs (terminology varies with the vendor), within a same region. In the event of an Availability Zone/VNET failure, VVR replicates data from one Availability Zone/VNET to the second Availability Zone/VNET.

### Replication across regions

In this scenario, replication is set between two on-cloud data centers that are located in two different regions. You can configure a primary site in one region and a secondary site in another region.

### Replication across cloud

In this scenario, replication is between Azure and AWS cloud. You can use one of these clouds as a primary site and the second cloud as a DR site.

# Replicating in AWS cloud environment

The deployment steps for setting up replication in an AWS cloud depends on the location where your primary data center and your secondary data center is located.

Depending on your deployment setup, follow the steps provided:

- Replication from on-premise data center to on-cloud data center
  See "Setting up replication from on-premise to AWS cloud" on page 388.
- Replication across Availability Zones within the same region
  See "Setting up replication across Availability Zones (AZ) within the same region" on page 390.
- Replication across regions
  See "Setting up replication across AWS regions" on page 392.

## Setting up replication from on-premise to AWS cloud

In this scenario, data is replicated from an on-premise data center to an on-cloud data center.

## Pre-requisites

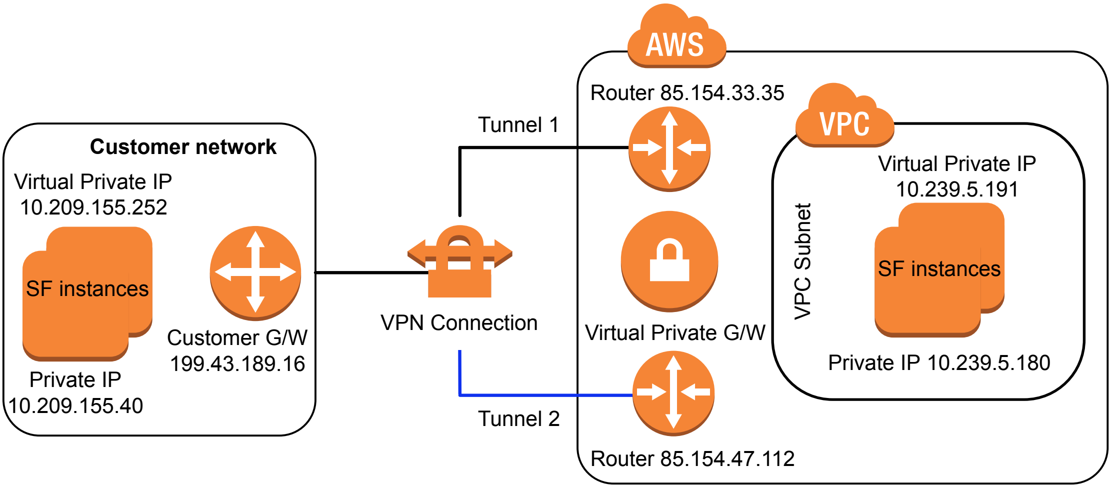
You must meet the following requirements before setting up replication from an on-premise data center to an on-cloud data center.

- Required ports are open for communication between an on-premise data center and an on-cloud data center.
  See "InfoScale ports and services" on page 425.
- The virtual private IP addresses are plumbed on both the nodes.
- The virtual private IP addresses are configured within the subnets.

## Sample configuration

The following diagram illustrates a sample configuration for setting up replication from an on-premise data center to an on-cloud data center.

**Figure 11-1** Replication from on-premise to on-cloud data center



## Setting up replication

Perform the steps in the following procedure to set up replication from an on-premise to an on-cloud data center.

**To set up replication from on-premise to on-cloud data centers**

1. Using AWS portal, create a Virtual Private Cloud (VPC) with a valid CIDR block, for example 10.239.0.0/16.
2. Create a subnet in the VPC created and assign it to an Availability Zone.
3. Create an EC2 instance and associate it with the VPC created.

4. Configure a virtual private gateway and associate it to the VPC.
5. Configure a gateway in the on-premise data center.
6. Create route table entries.
7. Associate the subnet with the route table.
8. Enable route propagation to automatically propagate the routes to the table.

   On the **Route Propagation** tab in the details pane, choose **Edit**, and select the virtual private gateway that you created.
9. Create a VPN connection.
10. Download the VPN configuration file.
11. Create a VPN tunnel between the on-premise network and the on-cloud network.
12. Install InfoScale Storage/InfoScale Enterprise on EC2 instances in both the data centers
13. Create VxVM disk groups, VxVM volumes, Replicated Volume Group (RVG), and RLinks.
14. Set up replication between the on-premise and on-cloud instances.
15. Verify the status of replication.

    ```
    # vradmin -g dg_name repstatus rvg_name
    ```

    Ensure that the replication status shows:

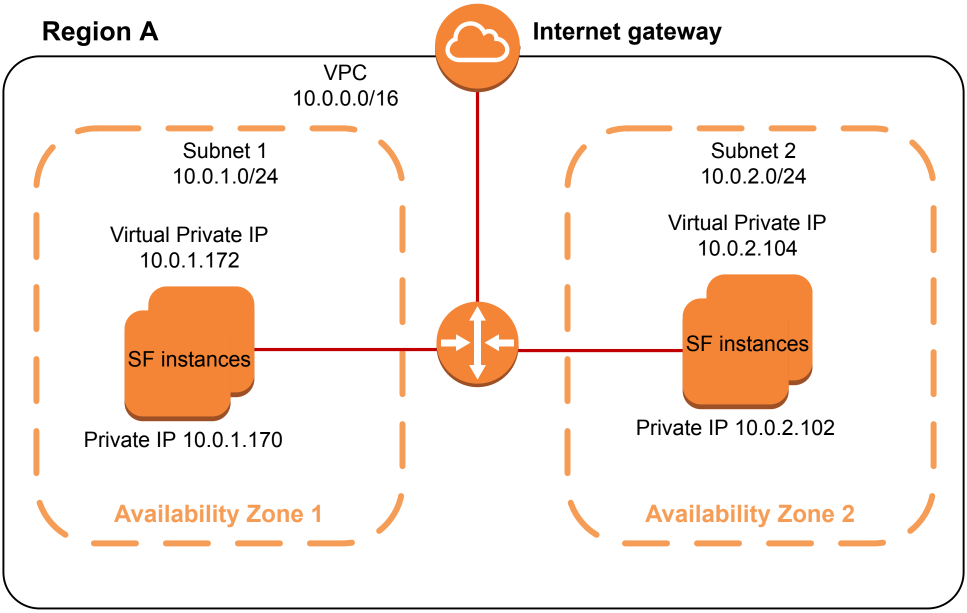
    ```
    Replication status: replicating (connected)
    ```

## Setting up replication across Availability Zones (AZ) within the same region

In this scenario, data is replicated between two on-cloud data centers that are located in different Availability Zones within a same region.

The following diagram illustrates a sample configuration for setting up replication between data centers that are located in different Availability Zones within a same region.

**Figure 11-2** Replication across Availability Zones



## Setting up replication across AZs within the same region

Perform the steps in the following procedure to set up replication across AZs within the same region.

**To set up replication across AZs within the same region**

1. Create a VPC with a valid CIDR block, for example, 10.0.0.0/16.
2. Create the internet gateway and attach it to the VPC.
3. Modify the VPC route table such that the two instances across availability zones can communicate with each other using private IP addresses.
4. Create two subnets—one subnet for the primary site in AZ1 and the second subnet for the secondary site in AZ2 with valid CIDR range, for example 10.0.1.0/24 and 10.0.2.0/24 respectively.
5. Launch the EC2 instances in the primary and secondary subnets. Install Veritas InfoScale on the instances.
6. Verify connectivity between the virtual private IP addresses of the instances.

   ```
   # ping PIP
   ```

7 Install InfoScale Storage/InfoScale Enterprise on EC2 instances in both the data centers

8 Create VxVM disk groups, VxVM volumes, Replicated Volume Group (RVG), and RLinks.

9 Set up replication between the instances using private IP address or virtual private IP address.

10 Verify the status of replication.

```
# vradmin -g dg_name repstatus rvg_name
```

Ensure that the status shows:

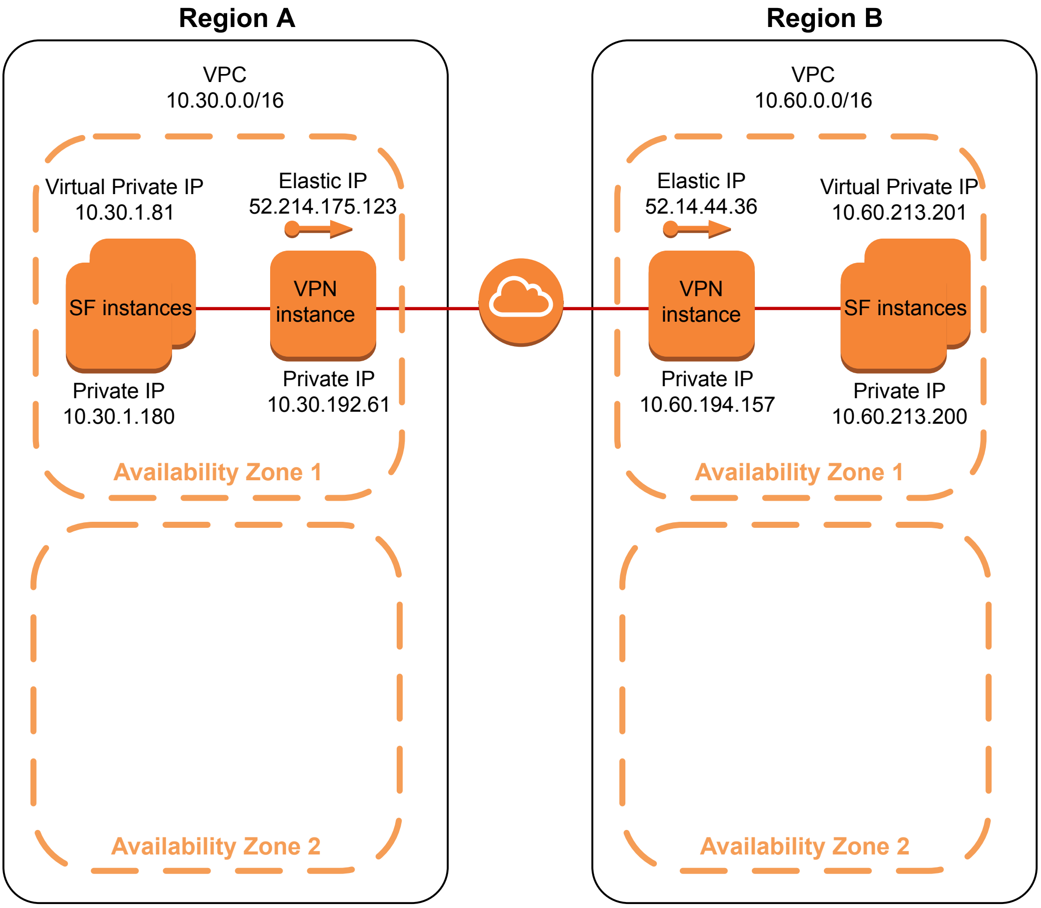
```
Replication status: replicating (connected)
```

## Setting up replication across AWS regions

In this scenario, replication is set up across Availability Zones configured in different regions. The configuration uses software VPN `Openswan/OpenVPN` to connect the VPCs across different regions.

The following diagram illustrates a sample configuration for setting up replication across regions.

**Figure 11-3** Replication across regions



## Setting up replication across regions

Perform the steps in the following procedure to set up replication across regions.

**To set up replication across regions**

1. Create two VPCs with valid CIDR blocks in different regions, for example, 10.30.0.0/16 and 10.60.0.0/16 respectively.
2. Create a primary site EC2 instance.
3. Create a primary site VPN instance, which belongs to the same VPC as that of the primary EC2 instance.

4. Modify the route table on the primary site. Ensure that the route table entry directs the secondary site traffic through the primary site VPN instance.
5. Create a secondary site EC2 instance.
6. Create a secondary site VPN instance, which belongs to the same VPC as that of the secondary EC2 instance.
7. Modify the route table on the secondary site. Ensure that the route table entry directs the primary site traffic through the secondary site VPN instance.
8. Set up connectivity across regions using software VPN.
9. Install InfoScale Storage/InfoScale Enterprise on EC2 instances in both the data centers
10. Create VxVM disk groups, VxVM volumes, Replicated Volume Group (RVG), and RLinks.
11. Set up replication between the instances using the private IP address or virtual private IP address.
12. Verify the status of replication.

    ```
    # vradmin -g dg_name repstatus rvg_name
    ```

    Ensure that the replication status shows:

    ```
    Replication status: replicating (connected)
    ```

# Replicating in Azure cloud environment

The deployment steps for setting up replication in an Azure cloud depends on the location where your primary data center and your secondary data center is located.

Depending on your deployment setup, follow the steps provided:

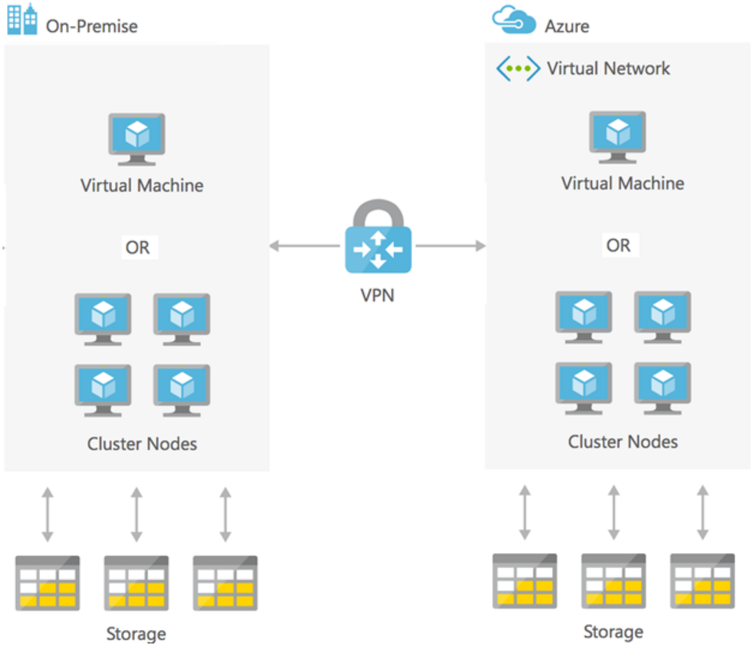
- Replication from on-premise data center to on-cloud data center
  See “Setting up replication from on-premise to Azure cloud” on page 395.
- Replication within or across VNet in the same region
  See “Setting up replication within same Azure region” on page 397.
- Replication across regions
  See “Setting up replication across Azure regions” on page 400.

# Setting up replication from on-premise to Azure cloud

The following diagram illustrates the sample configuration for setting up replication between an on-premise data center to Azure cloud (on-cloud data center):

**Note:** For the ease of use, the machines, whether virtual or physical are commonly mentioned as virtual machines. Your on-premise data center may include physical machines instead of the virtual machines. In any case, the steps to set up replication from on-premise data center to on-cloud data center remain the same for physical and well as virtual machines.

