

Dell EMC SRDF

Introduction

Notes, cautions, and warnings

 **NOTE:** A NOTE indicates important information that helps you make better use of your product.

 **CAUTION:** A CAUTION indicates either potential damage to hardware or loss of data and tells you how to avoid the problem.

 **WARNING:** A WARNING indicates a potential for property damage, personal injury, or death.

Figures.....	6
Tables.....	7
Preface.....	8
Chapter 1: Introduction.....	10
What is SRDF?.....	11
Disaster recovery.....	11
High availability.....	13
Data migration.....	14
SRDF concepts.....	15
SRDF device pairs.....	15
Invalid tracks in SRDF pairs.....	18
SRDF groups.....	19
Dynamic devices.....	20
SRDF modes of operation.....	21
SRDF device states.....	24
SRDF consistency.....	29
Director boards, links, and ports.....	29
Chapter 2: Disaster recovery.....	30
Two-site configurations.....	31
Three-site configurations.....	33
Concurrent SRDF solutions.....	35
Cascaded SRDF solutions.....	36
SRDF/Star solutions.....	36
Four-site configurations.....	44
Four-site FBA configurations.....	44
Four-site mainframe configurations (SRDF/SQAR).....	45
SRDF recovery scenarios.....	47
Planned failover (SRDF/S).....	47
Unplanned failover	48
Failback to the primary array.....	48
Recovery for a large number of invalid tracks.....	48
Temporary link loss.....	48
Permanent link loss (SRDF/A).....	49
SRDF/A session cleanup.....	49
Failback from R2 devices.....	49
Replication of VMWare vVols.....	50
vVol architecture.....	50
Remote replication in a vVols environment.....	51
Management facilities.....	52
Chapter 3: High availability.....	53
SRDF/Metro	54

SRDF/Metro life cycle.....	57
Initial provisioning and configuration setup.....	57
Add and remove devices.....	58
Remove the SRDF/Metro configuration.....	59
SRDF/Metro configuration changes.....	60
Add devices.....	60
Remove devices.....	61
SRDF/Metro resilience.....	62
Witness.....	62
Device Bias.....	68
Preventing data loss.....	69
Specifying the resiliency method.....	69
Mobility ID with ALUA.....	69
Disaster recovery facilities.....	70
Highly available disaster recovery (SRDF/Metro Smart DR).....	70
Independent disaster recovery.....	73
Deactivate SRDF/Metro.....	74
SRDF/Metro restrictions.....	75
Chapter 4: Data migration.....	76
Introduction to data migration using SRDF.....	77
Non-disruptive migration.....	77
Migration from VMAX array.....	77
Migration from VMAX All Flash or VMAX3.....	78
Migrating data with concurrent SRDF.....	80
Replacing R1 devices with new R1 devices.....	80
Replacing R2 devices with new R2 devices.....	81
Replacing R1 and R2 devices with new R1 and R2 devices.....	82
Migration-only SRDF.....	83
Device Migration operations requirements	84
Chapter 5: SRDF I/O operations.....	85
SRDF write operations.....	86
Write operations in synchronous mode.....	86
Write operations in asynchronous mode.....	86
Cycle switching in asynchronous mode.....	87
Write operations in cascaded SRDF.....	90
SRDF read operations.....	91
Read operations if R1 local copy fails.....	91
Read operations from R2 devices.....	91
SRDF/A resilience and performance features.....	92
Tunable cache.....	92
SRDF/A cache data offloading	92
Transmit Idle.....	93
Write folding.....	93
Write pacing	93
Chapter 6: Interfamily compatibility.....	95
Overview.....	96

SRDF supported features.....	96
Chapter 7: Management tools.....	98
Solutions Enabler.....	99
Unisphere.....	99
SRDF/TimeFinder Manager for IBM i.....	100
Mainframe management tools.....	101
Mainframe Enablers.....	101
Geographically Dispersed Disaster Restart (GDDR).....	102
Chapter 8: More information.....	103
Solutions Enabler CLI.....	104
Unisphere.....	104
Mainframe Enablers.....	104
GDDR.....	104
SRDF/TimeFinder Manager for IBM i.....	104
SRDF/Metro vWitness.....	104
SRDF Interfamily Compatibility.....	104
Storage arrays.....	105


1	Data replicated to one additional array.....	11
2	Data replicated to two arrays simultaneously.....	12
3	SRDF/Metro example.....	13
4	Migration example.....	14
5	R1 and R2 devices.....	15
6	R11 device in concurrent SRDF.....	16
7	R21 device in cascaded SRDF.....	17
8	R22 devices in cascaded and concurrent SRDF/Star.....	17
9	Host interface view and SRDF view of states.....	24
10	Concurrent SRDF topology.....	35
11	Cascaded SRDF topology.....	36
12	Concurrent SRDF/Star.....	37
13	Concurrent SRDF/Star with R22 devices.....	38
14	Cascaded SRDF/Star.....	39
15	R22 devices in cascaded SRDF/Star.....	40
16	Concurrent SRDF/Star with GDDR.....	41
17	Cascaded SRDF/Star with GDDR.....	42
18	SRDF 4-site FBA configuration.....	44
19	SRDF/SQAR with Autoswap environment.....	45
20	Planned failover: before personality swap.....	47
21	Planned failover: after personality swap.....	47
22	Failover to Site B, Site A and production host unavailable.....	48
23	The implementation of vVols in PowerMax OS.....	50
24	SRDF replication of vVols in PowerMaxOS.....	51
25	SRDF/Metro.....	54
26	SRDF/Metro Array Witness and groups.....	63
27	SRDF/Metro vWitness vApp and connections.....	64
28	SRDF/Metro Witness single failure scenarios.....	67
29	SRDF/Metro Witness multiple failure scenarios.....	68
30	SRDF/Metro Smart DR.....	70
31	Disaster recovery for SRDF/Metro.....	73
32	Migrating data and replacing the original primary array (R1).....	80
33	Migrating data and removing the original secondary array (R2).....	81
34	Migrating data and replacing the original primary (R1) and secondary (R2) arrays.....	82
35	Write I/O flow: simple synchronous SRDF.....	86
36	SRDF/A SSC cycle switching – multi-cycle mode.....	88
37	SRDF/A SSC cycle switching – legacy mode.....	88
38	SRDF/A MSC cycle switching – multicycle mode.....	89
39	Write commands to R21 devices.....	90

1	SRDF pair states.....	25
2	R1 device accessibility.....	27
3	R2 device accessibility.....	27
4	Possible SRDF device and link state combinations.....	28
5	SRDF two-site configurations.....	31
6	SRDF multisite solutions.....	33
7	Management responsibilities in a vVol environment.....	51
8	Limitations of the migration-only mode.....	83
9	SRDF features by hardware platform/operating environment.....	96
10	Unisphere tasks.....	99

Preface

As part of an effort to improve its product lines, Dell Technologies periodically releases revisions of its software and hardware. Therefore, some functions described in this document might not be supported by all versions of the software or hardware currently in use. The product release notes provide the most up-to-date information on product features.

Contact your Dell Technologies technical support professional if a product does not function properly or does not function as described in this document.

 **NOTE:** This document was accurate at publication time. Go to Dell Technologies Online Support (<https://dell.com/support/home>) to ensure that you are using the latest version of this document.

Purpose

This document provides an introduction to the Symmetrix Remote Data Facility (SRDF) and its uses in disaster recovery, high availability, and data migration applications.

Audience

This document is intended for Dell Technologies customers who want an overview of SRDF and its applications.

Related documentation

Information on the storage arrays that SRDF runs on is in the following publications:

- *Dell EMC PowerMax Family Product Guide*
- *Dell EMC VMAX All Flash Product Guide for VMAX 250F, 450F, 850F, 950F with HYPERMAX OS*
- *EMC VMAX3 Family Product Guide for VMAX 100K, VMAX 200K, VMAX 400K with HYPERMAX OS*

Where to get help

Dell Technologies support, product, and licensing information can be obtained as follows:

Product information	Dell EMC technical support, documentation, release notes, software updates, or information about Dell EMC products can be obtained at https://www.dell.com/support/home (registration required) or https://www.dell.com/support/home/en-us/documentation/vmax-all-flash-family.htm .
Technical support	To open a service request through the Dell EMC Online Support (https://www.dell.com/support/home) site, you must have a valid support agreement. Contact your Dell EMC sales representative for details about obtaining a valid support agreement or to answer any questions about your account.
Technical support	<p>Dell EMC offers various support options.</p> <ul style="list-style-type: none">• Support by Product: Dell EMC offers consolidated, product-specific information through the Dell EMC Online Support site. <p>The Support by Product web pages: https://www.dell.com/support/home, select Product Support. These pages offer quick links to Documentation, White Papers, Advisories (such as frequently used Knowledgebase articles) and Downloads. They also offer dynamic content such as presentations, discussion, relevant Customer Support Forum entries, and a link to Dell EMC Live Chat.</p> <ul style="list-style-type: none">• Dell EMC Live Chat: Open a Chat or instant message session with a Dell EMC Support Engineer.

Your comments

Your suggestions help improve the accuracy, organization, and overall quality of the documentation. Send your comments and feedback to: VMAXContentFeedback@emc.com

Introduction

This chapter introduces SRDF, lists its uses, and defines SRDF's concepts.

Topics:

- [What is SRDF?](#)
- [SRDF concepts](#)

What is SRDF?

The Symmetrix Remote Data Facility (SRDF) maintains real-time (or near real-time) copies of data on a production storage array at one or more remote storage arrays. SRDF has three primary applications:

- Disaster recovery
- High availability
- Data migration

This is an introduction to SRDF, its uses, configurations, and terminology. The rest of this section provides a summary of the applications for SRDF ([SRDF device pairs](#) on page 15 explains the device naming conventions used in the diagrams).

Disaster recovery

In disaster recovery, SRDF maintains a real-time copy of the data of one or more devices on a storage array at one or more additional arrays. This provides the means to restart host applications should the main array become unavailable for any reason. Typically, each array is on a separate site from all the others in the SRDF configuration.

For example, this diagram shows a configuration where data is replicated to one additional array:

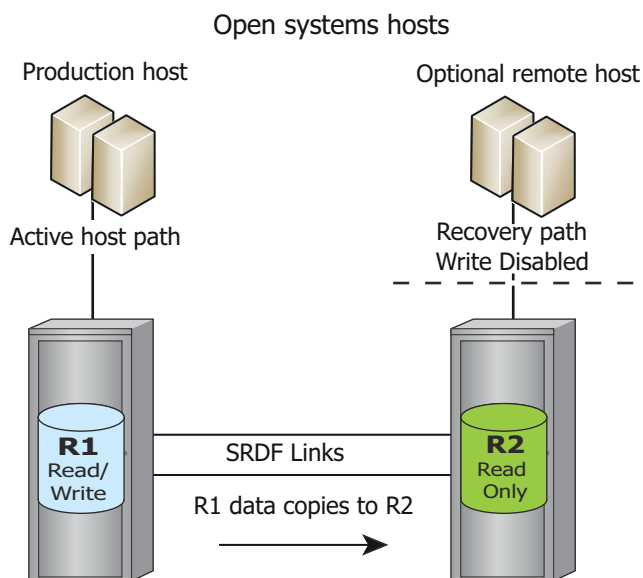


Figure 1. Data replicated to one additional array

The next example shows data being replicated to two additional arrays simultaneously. This improves redundancy and data security:

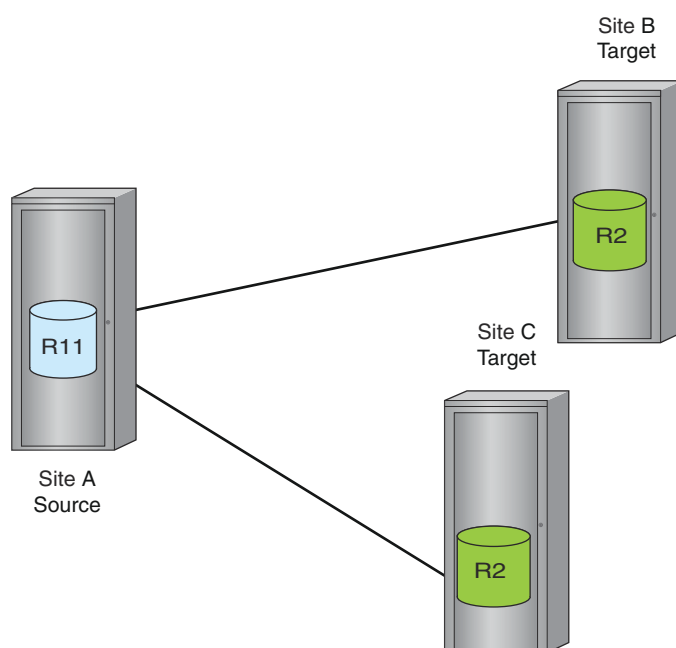


Figure 2. Data replicated to two arrays simultaneously

[Disaster recovery](#) on page 30 describes SRDF's disaster recovery facilities, and system configurations in more detail.

High availability

In other SRDF configurations, devices on the primary array are Read/Write accessible to the application host while devices on the additional arrays are Read Only/Write Disabled. However, in an SRDF high availability configuration:

- Devices on the additional array are Read/Write accessible to the application host.
- The application host can write to both sides of the device pair.
- The devices on the additional array assume the same external identity (such as geometry and device identifier) as the devices on the primary array.

This shared identity means that the devices appear to the application host as a single, virtual device across the two arrays. Using two devices improves the availability of the data they contain. One device can become unavailable without impacting on the host application as the second device continues to operate.

Such a configuration, known as SRDF/Metro, can be deployed in a single, multipathed host environment, or in a clustered environment as this diagram shows:

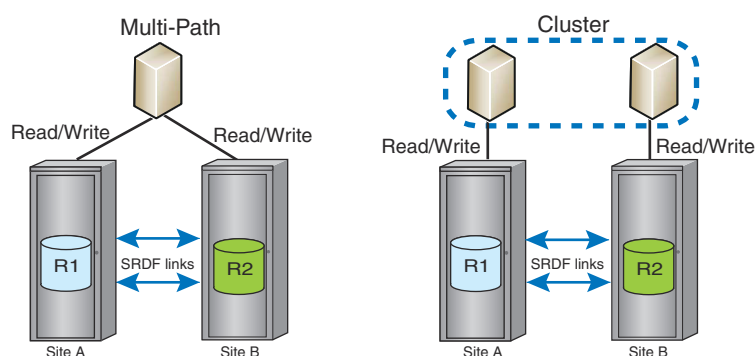


Figure 3. SRDF/Metro example

[High availability](#) on page 53 describes SRDF high availability and its system configurations in more detail.

Open systems (FBA) only

SRDF/Metro is available in open systems (FBA) and IBM i D910¹ environments only. The mainframe environment has its own high availability configuration called AutoSwap. The publications in the Mainframe and GDDR sections of [More information](#) on page 103 contain details of AutoSwap and its capabilities.

¹ IBM i D910 requires PowerMaxOS 5978.444.444 or later.

Data migration

The data migration capabilities of SRDF enable devices on either side of a 2-array configuration to be replaced with new devices without interrupting disaster recovery operations. To do this, the configuration is enhanced with a third array that contains the new devices and data is replicated to that in addition to the normal operation. Once the migration is complete the devices being replaced can be taken out of the configuration leaving one of the original arrays and the new one.

For example, this diagram shows the replacement of R2 devices with new devices using SRDF migration facilities:

- The initial two-array topology
- The interim three-array topology
- The final two-array topology

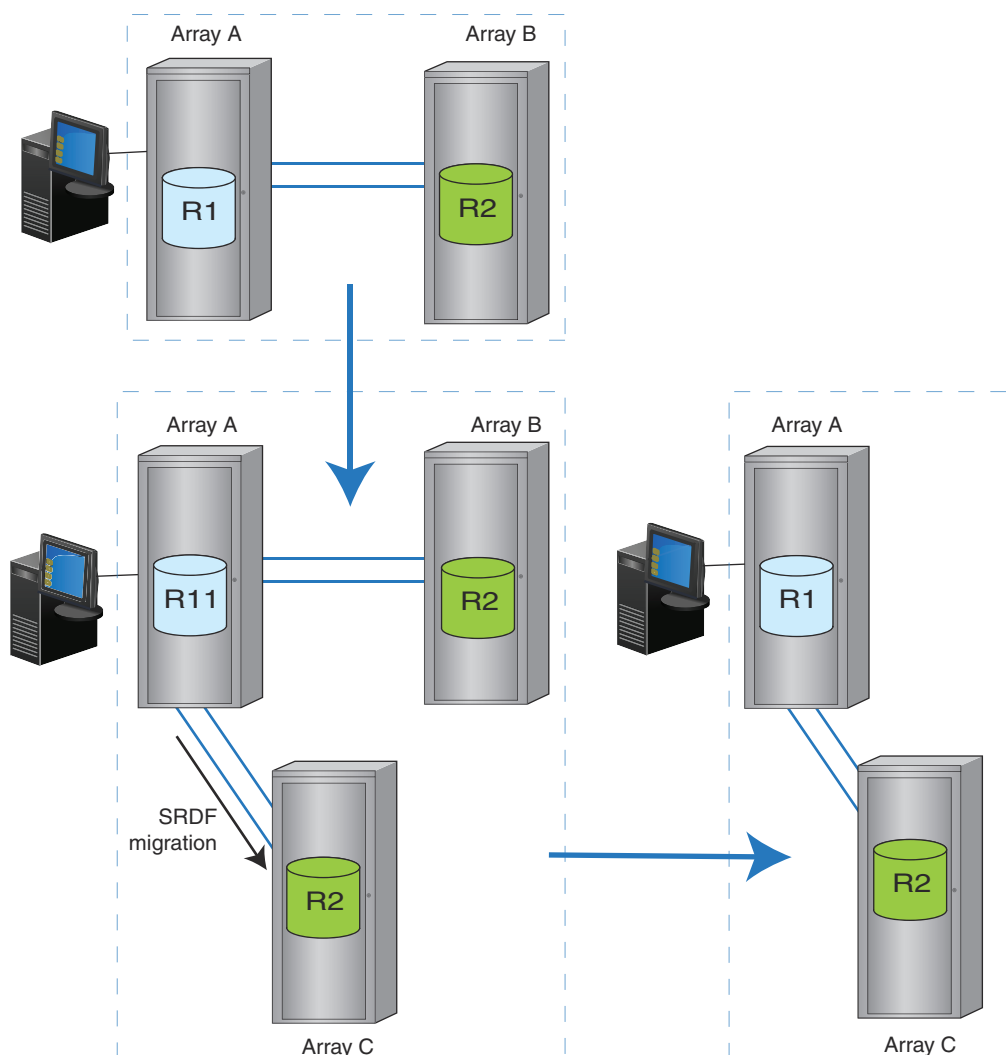


Figure 4. Migration example

[Data migration](#) on page 76 describes SRDF's migration capabilities and system configurations in more detail.

SRDF concepts

SRDF device pairs

An SRDF device pair is a logical device that is paired with another logical device that resides in a second array. The arrays are connected by SRDF links.

Encapsulated Data Domain devices that are used for Storage Direct cannot be part of an SRDF device pair.

R1 and R2 devices

An R1 device is the member of the device pair at the source (production) site. R1 devices are generally Read/Write accessible to the application host.

An R2 device is the member of the device pair at the target (remote) site. During normal operations, host I/O writes to the R1 device are mirrored over the SRDF links to the R2 device. In general, data on R2 devices is not available to the application host while the SRDF relationship is active. In SRDF synchronous mode, however, an R2 device can be in Read Only mode that allows a host to read from the R2.

In a typical environment:

- The application production host has Read/Write access to the R1 device.
- An application host connected to the R2 device has Read Only (Write Disabled) access to the R2 device.