**Figure 11-4** Sample configuration for setting up replication between an on-premise data center to on-cloud data center



## About setting up replication between an on-premise data center to on-cloud data center

Replication between an on-premise data center to on-cloud data center involves the following high-level steps:

1. Prepare the setup at on-premise data center
2. Prepare the setup at on-cloud data center
3. Establish a tunnel from on-premise data center to on-cloud data center

4. Deploy setup

The following sections provide details about performing each of these steps.

## Preparing the setup at on-premise data center

**Perform the following steps to prepare the setup at on-premise data center:**

1. Enable the ports that are used for inbound and outbound communication.

   See "InfoScale ports and services" on page 425.

2. Create a subnet and a local VPN gateway.

3. Note the address space that is allotted for the subnet and the public IP address that is allotted for the local VPN gateway.

## Preparing the setup at on-cloud data center

**Perform the following steps to prepare the setup at on-cloud data center:**

1. Using Microsoft Azure portal, create a resource group.

2. Create a VNet in the resource group created and specify an IP address space for the VNet.

   The IP address range must be diff on the on-premise subnet and on the on-cloud subnet.

3. Create a gateway subnet.

4. Create a VPN gateway and associate it with the created VNet.

   Note the public IP address that is allotted for the on-cloud VPN gateway.

5. Create a local network gateway.

   When you create the local network gateway, you must provide the on-premise subnet IP address range and the public IP address of the on-premise local VPN gateway.

6. Establish a tunnel from on-cloud to on-premise network.

   To establish the tunnel, create a connection of type **Site-to-Site (IPSec)** and choose the on-cloud VPN gateway and the local network gateway.

7. Provide a shared key (alpha-numeric key).

   A shared key is a pass-phrase. This pass-phrase is required when you establish a tunnel from on-premise data center to cloud data center.

### Establishing a tunnel from on-premise data center to on-cloud data center

To establish a tunnel from on-premise data center to on-cloud data center, use the following parameters:

- Public IP address that is allotted for the on-cloud VPN gateway
- Shared key (alpha-numeric key) that was provided while establishing a tunnel from on-cloud data center to on-premise data center
- On-cloud VPN gateway configuration type (Policy based or Route based)

### Deploying the setup

**Perform the following steps to deploy the setup (in both the data centers):**

1. Create virtual machines in the subnets created.
2. Provision storage.
3. Install InfoScale Storage/InfoScale Enterprise.
4. Create VxVM disk groups, VxVM volumes, Replicated Volume Group (RVG), and RLinks.
5. Flush the iptables on both the virtual machines.

   ```
   # iptable -F
   ```

6. Set up replication between the virtual machines using the private IP address or the virtual IP address.
7. Verify the replication status.

   ```
   # vradmin -g dg_name repstatus rvg_name
   ```

   Ensure that the replication status shows:

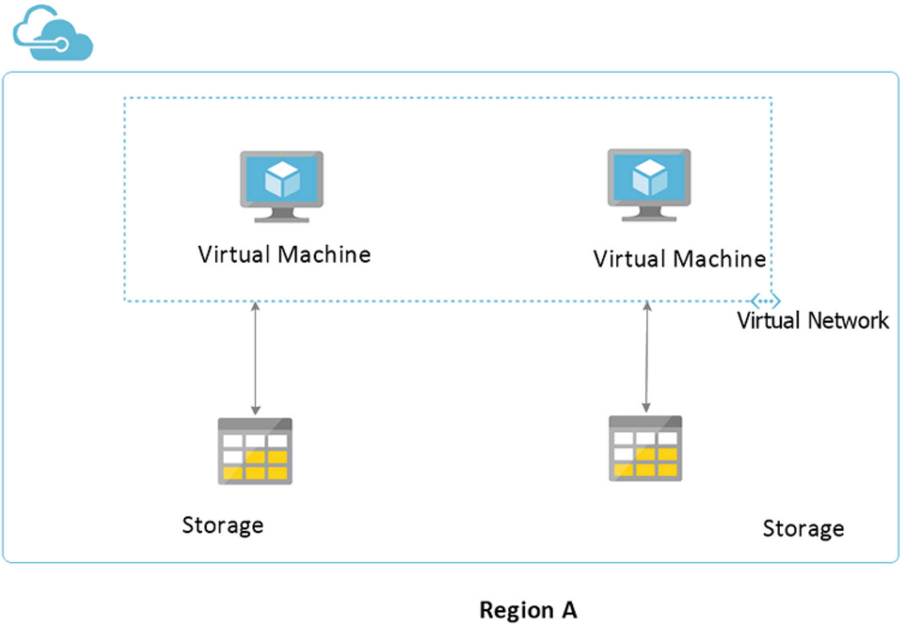
   ```
   Replication status: replicating (connected)
   ```

## Setting up replication within same Azure region

In an Azure cloud environment, in a single region, you can provision your setup across virtual networks (VNets) or within a VNet.

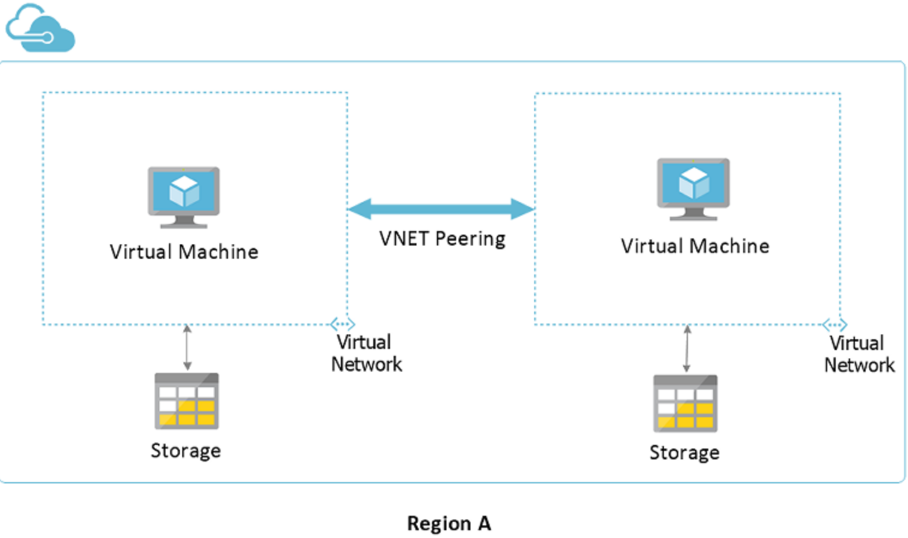
The following diagram illustrates the sample configuration for setting up replication between the same VNet:

**Figure 11-5** Sample configuration for setting up replication between same VNet



The following diagram illustrates the sample configuration for setting up replication across VNets:

**Figure 11-6** Sample configuration for setting up replication across VNets



## Setting up replication in the same VNet

### Perform the following steps to set up replication in the same VNet, within the same region

1. Enable the ports that are used for inbound and outbound communication.

   See "InfoScale ports and services" on page 425.

2. Using Microsoft Azure portal, create a VNet and specify an IP address space for the VNet.

3. Create a subnet in the VNet created.

   For details about creating a VNet, specifying an IP address space, and creating a subnet, refer to Microsoft documentation.

4. Create two virtual machines within the subnets and provision storage.

5. Install InfoScale Storage/InfoScale Enterprise on both the virtual machines.

6. Create VxVM disk groups, VxVM volumes, Storage Replicator Log (SRL), Replicated Volume Group (RVG), and RLinks.

7. Flush the iptables on both the virtual machines.

   ```
   # iptable -F
   ```

8 Set up replication between the virtual machines using the private IP address or the virtual IP address.

9 Verify the replication status.

```
# vradmin -g dg_name repstatus rvg_name
```

Ensure that the replication status shows:

```
Replication status: replicating (connected)
```

## Setting up replication across VNet

**Perform the following steps to set up replication across the VNets, within the same region**

1 Using Microsoft Azure portal, create two VNets and specify an IP address space for each VNet.

2 Set up VNet Peering between the two VNets.

3 Create a subnet in each VNet.

4 Create a virtual machine in each subnet and provision storage.

5 Install InfoScale Storage/InfoScale Enterprise on both the virtual machines.

6 Create VxVM disk groups, VxVM volumes, Storage Replicator Log (SRL), Replicated Volume Group (RVG), and RLinks.

7 Flush the iptables on both the virtual machines.

```
# iptable -F
```

8 Set up replication between the virtual machines using the private IP address or the virtual IP address.

9 Verify the replication status.

```
# vradmin -g dg_name repstatus rvg_name
```

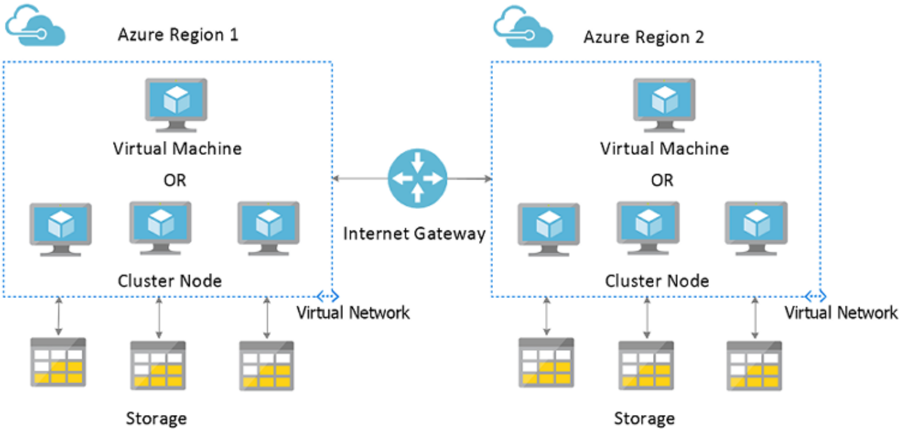
Ensure that the replication status shows:

```
Replication status: replicating (connected)
```

# Setting up replication across Azure regions

The following diagram illustrates the sample configuration for setting up replication across regions:

**Figure 11-7** Sample configuration for setting up replication across regions



## Setting up replication across regions

### Perform the following steps to set up replication across region

1 Enable the ports that are used for inbound and outbound communication.

See "InfoScale ports and services" on page 425.

2 Using Azure portal, create a resource group (RG) in both the regions.

3 Create a VNet in each region and specify a non-overlapping IP address space for each VNet.

4 Create a subnet and a gateway subnet under the VNets created in both the regions.

5 Create a virtual network gateway in both the regions.

6 Establish a connection in between the two virtual network gateways.

7 Create virtual machines in the subnets and provision storage.

8 Install InfoScale Storage/InfoScale Enterprise.

9 Create VxVM disk groups, VxVM volumes, Replicated Volume Group (RVG), and RLinks.

10 Flush the iptables on both the virtual machines.

```
# iptable -F
```

**11** Set up replication using the private IP address or the virtual IP address.

**12** Verify the replication status.

```
# vradmin -g dg_name repstatus rvg_name
```

Ensure that the replication status shows:

```
Replication status: replicating (connected)
```

# Replicating in cross-cloud environment

Replication across cloud is similar to setting up replication from an on-premise data center to an on-cloud datacenter, were one of the cloud can be considered as an on-premise data center and the second as an on-cloud data center.

It involves the following high-level steps:

1. Prepare the deployment setup in each cloud.

   See “Setting up replication from on-premise to Azure cloud” on page 395.

   See “Setting up replication from on-premise to AWS cloud” on page 388.

2. Set up a communication channel between the clouds. For example, you can set up a virtual network gateway or can configure a VPN-based connection.

3. Install InfoScale Storage/InfoScale Enterprise and configure VVR on the virtual machines created.

   For details about installing InfoScale Storage/InfoScale Enterprise, refer to *Veritas InfoScale Installation and Upgrade Guide*.

   For details about configuring replication, refer to *Veritas Volume Replicator Administrator's Guide*.

# Troubleshooting Volume Replicator

This chapter includes the following topics:

## About troubleshooting Volume Replicator

This chapter describes the process of recovering from various error conditions that may occur when replicating in a Volume Replicator environment. Recommendations and checks that can help in preventing errors are also provided.

## Recommendations and checks

This section describes some recommendations and checks that help you in avoiding some errors when working with the Volume Replicator.

See “Encrypted files on replicated volumes” on page 404.

See “Selecting the mode of replication ” on page 404.

See “Volume Replicator issues when Norton AntiVirus scan is performed” on page 405.

See “Monitor view does not display the RDS information” on page 405.

See “Preventing the connect problems” on page 406.

See “Configuration checks for RLINKS” on page 406.

See “Network, process, and operating system checks” on page 407.

See “Configuration checks for volume mappings” on page 408.

See “Troubleshooting Volume Replicator performance” on page 408.

See “Other information and checks” on page 410.

## Encrypted files on replicated volumes

Avoid using encrypted files in replicated volumes. Veritas recommends that you use secure networks, which can include private network, hardware assisted encryption, or other secured means to replicate your data.

Volume Replicator by design does not encrypt the data before replication. However, if you have encrypted data on the replicated volumes, Volume Replicator replicates this data, as is. The volume replication using Volume Replicator maintains exact replica (byte by byte) of the volume under replication, irrespective of the file system on the volume.

Since the decryption or re-encryption of an encrypted file requires write permissions and also the availability of the public or private keys, the encrypted file accessibility is limited to the local system or domain of the system. The Volume Replicator replication services doesn't provide any support or services for managing accessibility of the encrypted files on replicated volumes on the remote host.

However, if you have created encrypted file systems (EFS), it is possible to use the Certificate Export wizard and the Certificate Import wizard to transfer your certificate and private key to your user profile on the other computer.

## Selecting the mode of replication

As far as possible use the default synchronous override mode of replication. In synchronous override mode as long as the network is connected, the replication would be in synchronous mode and in cases of network failure, the updates can go to the Replicator Log without failing the updates.

If you need to use only the synchronous mode, then in cases when the network cannot be recovered, change the replication mode later to resolve the problem.

If an RVG has NTFS or ReFS mounted volumes and one of its associated Secondary hosts is in synchronous mode of replication, then in the case of a network failure all the writes to the replicated volume fail. The system may display a `Delayed write failed` error and even freeze or hang.

## Volume Replicator issues when Norton AntiVirus scan is performed

When Norton AntiVirus (NAV) is installed on a Volume Replicator host that has a Primary RVG with heavy updates, then, after some hours of antivirus scan for virus detection, the Volume Replicator host may experience a system freeze, hang, or be slow in responding to user interactions. This can also affect the behavior of the Volume Replicator replication services, such as Secondary disconnection or loss of data when virus scans are scheduled on the Primary host.

This can be because of a known issue with few versions of Norton AntiVirus (including Corporate Edition, version 7.0x) that causes kernel memory leaks during its virus scans. Such kernel memory leaks would severely degrade the system performance causing the system to become unstable, slow in responding, and at times may even cause a system crash.

Currently, Volume Replicator is not fully tested with edition 7.51 to ascertain that it is completely free from kernel memory leaks.

Refer *Norton Antivirus Release Notes* for details.

Schedule the Norton AntiVirus (NAV) scan at times when Volume Replicator and the application workload is lean.

Veritas recommends that to avoid such issues on a Volume Replicator host that has Norton AntiVirus installed, the scanning time of Norton AntiVirus should be scheduled in such a way that it should not overlap Volume Replicator updates on the Primary RVG.

When such a situation occurs, restart the system to recover Volume Replicator.

## Monitor view does not display the RDS information

The monitor view may not display the RDS information rows if the statistics information is unavailable or inadequate.

The reasons for this are as follows:

- The Primary RVG may be unavailable if it was deleted after the RDS was formed, or if the disk group of the RVG has been deported, or is failing over in the cluster.
- The statistical data cannot be exchanged within the Volume Replicator setup because the Primary host is down or the network between the Primary and Secondary is down.
- The Primary RVG has no Replicator Log.
  See “Monitoring replication using the VEA console” on page 131.

## Preventing the connect problems

The Secondary host may not connect for many reasons.

You can diagnose whether the Secondary is connected or not in the following ways:

- Secondary hosts stay in the `Activating` state, as displayed in VEA when replication is started.
- Secondary is `Primary paused` after the replication is started.
- The Primary RVG has a `Resynchronization paused` state.
- The replication status is `Active`, but there is no replication taking place.

## Configuration checks for RLINKS

The following sections give a checklist that can be used to troubleshoot the RLINKs that are not connected.

You may need to do certain configuration checks if you have created an RDS or changed some setting in the configuration through the CLI.

You may need to perform the following:

- On all nodes participating in the replication, run the following command:

  ```
  vxprint -lPV
  ```

  In the output, check for the following:

- Check whether the RLINKs are active or stale. For replication to begin they must be active.
- Primary `remote_rlink = Secondary rlink name`.
- Secondary `remote_rlink = Primary rlink name`.
- Primary `remote_dg = Secondary dg`.
- Secondary `remote_dg = Primary dg`.
- Primary `local_host = Secondary remote_host`.
- Secondary `local_host = Primary remote_host`.
- Primary `remote_host IP` is indeed the IP of the Secondary host.
- Secondary `remote_host IP` is indeed the IP of the Primary host.
- Verify that the Primary RLINK is ACTIVE.

## Network, process, and operating system checks

General problems like high latency, low-bandwidth, high collisions counts, and a high percentage of dropped packets also affects Volume Replicator.

Specific issues with networks are as follows:

- Check the status of communication between the Primary and Secondary nodes by using the replication path. To do this ping from Primary to Secondary and Secondary to Primary using remote_host fields in the RLINKs. There should be very minimal packet loss, if any.

  ```
  Run: ping <remote_host>Run: ping <remote_host_ip_address>
  ```

- Confirm whether the network can handle large packets using the `ping` command. The packet loss should be similar to that mentioned in the preceding point. In the `ping` command, specify `packet_size` value that the `vxprint -l <rlink-name>` command displays.

  ```
  Run: ping -l <packet_size> <remote_host>
  Run: ping -l <packet_size> <remote_ip_address>
  ```

- Check whether the connection server is started or not. You can confirm this, by checking the system event log. You should see an entry similar to the one given below:

  ```
  Connection Server started successfully (using port 6ae).
  ```

  If you do not see this entry, make sure that `vxsvc` service for the Veritas Enterprise Administration (VEA) is started. If it is not started then start the service, and check the log again.
  In the system event log, you may see entries similar to `Connection Server already started`. These messages do not indicate any problems.

- Run the following command on each node to make sure that the Volume Replicator connection server uses the port that is mentioned in the `vrport heartbeat` command.

  ```
  netstat -an | findstr <port-number mentioned in
   vrportheartbeat' output>
  ```

  The default port number is 4145. Check the output of the `vrport` command.
  See "Displaying or setting ports for heartbeats" on page 303.

## Configuration checks for volume mappings

Volume mapping errors can be displayed when starting replication, that is attaching the Primary RLINKs.

Configuration checks for volume mappings are as follows:

- Make sure that for each data volume that is associated with the Primary RVG, there is a corresponding Secondary volume associated with the Secondary RVG.
- Make sure that the size of each Secondary data volume is the same as the corresponding Primary data volume. The sizes should be the same in sectors or bytes. You can find the size of the volume in sectors using the Storage Foundation. To do this, select the volume and right-click. Select Properties > Size in Sectors to view the size of the volume in sectors. Alternatively, you can also run the following command from the command prompt:

  ```
  vxvol volinfo <volume name or drive letter>
  ```

## Troubleshooting Volume Replicator performance

To troubleshoot Volume Replicator performance and improve replication, you can perform certain checks which are explained below.

**To calculate, check, and improve the replication performance**

1. When the replication is active run the following command at the command prompt. Make sure that you run this command only on the Primary.

   ```
   vxrlink -i 5 stats <rlink_name>
   ```

   Note the values in the `Blocks` column. This value indicates the number of blocks that have been successfully sent to the remote node.

2. Compute replication throughput using the following formula:

   ```
   ((# of blocks sent successfully * block size) / stats interval)
   / 1024) KB.
   ```

   where block size is 512 bytes.

   Stats interval is the value of the time interval that is specified for the `-i` parameter with the `vxrlink stats` command. In the command example the time interval is 5 seconds.

3. If the throughput computed in step 2 above is not equivalent to the expected throughput, then do the following:

- Check if the DCM is active by checking the flags field in the output of the following command:

  ```
  vxprint -lPV
  ```

  If DCM is active, run the following command to resume replication:

  ```
  vxrvg -g <diskgroup> resync <rvg>
  ```

  You can also perform the Resync operation from VEA by selecting the **Resynchronize Secondaries** option from the Primary RVG right-click menu. Note that the Secondary becomes inconsistent during the DCM replay.

- Check if there are any pending writes using the following command:

  ```
  vxrlink -i 5 status <rlink_name>
  ```

  If the application is not write intensive it is possible that the RLINK is mostly up-to-date, and there are not many pending updates to be sent to Secondary.

  To determine the number of writes that are happening to the data volumes run the Performance Monitor tool. This tool is generally installed when the operating system is installed.

  To launch the tool run `perfmon` from the command prompt. This launches the performance monitor. Select the **(+)** button to launch the Add Counters dialog. Select **Dynamic Volume** from the Performance Object drop-down list and select the **Write Block/Sec** from the Select counters from list pane.

- If there are pending writes in the Replicator Log, and replication is not using the expected bandwidth, check the `Timeout`, `Stream`, and `Memory` error columns in the output of the `vxrlink stats` command.

  If the number of time-out errors are high and the UDP protocol is used for replication, perform the following:

  If the network has a time relay component, change the replication packet size using the following command, to reduce the number of time-out errors and improve the replication throughput:

  ```
  vxrlink set packet_size=1400 <rlink_name>
  ```

  Some components in the network drop UDP packets larger than the MTU size, suspecting a denial of service (DoS) attack. Changing replication packet size to 1K should improve the performance in this case.

- If there are a number of memory errors, perform the following:

  Run the `vxtune` command. The output of the command displays the default values that are set for the following tunables:

```
C:\Documents and Settings\administrator.INDSSMG>vxtune
vol_max_nmpool_sz = 16384 kilobytes
vol_max_rdback_sz = 8192 kilobytes
vol_min_lowmem_sz = 1024 kilobytes
vol_rvio_maxpool_sz = 32768 kilobytes
compression_window = 0 kilobytes
max_tcp_conn_count = 64
nmcom_max_msgs = 512
max_rcvgap = 5
rlink_rdbklimit = 16384 kilobytes
compression_speed = 7
compression_threads = 10
msgq_sequence = 1
vol_maxkiocount = 1048576
force_max_conn = False
tcp_src_port_restrict = False
nat_support = False
```

Change the value of the NMCOM_POOL_SIZE (vol_max_nmpool_sz) tunable appropriately. The default (and minimum) value is 4192 (4MB) and maximum is 524288 (512MB).

After changing this value, restart the system so that the changes take effect.

Note that the value that is specified for the `NMCOM_POOL_SIZE` tunable is global to the system. Thus, if the node is a Secondary for two RLINKs (Primary hosts) then the value of the tunable must be set accordingly.

## Other information and checks

General information and checks in case of an error or problem on the Secondary data volumes can be done.

You can perform the following in case of a problem:

- If there is a problem on any of the Secondary data volumes, such as, a failure of the disk on which the Secondary data volumes are present, the corresponding Primary RLINK goes into FAIL state. You can check the replication status of the Secondary through the VEA console. In this case the VEA Secondary RVG view indicates replication status as FAILED, `Configuration error`. A Secondary represents one side of the RLINK. You can check the status by running the following command:

  ```
  vxprint -lPV
  ```

- If there is a configuration error where the Primary data volumes are larger than the corresponding Secondary volumes, then, the Secondary goes into the Secondary paused, `Secondary_config_err` state. The VEA for the Secondary RVG indicates replication status as `Failed`, `Configuration error` in this case. You can also check this in the Secondary RVG view or by running the following command:

  ```
  vxprint -lPV
  ```

  To verify whether the Secondary has gone into the configuration error state use the `vxrlink verify` command.

# Recovering from problems in a firewall or NAT setup

This section provides troubleshooting tips to recover from the problems that may occur when configuring replication in a firewall or NAT setup.

## Errors when replicating across a firewall

You may get the following error message when trying to replicate across a firewall:

```
Operation timed out. The configuration server may be busy or down.
```

When setting up replication across some firewalls, if the packet size is not set to 1400 bytes or 1 KB, you may encounter some errors. For example, when performing the Automatic Synchronization operation or changing the packet size you may see this message.

First check the firewall settings and the logs to verify if the packets are dropped, because the packet size exceeds the required value.

To fix the problem you may want to delete the RDS and recreate it. However, before doing so you must ensure that the firewall configuration is completed as required and the necessary ports have been opened.

To avoid such problems, when creating an RDS using the wizard, set the packet size to 1KB or 1400 bytes (default). If you still face the problem set the packet size to 1300 bytes.

# Recovering from problems during replication

This section provides troubleshooting tips to recover from the problems that may occur when configuring replication or performing Volume Replicator operations.

See “Permission denied errors when performing Volume Replicator Operations” on page 412.

See “Error when configuring the VxSAS Service” on page 414.

See “VEA Service is not started” on page 415.

See “Connecting to cluster having multiple IP addresses” on page 415.

See “Error when disabling data access to the RVG, creating Secondary RVG, adding volumes” on page 416.

See “Error when resizing volumes” on page 417.

See “Replica link already exists” on page 417.

See “Unable to perform delete RDS, add volume, delete volume” on page 418.

See “Removing the Replicator Log volume mirror” on page 418.

See “Pausing when writes are in progress” on page 419.

See “Unable to see volume name for associating Replicator Log” on page 419.

See “Unable to see the volume names for adding volumes to RDS” on page 419.

See “Adding logs to dissociated volumes” on page 420.

See “Using two commands in succession” on page 420.

See “Renaming dynamic disk group while importing” on page 421.

See “Problems when performing the snapshot operation” on page 422.

See “Operation time-out errors” on page 423.

## Permission denied errors when performing Volume Replicator Operations

You may get permission denied errors while Volume Replicator operations are carried out:

```
Failed to authenticate user credentials. Please verify the
 vxsasservice is running in proper account on all hosts in RDS.
```

This error can occur if the VxSAS service is not started, or, if it has been started using a logon account that is not valid as a `local administrator` on some of the Volume Replicator hosts, participating in the command.

An RDS is the logical unit that groups the RVGs present on different (local and remote) hosts. Volume Replicator uses the VxSAS service logon account as the account to be authenticated while performing remote RDS configuration operations. Volume Replicator provides many RDS-wide operations that can perform simultaneous updates of Volume Replicator configuration on multiple hosts. These operations can be initiated from the Primary or the Secondary, and can be successful only when the logon account(s) of the local host’s VxSAS service and Primary host's VxSAS service (if that is not the local host) has administrative privileges on all the remote participating hosts, failing which you may get this error.

To fix the problem, use the Volume Replicator Security Service Configuration Wizard to configure the VxSAS service remotely on multiple hosts from a single host.

Launch the wizard from **Start > All Programs > Veritas > Veritas Storage Foundation > Configuration Wizards > VVR Security Service Configuration Wizard** or, on Windows 2012 operating systems, from the **Apps** menu in the **Start** screen.

Optionally, run `vxsascfg.exe` from the command prompt.

Veritas recommends that you configure the VxSAS service now. Once correctly configured, it is not necessary to reconfigure a host unless you want to change the account name or the password.

When configuring the VxSAS service account make sure that all the hosts participating or being added in the RDS are configured, using an account that has administrative rights on all the other hosts. Another way to configure the `vxsas` service is through the Service Control Manager.

**To configure the VxSAS service directly from the Service Control Manager**

1. Select **Start > Settings > Control Panel > Administrative Tools > Services** or, on Windows 2012 operating systems, on the **Start** screen, click **Control Panel**.

   Select VxSAS service and right-click. Choose the **Properties** option from the menu that appears.

2. Click on the **Log On** tab and select the **This account** option. Specify your administrative password in the **This account** field and your password if any in the **password** field. Click **OK** to effect these changes.

3. Restart the `VxSAS` service.

4. Perform these steps for each host that is intended to be a part of the replication configuration.

   For cluster setups, Veritas recommends that each node of the cluster should share the same VxSAS logon account. This can either be a domain account that has been configured as a member of the local administrators group in the local security policy of each node, or a local administrative account that is configured with the same name and password on each node.

# Error when configuring the VxSAS Service

When configuring the VxSAS service, you may get the following error message:

```
Could not start the service due to logon failure.
```

If you try to configure the VxSAS service using an account that has administrative privileges, but does not have Log on as a service privilege, you may get this error message.

On Windows Server, the Log-on as a service privilege is not automatically updated for the `administrator` user account. Hence, on fresh setup, no service, including VxSAS, is able to log on using any local administrative account. Trying to do so can result in this error.

You can choose to configure the VxSAS service at any time by typing the command `vxsascfg.exe` at the command prompt. However, before invoking the utility, make sure that the account that is used as the logon account for VxSAS service, must have the Log on as a service privilege on the systems selected for configuration.

To fix the problem add the Log-on as a service privilege to the accounts that belong to the `Administrators` group. The VxSAS security service configuration wizard tries to add Log on as a service privilege to the specified accounts. However, if this fails, you need to follow the manual procedure that is given below to add the Log-on as a service privilege to the accounts.