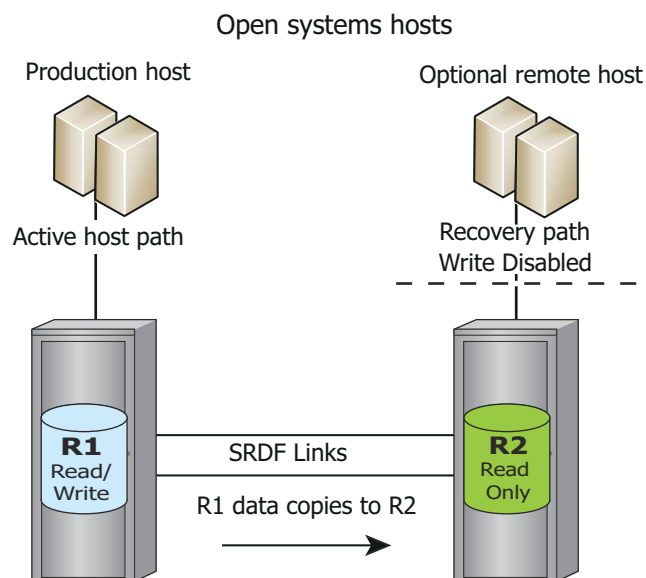


Figure 5. R1 and R2 devices

R11 devices

R11 devices operate as the R1 device for two R2 devices. Links to both R2 devices are active.

R11 devices are typically used in 3-site concurrent configurations where data on the R11 site is mirrored to two secondary (R2) arrays:

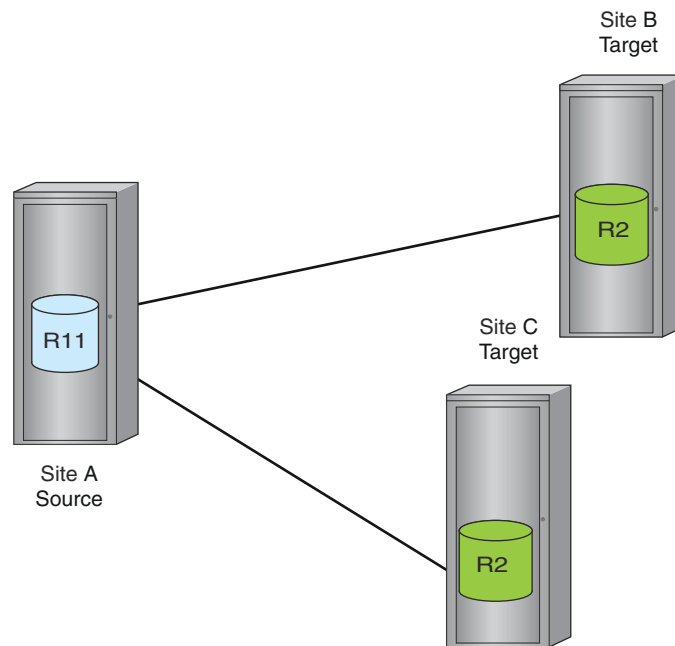


Figure 6. R11 device in concurrent SRDF

R21 devices

R21 devices have a dual role and are used in cascaded 3-site configurations where:

- Data on the R1 site is synchronously mirrored to a secondary (R21) site, and then
- Asynchronously mirrored from the secondary (R21) site to a tertiary (R2) site:

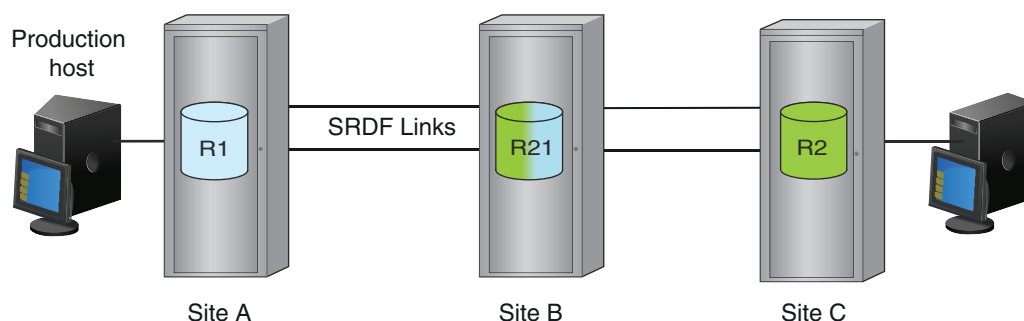


Figure 7. R21 device in cascaded SRDF

The R21 device acts as a R2 device that receives updates from the R1 device, and as a R1 device that sends updates to the R2 device.

When the R1->R21->R2 SRDF relationship is established, no host has write access to the R21 device.

In arrays that run Enginuity, the R21 device can be diskless. That is, it consists solely of cache memory and does not have any associated storage device. It acts purely to relay changes in the R1 device to the R2 device. This capability requires the use of thick devices. Systems that run PowerMaxOS or HYPERMAX OS contain thin devices only, so setting up a diskless R21 device is not possible on arrays running those environments.

R22 devices

R22 devices:

- Have two R1 devices, only one of which is active at a time.
- Are typically used in cascaded SRDF/Star and concurrent SRDF/Star configurations to decrease the complexity and time required to complete failover and failback operations.
- Let you recover without removing old SRDF pairs and creating new ones.

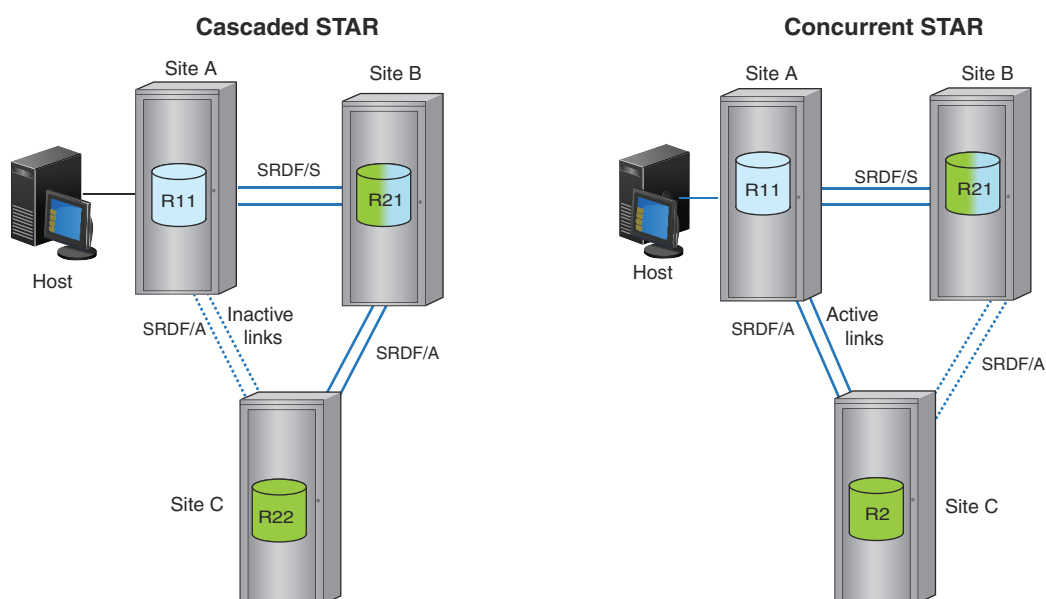


Figure 8. R22 devices in cascaded and concurrent SRDF/Star

Invalid tracks in SRDF pairs

Invalid tracks are tracks that are not synchronized between the two devices in a SRDF pair. They occur when either member of the pair cannot communicate with its partner; for example due to a failure of the SRDF link between the storage arrays. On both sides of the configuration, the storage arrays record the number of tracks that are owed to the other side.

Once the two devices can communicate once again, the invalid tracks need resolving between the pair. There are two ways to resolve the tracks:

- Copy the modified R1 tracks to the R2 side.
Any tracks that were modified on the R2 side are overwritten with the data for the corresponding tracks on the R1 side.
- Copy the modified R2 tracks to the R1 side.
Any tracks that were modified on the R1 side are overwritten with the data for the corresponding tracks on the R2 side.

Example: Unavailable SRDF link or R2 device

Here, the SRDF link is unavailable for some reason, the R2 device is unavailable, or both the link and the R2 device are unavailable. The R1 device, however, remains write accessible to the application host. While this situation exists, the R1 device receives I/O from the application host, and invalid tracks accumulate on the R1 array.

Once the SRDF link and the R2 device are available again, the array containing the R1 array sends the invalid tracks to the R2 device so that the two devices are synchronized once more.

Example: R1 unavailable

Here, the R1 device has become unavailable for some reason. To maintain service to the application host, processing is moved to the R2 device. That is, the R2 device is made write accessible to the application host, and it receives I/O from that host. While this situation exists, invalid tracks accumulate at the R2 array.

Once the R1 device is available again, the array containing the R2 device sends the invalid tracks to the R1 device. Once the two devices are fully synchronized, processing returns to the R1 device and the R2 device is made write protected to the application host.

SRDF groups

An SRDF group defines the logical relationship between SRDF devices and directors on both sides of a SRDF pair.

Group properties

The properties of an SRDF group are:

- Label (name)
- Set of ports on the local array used to communicate over the SRDF links
- Set of ports on the remote array used to communicate over the SRDF links
- Local group number
- Remote group number
- One or more pairs of devices

The devices in the group share the ports and associated CPU resources of the port's directors.

Advanced properties of an SRDF group include:

- Link Limbo mode — The amount of time that the array's operating environment waits after the SRDF link goes down before updating the link's status.
- Link Domino mode — Specifies whether to force SRDF devices into the Not Ready state to the application host if, for example, host I/Os cannot be delivered across the SRDF link.
- Autolink recovery — Specifies whether SRDF automatically restores the SRDF links when they become available again after an earlier failure.
- Compression — Specifies whether to use compression when sending data over the SRDF links. Both hardware and software compression are available and can be used independently or together.

Types of group

There are two types of SRDF group:

- Static
- Dynamic

Static groups are defined in the local array's configuration file. Dynamic groups are defined using SRDF management tools and their properties stored in the array's cache memory.

On arrays running PowerMaxOS or HYPERMAX OS all SRDF groups are dynamic.

Group membership

An SRDF device is a member of as many SRDF groups as there are mirrors of that device. So, in a simple, 2-site configuration (see [R1 and R2 devices](#) on page 15) that consists of R1 and R2 devices, each device is a member of one group. In a concurrent SRDF configuration (see [R11 device in concurrent SRDF](#) on page 16), the R11 device is a member of two groups, one for each R2 mirror. The R2 devices are each in a single group.

Dynamic devices

Dynamic SRDF devices are SRDF devices that allow flexible control over the SRDF solution. You can configure and control dynamic SRDF devices using the SRDF management tools. Dynamic device attributes are stored in the mirrored and protected array cache memory.

SRDF management tools can modify the attributes of dynamic SRDF devices in these ways:

- Create a new R1/R2 pair relationship from non-SRDF devices.
- Terminate and establish an SRDF relationship with a new R2 device.
- Swap personalities between R1 and R2 devices.
- Move R1/R2 pairs between SRDF groups.

R1/R2 personality swap

SRDF devices can dynamically swap “personality” between R1 and R2. After a personality swap:

- The R1 in the device pair becomes the R2 device, and
- The R2 becomes the R1 device.

Swapping R1/R2 personalities allows the application to be restarted at the remote site if an application fails at the production site. After a swap, the R2 side (now R1) can control operations while being remotely mirrored at the primary (now R2) site.

An R1/R2 personality swap is not possible:

- If the R2 device is larger than the R1 device.
- If the device to be swapped is participating in an active SRDF/A session.
- In SRDF/EDP topologies diskless R11 or R22 devices are not valid end states.
- If the device to be swapped is the target device of any TimeFinder or Dell EMC Compatible flash operations.

SRDF modes of operation

SRDF modes of operation address different service level requirements and determine:

- How R1 devices are remotely mirrored across the SRDF links.
- How I/Os are processed.
- When the host receives acknowledgment of a write operation relative to when the write is replicated.
- When writes “owed” between partner devices are sent across the SRDF links.

The mode of operation may change in response to control operations or failures:

- The primary mode (synchronous or asynchronous) is the configured mode of operation for a given SRDF device, range of SRDF devices, or an SRDF group.
- The secondary mode is adaptive copy. Adaptive copy mode moves large amounts of data quickly with minimal host impact. Adaptive copy mode does not provide restartable data images at the secondary site until no new writes are sent to the R1 device and all data has finished copying to the R2.

Use adaptive copy mode to synchronize new SRDF device pairs or to migrate data to another array. When the synchronization or migration is complete, you can revert to the configured primary mode of operation.

Synchronous mode

SRDF/Synchronous (SRDF/S) maintains a real-time mirror image of data between the R1 and R2 devices. The recommended distance between the devices is 200 km (125 miles) or less because application latency may rise to unacceptable levels at longer distances.

Host writes are written simultaneously to both arrays in real time before the application I/O completes. Acknowledgments are not sent to the host until the data is stored in cache on both arrays.

[Write operations in synchronous mode](#) on page 86 and [SRDF read operations](#) on page 91 have more information about I/O operations in synchronous mode.

Asynchronous mode

SRDF/Asynchronous (SRDF/A) maintains a dependent-write consistent copy between the R1 and R2 devices across any distance with no impact to the application.

Host writes are collected for a configurable interval into “delta sets”. Delta sets are transferred to the remote array in timed cycles.

SRDF/A operations vary depending on whether the SRDF session mode is single or multi-session with Multi Session Consistency (MSC) enabled:

- For single SRDF/A sessions, cycle switching is controlled by the array's operating environment. Each session is controlled independently, whether it is in the same or multiple arrays.
- For multiple SRDF/A sessions in MSC mode, multiple SRDF groups are in the same SRDF/A MSC session. Cycle switching is controlled by SRDF host software to maintain consistency.

[SRDF/A MSC cycle switching](#) on page 89 has more information on I/O operations in asynchronous mode.

Active mode

Active mode is used in SRDF/Metro high-availability configurations. When in Active mode the R1 and R2 devices of a configuration appear to the host as a single, federated device. The host can write to both of the devices and SRDF/Metro ensures that both sides hold identical data that is complete and up to date.

[High availability](#) on page 53 has more information about SRDF/Metro and high availability.

Adaptive copy modes

Adaptive copy modes:

- Transfer large amounts of data without impact on the host.
- Transfer data during data center migrations and consolidations, and in data mobility environments.
- Allow the R1 and R2 devices to be out of synchronization by up to a user-configured maximum *skew value*. If the maximum skew value is exceeded, SRDF starts the synchronization process to transfer updates from the R1 to the R2 devices.
- Are secondary modes of operation for SRDF/S. The R1 devices revert to SRDF/S when the maximum skew value is reached and remain in SRDF/S until the number of tracks out of synchronization is lower than the maximum skew.

There are two types of adaptive copy mode:

- [Adaptive copy disk](#) on page 22
- [Adaptive copy write pending](#) on page 22

Adaptive copy disk


In adaptive copy disk mode, write requests accumulate on the R1 device (not in cache). A background process sends the outstanding write requests to the corresponding R2 device. The background copy process scheduled to send I/Os from the R1 to the R2 devices can be deferred if:

- The write requests exceed the maximum R2 write pending limits, or
- The write requests exceed 50 percent of the primary or secondary array write pending space.

Adaptive copy write pending

In adaptive copy write pending mode, write requests accumulate in cache on the primary array. A background process sends the outstanding write requests to the corresponding R2 device.

Adaptive copy write-pending mode reverts to the primary mode if the device, cache partition, or system write pending limit is near, regardless of whether the maximum skew value specified for each device is reached.

 **NOTE:** Adaptive copy write pending mode is not available when the R1 side of an SRDF device pair is on an array running PowerMaxOS or HYPERMAX OS.

Domino modes

Under typical conditions, when one side of a device pair becomes unavailable, new data written to the device is marked for later transfer. When the device or link is restored, the two sides synchronize.

Domino modes force SRDF devices into the Not Ready state to the host if one side of the device pair becomes unavailable.

Domino mode can be enabled/disabled for any:

- Device (domino mode) — If the R1 device cannot successfully mirror data to the R2 device, the next host write to the R1 device causes the device to become Not Ready to the host connected to the primary array.
- SRDF group (link domino mode) — If the last available link in the SRDF group fails, the next host write to any R1 device in the SRDF group causes all R1 devices in the SRDF group to become Not Ready to their hosts.

Link domino mode is set at the SRDF group level and only impacts devices where the R1 is on the side where it is set.

Geometry Compatibility Mode

In Enginuity 5876, the track size of an FBA device is 64 KB, while in PowerMaxOS 5978 and HYPERMAX OS 5977 the track size is 128 KB. So an array running PowerMaxOS or HYPERMAX OS cannot create a device that is the same size as a device that has an odd number of cylinders on an array running Enginuity in a mixed SRDF configuration. However, SRDF requires that the devices in a device pair are the same size.

PowerMaxOS and HYPERMAX OS manage the difference in size automatically using the device attribute Geometry Compatibility Mode (GCM). A device with GCM set is presented as being half a cylinder smaller than its configured size. This enables full functionality in a mixed configuration for SRDF, TimeFinder, SnapVX, and TimeFinder emulations (TimeFinder Clone, TimeFinder VP Snap, and TimeFinder/Mirror) and ORS.

The GCM attribute can be set in two ways:

- Automatically on a target device when it is on an array running PowerMaxOS or HYPERMAX OS and the source device is on an array running Enginuity 5876
- Manually using the Solutions Enabler CLI, Mainframe Enablers SRDF Host Component, or Unisphere

Notes:

- Do not set GCM on devices that are mounted and under the control of a Local Volume Manager (LVM).
- Clear the GCM flag before mapping the device to a host. Otherwise, to clear the attribute, the device must be unmapped from the host which results in a data outage.
- The GCM setting for a device cannot be changed when the target of the data device is already part of another replication session.

SRDF device states

An SRDF device's state is determined by a combination of two views; host interface view and SRDF view, as shown in this diagram.

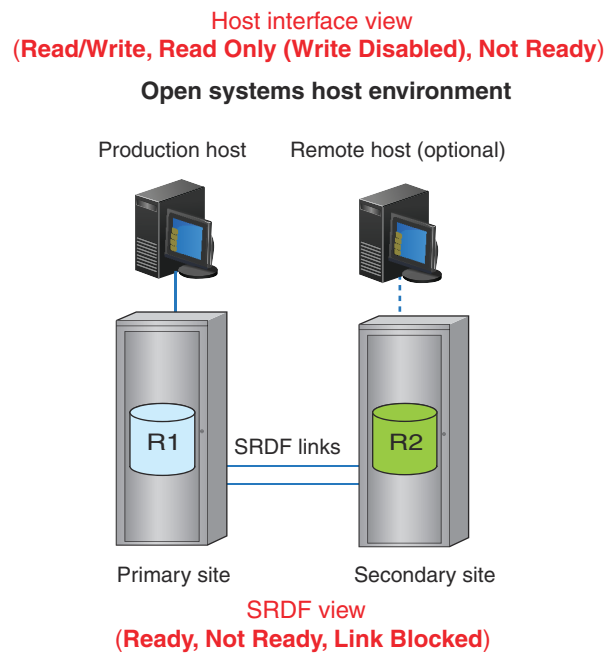


Figure 9. Host interface view and SRDF view of states

Host interface view

The host interface view is the SRDF device state as seen by the application host.

R1 device states

An R1 device presents one of the following states to a host connected to it:

- Read/Write (Write Enabled)—The R1 device is available for Read/Write operations. This is the default R1 device state.
- Read Only (Write Disabled)—The R1 device responds with Write Protected to all write operations to that device.
- Not Ready—The R1 device responds Not Ready to the host for read and write operations to that device.

R2 device states

An R2 device presents one of the following states to a host connected to it:

- Read Only (Write Disabled)—The R2 device responds Write Protected to the host for all write operations to that device.
- Read/Write (Write Enabled)—The R2 device is available for read/write operations. This state is possible in recovery or parallel processing operations.
- Not Ready—The R2 device responds Not Ready (Intervention Required) to the host for read and write operations to that device.

SRDF view

The SRDF view is composed of the SRDF state and internal SRDF device state. These states indicate whether the device is available to send data across the SRDF links, and able to receive software commands.

R1 device states

An R1 device can have the following states for SRDF operations:

- Ready—The R1 device is ready for SRDF operations.
The R1 device is able to send data across the SRDF links.
True even if local mirror(s) of the R1 device are Not Ready for I/O operations.
- Not Ready (SRDF mirror Not Ready)—The R1 device is Not Ready for SRDF operations.

NOTE: When the R2 device is placed into a Read/Write state to the host, the corresponding R1 device is automatically placed into the SRDF mirror Not Ready state.

R2 device states

An R2 device can have the following states for SRDF operations:

- Ready—The R2 device receives the updates propagated across the SRDF links and can accept SRDF host-based software commands.
- Not Ready—The R2 device can receive updates propagated from the primary array, but cannot accept SRDF host-based software commands.
- Link blocked (LnkBlk) — Applicable only to R2 SRDF mirrors that belong to R22 devices.

One of the R2 SRDF mirrors cannot receive writes from its associated R1 device. In normal operations, one of the R2 SRDF mirrors of the R22 device is in this state.

Device pair states

Device pairs that are part of an SRDF operation must be in the correct state. This table lists the states that a device pair can be in.

Table 1. SRDF pair states

Pair State	Description
SyncInProg	Synchronization is in progress between the R1 and the R2 devices. There are existing invalid tracks between the two pairs and the logical links between both sides of an SRDF pair are up.
Synchronized	The R1 and the R2 are in a synchronized state. The same content exists on the R2 as the R1 and there are no invalid tracks between the two pairs.
Split	The R1 and the R2 are ready to their hosts. However, the SRDF links are not ready or are write-disabled.
Failed Over	The R1 is not ready or write disabled. Operations have been failed over to R2.
R1 Updated	The R1 is not ready or write disabled to the host. There are no local invalid tracks on the R1 side, and the links are ready or write disabled.
R1 UpdInProg	The R1 is not ready or write disabled to the host.

Table 1. SRDF pair states (continued)

Pair State	Description
	There are invalid local (R1) tracks on the source side. So data is being copied from the R2 to the R1 device, and the links are ready.
ActiveActive	<p>The R1 and the R2 are in the default SRDF/Metro configuration which uses an Array Witness or Virtual Witness:</p> <ul style="list-style-type: none"> • There are no invalid tracks between the two pairs. • The R1 and the R2 are Ready (RW) to the hosts.
ActiveBias	<p>The R1 and the R2 are in an SRDF/Metro configuration using bias:</p> <ul style="list-style-type: none"> • The user has specified “use bias” during the establish or restore action, or the required Witness is not available. • There are no invalid tracks between the two pairs. • The R1 and the R2 are Ready (RW) to the hosts.
Suspended	<p>The SRDF links have been suspended and are not ready or write disabled.</p> <p>If the R1 is ready while the links are suspended, any I/O accumulates as invalid tracks owed to the R2.</p>
Partitioned	<p>The SYMAPI is unable to communicate through the corresponding SRDF path to the remote array.</p> <p>The Partitioned state may apply to devices within an SRDF group. For example, if SYMAPI is unable to communicate with a remote array from an SRDF group, devices in that group are in the Partitioned state.</p> <p>A half pair and a duplicate pair are also reported as Partitioned.</p>
Mixed	<p>A composite SYMAPI device group SRDF pair state.</p> <p>There are different SRDF pair states within a device group.</p>
Invalid	<p>This state is the default when no other SRDF state applies.</p> <ul style="list-style-type: none"> • The combination of the R1 device, the R2 device, and the SRDF link states do not match any other pair state. • This state may occur if there is a problem at the disk director level.
Consistent	<p>The R2 SRDF/A capable devices are in a consistent state.</p> <p>The consistent state signifies the normal state of operation for device pairs operating in asynchronous mode.</p>
Transmit Idle	The SRDF/A session cannot send data in the transmit cycle over the link because the link is unavailable.

R1/R2 device accessibility

Accessibility of a SRDF device to the application host depends on both the host and the array view of the SRDF device state.

[R1 device accessibility](#) on page 27 and [R2 device accessibility](#) on page 27 list application host accessibility for R1 and R2 devices.

Table 2. R1 device accessibility

Host interface state	SRDF R1 state	Accessibility
Read/Write	Ready	Read/Write
	Not Ready	Depends on R2 device availability
Read Only	Ready	Read Only
	Not Ready	Depends on R2 device availability
Not Ready	Any	Unavailable

Table 3. R2 device accessibility

Host interface state	SRDF R2 state	Accessibility
Write Enabled (Read/Write)	Ready	Read/Write
	Not Ready	Read/Write
Write Disabled (Read Only)	Ready	Read Only
	Not Ready	Read Only
Not Ready	Any	Unavailable

SRDF device and link state combinations

Control actions on an SRDF pair may change the SRDF pair state.

Additionally, the state of a device can change if its front-end or back-end directors detect a change in the SRDF links.

The following table lists:

- SRDF pair states resulting from the combination of the states of the source and target devices and the SRDF links.
- The possible R1 or R2 invalid tracks for each SRDF pair state.

Table 4. Possible SRDF device and link state combinations

SRDF pair state	Source (R1) SRDF state	SRDF link state	Target (R2) SRDF state	R1 or R2 invalid tracks
Synchronized	Ready (RW)	Ready (RW)	Not Ready or WD	0
Failed Over	Not Ready or WD	Not Ready	Ready (RW)	—
R1 Updated	Not Ready or WD	Ready (RW) or WD	Ready (RW)	0 ^a
R1 UpdInProg	Not Ready or WD	Ready (RW) or WD	Ready (RW)	>0 ^a
ActiveActive	Ready (RW)	Ready (RW)	Ready (RW)	0
ActiveBias	Ready (RW)	Ready (RW)	Ready (RW)	0
Split	Ready (RW)	Not Ready or WD	Ready (RW)	—
SyncInProg	Ready (RW)	Ready (RW)	Not Ready or WD	>0
Suspended	Any status ^b	Not Ready or WD	Not Ready or WD	—
Partitioned ^c	Any status	Not Ready	Not Available	—
Partitioned ^d	Not Available	Not Ready	Any status	—
Mixed	^e	^e	^e	—
Invalid ^e	Any status ^f	Any status	Any status	—
Consistent	Ready (RW) ^f	Ready (RW)	Not Ready or WD	0 or >0 ^a
Transmit Idle	Ready (RW) ^f	Ready (RW)	Not Ready or WD	—

a. Refers to invalid local (R1) tracks on source.

b. Any status value is possible (Ready, Not Ready, Write Disabled, or Not Available).

c. Viewed from the host locally connected to the source (R1) device

d. Viewed from the host locally connected to the target (R2) device.

e. When no other SRDF states apply, the state defaults to Invalid.

f. The combination of source SRDF, SRDF links, and target SRDF statuses does not match any other SRDF state; therefore, the SRDF state is considered Invalid.

SRDF consistency

Many applications (in particular, database management systems), use dependent write logic to ensure data integrity if a failure occurs. A dependent write is a write operation that the application does not issue unless some prior I/O has completed. If the writes are out of order, and an event such as a failure occurs at that exact time, unrecoverable data loss may occur.

An *SRDF consistency group* (SRDF/CG) contains SRDF devices with consistency enabled.

An SRDF consistency group preserves the dependent-write consistency of devices within the group. Consistency is maintained by monitoring data propagation from source devices to their corresponding target devices. If consistency is enabled, and SRDF detects any write I/O to a R1 device that cannot communicate with its R2 device, SRDF:

1. Suspends remote mirroring for all devices in the consistency group
2. Completes the intercepted I/O to the R1 device
3. Returns control to the application


In this way, SRDF/CG prevents a dependent-write I/O from reaching the secondary site if the previous I/O only gets as far as the primary site.

SRDF consistency allows you to quickly recover from certain types of failure or physical disasters by retaining a consistent, DBMS-restartable copy of your database.

SRDF consistency group protection is available for both SRDF/S and SRDF/A.

Director boards, links, and ports

SRDF links are the logical connections between SRDF groups and their ports. The ports are physically connected by cables, routers, extenders, switches and other network devices.

 **NOTE:** Two or more SRDF links per SRDF group are required for redundancy and fault tolerance.

The relationship between the resources on a director (CPU cores and ports) varies depending on the operating environment.

PowerMaxOS and HYPERMAX OS

On arrays running PowerMaxOS or HYPERMAX OS :

- The relationship between the SRDF emulation and resources on a director is configurable:
 - One director/multiple CPU cores/multiple ports
 - Connectivity (ports in the SRDF group) is independent of compute power (number of CPU cores). You can change the amount of connectivity without changing compute power.
- Each director has up to 16 front end ports, any or all of which can be used by SRDF. Both the SRDF Gigabit Ethernet and SRDF Fibre Channel emulations can use any port.
- The data path for devices in an SRDF group is not fixed to a single port. Instead, the path for data is shared across all ports in the group.

Mixed configurations: PowerMaxOS or HYPERMAX OS and Enginuity 5876

For configurations where one array is running Enginuity 5876, and the other array is running PowerMaxOS or HYPERMAX OS, these rules apply:

- On the 5876 side, an SRDF group can have the full complement of directors, but no more than 16 ports on the PowerMaxOS or HYPERMAX OS side.
- You can connect to 16 directors using one port each, 2 directors using 8 ports each or any other combination that does not exceed 16 per SRDF group.

Disaster recovery

This chapter provides more detail on the disaster recovery configurations of SRDF.

Topics:

- [Two-site configurations](#)
- [Three-site configurations](#)
- [Four-site configurations](#)
- [SRDF recovery scenarios](#)
- [Replication of VMWare vVols](#)

Two-site configurations

This table shows the two-site configurations for SRDF.

Table 5. SRDF two-site configurations

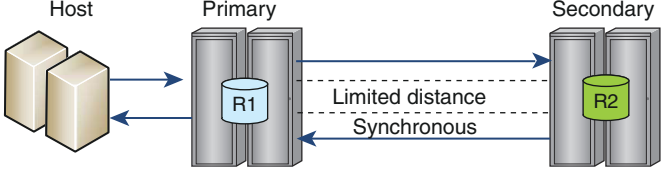
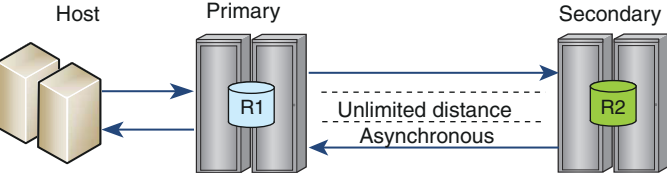
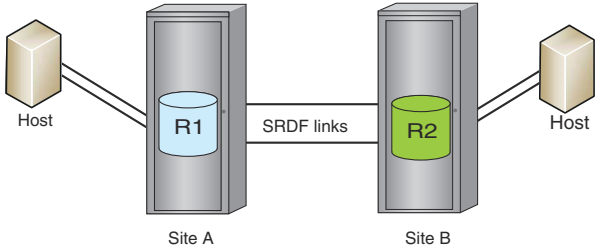
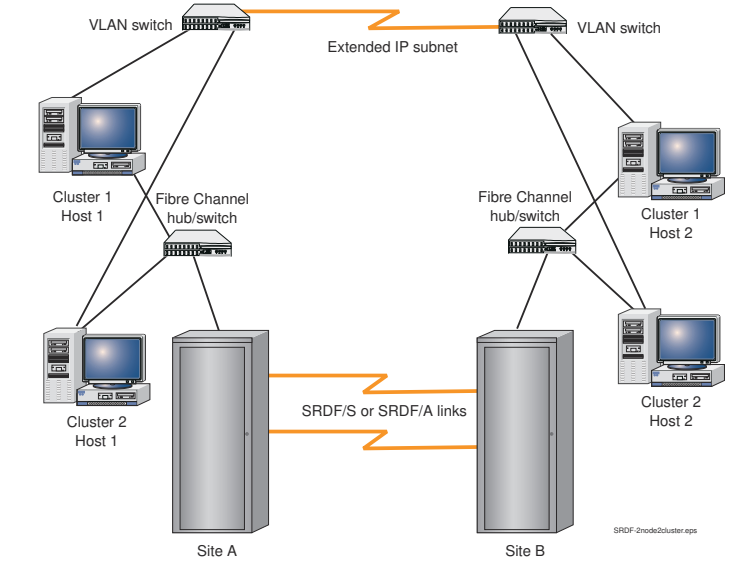
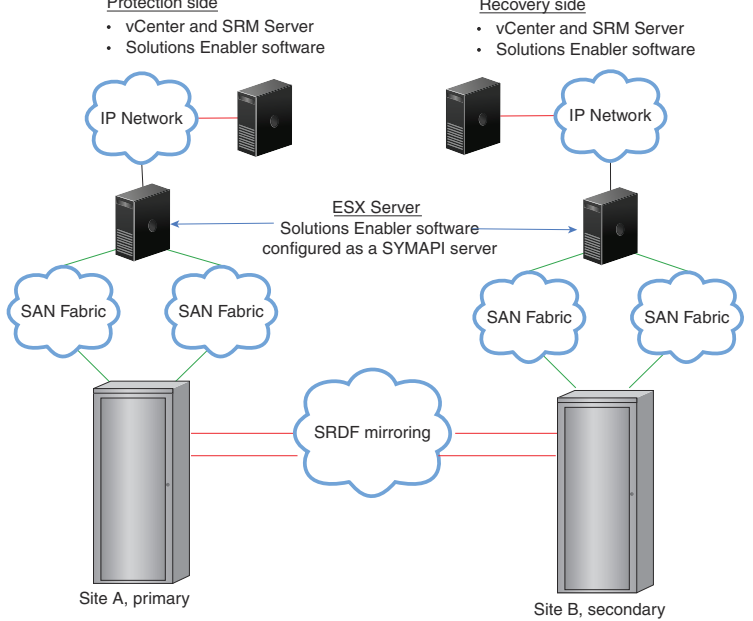
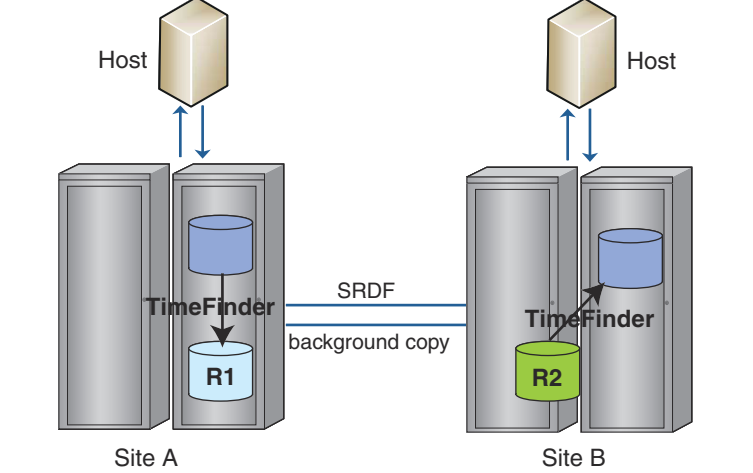
Solution highlights	Site topology
<p>SRDF/Synchronous (SRDF/S)</p> <p>Maintains a real-time copy of production data at a physically separated array.</p> <ul style="list-style-type: none"> • No data exposure • Ensured consistency protection with SRDF/Consistency Group • Recommended maximum distance of 200 km (125 miles) between arrays as application latency may rise to unacceptable levels at longer distances <p>NOTE: In some circumstances, using SRDF/S over distances greater than 200 km may be feasible. Contact your Dell Technologies representative for more information.</p>	 <p>The diagram shows a Host connected to a Primary storage array (R1). R1 is connected via a dashed line labeled 'Limited distance' to a Secondary storage array (R2). A solid line labeled 'Synchronous' connects R1 and R2, indicating real-time data replication.</p>
<p>SRDF/Asynchronous (SRDF/A)</p> <p>Maintains a dependent-write consistent copy of the data on a remote secondary site. The copy of the data at the secondary site is seconds behind the primary site.</p> <ul style="list-style-type: none"> • RPO seconds before the point of failure • Unlimited distance 	 <p>The diagram shows a Host connected to a Primary storage array (R1). R1 is connected via a dashed line labeled 'Unlimited distance' to a Secondary storage array (R2). A solid line labeled 'Asynchronous' connects R1 and R2, indicating delayed data replication.</p>
<p>SRDF/Data Mobility (SRDF/DM)</p> <p>This example shows an SRDF/DM topology and the I/O flow in adaptive copy mode.</p> <ul style="list-style-type: none"> • The host write I/O is received in cache in Site A • The host emulation returns a positive acknowledgment to the host • The SRDF emulation transmits the I/O across the SRDF links to Site B • Once data is written to cache in Site B, the SRDF emulation in Site B returns a positive acknowledgment to Site A <p>Operating Notes:</p> <ul style="list-style-type: none"> • The maximum skew value set at the device level in SRDF/DM solutions must be equal to or greater than 100 tracks • SRDF/DM is only for data replication or migration, not for disaster restart solutions 	 <p>The diagram shows two sites, Site A and Site B. Site A has a Host connected to a storage array (R1). Site B has a storage array (R2) connected to a Host. R1 and R2 are connected by a solid line labeled 'SRDF links'. A dashed line connects the Host at Site A to R2, and another dashed line connects R1 to the Host at Site B, illustrating data mobility.</p> <p>NOTE: Data may be read from the drives to cache before it is transmitted across the SRDF links, resulting in propagation delays.</p>

Table 5. SRDF two-site configurations (continued)

Solution highlights	Site topology
<p>SRDF/Cluster Enabler (CE)</p> <ul style="list-style-type: none"> Integrates SRDF/S or SRDF/A with Microsoft Failover Clusters (MSCS) to automate or semi-automate site failover. Complete solution for restarting operations in Microsoft Failover Cluster environments. Expands the range of cluster storage and management capabilities while ensuring full protection of the SRDF remote replication. 	
<p>SRDF and VMware Site Recovery Manager</p> <p>Completely automates storage-based disaster restart operations for VMware environments in SRDF topologies.</p> <ul style="list-style-type: none"> The Dell EMC SRDF Adapter enables VMware Site Recovery Manager to automate storage-based disaster restart operations in SRDF solutions. Can address configurations in which data are spread across multiple storage arrays or SRDF groups. Requires that the adapter is installed on each array to facilitate the discovery of arrays and to initiate failover operations. Implemented with: <ul style="list-style-type: none"> SRDF/S SRDF/A SRDF/Star TimeFinder 	
<p>SRDF/Automated Replication (SRDF/AR)</p> <ul style="list-style-type: none"> Combines SRDF and TimeFinder to optimize bandwidth requirements and provide a long-distance disaster restart solution. Operates in two-site solutions that use SRDF/DM in combination with TimeFinder. <p>SRDF/AR is available only for arrays that run Enginuity 5876 since it uses STD (thick) devices. Arrays that run PowerMaxOS 5978 or HYPERMAX OS 5977 have TDEV (thin) devices only.</p>	

Three-site configurations

This table shows the three-site configurations for SRDF.