**To add the Log-on as service privilege using the local security policy option**

1. Launch Local Security Policy from **Start > Settings > Control Panel > Administrative Tools > Local Security Policy** or, on Windows 2012 operating systems, on the **Start** screen, click **Control Panel**.
2. From the Local Security Settings dialog box, select **Local Policies > User Rights Assignment** from the tree view in the left panel.
3. Double-click on **Log on as a service** option from the right panel, to display the Local Security Policy Setting. In this window add `Administrators` group to the list of users.
4. Click **OK** to complete the procedure.

## Configuring the VxSAS using the Service Control Manager

You can also choose to configure the VxSAS service using the Service Control Manager.

**To configure VxSAS through Service Control Manager**

1. Select **Start > Settings > Control Panel > Administrative Tools > Services** or, on the **Start** screen, click **Control Panel**.
2. From the Services window select the VxSAS logon account and log on at least once using the appropriate account name and password.

   This manual procedure is equivalent to using the VxSAS configuration wizard, however, if it is done once, it automatically adds the Logs on as a service right to the account and the wizard can be used for successive modifications.

# VEA Service is not started

The following error message is displayed when VEA service is not started:

```
The Veritas Enterprise Administrator Service could not be started.
```

or

```
Could not connect to the server.
```

When using the Command Line Interface, sometimes the Volume Replicator commands may start failing indicating that the VEA service is not started. Sometimes, the `vxsvc` service cannot be started successfully in the first attempt and the error message is displayed.

To resolve this problem stop and restart the `vxsvc` service from the command line using the following commands:

```
net stop vxsvc
net start vxsvc
```

You can also use the Service Control Manager (SCM) to restart the Veritas Enterprise Administrator service.

# Connecting to cluster having multiple IP addresses

At any time, VEA can support only one connection to a particular host. When VEA is connected to a cluster node using virtual IP addresses or the corresponding network names (virtual server), it may also allow another connection to the same physical host using different IP addresses that the node supports. This behavior is because of the constraints which virtual IP address and its network name has on the queries.

In such cases, VEA cannot identify if multiple connections are made to a same cluster node as a Volume Replicator host. These occurrences should be avoided as it causes ambiguities in identification and other such problems in VEA.

To avoid these multiple connections to the same cluster node, you can do the following:

- Ensure that VEA has only one connection to the cluster node when using cluster virtual IP addresses to connect. If you want to have another connection through VEA to the same host use a separate instance of VEA.
- When setting up the RDS, if the required replication IP address or name does not appear in the list of possible Primary hosts, then disconnect VEA from all connected hosts, then try connecting again.
- If this does not solve the problem, close the VEA and reopen it. Then try connecting to the cluster IP name again.

## Error when disabling data access to the RVG, creating Secondary RVG, adding volumes

The following error message is displayed:

```
Failed to acquire lock on volume. Please close all applications
using volume(s) under replication and try this operation again.
```

This problem may occur when you try to disable data access to the RVG, creating the Secondary RVG, or when adding new volumes to the RVG. These operations first try to lock all the volumes under RVG. This holds true for both the Primary and Secondary RVG volumes.

This error may also occur when performing the migrate operations. These operations internally try to disable data access to the RVG. These operations also require that no application should use the volumes under replication.

Disabling data access to the RVG, creating the Secondary RVG or adding new volumes operations fails if it is unable to lock the volume because of the following reasons:

- If any application or file handles are still open on the volume it cannot be locked
- The volume drive letter should not be accessed through any explorer
- That drive letter should not be active in the command prompts.

**Note:** Use `chkdsk` to forcefully dismount the volumes only in situations where all other recommended actions do not work because the forced dismount causes all open handles on the volume to become invalid and may cause data loss. As a result, the applications that use these volumes may crash.

Workaround for avoiding these error messages are as follows:

- Ensure that the required volumes are not accessed through any of the Explorer windows or the command prompt. Also ensure that the application handles are closed on these volumes.
- Before disabling data access to the RVG the application must be stopped. Ensure that you provide sufficient time for the cached buffers to be flushed before performing these operations.
- You can also use the `vxrvg dismount` command to verify whether disabling or enabling data access succeeds.
- In some rare cases, even after closing all the applications which use the replicated volumes, the volume still can’t be dismounted because of some system or application problem. In this case, forcefully dismount the volume using the `chkdsk /x` command. After forcefully dismounting the volumes, the RVG dismount or disable data access succeeds.

## Error when resizing volumes

You may get the following error while trying to perform the resize volume operation:

```
Failed to extend file system on the volume.
```

You may get the above error when trying to resize the volumes that are part of an RVG, if the RVG has data access disabled. Check the state of the RVG in the RVG view or by running `vxprint -l <rvgName>` command. The command output displays the RVG status as disabled and detached. In this case, the volume is resized, but the file system is not resized as the RVG does not allow any writes to the volume. The resize operation is completed, however, if you run `chkdsk` command on the volume, you get the old volume size.

Enable data access to the RVG and then resize the volume.

## Replica link already exists

You may get the following error while adding a Secondary host to an RDS:

```
Error occurred on host <HOSTNAME>. Replica link for the same
 remote host already exists.
```

When trying to add the Secondary host to the RDS you may sometimes get the above error, if there is an RLINK already associated with the Primary RVG of the RDS and having the remote host field the same as the host, which you want to add as a Secondary. It could also occur if you try to associate an existing RLINK using the Command Line Interface. To check this run the command `vxprint -l <rvgName>` command on Primary where <rvgName> refers to the name of the

Primary RVG of the RDS. In the command output display, you see an RLINK which has the `remote_host` field that is set to the Secondary host which is the same one that you want to add as Secondary.

To solve this problem, at the command prompt of the host for which the error was displayed, run the following command:

```
vxprint -l <rvg>
```

This command displays the RLINKs associated to the specific RVG. Identify the RLINK for the host that you try to add as Secondary using the information in the field `remote_host`. For this RLINK run the command `vxrlink -f rm <rlink>`

Now you can add the Secondary or associate the RLINK.

## Unable to perform delete RDS, add volume, delete volume

The following error is displayed while trying to delete an RDS, add, or delete a volume:

```
Operation failed, target host not responding.
```

If you have lost one of the Secondary hosts permanently (due to disaster, or system failure or such other causes) or attempt certain operations after migration or takeover, above mentioned operations may fail. This is because you have RLINKs to the Secondary hosts which are no longer accessible.

Remove the links to Secondary hosts which are no longer accessible by running the following command on the host on which the error occurred:

```
vxprint -l <rvg>
```

This command displays the RLINKs to the specific RVG. Identify the RLINK to the Secondary hosts which are no longer accessible by using the information in the field `remote_host`. For these RLINKs, run the following command:

```
vxrlink -f rm <rlink>
```

You can now delete the RDS and add or delete the volumes.

## Removing the Replicator Log volume mirror

When a mirrored Replicator Log volume is removed, the following error message may get displayed:

```
Unable to perform operation as the mirror is either regenerating
or is in failed redundant state.
```

When you try to remove the mirrored Replicator Log volume you may get the above error if the mirrored volume is still in the process of being synchronized.

Wait till the resynchronization process is completed. You may be able to remove the mirror after the mirrored Replicator Log volumes are synchronized.

## Pausing when writes are in progress

If you perform the pause operation on the Primary when large number of writes are in progress, then the following error message is displayed:

```
Operation timed out. The configuration server may be busy or down.
```

If you try to pause the Primary when a large number of writes are in progress you may get the above error. This is because the Primary volumes are flushed for pausing and the writes also happen at the same time.

Before performing the disable data access, pause or migrate operations, it is recommended that you run the `vxrvg dismount` command.

## Unable to see volume name for associating Replicator Log

If you try to associate a new volume as a Replicator Log volume to the RVG in an RDS whose Replicator Log volume has been dissociated, you may be unable to see the volume name in the drop-down list of the Associate Replicator Log wizard.

This is because the volume may have a DCM or DRL log associated with it. To find out the type of the log associated with the volume, click on the volume name in the tree view. The right panel of the VEA displays the volume properties. Click the logs tab to see the type of log associated with the volume.

Remove all logs from the volume that you want to use as a Replicator Log volume.

## Unable to see the volume names for adding volumes to RDS

When trying to add volumes to the RDS, sometimes the volume names may not appear in the **Add Volumes** dialog box.

The reason for this is that the volume may have a Dirty Region Log (DRL) associated with it or the volume created may be a software RAID 5 volume or it may be missing. None of these volumes can be used for replication. A replicated volume can have only DCM logs associated with it. Similarly, a volume with a DCM log cannot be used as a Replicator Log.

To find out the type of the log associated with the volume that you want to add, click on the volume name in the tree view. The right panel of the VEA displays the volume properties. Click the logs tab to see the type of log associated with the volume.

Select the volume and right-click. Select the Log > Remove option from the menu that appears, to remove all the DRL logs. Now you can add the volume to the RDS.

## Adding logs to dissociated volumes

When adding logs to dissociated volumes, the following error message is displayed:

```
The requested operation is not valid.
```

If you try to add a log to a volume that has been dissociated from the RDS, you may get the above error. The volume may already have a DCM log associated with it.

The DCM log may have got added when you had added this volume to the RDS with the Add DCM log to all selected volumes option selected. Now that the volume is dissociated from the RDS, if you try to add the DRL log to it you may get the error message given above, as DRL and DCM logs cannot exist together.

To find out the type of the log associated with the volume that you want to add; click on the volume name in the tree view. The right panel of the VEA displays the volume properties. Click the logs tab to see the type of log associated with the volume.

If you need to add the dissociated volume back to the RDS later, do not add DRL log to the volume. Otherwise, you can remove the DCM logs and add the DRL logs by using the options that you get when you select the volume and right-click on it.

## Using two commands in succession

When two commands are run in succession, one immediately after the other, the following error message is displayed:

```
Could not complete operation. Please try again.
```

When using the Command Line Interface (CLI), if you use two commands one immediately after the other, you may get the above message.

This is because even before Volume Replicator has completed the first command, you have issued the second one. Therefore, Volume Replicator rejects the second command to prevent the Volume Replicator objects from operating in inconsistent states.

When using the command, wait for a few seconds after the first command completes before issuing the second command.

## Renaming dynamic disk group while importing

When importing a dynamic disk group, you may need to rename it, if a dynamic disk group of the same name has already been imported on the host. If the dynamic disk group that needs to be renamed contains Volume Replicator objects, then replication does not restart after importing the renamed disk group.

**Warning:** Volume Replicator objects are created even if some disks are unavailable while importing the dynamic disk group. However, the Volume Replicator configuration in such a case may be invalid and replication may not restart even after performing the steps that are given below. Importing a partial dynamic disk group after resetting the disk group `Host ID` may result in losing the integrity of dynamic disk group organization.

Use the following steps to enable replication after the disk group is imported. Ensure that all disks and volumes of the imported dynamic disk group are accessible and in a HEALTHY state before proceeding.

This example assumes that the following setups are already set:

- `host_imp`

  The host on which you import the dynamic disk group after renaming.

- `dg_name_imp`

  The new name of the dynamic disk group.

  All other hosts in the Volume Replicator configuration are referred to as remote host(s).

**To enable replication after the disk group is imported**

1 On host `host_imp`, find the RLINK objects to remote host(s) for the Volume Replicator configuration, using:

```
vxprint -g dg_name_imp -P
```

This displays the list of RLINKs (among other things), in the following format:

```
rl <rlink_name> attributes
```

2 For each RLINK object in the `dg_name_imp` dynamic disk group, find the corresponding remote objects, using:

```
vxprint -g dg_name_imp -l <rlink_name> | findstr "remote_"
```

This displays the remote objects in the following format:

```
remote_host = <remote-host-name or ip>
remote_dg = <name of the remote disk group>
remote_rlink = <name of the corresponding remote rlink>
```

3 For every RLINK on the remote host change the `remote_dg` attribute of the corresponding remote RLINK using the following commands:

Pause and Resume operations are permitted only on RLINK objects in ACTIVE state. If any of the remote RLINKs are not ACTIVE, then, the Pause operation fails. In such a case, do not perform the Resume operation.

```
vxrlink -g <remote_dg> pause <remote_rlink>
vxrlink -g <remote_dg> set remote_dg=dg_name_imp <remote_rlink>
```

If the pause operation above succeeded then run the following command:

```
vxrlink -g <remote_dg> resume <remote_rlink>
```

Run these commands on each of the remote hosts.

4 Verify the changes on every remote host, using:

```
vxrds -g <remote_dg> printrvg
```

This command should list the RVGs on `host_imp` as part of the RDS(s).

## Problems when performing the snapshot operation

If the DCM log for a volume and its snap ready plex exists on the same disk, then, the subsequent snapshot operation does not produce the desired results.

To work around the problem, before performing the snapshot operation you must manually move the DCM log plex to another disk.

## Operation time-out errors

The following error message is displayed in case of operation time-out errors:

```
Operation timed out. The configuration server may be busy or down.
```

### The `vxrvg stop` command displays operation time-out error

The `vxrvg stop` command is a three-step process; first it acquires an exclusive lock on the volumes of the RVG, then it flushes the data volumes in the RVG and finally it disables data access to the volumes. However, at times the process of flushing the data volumes may take a while and since the VEA waits for a fixed time, if it does not receive an acknowledgement for the operation within that time, this message is displayed.

Despite this message being displayed the `vxrvg stop` operation completes. Check the **VEA events** pane to verify whether the operation has been completed successfully.

### Pausing Secondary from Primary displays operation time-out error

When trying to pause the Primary RLINK, if the Primary is busy performing some other operation, then this error is displayed. This is because the VEA waits for a fixed time, that is, one minute to complete the operation. However, since the Primary is busy it is currently unable to service this request and hence the message is displayed. Also, during this time the Secondary is unable to get a response from the Primary and assuming that the Primary is unavailable the Secondary RDS splits. This is a timing issue.

Despite this message being displayed, once the Primary becomes free it proceeds with the pause operation. After the resume operation is performed the Secondary RVG gets added back to the RDS, automatically.

# Problems when configuring Volume Replicator in a VCS environment

This section provides troubleshooting tips to recover from the problems that may occur when configuring Volume Replicator in a VCS environment.

See “Application Service group does not failover correctly” on page 424.

## Application Service group does not failover correctly

At times, the applications that can failover locally do not failover to the remote host in case the application is configured in a Volume Replicator setup. VCS logs messages stating the application disk group contains unsteady volumes, and the MountV resource cannot come online.

Perform the following actions:

- Performing a Rescan or Refresh operation clears up this problem, but this requires user intervention and defeats the purpose of having an automated failover mechanism.
- Another workaround is increasing the value for the `OnlineRetryLimit` attribute to a sufficiently larger value (not just 1), depending on the time that is required for the objects to get refreshed, so that the online would succeed.
  The `OnlineRetryLimit` attribute specifies the number of times the online entry point for a resource is retried, if the attempt to online a resource is unsuccessful. This attribute applies only during the initial attempt to bring a resource online. If the `OnlineRetryLimit` attribute is set to a non-zero value, the agent attempts to restart the resource before declaring the resource as faulted.
  To reset the value of this attribute select the **Show all attributes** for the required resource type, then choose MountV.

# Problems when setting performance counters

When performance counters are set, certain issues are seen which are explained in this section.

## Volume Replicator objects are not displayed

When setting up a new file for logging or monitoring the Volume Replicator performance-related information, you may be unable to see the Volume Replicator Objects (Memory and Remote Hosts) in the **Performance objects** list when you click **Add Objects**. This could happen if the Volume Replicator have got unloaded.

Load the Volume Replicator counters by running the command:

```
lodctr %Installed Dir%\Veritas\Veritas Volume Manager\VM5INF\vvrperf.ini
```

# Services and ports

This appendix includes the following topics:

- InfoScale ports and services

## InfoScale ports and services

If you have configured a firewall, then ensure that the firewall settings allow access to the services and ports used by the InfoScale products.

The following table displays the services and ports used by InfoScale products.

Ensure that you enable the ports and services for both, inbound and outbound communication.

**Note:** The port numbers marked with an asterisk are mandatory for configuring the InfoScale products.

**Table A-1** InfoScale services and ports

| Component Name/Port | InfoScale Foundation | InfoScale Availability | InfoScale Storage | InfoScale Enterprise |
|---|---|---|---|---|
| vxsvc.exe<br>2148*, 3207/TCP/UDP<br>Veritas Enterprise Administrator (VEA) Server | ✓ | X | ✓ | ✓ |
| CmdServer.exe<br>14150*/TCP<br>Veritas Command Server | X | ✓ | ✓ | X |

**Table A-1** InfoScale services and ports *(continued)*

| Component Name/Port | InfoScale Foundation | InfoScale Availability | InfoScale Storage | InfoScale Enterprise |
|---|---|---|---|---|
| had.exe<br>14141*/TCP<br>Veritas High Availability Engine | X | ✓ | X | ✓ |
| Plugin_Host.exe<br>7419*/TCP<br>Veritas Plugin Host Service | X | ✓ | ✓ | ✓ |
| vcsauthserver.exe<br>14149/TCP/UDP<br>VCS Authentication Service | X | ✓ | X | ✓ |
| vras.dll<br>8199/TCP<br>Volume Replicator Administrative Service | X | X | ✓ | ✓ |
| vxrserver.exe<br>8989/TCP<br>Volume Replicator Resync Utility | X | X | ✓ | ✓ |
| vxio.sys<br>4145/TCP/UDP<br>Volume Replicator Connection Server | ✓ | X | ✓ | ✓ |
| VxSchedService.exe<br>4888/TCP<br>Veritas Scheduler Service<br>Use to launch the configured schedule. | ✓ | X | ✓ | ✓ |
| User configurable ports created at kernel level by vxio .sys file<br>49152-65535/TCP/UDP<br>Volume Replicator Packets | ✓ | X | ✓ | ✓ |

**Table A-1** InfoScale services and ports *(continued)*

| Component Name/Port | InfoScale Foundation | InfoScale Availability | InfoScale Storage | InfoScale Enterprise |
|---|---|---|---|---|
| Notifier.exe<br>14144/TCP/UDP<br>VCS Notification | X | ✓ | X | ✓ |
| wac.exe<br>14155/TCP/UDP<br>VCS Global Cluster Option (GCO) | X | ✓ | X | ✓ |
| xprtld.exe<br>5634/HTTPS<br>Veritas Storage Foundation Messaging Service | ✓ | X | ✓ | ✓ |

# Using the vxrsync utility

This appendix includes the following topics:

## About using the vxrsync utility

Using the vxrsync utility, you can synchronize or verify the data volumes on the source host with those on the target host. These data volumes can either be a set of volumes that are associated with an RVG or independent SFW volumes. Using this utility, you can perform full synchronization or difference-based synchronization. The set of volumes that you specify can either be a part of the RVG or may be separate volumes. However, it is important to note that when specifying volumes for the vxrsync operations the volumes must not be in use. You can use the vxrsync utility with its synchronization and verify options to complement Volume Replicator.

After performing either the synchronization or the data verification operation, the results are displayed in the current console from where the command is run. The display is a progressive display and is shown on both the source and the target hosts. As the operation proceeds the status on the display console changes.

## When to use vxrsync

Use the vxrsync utility to perform initial synchronization of the volumes between the source and target volumes. This utility is mainly used for performing synchronization when the target host is detached and you can perform either full or difference-based synchronization. You can choose the type of synchronization depending on the amount of data that has changed on the source volume. However, using this utility for difference-based synchronization when the number of changes are too many may not be very useful. The vxrsync utility can be used to synchronize volumes on multiple target hosts.

# Understanding how the utility works

Using the vxrsync utility, you can perform three different operations; full synchronization, difference-based synchronization, and data verification.

When performing full synchronization between volumes the utility copies all the data from the source to the destination volumes. For performing difference-based synchronization, the utility first calculates the checksum and then compares the checksums between volumes. Based on the result the utility copies only those blocks that have changed on to the target volumes.

When performing verify data operation, vxrsync first calculates the checksum for the volumes to find the change in data between the source and the target and then displays the difference on the console.

The vxrsync utility consists of two components, vxrclient and vxrserver. The vxrclient must be running on the source computer and the vxrserver must be running on the target computer. Note that the vxrserver must first be started before the vxrclient is started. The vxrclient and the vxrserver require either volume names, the RVG name, or a configuration file name as input. You must also specify the port number on which the vxrserver needs to listen for requests. If you use a configuration file as input, then, the port number that you specify for the vxrserver must be the same as that specified in the file. If no port number is specified for vxrserver, then, by default, it uses the port number 8989 and vxrclient uses this port to communicate with vxrserver.

If you choose to use the RVG name as input then you must ensure that the target or Secondary (RLINK) must be detached. Make sure that the target volumes are not in active use during the period the synchronization or verification is in progress. Otherwise, the synchronization process fails. If the -x option is specified with vxrclient then the source volumes are locked. Otherwise, a warning message is displayed, but synchronization still proceeds.

Also, note that all the specified volumes with the same names and sizes must be present on each host within the RDS.

The configuration file defines the relation between the source and target volumes which need to be synchronized or verified. If you want to use the configuration file as an input then you must first create it. Ensure that the file is created using a text editor and is available in the current directory from where you are running the command.

**Note:** The configuration file must be named using the format <groupname>.cfg. The groupname is the name that you have given the set of related volumes within the configuration file. The file must have an extension .cfg.

The configuration file can be used both for synchronizing the data volumes or for verifying the data. However, for the utility to complete the specified operation successfully, the configuration file must be exactly the same on the source and the target. Blank lines and lines starting with a # character are considered as comments.

**Note:** The vxrsync utility can accept only SFW volumes having a name as input. Any other volumes cannot be used.

# Layout of the configuration file

The configuration file defines the relation between the source and target volumes that need to be synchronized or verified. After you have created the configuration file make sure that it is available in the directory from where you intend to run the vxrclient or vxrserver command.

Each host system must have a configuration file that contains the following information:

- A description of all host systems and the local host
- Association of the volumes between each host system that is linked for synchronization
- Association between the volumes on different host systems

  To facilitate managing multiple volumes present on one or more hosts with ease, vxrclient associates one or more related volumes into an organizational construct that is called a group. A group is identified by its <group name>. You can use the concept of a group to synchronize a number of volumes in one operation.

## Sample configuration file layout

The configuration file layout is similar to the sample that is shown below:

```
GROUPNAME:  <group_name>   HOST: <hostname_or_IP> [<port_number>]
   VOLUME:  <virtual_volume_name> <physical_volume_path>
   VOLUME:  <virtual_volume_name> <physical_volume_path>
```

```
.
.
.
HOST: <hostname_or_IP> [<port_number>]
VOLUME:  <virtual_volume_name> <physical_volume_path>
VOLUME:  <virtual_volume_name> <physical_volume_path>
```

where,

`<group_name>` is name of a group, `virtual_volume_name` indicates the name of the volume, for example `NAME_TBLSPCphysical_volume_path` indicates the drive letter or mount path, for example, `\\.\Z:`

## Using the vxrsync utility with the vxrclient component

The `vxrclient` is executed on the source system whereas the `vxrserver` is expected to be running on the target or remote system. When the `vxserver` is run for the first time on the target system make sure that you run it with the `-spawn` option. This ensures that each time a `vxrclient` requests some operation, a new instance of the `vxserver` is automatically spawned for every new request. If the vxserver is run without the `-spawn` option then it can serve only one request from the client and then it gets terminated.

### vxrclient

The `vxrsync` utility contains the `vxrclient` component.

This component of the `vxrsync` utility can be used to:

- Synchronize the remote systems with the source on which the `vxrclient` is running
- Verify the data on the volumes between the source and the target.

The command syntax varies depending on the options that it is used with. To perform full synchronization you must use the `-full` option. To verify the data between the source and the target systems, use the `-verify` option. Using the command without either of these options, which is the default, results in difference-based synchronization.

The `vxrclient` command, by default, does not lock all the volumes before synchronizing or verifying the volumes. If you choose to proceed with the default, a warning message is displayed.

**Note:** Veritas recommends that you use the command with the -x option to make sure that the all volumes in the RDS are locked before performing any operation on them.

Syntax for the vxrclient command:

```
vxrclient [-noreport] [-reportinterval <secs>]
   [-full|-swiftsync] [-blocksize|-bs <blksize_KB>]
   [-blockgroupcount|-bc <numblocks>] [-x]will be
   [-use <host>] -to <host> [[<host>]...] [-port
   <serverportnumber>]
   {-for | -g} <groupname> | -r <rvgname> | -vol
<volumename>[[,<volumename>] ...][-dg <diskgroupname>]
```

**Note:** You can use the swiftsync option to leverage the NTFS or ReFS file systems.

Syntax for verifying the vrclient command against remote or target host:

```
vxrclient -verify|-quick[verify] [-noreport]
   [-reportinterval <secs>] [-blocksize|-bs <blksize_KB>]
   [-blockgroupcount|-bc <numblocks>] [-x]
   [-use <host>] -with <host> [[<host>]...] [-port
   <serverportnumber>]
   {-for | -g} <groupname> | -r <rvgname> | -vol
<volumename>[[,<volumename>] ...]
            [-dg <diskgroupname>]
```

The following table describes the vxrclient command usage with the basic options to perform the required operations:

**Table B-1** Command usage for vxrclient

| Operation | Command | Description |
|---|---|---|
| Full synchronization | `vxrclient -full -use <host> -to <host> -r <rvgname>` | This command enables you to perform full synchronization between the source and target volumes. The RVG name is used as input by the command.<br>The -x parameter can be optionally specified if you want all the volumes in the RDS to be locked. |

**Table B-1** Command usage for vxrclient *(continued)*

| Operation | Command | Description |
|---|---|---|
| | `vxrclient -full -use <host> -to <host> -vol <vol1>, <vol2>` | This command enables you to perform a full synchronization between the source and target volumes. A comma-separated list of volumes is used as input to the command. |
| | `vxrclient -full -use <host> -to <host> -g <groupname>` | This command enables you to perform a full synchronization between the source and target volumes. A configuration file is used as input to the command. |
| Difference-based synchronization | `vxrclient -use <host> -to <host> -r <rvgname>` | This command enables you to perform a difference-based synchronization between the source and target volumes. The RVG name is used as input to the command. |
| | `vxrclient -use <host> -to <host> -vol <vol1>, <vol2>` | This command enables you to perform a difference-based synchronization between the source and target volumes. A comma-separated list of volumes is used as input to the command. |
| | `vxrclient -use <host> -to <host> -g <groupname>` | This command enables you to perform a difference-based synchronization between the source and target volumes. A configuration file is used as input to the command. |

**Table B-1** Command usage for `vxrclient` *(continued)*

| Operation | Command | Description |
|---|---|---|
| `Swiftsync` synchronization | `vxrclient swiftsync -use <host> -to <host> -r <rvgname>` | This command enables you to perform a swiftsync synchronization between the source and target volumes. The RVG name is used as input to the command.<br>**Note:** If you use this option with an NTFS or ReFS file system, only the used blocks are synced between the source and target volumes. |
| Data Verification | `vxrclient -verify -use <host> -with <host> -r <rvgname>` | This command enables you to perform data verification between the source and target volumes. The RVG name is used as input to the command. |
| | `vxrclient -verify -use <host> -with <host> -vol <vol1>, <vol2>` | This command enables you to perform data verification between the source and target volumes. A comma-separated list of volumes is used as input to the command. |
| | `vxrclient -verify <host> -with <host> -g <groupname>` | This command enables you to perform data verification between the source and target volumes. A configuration file is used as input to the command. |

The following table lists the command options available with `vxrclient`.

**Table B-2** Command options for `vxrclient`

| Option | Description |
|---|---|
| `-v\|-version` | Prints the version number of `vxrclient` command. |
| `-?\|/?\|-h\|-help` | Prints a brief summary of command line options. |
| `-longhelp` | Prints a detailed summary of command line options and an explanation of the operation of `vxrclient` command. |

**Table B-2** Command options for vxrclient *(continued)*

| Option | Description |
|---|---|
| -noreport | Specifies that the performance and progress information does not require to be printed. |
| -reportinterval <secs> | Updates the performance and progress information every <secs> seconds where you can specify the value for the report interval. The default value is 10 seconds. |
| -full | Copies all the data, and not just differences from the source volume to the target volume. This option is useful to create the initial volume copies. The default is to transfer only data differences. |
| -swiftsync | Acts like -full option, if the volume is not an NTFS or ReFS volume. For NTFS or ReFS volumes, only the blocks that are used by these volumes are transferred. This option is useful for creating the initial volume copies. |
| -blocksize\|-bs <KB> | Sets the size of the block of data to be examined, and then transfers it as a unit. The default is 8 KB. |
| -blockgroupcount\|-bc <numblocks> | Sets the number of blocks of size that is specified in -blocksize\|-bs option, that are sent in one network message. The default is 200 blocks. |
| -x | Specifies that all the volumes on the client system are locked. |
| -use <hostname_or_ip> | If the client system has more than one network interface card (NICs), specifies which interface to use when connecting to the required server systems either by providing the host name or IP address of the local network connection to use. |
| -to <host> [[<host]...] | Synchronizes one or more remote host systems from this client system. All the host names with the corresponding information must be found in the configuration file. |
| -port | Specifies the port number on which the server listens for requests from the client. This parameter does not need to be specified if the configuration file is used as input. |
| -verify | Verifies the client's volumes with one or more remote host systems and lists any differences that are found. |
| -quick[verify] | Verifies the client's volumes with one or more remote host systems. Halt this operation upon detection of any difference. This option does not perform any synchronization. |
| -with <host> [[<host>]...] | Specifies the host name or IP of the remote host system(s) with which this client system's volumes should be verified. |

**Table B-2** Command options for vxrclient *(continued)*

| Option | Description |
|---|---|
| -for\|-g <groupname> | Identifies the group of volumes for this operation. The group name corresponds to an ASCII configuration file that describes all possible host systems and the relationship and paths of the volumes that should be synchronized or verified together as a unit. |
| -dg <disk group name> | Identifies the SFW disk group name. The disk group name is used to uniquely identify the specified RVG or volumes. |
| -r <rvgname> | Identifies the RVG whose volumes are used for the required operation. |
| -vol <volumename> [[,volumename].....] | Identifies the volumes that are used for the required operation. If there is more than one volume, they are indicated by a comma-separated list.<br>**Note:** A volume or a set of volumes that are synchronized using swiftsync option, when verified through vxrclient and vxrserver, would show differences. This is because the swiftsync option has synchronized only the NTFS or ReFS used blocks, it has ignored the rest of the blocks, even though they may be different between source and destination volumes.<br>See "vxrserver" on page 436. |

## vxrserver

This component of the vxrsync utility is used as the remote utility server when the client initiates the synchronize or verify operations. This component must be running on the target or remote systems when the vxrclient command is run on the source system.