Table 6. SRDF multisite solutions

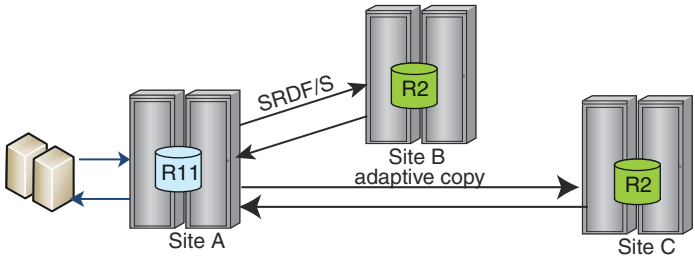
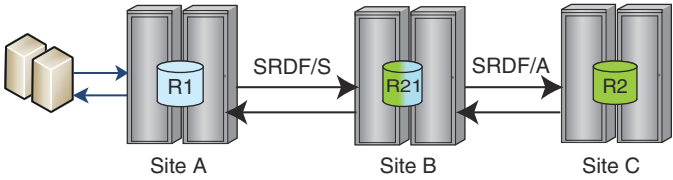
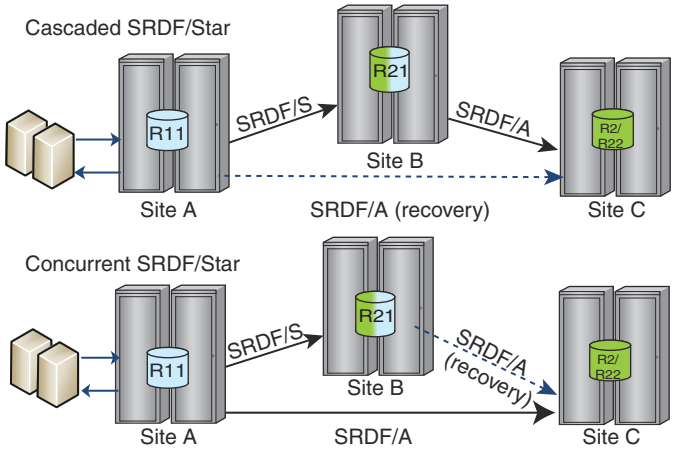
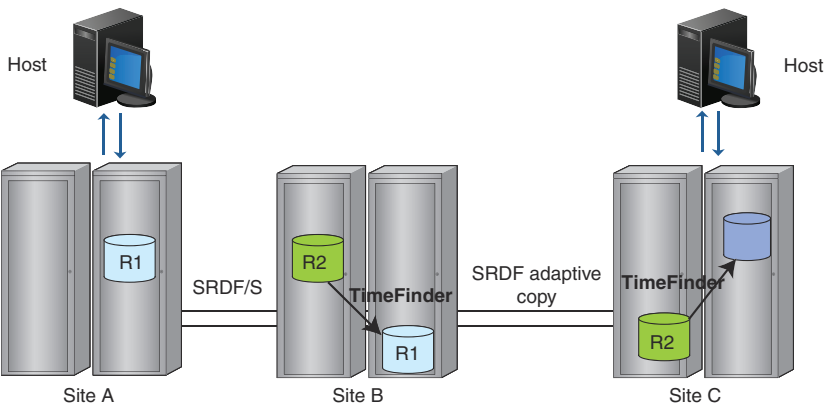
Solution highlights	Site topology
<p>Concurrent SRDF</p> <p>Three-site disaster recovery and advanced multisite business continuity protection.</p> <ul style="list-style-type: none"> • Data on the primary site is concurrently replicated to 2 secondary sites. • Replication to a remote site can use SRDF/S, SRDF/A, or adaptive copy. 	
<p>Cascaded SRDF</p> <p>Three-site disaster recovery and advanced multisite business continuity protection.</p> <ul style="list-style-type: none"> • Data on the primary site is synchronously mirrored to a secondary (R21) site. The data is then asynchronously mirrored from the secondary (R21) site to a tertiary (R2) site. • First “hop” is SRDF/S. Second hop is SRDF/A. 	
<p>SRDF/Star</p> <p>Three-site data protection and disaster recovery with zero data loss recovery, business continuity protection, and disaster-restart.</p> <ul style="list-style-type: none"> • Available in 2 configurations: <ul style="list-style-type: none"> ◦ Cascaded SRDF/Star ◦ Concurrent SRDF/Star • Differential synchronization allows rapid reestablishment of mirroring among surviving sites in a multisite disaster recovery implementation. • Implemented using SRDF consistency groups (CG) with SRDF/S and SRDF/A. 	

Table 6. SRDF multisite solutions (continued)

Solution highlights	Site topology
<p>SRDF/Automated Replication (SRDF/AR)</p> <ul style="list-style-type: none"> Combines SRDF and TimeFinder to optimize bandwidth requirements and provide a long-distance disaster restart solution. Operates in three-site solutions that use a combination of SRDF/S, SRDF/DM, and TimeFinder. <p>SRDF/AR is available only for arrays that run Enginuity 5876 since it uses STD (thick) devices. Arrays that run PowerMaxOS 5978 or HYPERMAX OS 5977 have TDEV (thin) devices only.</p>	 <p>The diagram illustrates a three-site topology for SRDF/AR. It consists of three sites: Site A, Site B, and Site C. Each site has a Host connected to a storage array. Site A has a Host connected to an array containing a blue cylinder labeled R1. Site B has a Host connected to an array containing a green cylinder labeled R2 and a blue cylinder labeled R1. Site C has a Host connected to an array containing a green cylinder labeled R2 and a blue cylinder. The connections between sites are as follows: Site A and Site B are connected via a line labeled 'SRDF/S'. Site B and Site C are connected via a line labeled 'SRDF adaptive copy'. Additionally, there are arrows labeled 'TimeFinder' pointing from the R2 cylinder in Site B to the R1 cylinder in Site B, and from the R2 cylinder in Site C to the blue cylinder in Site C.</p>

Concurrent SRDF solutions

Concurrent SRDF is a three-site disaster recovery solution using R11 devices that replicate to two R2 devices. The two R2 devices operate independently but concurrently using any combination of SRDF modes:

- Concurrent SRDF/S to both R2 devices if the R11 site is within synchronous distance of the two R2 sites
- Concurrent SRDF/A to sites at extended distances from the workload site

Either of the R2 devices can restore data to the R11 device. Similarly, an R2 device can restore both the R11 and the other R2 device.

Concurrent SRDF can also replace an existing R11 or R2 device with a new device. Migrate data from the existing device to a new device using adaptive copy disk mode, and then replace the existing device with the newly populated device.

Concurrent SRDF topologies use Fibre Channel and Gigabit Ethernet.

This example shows:

- The R11 -> R2 in Site B in synchronous mode
- The R11 -> R2 in Site C in adaptive copy mode

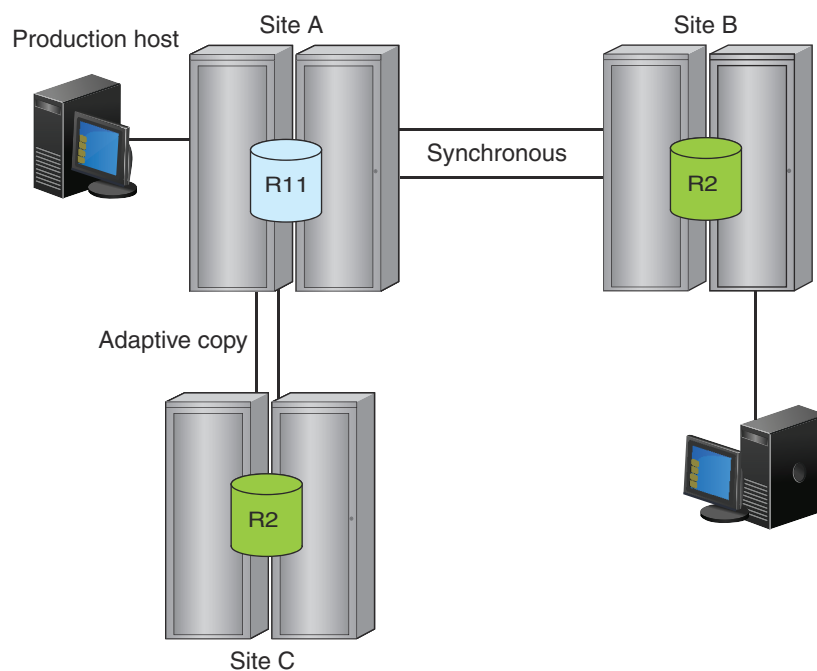


Figure 10. Concurrent SRDF topology

Concurrent SRDF/S with Enginuity Consistency Assist

If both legs of a concurrent SRDF configuration use SRDF/S, you can leverage the independent consistency protection feature. This feature is based on Enginuity Consistency Assist (ECA) and enables you to manage consistency on each concurrent SRDF leg independently.

If consistency protection on one leg is suspended, consistency protection on the other leg can remain active and continue protecting the primary site.

Cascaded SRDF solutions

Cascaded SRDF provides a zero data loss solution at long distances in the event that the primary site is lost.

In cascaded SRDF configurations, data from a primary (R1) site is synchronously mirrored to a secondary (R21) site, and then asynchronously mirrored from the secondary (R21) site to a tertiary (R2) site.

Cascaded SRDF provides:

- Fast recovery times at the tertiary site
- Tight integration with TimeFinder product family
- Geographically dispersed secondary and tertiary sites

If the primary site fails, cascaded SRDF can continue mirroring, with minimal user intervention, from the secondary site to the tertiary site. This enables a faster recovery at the tertiary site.

Both the secondary and the tertiary site can be failover sites. Open systems solutions typically fail over to the tertiary site.

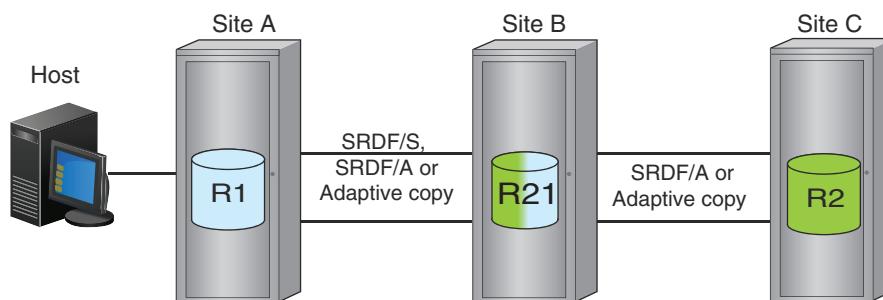


Figure 11. Cascaded SRDF topology

SRDF/Star solutions

SRDF/Star is a disaster recovery solution that consists of three sites: primary (production), secondary, and tertiary. The secondary site synchronously mirrors the data from the primary site, and the tertiary site asynchronously mirrors the production data.

NOTE: In mainframe environments, GDDR is required to implement SRDF/Star. For more information, see [SRDF/Star for mainframe systems](#) on page 41 and the appropriate GDDR product guide listed in [More information](#) on page 103.

If an outage occurs at the primary site, SRDF/Star enables you to quickly move operations and re-establish remote mirroring between the remaining sites. When conditions permit, you can quickly rejoin the primary site to the solution, resuming the SRDF/Star operations.

SRDF/Star operates in concurrent and cascaded environments that address different recovery and availability objectives:

- Concurrent SRDF/Star—Data is mirrored from the primary site concurrently to two R2 devices. Both the secondary and tertiary sites are potential recovery sites should the primary site fail. Differential resynchronization is used between the secondary and the tertiary sites.
- Cascaded SRDF/Star—Data is mirrored first from the primary site to a secondary site, and then from the secondary to a tertiary site. Both the secondary and tertiary sites are potential recovery sites. Differential resynchronization is used between the primary and the tertiary site.

Differential synchronization between two remote sites:

- Allows SRDF/Star to rapidly reestablish cross-site mirroring should the primary site fail.
- Greatly reduces the time required to remotely mirror the selected production site.

If a rolling disaster affects the primary site, SRDF/Star helps you determine which remote site has the most current data. You can select which site to operate from and which site's data to use when recovering from the primary site failure.

If the primary site fails, SRDF/Star enables you to resume asynchronous protection between the secondary and tertiary sites, with minimal data movement.

Concurrent SRDF/Star

In concurrent SRDF/Star solutions, production data on R11 devices replicates to two R2 devices in two remote arrays.

In this example:

- Site B is a secondary site using SRDF/S links from Site A.
- Site C is a tertiary site using SRDF/A links from Site A.
- The (normally inactive) recovery links are SRDF/A between Site C and Site B.

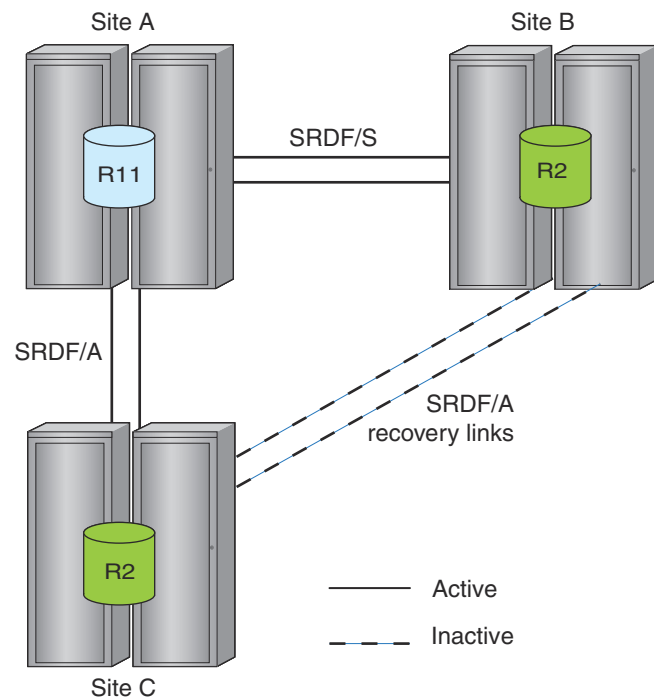


Figure 12. Concurrent SRDF/Star

Concurrent SRDF/Star with R22 devices

SRDF supports concurrent SRDF/Star topologies using R22 devices. R22 devices have two SRDF mirrors, only one of which is active on the SRDF links at a given time. R22 devices improve the resiliency of the SRDF/Star application, and reduce the number of steps for failover procedures.

This example shows R22 devices at Site C.

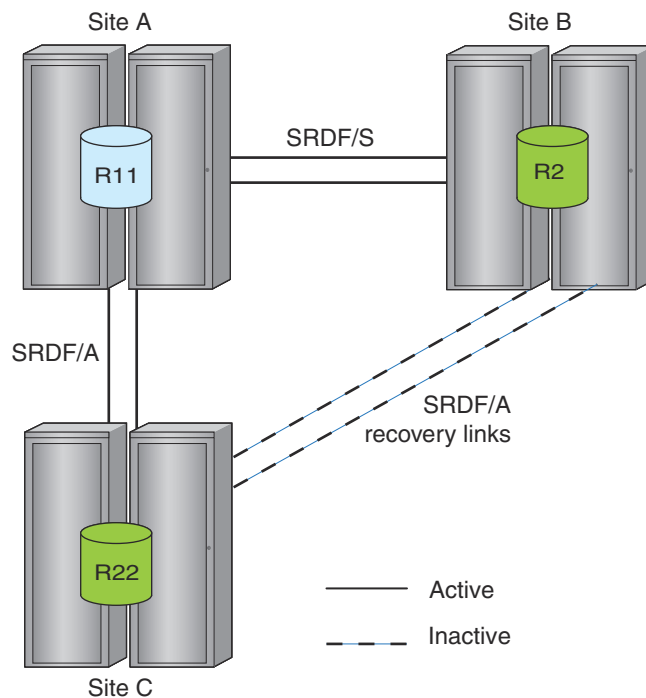


Figure 13. Concurrent SRDF/Star with R22 devices

Cascaded SRDF/Star

In cascaded SRDF/Star solutions, the synchronous secondary site is always more current than the asynchronous tertiary site. If the synchronous secondary site fails, the cascaded SRDF/Star solution can incrementally establish an SRDF/A session between primary site and the asynchronous tertiary site.

Cascaded SRDF/Star can determine when the current active R1 cycle (capture) contents reach the active R2 cycle (apply) over the long-distance SRDF/A links. This minimizes the amount of data that must be moved between Site B and Site C to fully synchronize them.

This example shows a basic cascaded SRDF/Star solution.

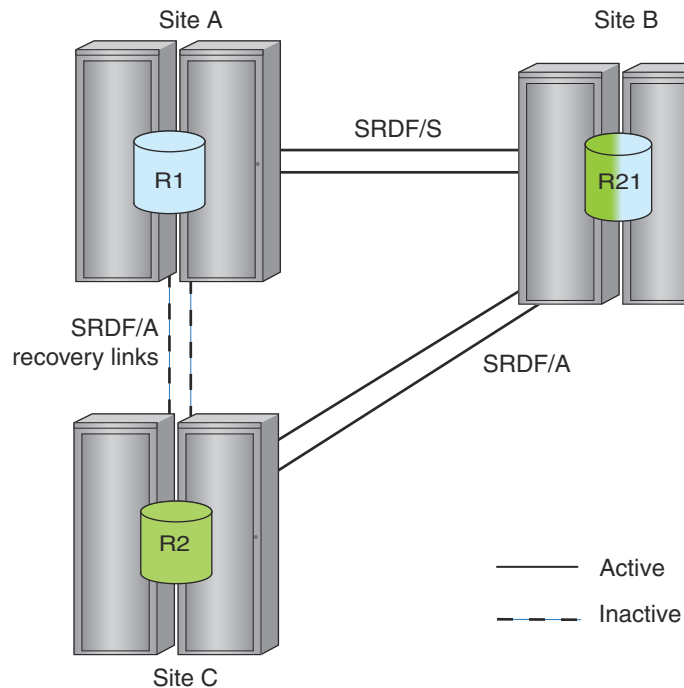


Figure 14. Cascaded SRDF/Star

Cascaded SRDF/Star with R22 devices

You can use R22 devices to pre-configure the SRDF pairs required to incrementally establish an SRDF/A session between Site A and Site C in case Site B fails.

This example shows cascaded R22 devices in a cascaded SRDF solution.

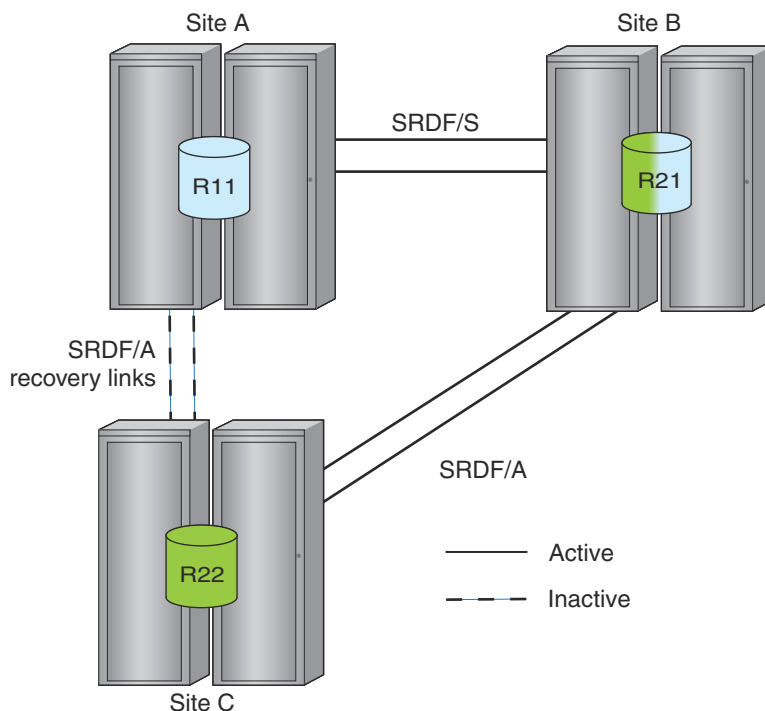


Figure 15. R22 devices in cascaded SRDF/Star

In cascaded SRDF/Star configurations with R22 devices:

- All devices at the production site (Site A) must be configured as concurrent (R11) devices paired with R21 devices (Site B) and R22 devices (Site C).
- All devices at the synchronous site in Site B must be configured as R21 devices.
- All devices at the asynchronous site in Site C must be configured as R22 devices.

Requirements/restrictions

Cascaded and Concurrent SRDF/Star configurations (with and without R22 devices) require the following:

- All SRDF/Star device pairs must be of the same geometry and size.
- All SRDF groups including inactive ones must be defined and operational prior to entering SRDF/Star mode.
- It is strongly recommended that all SRDF devices be locally protected and that each SRDF device is configured with TimeFinder to provide local replicas at each site.

SRDF/Star for open systems

Solutions Enabler controls, manages, and automates SRDF/Star in open systems environments. Session management is required at the production site.

Host-based automation is provided for normal, transient fault, and planned or unplanned failover operations.

Dell EMC Solutions Enabler Symmetrix SRDF CLI Guide provides detailed descriptions and implementation guidelines.

In cascaded and concurrent configurations, a restart from the asynchronous site may require a wait for any remaining data to arrive from the synchronous site. Restarts from the synchronous site requires no wait unless the asynchronous site is more recent (the latest updates need to be brought to the synchronous site).

SRDF/Star for mainframe systems

The SRDF Host Component for z/OS creates and manages the standard two-site SRDF configurations along with Cascaded and Concurrent three-site configurations. Geographically Dispersed Disaster Restart (GDDR) creates and manages SRDF/Star configurations in a mainframe environment.

The structure of SRDF/Star configurations differs when GDDR is involved as each site has a GDDR control system. Each control system monitors the overall SRDF/Star configuration, detects failure conditions, and acts on those conditions (for example, by restarting operations at an alternate site).

This is an example of a Concurrent SRDF/Star configuration with GDDR:

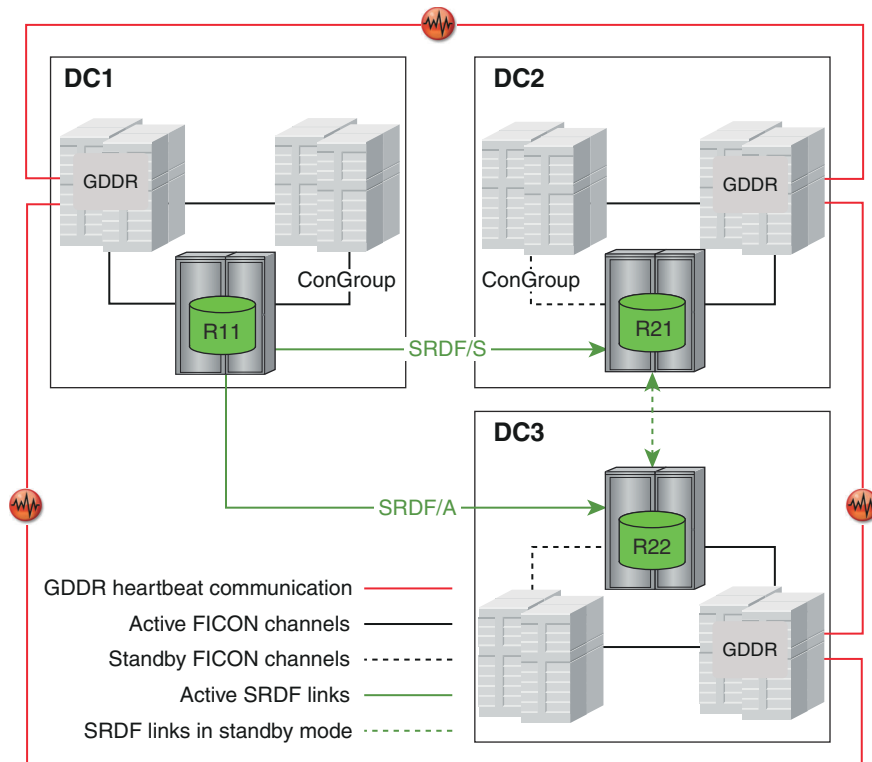


Figure 16. Concurrent SRDF/Star with GDDR

This is an example of a Cascaded SRDF/Star configuration with GDDR:

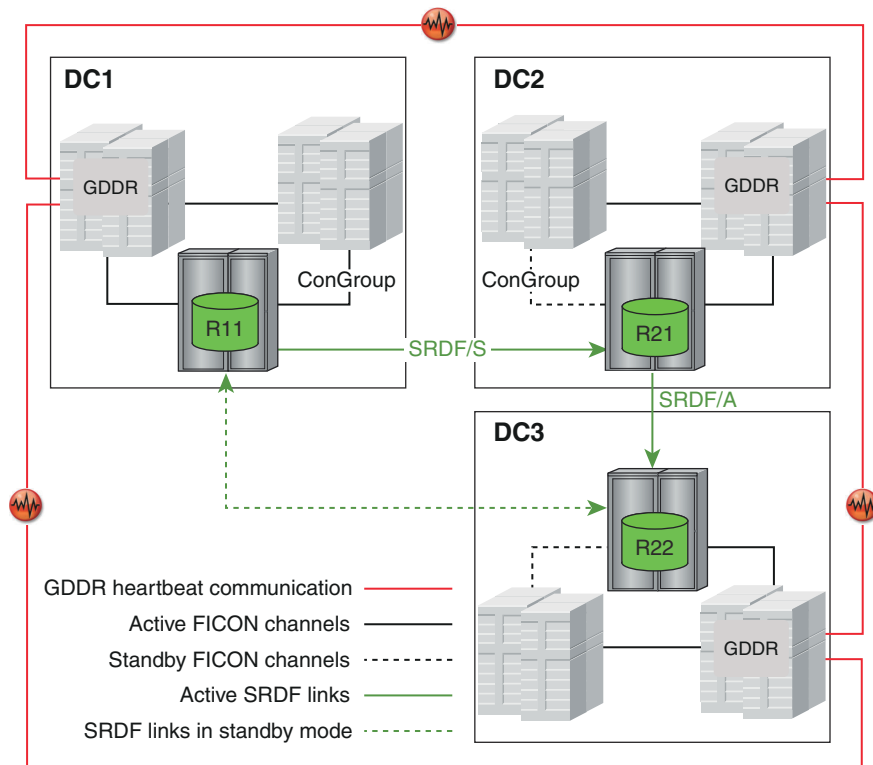


Figure 17. Cascaded SRDF/Star with GDDR

Mainframe-specific variations of SRDF/Star

GDDR provides some variations of SRDF/Star configurations that are unique to the mainframe environment.

Two-site SRDF/Star

In the two-site SRDF/Star configuration, there are three storage arrays as in any other SRDF/Star configuration, but there are GDDR controls systems at DC1 and DC3 only. This means that if there is a failure at DC1 (the primary site), operations can be restarted at DC3 only.

Asynchronous SRDF/Star (Star-A)

Asynchronous SRDF/Star configurations are similar to other SRDF/Star configurations but all SRDF links use SRDF/A. This enables both DC2 in addition to DC3 to be remote from DC1 and so provide better protection from a site failure at DC1.

SRDF/Star with Autoswap

Autoswap is a facility to move (swap) workloads from volumes in one set of storage arrays to volumes in another set of arrays without interrupting host processing. The combination of Autoswap with SRDF/Star provides near-continuous availability through device failover between sites DC1 and DC2, while also providing disaster restart capabilities at site DC3.

SRDF/Star restrictions

- GNS Remote Mirroring is NOT supported with SRDF/STAR configurations .
- Devices that are part of an RecoverPoint configuration, cannot at the same time, be part of an SRDF/Star configuration.
- The RDF groups that are part of an SRDF/Star CG cannot contain any devices that are not part of the SRDF/Star CG.
- Devices that are part of a SRDF/Star CG should not be controlled outside of `symstar` commands.
- Devices that are part of an SRDF/Metro configuration cannot at the same time be part of an SRDF/Star configuration.
- If any array in a SRDF/Star configuration is running HYPERMAX OS, Solutions Enabler 8.1 or higher is required in order to manage that configuration.
- If any array in a SRDF/Star configuration is running PowerMaxOS, Solutions Enabler 9.0 or later is required in order to manage that configuration.
- Each SRDF/Star control host must be connected to only one site in the SRDF/Star triangle. An SRDF/Star control host is where the `symstar` commands are issued.
- A minimum of one SRDF daemon must be running on at least one host attached locally to each site. This host must be connected to only one site in the SRDF/Star triangle. The host could be the same as the Star control host but is not required unless using `symstar modifycg`.

Dell Technologies strongly recommends running redundant SRDF daemons on multiple hosts to ensure that at least one SRDF daemon is available to perform time-critical, consistency monitoring operations. Redundant SRDF daemons avoid service interruptions caused by performance bottlenecks local to a host.

- SRDF/A recovery links are required.
- SRDF groups cannot be shared between separate SRDF/Star configurations.
- R22 devices are required in SRDF/Star environments that include VMAX 10K arrays.
- CKD striped metadevices are not supported.
- R2 devices cannot be larger than their R1 devices.
- Composite groups consisting of device groups are not supported.
- Devices enabled as part of consistency groups cannot at the same time be part of an SRDF/Star configuration.
- Devices cannot be BCV devices.
- Every device must be dynamic SRDF (R1 and R2 capable).
- BCV device management must be configured separately.

NOTE:

Dell Technologies strongly recommends that you have BCV device management available at both the synchronous and asynchronous target sites.

- With Engenuity 5876.159.102 and higher, a mixture of thin and (non-diskless) thick devices is supported.

Four-site configurations

Four-site configurations provide extra data protection. There are configurations for both FBA and mainframe environments.

Four-site FBA configurations

The four-site SRDF solution for open systems host environments:

- Replicates FBA data by using both concurrent and cascaded SRDF topologies
- Is a multi-region disaster recovery solution with higher availability, improved protection, and less downtime than concurrent or cascaded SRDF solutions
- Offers multi-region high availability by combining the benefits of concurrent and cascaded SRDF solutions

If two sites fail because of a regional disaster, a copy of the data is available, and there is protection between the remaining two sites. An existing two-site or three-site SRDF topology is the basis for a four-site topology. Four-site SRDF can also be used for data migration.

This is an example of the four-site SRDF solution.

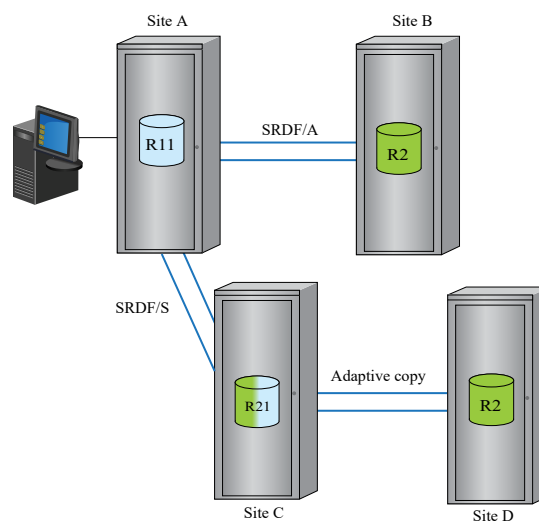


Figure 18. SRDF 4-site FBA configuration

Four-site mainframe configurations (SRDF/SQAR)

SRDF/SQAR (Symmetrix Quadrilateral Asynchronous Replication) is a four-site implementation of SRDF/S and SRDF/A that enables differential resynchronization between sites along the perimeter of a 'square' SRDF topology. Dell EMC GDDR is required to be able to implement SRDF/SQAR.

The SRDF/SQAR configuration provides the ability to recover from a single or dual unplanned site outage in one region with SRDF/S protection established differentially between the recover sites in another region. This enables rapid resumption of a workload with SRDF/S and Autoswap protection in another region. In certain failure scenarios, it also provides zero data loss recovery across regions.

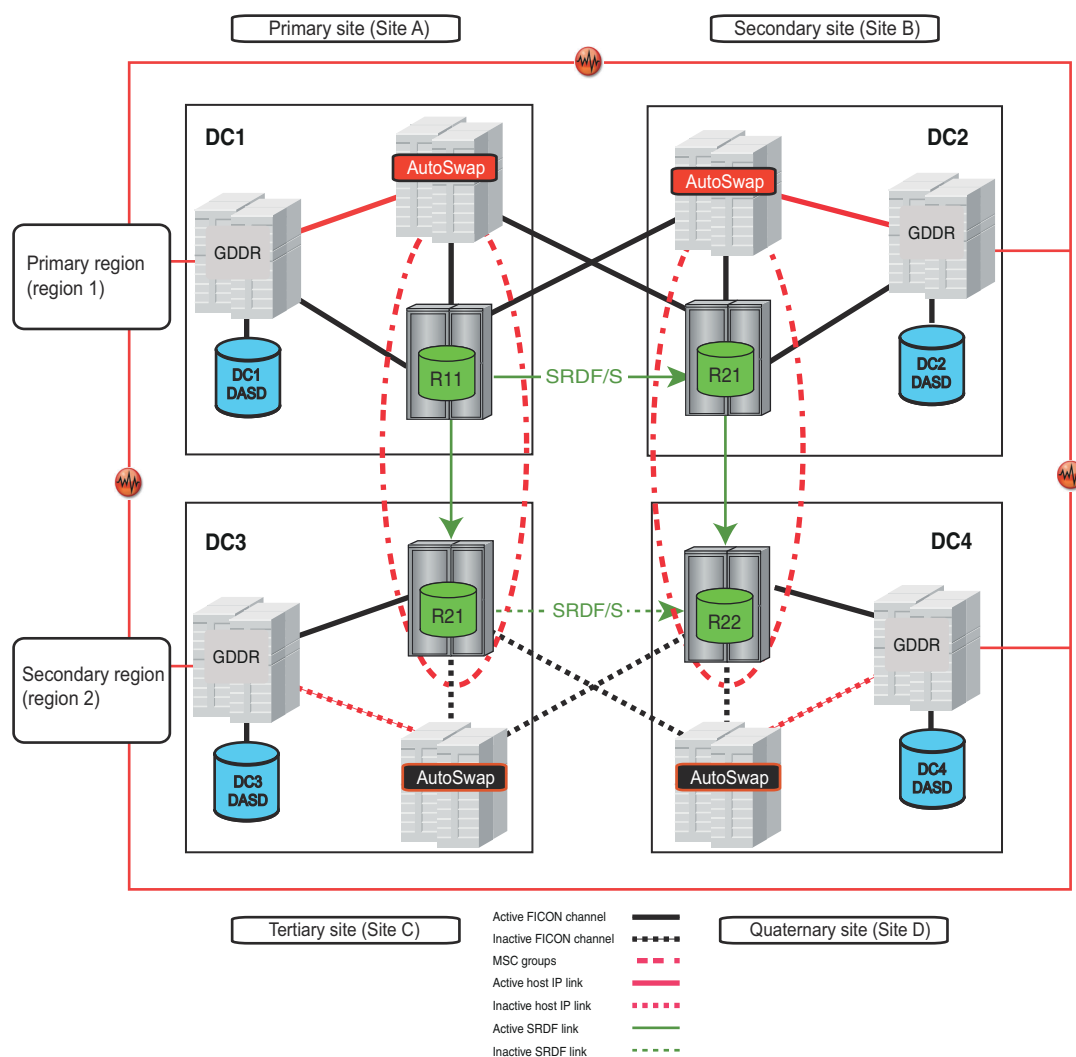


Figure 19. SRDF/SQAR with Autoswap environment

The diagram shows four Dell EMC GDDR control systems with their independent heartbeat communication paths, separate from the production disk and computer facilities. Each of the managed z/OS systems has Dell EMC Autoswap and Dell EMC Consistency Groups (ConGroup) installed.

Each GDDR SRDF/SQAR environment manages two consistency groups (one active, one defined) and two Multi-Session Consistency MSC groups (both active). A consistency group is a named group of source (R1) volumes managed by the ConGroup application as a unit. An MSC group is a named group consisting of multiple SRDF groups operating in SRDF/A mode, managed by the Dell EMC MSC control software feature as a single unit. The relationship between Site A (DC1) and Site B (DC2) is maintained through SRDF/S replication of primary disk images at DC1 to DC2, while SRDF/A replication maintains out of region, mirrored data at Site C (DC3) and Site 4 (DC4).

Requirements and restrictions

- SRDF/SQAR is required to be configured with the MSC High Availability feature with a second SCF instance and MSC configured using weight factor of 2.
- SRDF Host Component actions that change the devices defined to the SQAR MSC groups require the MSC tasks to be inactive at the time of the change.
- Connectivity is provided only along the perimeter of the SQAR topology. Cross site connectivity (for example, Site A to Site D) is not supported. Therefore, traditional three-site SRDF/Star as a recovery configuration is not available.
- In the case of a single site failure it is important to know which SRDF/A site is more current. The existing SRDF/A secondary time-of-day value is used to determine which site is ahead.

SRDF recovery scenarios

This section describes recovery scenarios in 2-site SRDF configurations.

Planned failover (SRDF/S)

A planned failover moves production applications from the primary site to the secondary site in order to test the recovery solution, upgrade or perform maintenance at the primary site.

This diagram shows a two-site SRDF configuration before the R1 <-> R2 personality swap:

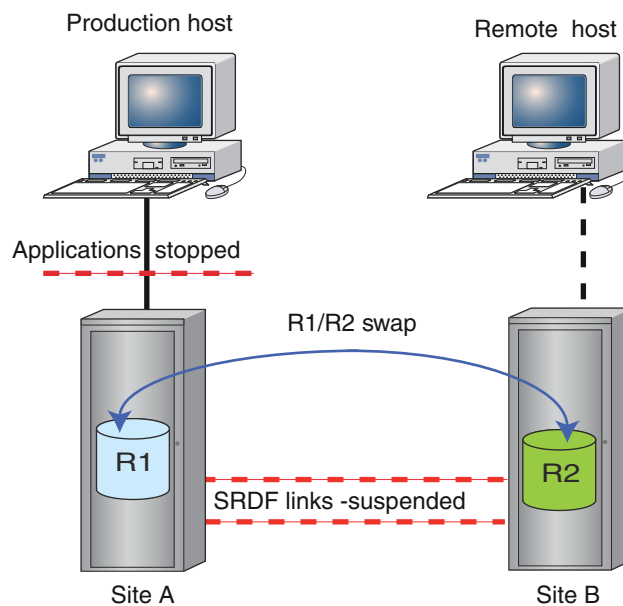


Figure 20. Planned failover: before personality swap

- Applications on the production host are stopped.
- SRDF links between Site A and Site B are suspended.
- If SRDF/CG is used, consistency is suspended.

The next diagram shows a two-site SDRF configuration after the R1 <-> R2 personality swap.

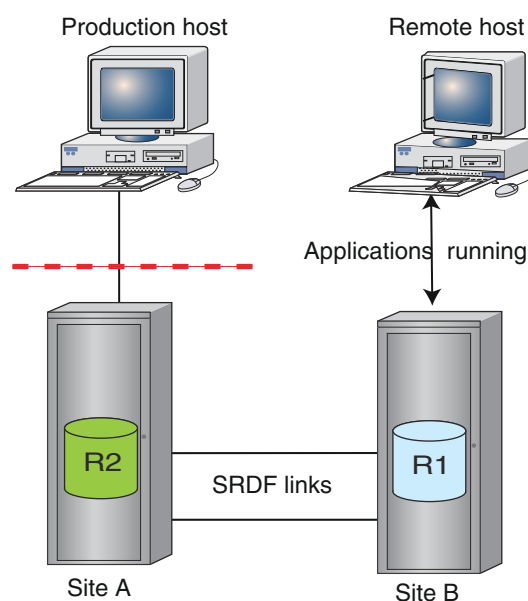


Figure 21. Planned failover: after personality swap

When the maintenance, upgrades or testing procedures are complete, a similar procedure returns production to Site A.

Unplanned failover

An unplanned failover moves production applications from the primary site to the secondary site after an unanticipated outage at the primary site.

Failover to the secondary site in a simple configuration can be performed in minutes. You can resume production processing as soon as the applications are restarted on the failover host connected to Site B.

Unlike the planned failover operation, an unplanned failover resumes production at the secondary site, but without remote mirroring until Site A becomes operational and ready for a failback operation.

This diagram shows failover to the secondary site after the primary site fails.

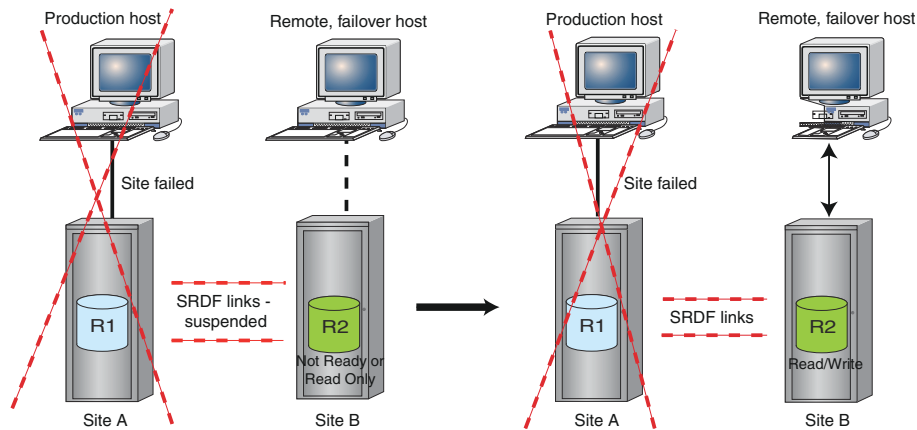


Figure 22. Failover to Site B, Site A and production host unavailable.