**Note:** The vxrserver must be started before running the vxrclient.

The command syntax varies depending on the options that it is used with. Following is the command usage to start the server and launch multiple instances as required. All the options that can be used with the command are explained in the following table:

```
vxrserver -spawn
```

The following table lists the command options for vxrserver -spawn.

**Table B-3** Command options for `vxrserver -spawn`

| Option | Description |
|---|---|
| `-v\|-version` | Prints the version number of `vxrserver` command. |
| `-?\|/?\|-h\|-help` | Prints a brief summary of command line options. |
| `-port <tcp_listening_port>` | Specifies the port number on which the server listens for request from the client. If no port number is specified for `vxrserver`, then it uses the 8989 port by default. |
| `-use <hostname_or_ip>` | If the server system has more than one network interface card (NICs), specifies which interface to use when connecting to the required client systems either by providing the host name or IP address of the local network connection to use. |
| `-spawn` | Spawns a new instance of `vxrserver` after connection. |

# Example: Using `vxrsync` for difference-based synchronization

The `vxrsync` utility can be used for synchronizing the Secondary after a break in the replication. This utility provides you the option of performing difference-based synchronization, instead of complete synchronization.

For information about alternative methods to synchronize the Secondary faster, See “Alternative methods to synchronize the Secondary faster” on page 213.

If for some reason the replication between `london` and `seattle` stops, then you need to start replication with complete synchronization. This can be time consuming. However, using the `vxrsync` utilities you can perform difference-based synchronization to send only those data blocks that are different from the Secondary.

**Note:** The following steps assume that the Primary and Secondary RLINKs are detached.

**To use vxrsync utility for difference-based synchronization**

1 On the Primary host london, checkstart the Primary RVG using the following command:

```
vxrvg -g vvr_dg -c checkpt2 checkstart vvr_rvg
```

2 Start vxrsync server on the Secondary host seattle by running the command:

```
vxrserver
```

3 Start the vxrsync client on the Primary host london:

```
vxrclient -use london -r vvr_rvg -to seattle
```

In this command the RVG name is provided as input, however you can also provide the volume names or a configuration file as inputs. This starts the difference-based synchronization process. Progress is displayed periodically at the client side that is on host london.

4 After the synchronization completes, perform the following:

- On the Primary host london, checkend the Primary RVG

  ```
  vxrvg -g vvr_dg checkend vvr_rvg
  ```

- Start the replication to Secondary using the checkpoint that you have created.

  ```
  vxrds -g vvr_dg -c checkpt2 startrep vvr_rvg seattle
  ```

  This command starts replication to Secondary after synchronizing from the mentioned checkpoint and the replication status is now ACTIVE.

Appendix C

# VR Advisor (VRAdvisor)

This appendix includes the following topics:

## Introducing Volume Replicator Advisor (VRAdvisor)

Volume Replicator Advisor (VRAdvisor) is a planning tool that helps you determine an optimum Volume Replicator configuration.

This appendix provides information about installing and using this tool on different platforms. Wherever applicable, the information that is specific to a platform has been appropriately indicated. For Windows, the Veritas Volume Manager (VxVM) has been renamed toStorage Foundation (SFW) from Release 4.1 onwards.

This appendix is intended for system administrators who are responsible for setting up replication using Volume Replicator. This appendix assumes that the user has:

- An understanding of system administration.
- A working knowledge ofVolume Replicator.

This appendix guides you through the process of installing VRAdvisor and then evaluating various parameters using the data collection and data analysis process.

It describes procedures using both the graphical user interface and the command line interface on the different platforms.

## Overview of VRAdvisor

Planning is the key to successfully configuring Volume Replicator. To set up an optimum configuration, you must understand the components of Volume Replicator and their interactions with each other. In addition, you must consider the factors that are specific to your environment while planning your Volume Replicator configuration.

The important factors to consider when you plan your Volume Replicator configuration include:

- The needs and constraints of the business
- Application characteristics
- The mode of replication
- Network characteristics

These factors are dependent on each other and these dependencies must be considered during planning. For example, if your business requires the data on the Secondary to be as up to date with the Primary as possible, you must choose synchronous mode and provide enough network bandwidth to handle the peak application write rate on the Primary. Or, if the available network bandwidth is less than the peak write rate of the application, you must choose asynchronous mode of replication. Also, the size of the Storage Replicator Log (SRL) must be able to handle the Secondary outages and network outages for the given application characteristics. VRAdvisor considers these dependencies and enables you to determine the parameters to suit your Volume Replicator environment.

VRAdvisor does the following:

- Collects a sample of data that reflects the application characteristics.
- Analyzes the sample of the application characteristic and calculates the size of the SRL and the network bandwidth that is required for replication.
- Enables you to perform a What-if Analysis by varying the needs and constraints of your business, based on your future requirements.

**Note:** The replication log of Volume Replicator is referred to as SRL (Storage Replicator Log) on UNIX and as Replicator Log on Windows. The terms SRL and Replicator Log are used interchangeably in the appendix.

# How VRAdvisor works

Using VRAdvisor for planning involves collecting a sample of data. This data represents the application write rate and analyzing this sample of data based on factors, such as the network bandwidth and network outage. When VRAdvisor analyzes the data, it considers the worst case situations , which results in an optimum configuration for Volume Replicator.

Working with VRAdvisor involves multiple tasks.

See "Data collection" on page 441.

See "Data analysis" on page 441.

See "What-if analysis" on page 442.

## Data collection

VRAdvisor uses a sample of data for analysis; the sample of data must be available in an appropriate format that VRAdvisor requires. To collect a sample of data that represents the application write rate, we recommend that you collect the sample of data for a period of seven to fourteen days. Make sure that the collection period includes times of peak usage for your application, so that the collected data reflects your environment.

In the data collection mode, VRAdvisor collects the sample of data in the appropriate VRAdvisor format. You can also collect the sample of data using the data collection script provided. The data collection script uses the appropriate command at the operating system level to collect the data, and also converts the data to the appropriate format. For more information, See "Collecting the sample of data" on page 443.

## Data analysis

In the data analysis mode, VRAdvisor analyzes the sample of data that you have collected, based on the following factors specified by you:

- Available network bandwidth
- Network outage duration
- Secondary outage duration

After analyzing the data, VRAdvisor displays a graphical as well as textual representation of the results in a separate window. For more information, See "Analyzing the sample of data" on page 447.

## What-if analysis

The What-if analysis feature enables you to perform additional calculations, to plan for future requirements or alternative scenarios. You can vary the parameters and recalculate the results according to different criteria. For example, you can vary the network bandwidth parameter to see what effect it would have on the SRL size. Or, you can specify a potential SRL size and see how much network bandwidth would be required for that SRL size. For more information, See "Performing What-if analysis" on page 453.

# Installing Volume Replicator Advisor (VRAdvisor)

This section explains how to install Volume Replicator Advisor on a Windows operating system.

---

**Note:** VRAdvisor is not installed as a part of the common installation process that uses the product installer. Although VRAdvisor is supported in a non-English locale, the wizards are still displayed in English.

---

**To install VRAdvisor**

1. If a previous version of VRAdvisor is installed, remove the existing VRAdvisor before installing VRAdvisor.
2. Navigate to the `storage_foundation` directory under the `Tools` directory in the software package.
3. Run the `vrtsvradv.msi` from the `storage_foundation` directory.

   The installation wizard is launched. A message indicates that the VRAdvisor setup file checks for the necessary parameters before starting the installation process.
4. On the **Welcome** panel, click **Next**.
5. On the **Customer Information** panel, enter your user name and organization, and click **Next**.
6. The **Destination Folder** panel appears. Provide the following information:
   - To install VRAdvisor in the default directory `C:\Program Files (x86)\Veritas\Volume Replicator Advisor`, click **Next**.
     OR
   - To choose another location for installing VRAdvisor, click **Change**.
   - On the **Change Current Destination Folder** panel, in the **Folder name** field, enter the complete path to the directory where you want the VRAdvisor

package to be installed. You can also use the browse option to navigate to the required directory. Click **OK**.

- On the **Destination Folder** panel, click **Next**.

7 On the **Ready to Install the Program** panel, click **Install** to proceed with the installation.

The Installing **Volume Replicator Advisor** panel appears. This panel displays a progress bar to indicate that the installation is in progress. After the installation completes, a message indicates that the installation was successful.

8 Click **Finish**.

9 If required, a message prompts you to restart the computer. Click **Yes** to restart the computer now. Click **No** to restart it later. On computers running Windows XP, a restart is not required to enable disk performance counters.

# Uninstalling VRAdvisor on Windows

**To uninstall VRAdvisor**

1 To uninstall VRAdvisor, go to **Start > Settings > Control Panel > Add or Remove Programs** or, on Windows Server 2012 operating systems, on the **Start** screen, click **Control Panel**.

2 Select **Veritas Volume Replicator Advisor** from the list of programs.

3 Click **Remove**. Windows prompts you to confirm that you want to remove Volume Replicator Advisor.

4 Click **Yes**. The **Veritas Volume Replicator Advisor** dialog box appears.

The progress bar on the **Veritas Volume Replicator Advisor** dialog box indicates that the removal is in progress.

After the uninstallation procedure completes, the Add or Remove Programs dialog box indicates that the Veritas Volume Replicator Advisor program has been removed successfully.

# Collecting the sample of data

You need to collect the data write samples that can be used with the VRAdvisor Wizard.

**Best practices**:

- Veritas recommends that you collect the sample data using the volumes that are part of the Volume Replicator configuration that you plan to set up.

- To collect a representative sample of data, it is recommended that you collect the sample of data over a period of 7 to 14 days.

  **Note:** The data must be collected for a minimum of 7 days.

- Make sure that the collection period includes times of peak usage for your application, so that the collected data reflects your actual requirements.

  VRAdvisor calculates an optimum size of the Storage Replicator Log (SRL) and the network for your Volume Replicator configuration using a sample of the write statistics.

  Depending on the operating system on which you collect data, you can either collect the sample of data using the VRAdvisor Wizard, commands, or the data collection script. For details, refer to the section for your platform.

You can collect data using the VRAdvisor Wizard or the `diskStats` command. To use VRAdvisor to collect data, you must install VRAdvisor on your system. If you do not plan to install VRAdvisor on your system, use the `diskStats` command to collect data.

On Windows, collect the sample data using one of the following methods:

- See "Collecting sample data using the VRAdvisor Wizard" on page 445.
- See "Collecting the sample data using the diskStats command" on page 446.

**Prerequisite**:

- If you use SFW volumes, then ensure that you import the disk group containing the required volumes onto your system.

# Collecting sample data using the VRAdvisor Wizard

**To collect data using the VRAdvisor Wizard**

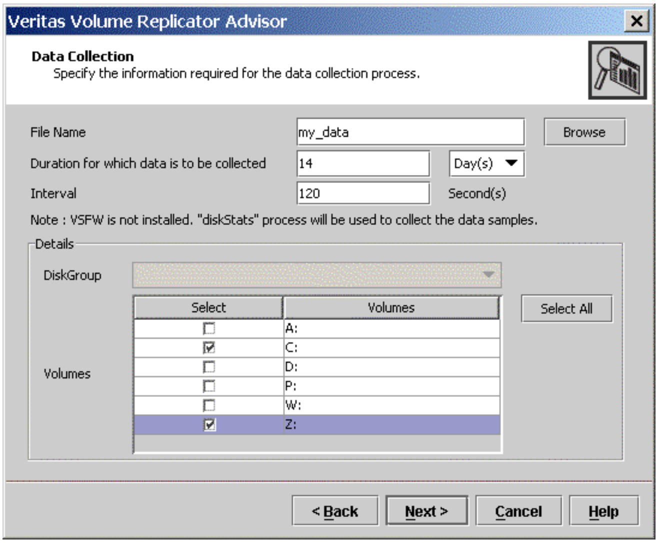
1 Launch the VRAdvisor Wizard on Windows from **Start** > **All Programs** > **Veritas** > **Volume Replicator Advisor** > **VRAdvisor Wizard** or, on Windows 2012 operating systems, from the **Apps** menu in the **Start** screen.

2 On the **Welcome** panel, select **Data Collection**, and then click **Next**. The Data Collection panel appears.

---

**Note:** On Windows, only the `diskStats` command is used to collect data.

---



**Data Collection**

Specify the information required for the data collection process.

File Name: my_data

Duration for which data is to be collected: 14 Day(s)

Interval: 120 Second(s)

Note : VSFW is not installed. "diskStats" process will be used to collect the data samples.

Details

DiskGroup:

Volumes:

| Select | Volumes |
|---|---|
| ☐ | A: |
| ☑ | C: |
| ☐ | D: |
| ☐ | P: |
| ☐ | W: |
| ☑ | Z: |

3 Complete the Data Collection panel as follows:

| | |
|---|---|
| **File Name** | Enter the name of the file where the data write samples are collected.<br>Make sure that no other application uses the name.<br>If a file already exists with that file name or if the path is incorrect, a message is displayed. |
| **Duration for which data is to be collected** | Enter the duration in days or hours. The default value is 14 days. The maximum duration is 30 days. |
| **Interval** | Enter a value in seconds to indicate the frequency at which you want the data to be collected. The default value is 120 seconds. |
| **Details** | Select the required volumes individually, or click **Select All** to select all of the available volumes in the selected disk group.<br>Only volumes with drive letters are displayed.<br>On Windows, the **DiskGroup** field is not available. |

4 Click **Next**. The Confirmation message appears.

5 To start the data collection process immediately, click **Yes**. To go back and make any changes, click **No**.

The **Data Collection Summary** panel indicates that the data collection has started. It also displays a summary of the specifications you entered for the data collection.

6 Click **Finish**. VRAdvisor continues to collect data for the specified duration, although the wizard window closes. The data collection wizard displays an error message if it is unsuccessful in starting the data collection process. Select **Cancel**, fix the reported error and launch the data collection wizard again.

After the data collection completes, the specified file contains the sample of data in a format that can be used for analysis by VRAdvisor. For more information, See “Analyzing the sample of data” on page 447.