Failback to the primary array

When the primary host and array containing the primary (R1) devices are operational once more, an SRDF failback allows production processing to resume on the primary host.

Recovery for a large number of invalid tracks

If the R2 devices have handled production processing for a long period of time, there may be a large number of invalid tracks owed to the R1 devices. SRDF control software can resynchronize the R1 and R2 devices while the secondary host continues production processing. Once there is a relatively small number of invalid tracks owed to the R1 devices, the failback process can take place.

Temporary link loss

In SRDF/A configurations, if a temporary loss (10 seconds or less) of all SRDF/A links occurs, the SRDF/A state remains active, and data continues to accumulate in global memory. This may result in an elongated cycle, but:

- The secondary array dependent-write consistency is not compromised.
- The primary and secondary array device relationships are not suspended.

[Transmit Idle](#) on page 93 can keep SRDF/A in an active state during all links lost conditions.

In SRDF/S configurations, if a temporary link loss occurs, writes are stalled (but not accumulated) in hopes that the SRDF link comes back up, at which point writes continue.

Reads are not affected.

NOTE: Switching to SRDF/S mode with the link limbo parameter configured for more than 10 seconds could result in an application, database, or host failure.

Permanent link loss (SRDF/A)

If all SRDF links are lost for more time than link limbo or Transmit Idle can manage:

- All of the devices in the SRDF group are set to a Not Ready state.
- All data in capture and transmit delta sets is changed from write pending for the R1 SRDF mirror to invalid for the R1 SRDF mirror and is therefore owed to the R2 device.
- Any new write I/Os to the R1 device are also marked invalid for the R1 SRDF mirror.

These tracks are owed to the secondary array once the links are restored.

When the links are restored, normal SRDF recovery procedures are followed:

- Metadata representing the data owed is compared and merged based on normal host recovery procedures.
- Data is resynchronized by sending the owed tracks as part of the SRDF/A cycles.

Data on non-consistency exempt devices on the secondary array is always dependent-write consistent in SRDF/A active/consistent state, even when all SRDF links fail. Starting a resynchronization process compromises the dependent-write consistency until the resynchronization is fully complete and two cycle switches have occurred.

For this reason, it is important to use TimeFinder to create a gold copy of the dependent-write consistent image on the secondary array.

SRDF/A session cleanup

When an SRDF/A single session mode is dropped, SRDF:

- Marks new incoming writes at the primary array as being owed to the secondary array.
- Discards the capture and transmit delta sets, and marks the data as being owed to the secondary array. These tracks are sent to the secondary array once SRDF is resumed, as long as the copy direction remains primary-to-secondary.
- Marks and discards only the receive delta set at the secondary array, and marks the data as tracks owed to the primary array.

NOTE: It is very important to capture a gold copy of the dependent-write consistent data on the secondary array R2 devices prior to any resynchronization. Any resynchronization compromises the dependent-write consistent image. The gold copy can be stored on a remote set of BCVs or Clones.

Failback from R2 devices

If a disaster occurs on the primary array, data on the R2 devices represents an older dependent-write consistent image and can be used to restart the applications.

After the primary array has been repaired, production operations can return to the primary array as described in [SRDF recovery scenarios](#) on page 47.

If the failover to the secondary site is an extended event, the SRDF/A solution can be reversed by issuing a personality swap. SRDF/A can continue operations until a planned reversal of direction can be performed to restore the original SRDF/A primary and secondary relationship.

After the workload has been transferred back to the primary array hosts, SRDF/A can be activated to resume normal asynchronous mode protection.

Replication of VMWare vVols

A vVol is a VMWare object that, at a basic level, equates to a disk associated with a VMWare virtual machine (VM). The VMWare administrator creates and manages vVols through the VMWare vSphere product.

This section contains an overview of the architecture of vVols and their implementation. It then shows how vVols can be replicated for disaster recovery purposes using SRDF/A.

vVol architecture

This diagram shows the architecture of the vVol implementation in a PowerMax or VMAX All Flash environment.

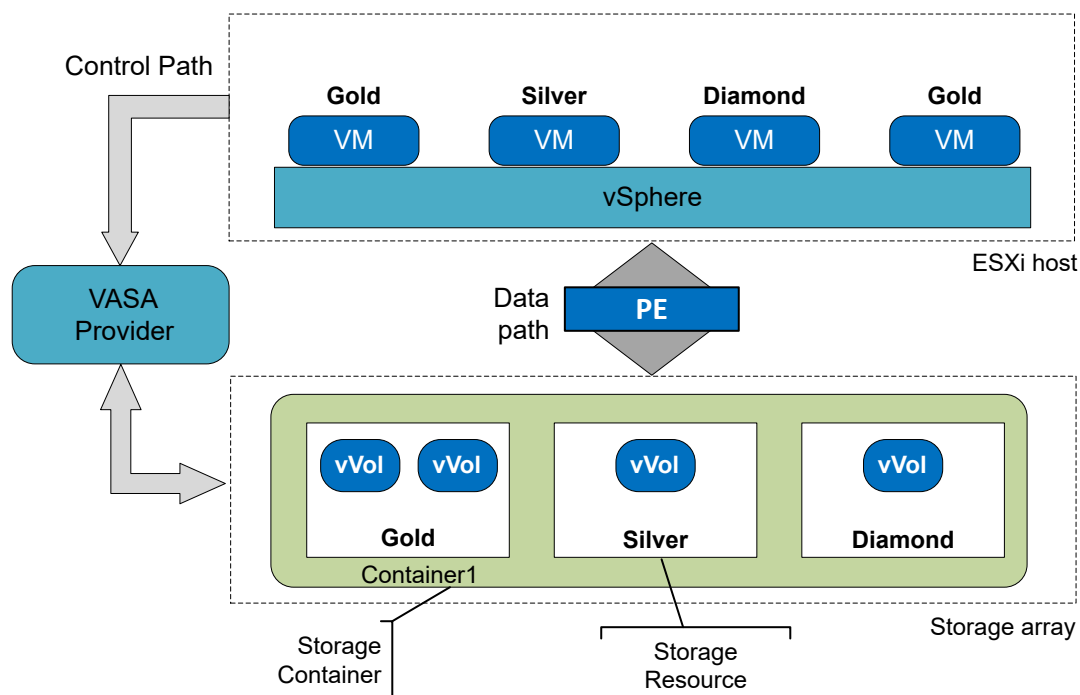


Figure 23. The implementation of vVols in PowerMax OS

The diagram shows the VASA Provider separate from the storage array. PowerMaxOS 5978.669.669 and later includes an embedded VASA Provider that runs as a guest management application on the PowerMax array.

Components

The major components in the implementation of vVols are:

Virtual Machine (VM)	A virtual machine on an ESXi host managed through the VMWare vCenter product.
vVol	A virtual disk associated with a specific VM.
Storage Container	A pool of storage on an array set aside for use by vVols.
Storage Resource	A subdivision of a storage container that has a specified capacity and has attributes like Service Level, Workload, and data reduction. There are one or more Storage Resources in each Storage Container. The vSphere administrator creates vVols in the Storage Resources according to the capabilities that the VM requires.
Protocol endpoint	A thin device that acts as a data path between the vVols on a storage array and their virtual machines on the ESXi host.

VASA Provider A software component that implements vSphere APIs for Storage Awareness (VASA). The vSphere administrator manages the resources on the array that are allocated to the VMs through the VASA Provider.

Management responsibilities

This table shows who creates and manages each component.

Table 7. Management responsibilities in a vVol environment

Component	Manager
Virtual Machine	vSphere administrator
vVol	vSphere administrator
Storage Container	Storage administrator
Storage Resource	Storage administrator
Protocol Endpoint	Storage administrator

Remote replication in a vVols environment

PowerMaxOS 5978.669.669 and later can remotely replicate vVols using SRDF/A. This diagram shows the architecture of a replicated vVol environment.

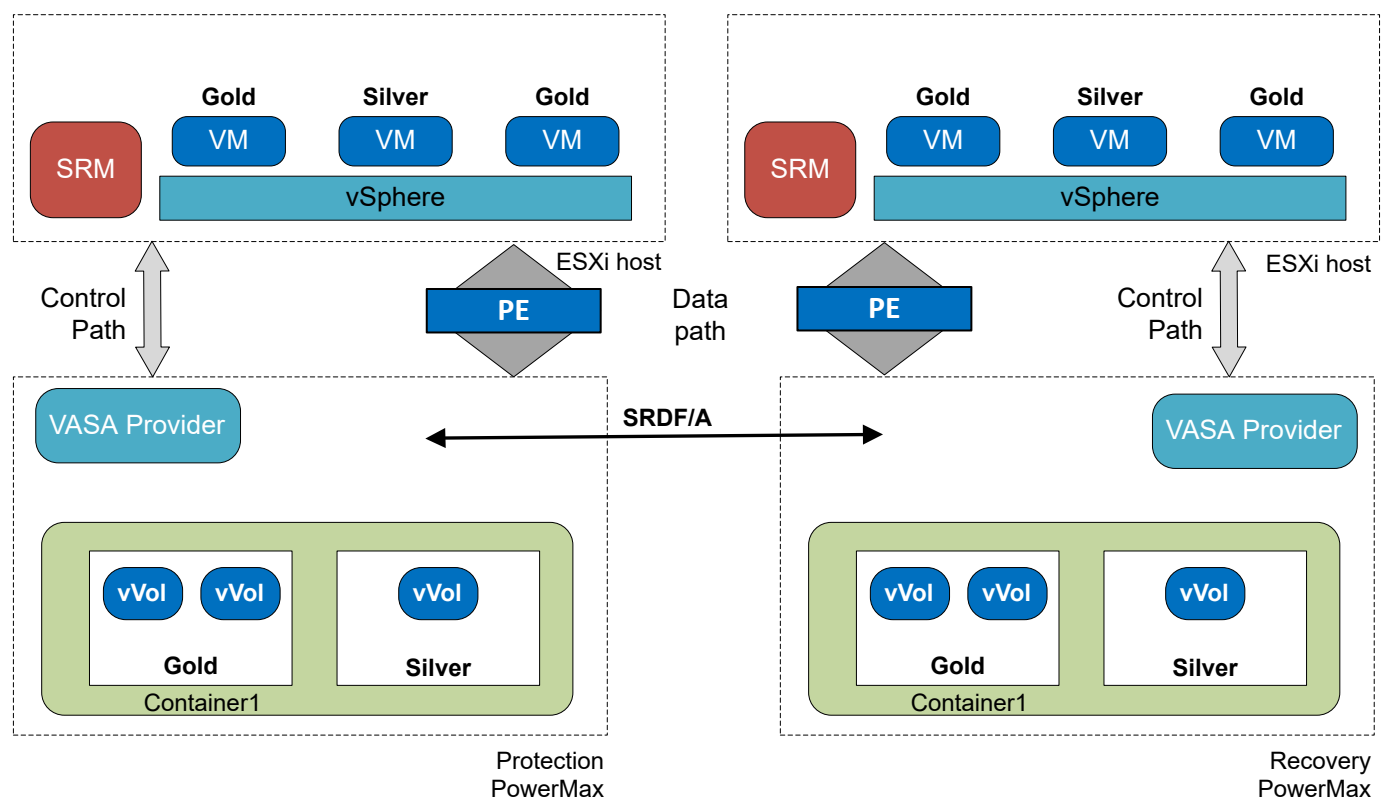


Figure 24. SRDF replication of vVols in PowerMaxOS

As the diagram shows, there is one extra component in this environment: VMWare Site Recovery Manager (or SRM). SRM is a disaster recovery product that orchestrates the failover and failback operations between the production environment and the disaster recovery environment. The vSphere administrator uses SRM to manage the failover and failback of vVols. As with other SRDF configurations failover, and the subsequent failback, can be both planned and unplanned.

Establish the replication configuration

Both the storage administrator and the vSphere administrator are involved in establishing a replication configuration.

The storage administrator:

1. Creates the Storage Containers and Storage Resources on the production environment
2. Provisions Protocol Endpoints for each ESXi host that uses the array
3. Creates Storage Containers on the disaster recovery environment
4. Creates Replication Groups that contain Storage Containers on both the production and disaster recovery environments

This step includes selecting SRDF/A as the communications protocol.

Once the storage administrator has set up the replication configuration at the two sites, the vSphere administrator:

1. Registers the VASA Provider
2. Creates vVols in the Storage Resources on the production environment
3. Sets up Storage Resource Policies

These policies define which vVols are replicated to the disaster recovery environment. The policies also define whether point-in-time copies of vVols are made and how frequently the copies are taken.

4. When necessary, uses SRM to carry out failover and failback operations between the production and disaster recovery sites

The vSphere administrator defines the replication services that are necessary. The storage array provides those services.

Requirements

Remote replication of a vVol environment requires specific versions of several software components on both the storage array and in the ESXi environment.

Storage array:

- PowerMaxOS 5978.669.669 and later
- Solutions Enabler Version 9.2 and later
- Unisphere for PowerMax Version 9.2 and later

ESXi environment:

- vSphere Version 7.0
- SRM Version 8.3

Management facilities

Solutions Enabler and Unisphere for PowerMax have facilities to:

- Create, manage, and remove Storage Containers, Storage Resources, and Protocol Endpoints.
- Register with a VASA Provider.
- Create, manage, and remove VASA Replication Groups.

High availability

This chapter provides more detail on the high availability configurations that SRDF/Metro provides for open systems (FBA) and IBM i D910 application hosts.

Topics:

- [SRDF/Metro](#)
- [SRDF/Metro life cycle](#)
- [SRDF/Metro configuration changes](#)
- [SRDF/Metro resilience](#)
- [Mobility ID with ALUA](#)
- [Disaster recovery facilities](#)
- [Deactivate SRDF/Metro](#)
- [SRDF/Metro restrictions](#)

SRDF/Metro

In traditional SRDF, R1 devices are Read/Write accessible. R2 devices are Read Only or Write Disabled. However, in the high-availability SRDF/Metro configuration:

- R2 devices are Read/Write accessible to application hosts.
- Application hosts can write to both the R1 and R2 side of the device pair.
- R2 devices assume the same external device identity (geometry, device WWN) as the R1 devices.

This shared identity means that the R1 and R2 devices appear to application hosts as a single, virtual device across the two arrays.

SRDF/Metro can be deployed in either a single, multipathed host, or in a clustered host environment.

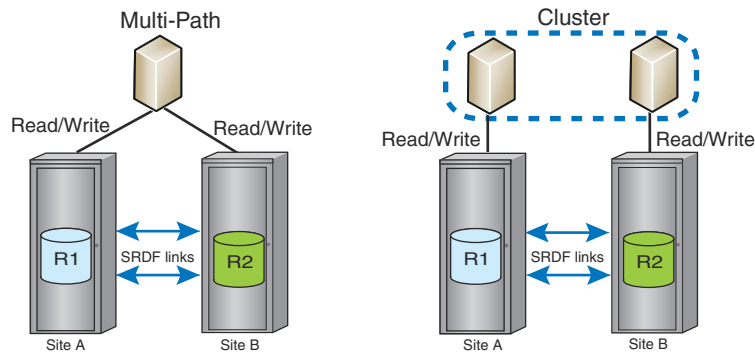


Figure 25. SRDF/Metro

Hosts can read and write to both the R1 and R2 devices in an SRDF/Metro configuration:

- In a single host configuration, a single host issues I/O operations. Multipathing software directs parallel reads and writes to each array.
- In a clustered host configuration, multiple hosts issue I/O operations to both sides of the SRDF device pair. Each cluster node has dedicated access to an individual storage array.
- In both single host and clustered configurations, writes to the R1 or R2 devices are synchronously copied to the paired device. SRDF/Metro software resolves write conflicts to maintain consistent images on the SRDF device pairs. The R1 device and its paired R2 device appear to the host as a single virtualized device.

Other characteristics of SRDF/Metro are:

- SRDF/Metro is managed using either Solutions Enabler or Unisphere.
- SRDF/Metro requires a license on both arrays.
- Storage arrays can simultaneously contain SRDF groups for SRDF/Metro operations and traditional SRDF groups.
- The arrays can be up to 200 km (125 miles) apart.
- The arrays are typically in separate fault domains for extra resilience.


Key differences in SRDF/Metro compared to traditional SRDF

In SRDF/Metro configurations:

- R2 device is Read/Write accessible to the host.
- The R1 and R2 devices have federated personalities. That is, both sides of the SRDF device pair appear to the hosts as the same device. When the device pair becomes RW on the SRDF link, the R2 device assumes the personality of its R1 partner (such as geometry and device WWN).

The R2 device retains this assumed identity until the device pair is removed from the SRDF/Metro configuration. From then on the two devices have unique personalities to avoid device collusion. The user can reset the device personality manually, but this is not required.

- Both arrays in the configuration monitor the health of the devices in the SRDF group and of the SRDF link between the arrays. If any of the following occur, the arrays cooperate to ensure that data remains available for reads and writes without allowing inconsistencies to occur between them.
 - A device fails.
 - The SRDF link fails.
 - Devices are made Not Ready on the SRDF link.
- Devices in a SRDF/Metro configuration operate in Active SRDF mode. [SRDF modes of operation](#) on page 21 contains information about SRDF modes.
- There are two extra SRDF pair states that only SRDF/Metro uses. These states indicate that the configuration is ready to provide high data availability:
 - ActiveActive for configurations using the Witness options (Array and Virtual)
 - ActiveBias for configurations using bias

 **NOTE:** R2 devices should not be presented to the cluster until they reach one of these two states at which time both devices present the same WWN.

[SRDF/Metro resilience](#) on page 62 has more information about the Witness and bias mechanisms.

Device management

All device pairs in an SRDF/Metro group are managed together for all supported operations, except when changing the contents of the group:

- Create pair and move pair operations can add devices to the SRDF/Metro group.
- Delete pair and move pair operations can remove devices from the SRDF/Metro group.

Failure resilience

While providing high data availability, SRDF/Metro ensures that the data is consistent on both sides of the configuration. If data consistency can no longer be guaranteed due to, for example, the failure of a device or the SRDF link, SRDF/Metro:

- Allows one side to remain accessible to the host (the winner)
This side continues to service I/O requests from the host.
- Makes the other side inaccessible to the host (the loser)

Making only one side accessible to the host eliminates the possibility of a split brain situation occurring. A split brain scenario could result in inconsistent data accumulating on both sides of the SRDF/Metro group.

SRDF/Metro provides two mechanisms for determining the winner:

- **Witness option:** A Witness is a third party that mediates between the two sides to help decide which remains available to the host if there is a failure. The Witness method provides intelligent selection of the winner at the time of failure to facilitate continued host availability.

The Witness option is the default.

SRDF/Metro provides two types of Witnesses, Array and Virtual:

- **Array Witness :** The operating environment on a third array acts as the mediator to decide the side of the device pair that remains R/W accessible to the host. It gives priority to the winner, but should that side be unavailable the other side remains available.
- **Virtual Witness (vWitness) option:** vWitness provides the same functionality as the Array Witness option, but it is packaged to run in a virtual appliance, rather than on the array.

The vWitness and Array Witness options are treated the same in the operating environment, and can be deployed independently or simultaneously. When deployed simultaneously, SRDF/Metro favors the Array Witness option over the vWitness option, as the Array Witness option has better availability.

- **Device Bias option:** Device pairs for SRDF/Metro have a *bias* attribute. SRDF/Metro designates the R1 side of a device pair at the time the devices become Ready on the SRDF link as the winner. If the device pair becomes Not Ready (NR) on the link, the R1 (bias side) remains accessible to the host. The R2 (nonbias side) is inaccessible to the hosts.

The Witness option has key advantages over the Device Bias option:

- The Witness option provides intelligent selection of the winner and loser at the time of a failure.
The Device Bias option, on the other hand, decides on the winner and loser in advance of any failure.
- As a result, the Witness option can designate as the winner the side that appears most capable of providing continued data availability to the host.
The Device Bias option can only designate the bias (R1) side as the winner. Device Bias cannot make the R2 side available to the host. If the R1 side is unavailable, the host loses all data availability that the devices provide.

It is recommended that you do not use Device Bias in a production environment.

[SRDF/Metro resilience](#) on page 62 has more information about these failure-recovery mechanisms.

Disaster recovery

SRDF/Metro provides high availability while traditional SRDF provides disaster recovery. The two technologies can exist together to provide both high availability and disaster recovery. [Disaster recovery](#) on page 30 has more information about using disaster recovery in an SRDF/Metro configuration.

Operating environment

Each array in a SRDF/Metro configuration runs HYPERMAX OS 5977.691.684 or later, or any version of PowerMaxOS 5978.

Some features of SRDF/Metro require later versions of HYPERMAX OS or PowerMaxOS. Where this applies, the following sections identify the versions required.

SRDF/Metro life cycle

The following demonstrates the life cycle of an SRDF/Metro configuration in relation to the life cycle of an application:

1. Initial provisioning and configuration setup
Allocate devices for the application on two arrays, turn those devices into an SRDF/Metro configuration, and activate the configuration.
2. Adding and removing devices
As the storage needs of the application change, add devices to and remove them from the SRDF/Metro configuration.
3. Removing the SRDF/Metro configuration
When the application reaches the end of its life, the SRDF/Metro configuration becomes obsolete and is dismantled.

Initial provisioning and configuration setup

The application requires storage that must be provisioned and made available to the application host:

1. Create the SRDF/Metro configuration.
2. Synchronize each device pair in the configuration.
3. Enter high-availability mode.

Create the SRDF/Metro configuration

First, the storage administrator determines the storage needs of the application and identifies an appropriate set of devices to meet those needs. The administrator also identifies another, identical set of devices on the remote array. These devices are the other side of the SRDF/Metro device pairs.

To set up the SRDF/Metro configuration, the administrator:

1. Creates two, empty SRDF groups, one on each array
2. Uses the create pair operation to create each device pair and add it to the two groups
When adding a pair, the administrator indicates that it is a SRDF/Metro device pair which marks the group as an SRDF/Metro group. In response, SRDF/Metro sets each pair to operate in Active mode.
3. Makes the device pairs Ready on the SRDF link between the arrays
As part of this step, the administrator defines the failure resilience mechanism that each pair is to use (Witness or Device Bias). Since Witness is the default, the administrator specifies a mechanism only if a pair uses Device Bias.

Synchronize each device pair

Next, each device pair synchronizes its data in preparation for providing high availability:

- The pair state becomes SyncInProg.
- Invalid tracks synchronize.
- R2 copies the device personality (for example, geometry and identity) of the R1 device.
- R2 adopts the personality of the R1 device.

While the devices are synchronizing, the application can run. However, only the R1 side can process I/O requests. The R2 side is inaccessible to the application host.

Enter high-availability mode

Once synchronization is complete, the final steps occur to provide high availability. SRDF/Metro:

- Makes the R2 devices accessible to the host using the federated R1 personalities
- Sets the state of each pair to one of:
 - ActiveActive—if a Witness resiliency mechanism is in use.
 - ActiveBias—is the Device Bias resiliency mechanism is in use.

From now on, SRDF/Metro monitors the SRDF link and the devices on that link. If the link fails, or either partner in a device pair becomes Not Ready, SRDF/Metro decides on the winning side according to the resiliency mechanism in use. The winning side remains accessible to the host while the losing side becomes inaccessible.

NOTE: When using one of the Witness options, SRDF/Metro may change the winning and losing sides at any time, based on the relative robustness of the two sides. SRDF/Metro always reports the winning side as the R1 device and the losing side as the R2 device. So each switch causes an apparent swap of the R1 and R2 personalities in a device pair.

The application host now discovers the R2 devices through their federated personalities. Once they are discovered, the configuration is providing high data availability:

- The application host can send write operations to either side. When SRDF/Metro receives a write operation it sends the I/O to the other array in the configuration. SRDF/Metro confirms the I/O to the application host only when that other array acknowledges receipt of the Write operation.
- Reading from either side always returns the same data since both sides have identical content.

Add and remove devices

The storage needs of the application change during its lifetime. To accommodate these changes, the storage administrator adds devices to or removes them from the SRDF/Metro group.

Add devices

The storage administrator uses one of two operations to add devices to an SRDF/Metro group:

- Move pair—to add an existing SRDF device pair
- Create pair—to create a pair from two non-SRDF devices

For the purposes of this example, assume that the administrator is moving existing SRDF device pairs into a SRDF/Metro group. The criteria that the devices must fulfill are:

- The devices must be in an SRDF session between the same two arrays that SRDF/Metro uses.
- Each device pair operates in synchronous or adaptive copy mode.
- The devices can be Ready or Not Ready on the SRDF link.
- All device pairs to move into the SRDF/Metro group have their R1 and R2 sides aligned.

The alignment does not have to match the alignment in the SRDF/Metro group. If the alignment is different, the R2 sides cannot owe data to the R1 side.

Once the devices are in the SRDF/Metro group, they synchronize:

1. The devices are set to Not Ready on the SRDF link.
2. The devices are set to operate in Active mode.
3. The devices are set to Ready on the SRDF link.
4. The SRDF pair state for each of the newly added pairs changes to SyncInProg.

The SRDF pair state of the pairs that were already in the SRDF/Metro group remains as ActiveActive or ActiveBias.

5. Invalid tracks synchronize from the R1 to the R2 side of each newly added pair.
6. SRDF/Metro switches the R1 and R2 sides of the newly added pairs if they differ from the R1 and R2 polarity of pairs already in the group.
7. SRDF/Metro copies the device personality (for example, identity and geometry) of the R1 device to the R2 device in each of the newly added pairs.
8. The R2 devices in each newly added pair set their identities to that of their R1 partners.

While this process is in progress, the R1 device in each newly added pair is accessible to the application host. The R2 devices are inaccessible.

Once synchronization is complete, SRDF/Metro:

1. Makes the R2 devices in each newly added pair accessible to the application host using the federated R1 personality copied from the R1 side
2. Sets the SRDF pair state of each pair to ActiveActive or ActiveBias, depending on the failure resiliency mechanism in use
3. Monitors the SRDF link and the newly added devices on both sides of the SRDF/Metro configuration

The application host now discovers the newly added R2 devices, that have the identity of their R1 partners. Now high data availability is in force.

Remove devices

The storage administrator can use two operations to remove devices from an SRDF/Metro configuration:

- Delete pair—delete the SRDF associated between a device pair, remove the two devices from their SRDF/Metro groups. The devices are now regular devices one of which is still available to the application host.
- Move pair—move the pair out of the SRDF/Metro configuration into a specified SRDF configuration between the same pair of arrays. The devices retain their SRDF relationship.

For the purposes of this example assume that:

- The SRDF/Metro configuration is providing high data availability.
- A device pair is to move to another SRDF configuration.

For the device pair to move, SRDF/Metro:

- Sets the mode of the device pair to synchronous SRDF in the destination SRDF group
- Makes the devices Not Ready on the SRDF link

In some circumstances, the storage administrator can specify which device in the pair remains accessible to the application host after the move. That is, which device is the R1 device in the destination SRDF group. The user can specify the R1 device if the device pair was Ready on the SRDF link when the move occurred. Otherwise, the R1 device at the time of the move remains the R1 device and is accessible to the application host. The R2 device is inaccessible to the application host.

Remove the SRDF/Metro configuration

Removing all the devices from a SRDF/Metro group removes the corresponding SRDF/Metro configuration. The storage administrator can use the delete pair or move pair to remove all the device pairs except the final one. To remove that pair, and the SRDF/Metro configuration, the storage administrator uses the delete pair operation.

Once the SRDF/Metro group is empty, it is no longer considered to be a SRDF/Metro group. The group is now available for other uses such as a group of traditional SRDF devices.

SRDF/Metro configuration changes

During its lifetime, the content of an SRDF/Metro group can change by adding and removing devices.

Add devices

There are two ways of adding devices to an SRDF/Metro group:

- Create a pair of devices using devices that are not in an SRDF relationship.
- Move an existing SRDF pair into the group.

Create pair

When creating a pair from two existing devices, you can specify which device retains its data. As part of the pair creation process, SRDF/Metro copies the data from that device to the other, replacing any data that it contained. The device that retains its data becomes the R1 device in the pair.

When adding devices that contain no data, SRDF/Metro can format the devices when adding them to the group as a device pair. Formatting removes any data on the devices, and often completes faster than when data has to be copied between the device pair. To be able to format devices, they:

- Must be unmapped or user-not-ready so that no I/O is occurring
- Cannot be part of another replication session

Move pair

When moving an existing pair into the group, the R1 and R2 sides do not have to align with the devices already in the group. SRDF/Metro switches the R1 and R2 sides when making the new pair ready on the SRDF link.

Process

The process of adding a device pair to a SRDF/Metro group is:

1. The R1 side of each device pair is made accessible to the host. The R2 side is inaccessible to the host.
SRDF/Metro uses a slightly different approach when it formats a device pair. In this case, both sides are inaccessible to the host until the formatting is complete. Only then does the R1 side become accessible to the host.
2. Once the devices are Ready on the SRDF link, the device pair enters the SyncInProg state while the two sides synchronize their data.
That is, SRDF/Metro aligns the data on the R2 device to match that on the R1 device. So when moving an existing pair into a SRDF/Metro group, the R2 side cannot owe data to the R1 side.
3. If necessary, SRDF/Metro switches the R1 and R2 sides so that they align with the device pairs already in the group, when synchronization is complete.
4. SRDF/Metro copies the personality attributes, such as geometry and identity, from the R1 side to the R2 side.
5. Once both devices have the same identity, SRDF/Metro:
 - Makes the R2 device accessible to the host
 - Sets the SRDF pair state of the added devices to ActiveActive or ActiveBias to match the state of all other device pairs in the SRDF/Metro group

Remove devices

There are two ways of removing devices from an SRDF/Metro group:

- Delete—remove the SRDF relationship between a pair of devices and move the two devices out of the SRDF/Metrogroup.
- Move—move a device pair from the SRDF/Metro group to another SRDF group.

In either case, the R1 device at the time the pair are removed remains accessible to the host. However, you can control which device in the pair remains accessible to the host. So if you prefer the R2 side remains accessible. Making the R2 accessible, switches the R1 and R2 sides. Thus, the side that was R2 now becomes R1. To specify the side that remains available to the host requires that the devices are Ready on the SRDF link.

Delete pair

When deleting a pair:

- The SRDF pairing between the two devices is removed. They are no longer a device pair but rather two non-SRDF devices. The device that was the R1 device remains accessible to the host as a regular device.
- It is possible to delete the last device in an SRDF/Metro group when that device is Not ready on the SRDF link.
- To make the R2 device accessible to the host, requires that the device pair was in the ActiveActive or ActiveBias state when deleted.
- After being removed from the SRDF/Metro group, the devices retain the identities that they last used in that group. To be able to reuse devices that are inaccessible to the host requires that the device identities are reset manually.

Move pair

The move pair operation moves a device pair from the SRDF/Metro group to another SRDF group:

- The device pairing is retained allowing replication to continue within the destination SRDF group.
- The destination group must be Synchronous. It cannot be active (in another SRDF/Metro group) or Asynchronous.
- The move pair operation must leave at least one device in the SRDF group.
- To make the R2 device accessible to the host, requires that the device pair was in the ActiveActive or ActiveBias state when deleted.

SRDF/Metro resilience

In normal operation, PowerMaxOS and HYPERMAX OS use the SRDF links to maintain consistency of the data on both sides of a SRDF/Metro pair. That consistency of data cannot be ensured when hosts have write access to both sides and either of the following occurs:

- The device pair becomes Not Ready (NR) on the SRDF link.
- One of the devices in the pair fails.

In either situation, SRDF/Metro selects one device in the pair to remain accessible to the host (the winner) and make the other device inaccessible (the loser). This decision applies to **all** device pairs in the SRDF/Metro configuration.

SRDF/Metro provides two ways of selecting the winner and loser:

- Witness
- Device bias

The following sections explain these methods and provide background information about how SRDF/Metro ensures that no data is lost due to the failure.

i **NOTE:** The selected method comes into effect only when a failure occurs and the devices pairs are ActiveActive or ActiveBias. Before transitioning to one of those states, only one side of a pair is accessible to the host. So there is no need for SRDF/Metro to take any special action.

Witness

The Witness method uses a third party (a witness) to enhance the response of an SRDF/Metro configuration to a failure. The witness intelligently selects the winner and loser sides at the time the failure occurs to provide continued host availability. The primary, and recommended, form of resilience is Witness.

The pair state of a device pair that uses a Witness is ActiveActive when the pair can provide high data availability. Once the devices in a configuration are ready to move to the ActiveActive state, the arrays on the R1 and R2 sides:

- Negotiate which Witness they are to use.
- Periodically renegotiate the winning and losing sides, based on the relative robustness of the two sides in the configuration.

This capability is available only on arrays that run PowerMaxOS.

[Witness negotiation and selection](#) on page 65 describes how the arrays initially select a Witness and how they regularly renegotiate the winning and losing sides.

If a failure occurs, the arrays on both sides of the SRDF/Metro group attempt to contact the Witness. The purpose of the Witness is to direct one array to act as the winner and the other side as the loser. The winner continues to be accessible to the host.

In determining the array that remains accessible to the host, the Witness gives preference to the winner chosen during the last negotiation between the two arrays. However, should that array not communicate with the Witness, the Witness turns what was the loser into the winner, and that array remains host-accessible. This mechanism helps to continue to provide the host with data accessibility should the winner fail.

The following sections describe:

- The two forms of witness:
 - Array Witness
 - Virtual Witness (vWitness)
- The process used to select a witness
- Witness failure scenarios

Array Witness

When using the Array Witness method, SRDF/Metro uses a third "witness" array to determine the winning side. The witness array runs one of these operating environments:

- PowerMaxOS 5978.144.144 or later
- HYPERMAX OS 5977.945.890 or later
- HYPERMAX OS 5977.810.784 with ePack containing fixes to support SRDF N-x connectivity
- Enginuity 5876 with ePack containing fixes to support SRDF N-x connectivity

The Array Witness must have SRDF connectivity to both the R1-side array and R2-side array. SRDF remote adapters (RA's) are required on the witness array with applicable network connectivity to both the R1 and R2 arrays.

For redundancy, there can be multiple witness arrays but only one witness array is used by an individual SRDF/Metro group at any one time. The two sides of the SRDF/Metro group agree on the witness array to use when the devices in the group are ready to transition to the ActiveActive state. If the auto configuration process fails and no other applicable witness arrays are available, SRDF/Metro uses the Device Bias method.

The Array Witness method requires two SRDF groups: one between the R1 array and the witness array, and a second between the R2 array and the witness array. A witness group:

- Is an SRDF group that enables an array act as a Witness for any or all SRDF/Metro sessions that are connected to the array at the other side of the witness group. The group has no other purpose.
 - There can be only one witness group between any pair of arrays.
 - A witness group does not contain any devices.
 - It is not possible to remove a witness group that is protecting a SRDF/Metro configuration unless there is another Witness that the configuration can use.
- Must be online when required to act as a Witness.

Both witness groups must be online when the device pairs in the SRDF/Metro configuration become Ready on the SRDF link.

NOTE: The term *witness array* is relative to a single SRDF/Metro configuration. While the array acts as a Witness for that configuration, it may also contain other SRDF groups, including SRDF/Metro groups.

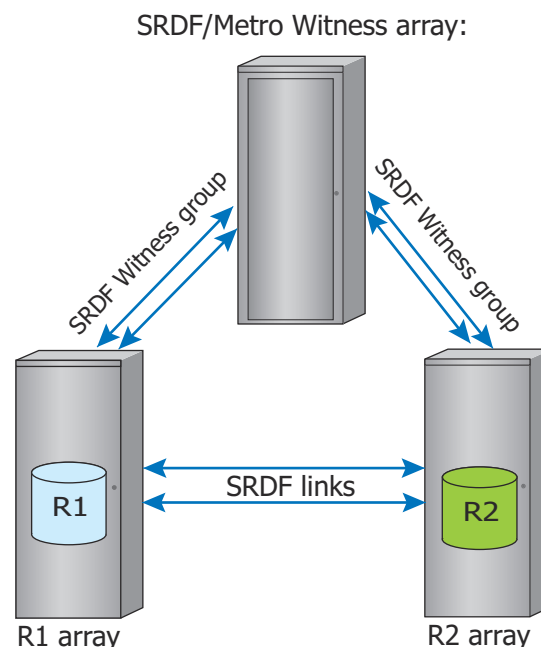


Figure 26. SRDF/Metro Array Witness and groups

SRDF/Metro management software checks that the Witness groups exist and are online when carrying out establish or restore operations. SRDF/Metro determines which witness array an SRDF/Metro group is using, so there is no need to specify the Witness. Indeed, there is no means of specifying the Witness.

When the Array Witness method is in operation, the state of the device pairs is ActiveActive.

If the witness array becomes inaccessible from both the R1 and R2 arrays, the state of the device pairs becomes ActiveBias.

Virtual Witness (vWitness)

vWitness has similar capabilities to the Array Witness. However, it is packaged to run in a virtual appliance (vApp) on a VMware ESXi server, not on an array. For redundancy, there can be up to 32 vWitnesses.

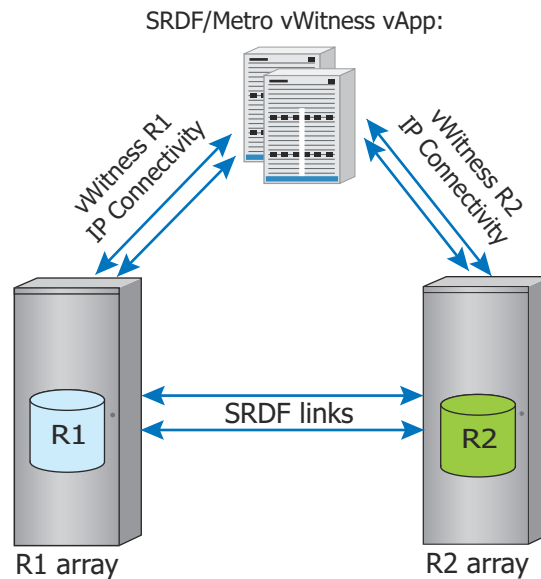


Figure 27. SRDF/Metro vWitness vApp and connections

The management guests on the R1 and R2 arrays maintain multiple IP connections to redundant vWitness virtual appliances. These connections use TLS/SSL for secure connectivity.

After creating IP connectivity to the arrays, the storage administrator can use SRDF management software to:

- Add a vWitness to the configuration. This action does not affect any existing vWitnesses. Once the vWitness is added, it is enabled for participation in the vWitness infrastructure.
- Query the state of a vWitness configuration.
- Suspend a vWitness. If the vWitness is servicing an SRDF/Metro session, this operation requires a force flag. The SRDF/Metro session is in an unprotected state until it renegotiates with another witness, if available.
- Remove a vWitness from the configuration. Once the vWitness is removed, SRDF/Metro breaks the connection with vWitness. The administrator can only remove vWitnesses that are not servicing active SRDF/Metro sessions. Nor can they remove a vWitness when there is no alternative Witness (array or vWitness) for the SRDF/Metro configuration to use.

Witness negotiation and selection

SRDF/Metro checks that at least one witness is available before setting the devices in a SRDF/Metro pair Ready on the SRDF link. The two arrays that host the device pair carry out the negotiation of which witness to use. This negotiation occurs when the devices are ready to move to the ActiveActive pair state.

NOTE: The process of selecting a witness is autonomous. The administrator cannot specify the witness that any SRDF/Metro pair is to use.

Witness negotiation

Each side of the SRDF/Metro configuration maintains a list of witnesses, that the administrator sets up. To begin the negotiation process, the nonbias side sends its list of witnesses to the bias side. On receiving the list, the bias side compares it with its own list of witnesses. The first matching witness definition is selected as the witness and the bias side sends its identification back to the nonbias side. The two sides then establish communication with the selected witness.