# Collecting the sample data using the diskStats command

On Windows, use the `diskStats` command to collect the data that is required for analysis. This command can be used to collect data whether or not the InfoScale product is installed on the system. The `diskStats` utility is installed in the following location:

```
Veritas\Volume Replicator Advisor\bin\diskStats.exe
```

**To collect data using the `diskStats` command**

1. Navigate to the specified path:

   ```
   Veritas\Volume Replicator Advisor\bin
   ```

2. At the prompt, enter the following command with exactly the parameters shown:

   ```
   diskStats [-i interval [-c count]] \
   
   <drive 1> [[drive 2][drive 3]... ]
   ```

   The command displays the output on the console.

**Note:** The `diskStats` command can accept only drive letters of the volumes as inputs. Volume names are not supported. The `diskStats` command also supports the volumes that an application creates.

To save the output to a file, you can redirect the output to a named file using the command:

```
diskStats [-i interval [-c count]] \

<drive 1> [[drive 2][drive 3]... ] > <filename>
```

After data collection completes, the file `filename` contains the sample data in`diskStats` format, which can be used for analysis by VRAdvisor. For more information, See “Analyzing the sample of data” on page 447.

# Analyzing the sample of data

VRAdvisor analyzes the sample data according to parameters that you specify such as available network bandwidth and network outage. In addition, VRAdvisor enables you to perform a What-If analysis by varying the values of the parameters. The output of the analysis gives the network the bandwidth that is required to replicate in synchronous mode, and the SRL (Storage Replicator Log) size that is required for a given bandwidth and for the given outages to replicate in asynchronous mode. The results of the analysis help you to set up an optimum configuration for Volume Replicator. For more information about some of the considerations and formulas that determine the size of the SRL, See “Sizing the SRL” on page 456.

VRAdvisor enables you to analyze the data that is collected on any of the supported platforms. For more information, See “Collecting the sample of data” on page 443.

However, to analyze the data, you must install and use VRAdvisor on a Windows operating system.

**Prerequisites**:

- All the files to be analyzed must be present in a single directory.
- The sample data must be available in one of the following formats that VRAdvisor accepts:
  - `vxstat` output
  - `diskStats` output
  - VRAdv CSV format (used by VRAdvisor Wizard or the UNIX data collection script)

**To analyze the collected data using the VRAdvisor Wizard**

1. Launch the VRAdvisor Wizard on Windows from **Start** > **All Programs** > **Veritas** > **Volume Replicator Advisor** > **VRAdvisor Wizard** or, on Windows 2012 operating systems, from the **Apps** menu in the **Start** screen.
2. On the **Welcome** panel, select **Analysis**, and then click **Next**.
3. On the **Directory Specification** panel, enter the name of the directory containing the data files to be analyzed. All files to be analyzed must be present in the same directory.

   The specified directory must contain the data files and any metadata files that are associated with each data file. The associated metadata and data files must have the same name except for the extension. Metadata files must have the extension `.meta`.
4. On the **File Selection** panel, VRAdvisor displays the list of files in a table. Select the files to be analyzed.

   **Note:** Files containing the information from the nodes that use the same network bandwidth for replication should be analyzed together. Otherwise, files should not be selected together. In order for the files to be analyzed together, the data collection for each node must start at the same time.

   - Provide the disk group name, Node name, and Cluster ID, if necessary.
5. On the **Block Size and Collection Interval Specification** panel, specify the metadata.

   If the data was collected using the data collection script for UNIX platforms, the generated files contain metadata such as block size, and data collection interval.

If the files do not contain metadata, because the data was collected using operating system commands or the VRAdvisor Wizard, enter the appropriate metadata:

- Specify the block size, if required.
- If no timestamps are present in the file, or if VRAdvisor is unable to parse the timestamps, specify the interval that is used during the data collection.

6 On the **Volume or Disk Selection** panel, select the tab for each selected file. For each file, the wizard lists the disks or volumes for which data has been collected.

When selecting disks or volumes, ensure that you do not select:

- RAID-5 volumes because these are not supported.
- Sub-level volumes (if the volumes are layered volumes). Select only the top-level volumes.
- The volume that you intend to use as the SRL.
- Drives or volumes containing high-activity data that is not be replicated.

  Using VRA to analyze data from drives or volumes containing high-activity data that is not to be replicated, may lead to erroneous results.

Select the volumes or disks to be analyzed, and then click **Next**.

7 The RVG Summary panel displays the disks or volumes that were selected for analysis. The disks or volumes for each analyzed file are grouped under an RVG name.

Click **Back** to modify the selections, or click **Next** to continue.

8 On the **Network Parameters for Analysis** panel, specify the parameters that apply to all defined RVGs.

- **Network Bandwidth Available for Replication** indicates the total bandwidth of the network across which you are replicating. Enter the network bandwidth that will be available for replication. Select the unit for the network bandwidth from the drop-down list. The default is 100 Mbps.

  **Note:** Before specifying the network bandwidth you must also consider the loss of available bandwidth because of the TCP-IP/UDP headers, because VRAdvisor does not handle this.

- **Network Outage Duration** indicates the maximum expected outage times applicable for all defined RVGs. For example, the time during which the network link is unavailable for the network that all the RVGs use for replication. Enter the duration of the network outage in days, hours, or minutes. The default is zero.

Click **Next**.

9 The **RVG Specific Parameters** panel appears. For each RVG, select the tab, and then specify the following parameters:

- **Bandwidth Limit** indicates the bandwidth throttling for that RVG. The default is 0 (zero), which indicates that no bandwidth limit applies.
- **Secondary Outage Duration** indicates the maximum expected outage times specific to that RVG, for example, the time during which the Secondary host for the RVG is unavailable. Enter the outage duration in days, hours, or minutes. The default is one hour.
- **Apply to all RVG(s)** indicates that the same bandwidth limit and outage duration apply to all RVGs. Select this check box to enable the All tab and disable the RVG-specific tabs.

Click **Next**.

10 The **Summary of Inputs** panel appears. The Total Outage Duration column shows the sum of the Network Outage Duration and the Secondary Outage for that RVG.

Click **Back** to modify the parameters, or select **Analyze** to start the analysis. VRAdvisor displays the results of the analysis for the selected data files.

# Understanding the results of the analysis

After the analysis completes, VRAdvisor displays the result. You can also change some parameters and recalculate the result.

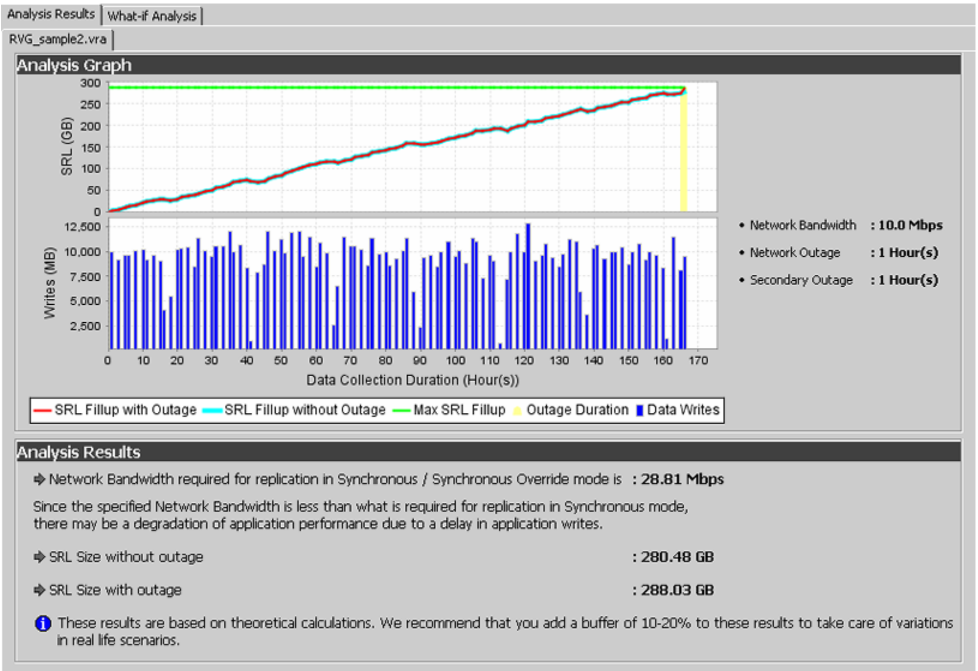
See “Viewing the analysis results” on page 450.

See “Recalculating the analysis results” on page 452.

See “Recording and viewing the results” on page 455.

## Viewing the analysis results

After the analysis completes, the **Analysis Results** panel is displayed by default.



- Network Bandwidth : 10.0 Mbps
- Network Outage : 1 Hour(s)
- Secondary Outage : 1 Hour(s)

➾ Network Bandwidth required for replication in Synchronous / Synchronous Override mode is : 28.81 Mbps

Since the specified Network Bandwidth is less than what is required for replication in Synchronous mode, there may be a degradation of application performance due to a delay in application writes.

➾ SRL Size without outage : 280.48 GB

➾ SRL Size with outage : 288.03 GB

ⓘ These results are based on theoretical calculations. We recommend that you add a buffer of 10-20% to these results to take care of variations in real life scenarios.

The **Analysis Results** panel displays the result of the analysis for each RVG. Select the tab for an RVG to display the result for that particular RVG. The results panel displays the following information:

## Analysis graph

The Analysis Graph section shows the following information:

- The top graph shows the SRL (Storage Replicator Log) fillup in megabytes (MB) on the y-axis. The fillup rate is shown both with specified outages and no outages. The x-axis shows the data write duration values. The peak SRL fillup size is shown against a max outage window that is displayed in yellow, which indicates a worst case scenario.

  **Note:** If SRL fillup value in the graph steadily increases up to maximum during the last data write duration, it indicates that you do not have sufficient network bandwidth for the number of data writes contained in the sample data.

- The bar graph shows the value of the Application Writes in bytes for the y-axis. The x-axis shows the data write duration values.

- To the right of the graphs, the panel displays the values that are specified for network bandwidth and outage parameters.

### Analysis results

The Analysis Results section displays the following information:

- The required network bandwidth synchronous replication. If the required bandwidth is more than the bandwidth that you specified, then VRAdvisor displays a message to indicate that the performance of the application writing to the disk writes is affected.
- The required SRL size with the specified outage.
- The required SRL size with no outage.

**Note:** Veritas recommends that you add a 10-20 percent buffer to these values when setting up the Volume Replicator configuration. VRAdvisor analyzes the data based on the specified values, which can be affected by factors that VRAdvisor does not consider, such as TCP/IP headers overhead or network congestion.

## Recalculating the analysis results

The following topics describe the ways in which you can recalculate the analysis results.

See "Applying different parameters to the existing sample of data" on page 452.

See "Performing What-if analysis" on page 453.

### Applying different parameters to the existing sample of data

You can recalculate the analysis results by changing the values you specified for the network bandwidth and the outage durations.

**To recalculate the analysis results**

1. To change the values you specified, select **File > Change Inputs**.
2. On the **Network Parameters for Analysis** panel, specify new values for any of the fields as required. Click **Next** to specify RVG- specific parameters or click **Back** to change volume or disk selection.
3. Continue using the **Next** and **Back** buttons to navigate through the input panels and change values as required.

4 When the values are correct, click **Next** to navigate to the **Summary of Inputs** panel.

5 Click **Analyze** to start the analysis.

VRAdvisor performs the analysis of the data using the changed values and displays the results.

## Performing What-if analysis

After checking the analysis results, you can use the What-if Analysis panel to do additional calculations, to plan for future requirements or alternative scenarios.

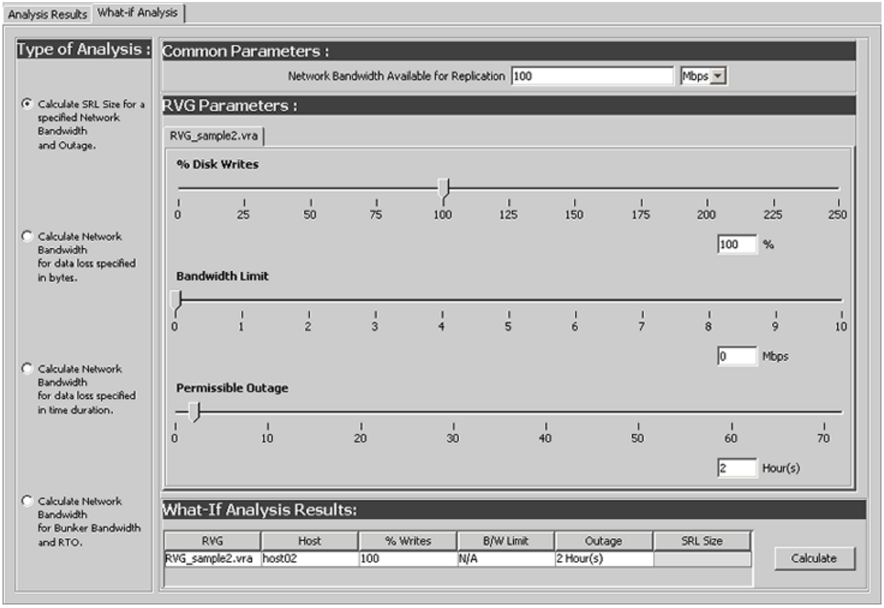
You can vary the parameters and recalculate the results according to different criteria. For example, you can vary the network bandwidth parameter to see what effect it would have on the SRL size or you can specify a potential SRL size and see how much network bandwidth would be required for that SRL size.

**Note:** Before specifying the network bandwidth, you must also consider the loss of available bandwidth due to the TCP-IP/UDP headers as VRAdvisor cannot manage this.

What-if Analysis also enables you to vary the percentage of disk writes as compared to the sample of data that was analyzed. For example, if you anticipate that your future needs will involve 20 percent more disk writes, set the percentage of disk writes to 120% and recalculate.

### To recalculate results using the What-If Analysis

1 Select the **What-If Analysis** tab.



● Calculate SRL Size for a specified Network Bandwidth and Outage.

○ Calculate Network Bandwidth for data loss specified in bytes.

○ Calculate Network Bandwidth for data loss specified in time duration.

○ Calculate Network Bandwidth for Bunker Bandwidth and RTO.

Network Bandwidth Available for Replication 100 Mbps

% Disk Writes 100 %

Bandwidth Limit 0 Mbps

Permissible Outage 2 Hour(s)

| RVG | Host | % Writes | B/W Limit | Outage | SRL Size |
|---|---|---|---|---|---|
| RVG_sample2.vra | host02 | 100 | N/A | 2 Hour(s) | |

2 To recalculate the results, select the appropriate option on the left side of the What-If Analysis panel as follows:

- **Calculate SRL Size for a specified Network Bandwidth and Outage**

  Use this option to calculate the SRL size for a specified network bandwidth and outage duration.

  Available parameters for this option are % Disk Writes and Permissible Outage.

- **Calculate the Network Bandwidth for data loss specified in bytes**

  Use this option to calculate the network bandwidth that would be required to minimize the amount of data loss at the Primary host.

  Available parameters for this option are % Disk Writes and Data loss in bytes.

- **Calculate Network Bandwidth for data loss specified in time duration**

  Use this option to calculate the network bandwidth that would be required to minimize the amount of data loss at the Primary host.

  Available parameters for this option are % Disk Writes and Data loss in time.

- **Calculate Network Bandwidth for Bunker and RTO**

  In a Bunker replication setup, the available bandwidth determines the RPO (Recovery Point Objective) and the RTO (Recovery Time Objective) that can be achieved after a disaster. Use this option to calculate the required bandwidth for a Primary and Secondary site and between a Bunker and Secondary based on the desired RPO and RTO.

  Available parameters for this option are % Disk Writes and RTO. The **Have Bunker** check box indicates that the RVG has a bunker attached. The right side of the panel displays the parameters you can specify for each option and the corresponding slider bars.

3. In the Common Parameters section, change the bandwidth value that is shared by all RVGs.
4. In the RVG Parameters section, select the tab for the RVG that you want to change, and then use the slider bar to specify the value for each parameter. Each slider has a default range of values, which can be customized using the **Preferences** option that is available from the **File** menu. For more information, See "Changing the value ranges on the slider bar" on page 455.
5. Click **Calculate** at the lower region of the panel. The What-if Analysis Results are displayed in this section.

Follow the steps that are given below to change the value ranges for the slider bars.

**Changing the value ranges on the slider bar**

1. Make sure the option for which you want to change the value ranges is selected on the left side of the What-if Analysis panel.
2. Select the **File** > **Preferences** option to display the Preferences panel.

   **Note:** The Preferences dialog box displays parameters corresponding to the calculate the option that you selected.

3. Change the values on the Preferences page as required:
   - Select the Unit for each option from the drop-down box.
   - Specify the appropriate values in the **Maximum** and **Minimum** fields. These values are used to indicate the range of values available on the slider bar.
4. Click **Ok**.

## Recording and viewing the results

VRAdvisor records values that you specified during the analysis phase and the results of the What-if Analysis to a file, which uses the following naming convention:

`VRAdvResults_Datestamp_and_Timestamp.txt`

The file is located at the `Veritas/Volume Replicator Advisor/results` subdirectory.

Every time you start the Analysis wizard, this file is automatically created, and can be referenced later.

# Sizing the SRL

This section provides information about sizing the Storage Replicator Log (SRL). The size of the SRL is critical to the performance of replication. You can use VRAdvisor to help determine the appropriate SRL size. This section describes some of the considerations in determining the size of the SRL. VRAdvisor uses the formulas that are described in this section to determine the appropriate SRL size.

**Note:** The terms Replicator Log and Storage Replicator Log (SRL) mean the same.

## Overview

When the SRL overflows for a particular Secondary, the RLINK corresponding to that Secondary is marked `STALE` and becomes out of date until a complete resynchronization with the Primary is performed. Because resynchronization is a time-consuming process and during this time the data on the Secondary cannot be used, it is important to avoid SRL overflows. The SRL size needs to be large enough to satisfy four constraints:

- It must not overflow for asynchronous RLINKs during periods of peak usage when replication over the RLINK may fall far behind the application.
- It must not overflow while a Secondary RVG is synchronized.
- It must not overflow while a Secondary RVG is restored.
- It must not overflow during extended outages (network or Secondary node).

**Note:** The size of SRL must be at least 110 MB. If size specified for SRL is less than 110 MB, Volume Replicator displays an error message that prompts to specify a value that is equal to or greater than 110 MB.

To determine a size of the SRL, you must determine the size that is required to satisfy each of these constraints individually. Choose a value at least equal to the maximum so that all constraints are satisfied. To perform this analysis, you need the following information:

- The maximum expected downtime for Secondary nodes
- The maximum expected downtime for a network connection
- The method for synchronizing Secondary data volumes with data from Primary data volumes. If the application is shutdown to perform the synchronization, the SRL is not used and the method is not important. Otherwise, this information can include the time that is required to copy data over the network or to copy it to a tape or disk, to send the copy to the Secondary site, and to load the data onto the Secondary data volumes.

**Note:** If Automatic Synchronization option is used to synchronize the Secondary, the above-mentioned step is not a concern.

To perform Secondary backups to avoid complete resynchronization in case of Secondary data volume failure, the following information is required:

- The frequency of Secondary backups
- The maximum expected delay to detect and repair a failed Secondary data volume
- The expected time to reload backups onto the repaired Secondary data volume

## Peak usage constraint

For some configurations, it might be common for replication to fall behind the application during certain period and catch up during others. For example, an RLINK might fall behind during business hours and catch up overnight if its peak bandwidth requirements exceed the network bandwidth. However, for synchronous RLINKs this does not apply as a shortfall in network capacity would cause each application write to be delayed. This in turn causes the application to run more slowly.

For asynchronous RLINKs, the only limit to how far replication can fall behind is the size of the SRL. If it is known that the peak write rate requirements of the application exceed the available network bandwidth, then it becomes important to consider this factor when sizing the SRL.

Assuming that data is available providing the typical application write rate over a series of intervals of equal length, it is simple to calculate the SRL size that is needed to support this usage pattern:

1. Calculate the network capacity over the given interval ($BW_N$).
2. For each interval *n*, calculate the SRL log volume usage ($LU_n$) as the excess of application write rate ($BW_{AP}$) over network bandwidth ($LU_n = BW_{AP(n)} – BW_N$).
3. For each interval, accumulate all the SRL usage values to find the cumulative SRL log size (LS):

$$LS_n = \sum_{i=1 \ldots n} LU_i$$

The largest value that is obtained for any $LS_n$ is the value that should be used for SRL size as determined by the peak usage constraint. For an example of this calculation, See Table C-1 on page 458. The third column, Application, contains the maximum likely application write rate per hour. The fourth column Network shows the network bandwidth. The fifth column SRL Usage shows the difference between application write rate and network bandwidth that is obtained for each interval. The sixth column Cumulative SRL Size shows the cumulative difference every hour. The largest value in column 6 is 37 gigabytes. The SRL should be at least this large for this application.

Several factors can reduce the maximum size to which the SRL can fill up during the peak usage period. The factors that need to be considered are:

- The `latencyprot` characteristic can be enabled to restrict the amount by which the RLINK can fall behind, slowing down the write rate.
- The network bandwidth can be increased to handle the full application write rate. In this example, the bandwidth should be 15 gigabytes/hour—the maximum value in column three.

**Table C-1** Example calculation of SRL size required to support peak usage period

| Hour Starting | Hour Ending | Application (GB/hour) | Network (GB/hour) | SRL Usage (GB) | Cumulative SRL Size (GB) |
|---|---|---|---|---|---|
| 7:00 A.M. | 8:00 A.M. | 6 | 5 | 1 | 1 |
| 8 | 9 | 10 | 5 | 5 | 6 |
| 9 | 10 | 15 | 5 | 10 | 16 |
| 10 | 11 | 15 | 5 | 10 | 26 |

**Table C-1** Example calculation of SRL size required to support peak usage period *(continued)*

| Hour Starting | Hour Ending | Application (GB/hour) | Network (GB/hour) | SRL Usage (GB) | Cumulative SRL Size (GB) |
|---|---|---|---|---|---|
| 11 | 12:00 P.M. | 10 | 5 | 5 | 31 |
| 12:00 P.M. | 1 | 2 | 5 | -3 | 28 |
| 1 | 2 | 6 | 5 | 1 | 29 |
| 2 | 3 | 8 | 5 | 3 | 32 |
| 3 | 4 | 8 | 5 | 3 | 35 |
| 4 | 5 | 7 | 5 | 2 | 37 |
| 5 | 6 | 3 | 5 | -2 | 35 |

## Synchronization period constraint

When a new Secondary is added to an RDS, its data volumes must be synchronized with those of the Primary unless the Primary and the Secondary data volumes have been zero initialized and the application has not yet been started. You also need to synchronize the Secondary after a Secondary data volume failure, in case of SRL overflow or after replication is stopped.

This section applies if you choose *not* to use the automatic synchronization method to synchronize the Secondary. Also, this constraint does not apply if you choose to use a method other than automatic synchronization and if the application on the Primary can be shut down while the data is copied to the Secondary. However, in most cases, it might be necessary to synchronize the Secondary data volumes with the Primary data volumes while the application is still running on the Primary.

If SRL overflows during the synchronization period when the application is running and data is getting accumulated in the SRL, then you must restart the synchronization process. To ensure that the SRL does not overflow during such periods, it is necessary to appropriately size the SRL so that it can hold as much data as the application writes. After replication is started, this data is replicated and the Secondary eventually catches up with the Primary.

Depending on your needs, it may or may not be possible to schedule synchronization during periods of low application write activity. If it is possible to complete the synchronization process during a period of low application write activity, then you must ensure that the SRL is sized such that it can hold all the incoming writes during this period. Otherwise, the SRL may overflow. Using VRAdvisor enables you to arrive at an optimum SRL size.

## Secondary backup constraint

Volume Replicator provides a mechanism to perform periodic backups of the Secondary data volumes. In case of a problem that would otherwise require a complete resynchronization using one of the methods that are described in See “Synchronization period constraint” on page 459., a Secondary backup, if available, can be used to bring the Secondary online much more quickly.

A Secondary backup is made by defining a Secondary checkpoint and then making a raw copy of all the Secondary data volumes. If a failure occurs, then the Secondary data volumes are restored from this local copy and replication proceeds from the checkpoint. Data is replayed from the checkpoint to the present.

The constraint that is introduced by this process is that the Primary SRL must be large enough to hold all the data that is logged in the Primary SRL after the creation of the checkpoint corresponding to the most recent backup. This depends largely on three factors:

- The application write rate.
- The frequency of Secondary backups.
- Minimum SRL size.

  You need to consider an application’s write rate and frequency of Secondary backups to calculate the minimum SRL size. An extra margin should be added to an estimate to cover other possible delays including:

  - Maximum delay before a system administrator detects a data volume failure.
  - Maximum delay to repair or replace the failed drive.
  - Delay to reload disk with data from the backup tape.

  To arrive at an estimate of the SRL size that is needed to support this constraint, first determine the total time period the SRL needs to support by adding the period planned between Secondary backups to the expected time for the three factors mentioned above. Then use the application write rate data to determine for the worst case scenario the amount of data the application can generate over this time period.

  **Note:** Even if only one volume fails, all other volumes need to be restored.

## Secondary downtime constraint

When the network connection to a Secondary node or the Secondary node itself goes down, the RLINK on the Primary node detects the broken connection and responds. If the RLINK has its `synchronous` attribute set to `fail`, the response is to fail all subsequent write requests until the connection is restored. In this case, the SRL does not grow and hence, the downtime constraint is irrelevant. For all other types of RLINKs, incoming write requests accumulate in the SRL until the connection is restored. Thus, the SRL must be large enough to hold the maximum output that the application can be expected to generate over the maximum possible downtime.

Maximum downtimes may be difficult to estimate. In some cases, the vendor may guarantee that failed hardware or network connections would be repaired within a stipulated period. However, if the repair is not completed within the guaranteed period, then SRL may overflow. Hence, it is recommended that a safety margin should always be added to any such arrived estimate.

To arrive at an SRL size estimate to support this constraint, first obtain estimates for the maximum downtimes which the Secondary node and network connections can reasonably be expected to incur. Then, use the application write rate data to determine, for the worst case scenario, the amount of data the application can generate over this time period. With the introduction of the `autodcm` mode of SRL overflow protection, sizing the SRL for downtime is not essential to prevent SRL overflow because the changed blocks are no longer stored in the SRL. However, note that the Secondary is inconsistent during the replay of the DCM, and hence it is still important for the SRL to be large enough to cover most eventualities.

## Additional factors

Once estimates of required SRL size have been obtained under each of the constraints described above, several additional factors must be considered.

For the synchronization period, downtime and Secondary backup constraints, it is likely that a period of peak usage may follow any of these situations. In this case, the Secondary can continue to fall further behind rather than catching up during the peak usage period. As a result, it might be necessary to add the size that is obtained from the peak usage constraint to the maximum size that is obtained using the other constraints. Note that this applies even for synchronous RLINKs, which are not normally affected by the peak usage constraint as after a disconnect they act as asynchronous RLINKs until caught up.

It is also possible that other situations can occur requiring additions to constraints. For example, a long network failure can follow a synchronization period or a Secondary node failure can follow a network failure. Whether and to what degree to plan for unlikely occurrences requires weighing the cost of additional storage against the cost of additional downtime that is caused by SRL overflow.

Once an estimate has been computed, one more adjustment must be made to account for the fact that all data that is written to the SRL also includes some header information. This adjustment must take into account the typical size of write requests. Each request uses at least one additional disk block for header information.

For AIX, Linux, and Solaris, the adjustments are as follows:

**Table C-2**

| If Average Write Size is: | Add This Percentage to SRL Size: |
|---|---|
| 512 bytes | 100% |
| 1K | 50% |
| 2K | 25% |
| 4K | 15% |
| 8K | 7% |
| 10K | 5% |
| 16K | 4% |
| 32K or more | 3% |

For HP-UX, the adjustments are as follows:

**Table C-3**

| If Average Write Size is: | Add This Percentage to SRL Size: |
|---|---|
| 1K | 100% |
| 2K | 50% |
| 4K | 25% |
| 8K | 13% |
| 10K | 10% |
| 16K | 6% |
| 32K or more | 3% |

## Example

This section shows how to calculate the SRL size for a Volume Replicator configuration. First, collect the relevant parameters for the site as follows:

**Table C-4**

| | |
|---|---|
| Application peak write rate | 1 Gigabyte/hour |
| Duration of peak | 8:00 A.M. - 8:00 P.M. |
| Application off-peak write rate | 250 megabytes/hour |
| Average write size | 2 kilobytes |
| Number of Secondary sites | 1 |
| Type of RLINK | Synchronous=override |
| Synchronization Period: | |
| Application shutdown | No |
| Copy data to tape | 3 Hours |
| Send tapes to Secondary site | 4 Hours |
| Load data | 3 Hours |
| Total | 10 Hours |
| Maximum downtime for Secondary node | 4 Hours |
| Maximum downtime for network | 24 Hours |
| Secondary backup | Not used |

Because synchronous RLINKs are to be used, the network bandwidth must be sized to handle the peak application write rate to prevent the write latency from growing. Thus, the peak usage constraint is not an issue and the maximum constraint is that the network can be out for 24 hours. The amount of data accumulating in the SRL over this period would be:

(Application peak write rate x Duration of peak) +

(Application off-peak write rate x Duration of off-peak).

In this case, the calculation would appear as follows:

1 GB/hour x 12 hours + 1/4 GB/hour x 12 = 15 GB

An adjustment of 25% is made to handle header information. Since the 24-hour downtime is already an extreme case, no additional adjustments are needed to handle other constraints. The result shows that the SRL should be at least 18.75 gigabytes.