Before allowing the devices in a pair to become Ready on the SRDF link, SRDF/Metro checks that the arrays have at least one witness definition in common. If there is no common witness, SRDF/Metro does not allow the devices to become Ready on the link. In this situation, the administrator reconfigures either or both arrays so that they have at least one witness definition in common.

NOTE: If the arrays cannot agree on a witness, the session operates in Device Bias mode. However, should a common witness subsequently become available, the two sides negotiate the use of that witness. The sides then continue in witness mode.

Intelligent witness management

When both sides run PowerMaxOS, the negotiation process is enhanced to include a decision on the winning side if there is a failure. The selection of the winning side is based on (in priority order):

1. The side that has connectivity to the application host (requires PowerMaxOS 5978.444.444 or later)
2. The side that has a write pending (WP) value that is less than 80% of the System WP Limit (requires PowerMaxOS 5978.669.669 or later)
3. The side that has a SRDF/A DR leg
4. Whether the SRDF/A DR leg is synchronized
5. The side that has an active SRDF/A DR leg
6. The side that has a ready mirror on the SRDF/A DR leg
7. The side that has more than 50% of the RA or FA directors that are available
8. The side that is the bias side

The two sides regularly repeat this selection process for each SRDF/Metro group to ensure that the winning side remains the one that is most preferable. So the winning side may change during the SRDF/Metro session. SRDF/Metro always reports the winning side as the R1 device. It reports the losing side as R2. So each switch in the winning side causes an apparent swap in the R1 and R2 personalities in the session.

The assessment of the winning and losing side occurs separately for each SRDF/Metro group that exists between two arrays. So, on a particular array, some devices could be R1 devices while others are R2 devices. Which are R1 and which are R2 depends on the outcome of assessing their respective SRDF/Metro groups.

Witness connection failure

While using a witness, an error could cause either or both sides of a SRDF/Metro session to lose contact with that witness. This table summarizes the actions that SRDF/Metro takes in these cases:

Error	SRDF/Metro action
One side loses contact with the witness.	<p>The session continues to run, but in a degraded, high-availability condition. The side that remains in contact with the witness is designated the winner. That side remains available to the application host should the two sides lose contact with each other.</p> <p>While only one side has contact with the witness, the witness status is Degraded.</p> <p>If contact with the witness resumes, the session operates in the normal way.</p>
Both sides lose contact with the witness.	<p>If at least one other witness is available, the two sides negotiate the use of another witness.</p> <p>When there is no other witness available:</p> <ul style="list-style-type: none">• The state of the device pair changes from ActiveActive to ActiveBias. SRDF/Metro considers the session to still be in witness mode. So the ActiveBias pair state represents a degraded, high availability condition.• The R1 device is designated the winner. That side remains available to the application host should the two sides lose contact with each other.• The witness status is Failed.

[Witness failure scenarios](#) on page 67 has more on how SRDF/Metro reacts to various failure scenarios.

Characteristics of witness negotiation

As a result of witness negotiation:

- An SRDF/Metro configuration uses one witness at a time.
- An individual witness may simultaneously protect multiple SRDF/Metro sessions.
- When multiple witnesses are available, different SRDF/Metro sessions between two arrays may use different witnesses. Alternatively they may all use the same witness.
- An error may cause the devices in an SRDF/Metro session to become Not Ready on the SRDF link. When the devices subsequently become Ready on the link, the session might use a different witness.

Witness failure scenarios

These diagrams show how SRDF/Metro reacts to various failure scenarios when either Witness option is in use.

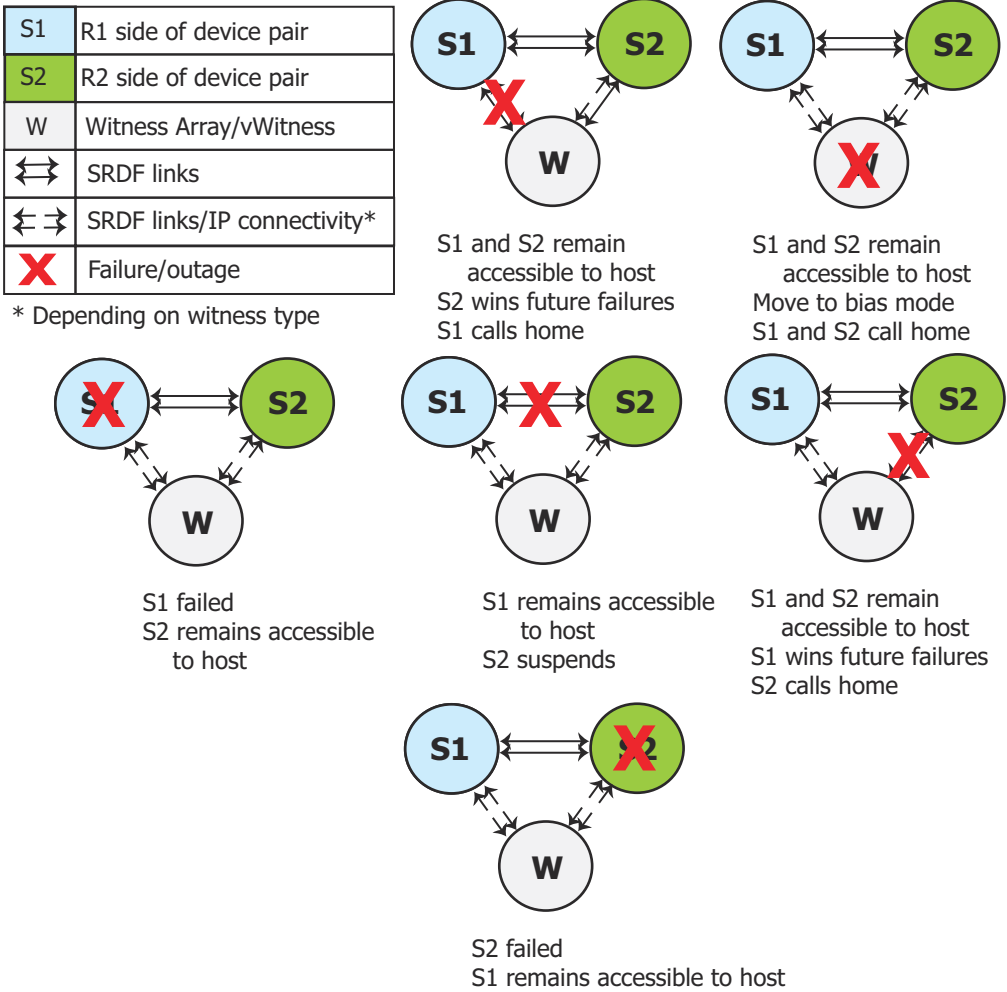


Figure 28. SRDF/Metro Witness single failure scenarios

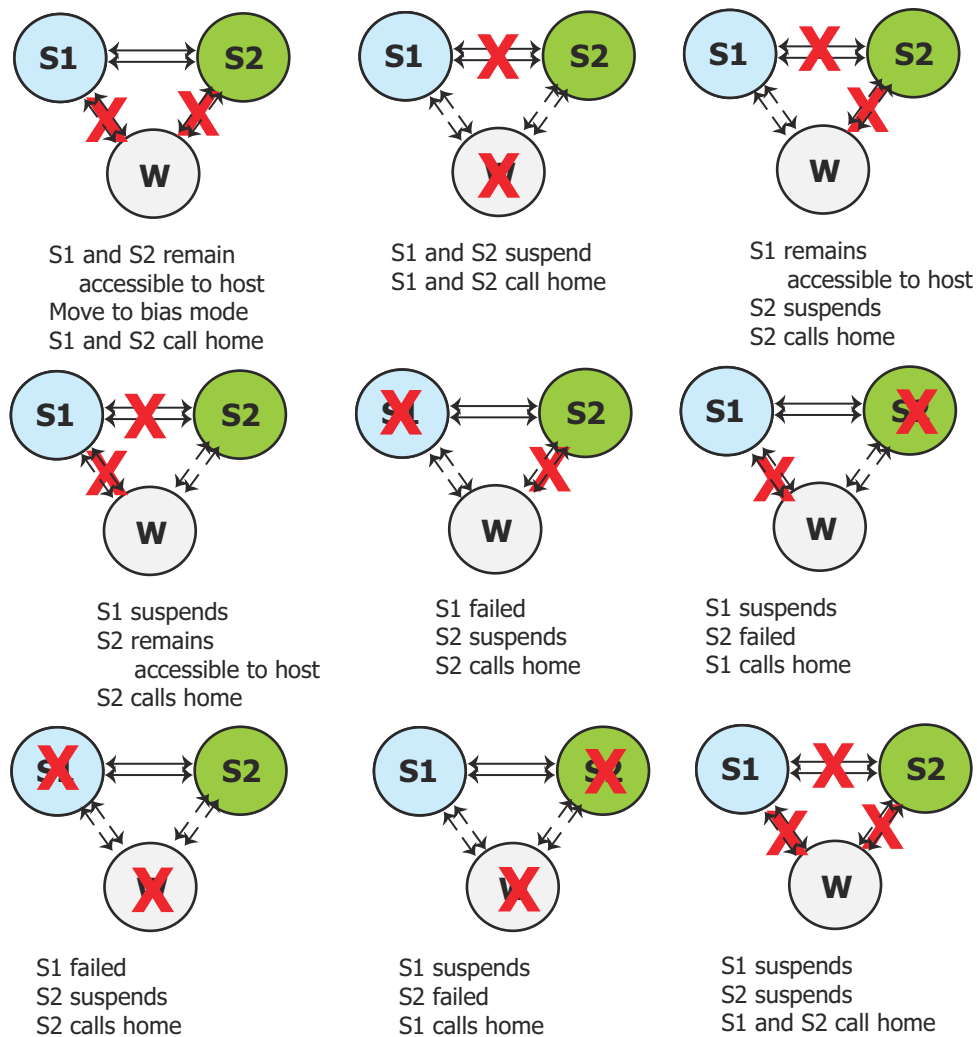


Figure 29. SRDF/Metro Witness multiple failure scenarios

Device Bias

When the devices in the SRDF/Metro group become Ready on the SRDF link, PowerMaxOS and HYPERMAX OS:

- Mark the array that contains the R1 devices as the bias side and the other array as the nonbias side.
- Designate each device on the bias side as the winner.
- Designate each device on the nonbias side as the loser.

If a device pair becomes Not Ready on the SRDF link, or either device in a pair fails, PowerMaxOS and HYPERMAX OS:

- Let all devices on the winner side remain accessible to the host.
- Make all devices on the loser side inaccessible to the host.

A pair that uses device bias has the ActiveBias state when the pair can provide high data availability.

When a pair is in the ActiveBias state, it is possible to change the bias to the other array in the configuration. Changing the bias also changes the R1 and R2 designations. That is, devices that were previously R1 devices are now R2 devices and R2 devices are now R1 devices. The winner and loser designations also change.

Note that if there is a failure on the R1 side, the host loses all connectivity to the devices in the SRDF/Metro group. The Device Bias method cannot make the R2 devices available to the host.


Preventing data loss

When a failure occurs, SRDF/Metro must ensure that there is no loss of data because of the failure, as well as deciding which side that remains host-accessible (the winner). Having decided on the winner, the steps that SRDF/Metro takes to prevent data loss are:

1. Briefly stall all I/O operations to all devices on both sides of the SRDF/Metro group.
2. Hold I/O operations to both sides of all pairs in the SRDF/Metro group without processing them.
3. Set all devices on both sides of the SRDF/Metro group Not Ready on the SRDF link.
4. Make all devices on the loser side of the group inaccessible to the host.

Devices on with winner side remain accessible to the host.

5. Clear the Stalled state.

 **NOTE:** The Stalled state is transient and typically transparent to applications. So, it does not appear as a discrete state, but is incorporated in the Suspended and Partitioned states.

6. Reject all the I/O operations that were held in step 2 with a retry sense code.

The use of the retry sense code tells the host to send these I/O operations again.

Subsequent I/O operations are accepted by devices on the winner's side, and rejected by devices on the loser's side. This applies to all I/O operations, whether they are new or ones that are retried.

When the failure is resolved, SRDF/Metro:

1. Sets the devices in the SRDF/Metro group to Ready on the SRDF link.
2. Re-synchronizes each device pair in the SRDF/Metro group.
3. Sets each pair state to ActiveActive or ActiveBias once the pair can provide high data availability.

Specifying the resiliency method

Selection of the resiliency method to use for a SRDF/Metro group occurs when the group's devices are made Ready on the SRDF link. The default resiliency method is Witness. So, the administrator specifies the method only if a group is to use the Device Bias method. To change the resiliency method, set the devices to Not Ready on the SRDF link. Then set them to Ready, specifying the desired resiliency method.

Mobility ID with ALUA

Mobility ID with Asymmetric Logical Unit Access (ALUA) assigns a unique identifier to a device in a system. This identifier enables the device to be moved between arrays without the need for any reconfiguration on the host. PowerMaxOS brings Mobility ID with ALUA capabilities to SRDF/Metro. So, when both sides run PowerMaxOS you can specify the Mobility ID in the create pair operation in place of the regular device identifier.

Disaster recovery facilities

Devices in SRDF/Metro groups can simultaneously be in other groups that replicate data to a third, disaster recovery site. There are two replication solutions. The number available in any SRDF/Metro configuration depends on the version of the operating environment that the participating arrays run:

- Highly-available disaster recovery – in configurations that consist of arrays that run PowerMaxOS 5978.669.669 and later
- Independent disaster recovery – in configurations that run all supported versions of PowerMaxOS 5978 and HYPERMAX OS 5977

Highly available disaster recovery (SRDF/Metro Smart DR)

SRDF/Metro Smart DR maintains a single, disaster recovery (DR) copy of the data in a SRDF/Metro pair on a third, remote array. This diagram shows the SRDF/Metro Smart DR configuration:

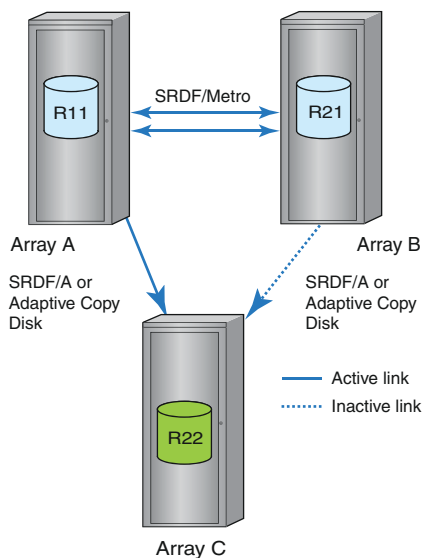


Figure 30. SRDF/Metro Smart DR

Notice that the device names differ from a standard SRDF/Metro configuration. This difference reflects the change in the device functions when SRDF/Metro Smart DR is in operation. For instance, as the diagram shows the R1 side of the SRDF/Metro on Array A now has the name R11, because it is the R1 device to both the:

- R21 device on Array B in the SRDF/Metro configuration
- R22 device on Array C in the SRDF/Metro Smart DR configuration

Arrays A and B both have SRDF/Asynchronous or Adaptive Copy Disk connections to the DR array (Array C). However, only one of those connections is active at a time (in this example the connection between Array A and Array C). The two SRDF/A connections are known as the active and standby connections.

If a problem prevents Array A replicating data to Array C, the standby link between Array B and Array C becomes active and replication continues. Array A and Array B keep track of the data replicated to Array C to enable replication and avoid data loss.

Resilience

The SRDF/Metro pair must use a Witness resilience mechanism. If the pair uses an Array Witness, the version of the operating environment on the witness array must be compatible with that running on each SRDF/Metro array. The *SRDF Interfamily Connectivity Information* document defines the versions of the operating environment that are compatible.

Operating environment

The three arrays in an SRDF/Metro Smart DR configuration run PowerMaxOS 5978.669.669 or later.

Management facilities

To set up and manage an SRDF/Metro Smart DR configuration requires either or both of:

- Unisphere for PowerMax version 9.2 or later
- Solutions Enabler version 9.2 or later

The facilities for managing SRDF/Metro Smart DR configurations are in three categories:

- The SRDF/Metro Smart DR environment
- The SRDF/Metro session
- The DR session

This table summarizes the management facilities:

Facility	Description
SRDF/Metro Smart DR environment	
Establish	Create a SRDF/Metro Smart DR environment from a concurrent SRDF environment. In the concurrent environment, one leg is the SRDF/Metro session and the other is an SRDF/A or Adaptive Copy Disk DR session.
Remove	Remove a SRDF/Metro Smart DR environment by removing one of the DR connections.
List	Display the high-level status of all SRDF/Metro Smart DR environments on a specified array.
Show	Display the configuration of a specified SRDF/Metro Smart DR environment.
Query	Display the status of all SRDF/Metro Smart DR environments on an array. For each environment, the display includes the status of its SRDF/Metro session, and the status of its DR session.
SRDF/Metro session	
Establish	Make the devices in a SRDF/Metro session RW on the SRDF link, and start an incremental resynchronization of data from R11 to R21. While this work is going on, only R11 is accessible to the hosts. When both devices contain identical data, R21 is also write accessible to the host.
Restore	Restore data from R21 to R11 in a SRDF/Metro session. While this work is going on R11 is not accessible to the host, and R21 is RW accessible to the host. Once both devices contain identical data, R11 is also write accessible to the host.
Suspend	Set the SDRF link to Not Ready, and make either R11 or R21 inaccessible to the hosts. The state of the SRDF/Metro becomes Suspended.
DR session	
Establish	Make the devices on the DR session RW on the SRDF link. Then start an incremental restore from the R11 array in the SRDF/Metro session to the DR (R22) array. While this work is going on, R11 is inaccessible to the host. R21 is accessible to the host if the state of the SRDF/Metro session is ActiveActive. Otherwise, R21 is inaccessible to the host. When the synchronization is complete, the state of the DR session become Consistent.
Restore	Restores data from R22 to R11. While this work is going on, R11 is inaccessible to the host.
Suspend	Make R22 Not Ready on the SRDF link, stopping data synchronization between the Metro session and R22. R11 remains accessible to the host. R21 is accessible to the host if the state of the SRDF/Metro session is ActiveActive. Otherwise, R21 is inaccessible to the host. The state of the DR session is Suspended.
Split	Make the devices in the DR session Not Ready on the SRDF link. If R11 is mapped to the host, it remains accessible to the host. R21 remains accessible to the host if the state of the SRDF/Metro session is ActiveActive. Otherwise R21 is inaccessible to the host. The state of the DR session is Inactive.
Failover	Make the devices in the DR session Not Ready on the SRDF link, stop synchronization between R11 or R21 and R22. Also, adjust R22 to allow the application to start on the DR side. The state of the DR session is Inactive.

Facility	Description
Failback	Make the devices in the DR session RW on the SRDF link and start an incremental resynchronization from R22 to R11. Also, make the devices in the SRDF/Metro RW on the SRDF link and begin an incremental resynchronization of data from R11 to R21.
Update R1	Make the R11 and R22 devices RW on the SRDF link and start an update of R11, while R22 is accessible to the host.
Set mode	Set the mode of the SRDF link in the DR session to either SRDF/A or Adaptive Copy Disk.

Restrictions

- The devices in each R11, R21, and R22 triple must have the same capacity.
- Devices in a SRDF/Metro Smart DR configuration cannot be:
 - CKD devices
 - BCV devices
 - Encapsulated devices
 - RecoverPoint devices
 - Data Domain devices
 - In a PPRC configuration
 - Part of a SRDF/STAR configuration
 - Part of a SRDF/SQAR configuration
 - Part of a data migration session
 - Enabled for MSC

Independent disaster recovery

Devices in SRDF/Metro groups can simultaneously be part of device groups that replicate data to a third, disaster-recovery site.

Either or both sides of the Metro region can be replicated. An organization can choose which ever configuration that suits its business needs. The following diagram shows the possible configurations:

NOTE: When the SRDF/Metro session is using a witness, the R1 side of the Metro pair can change based on the witness determination of the preferred side.

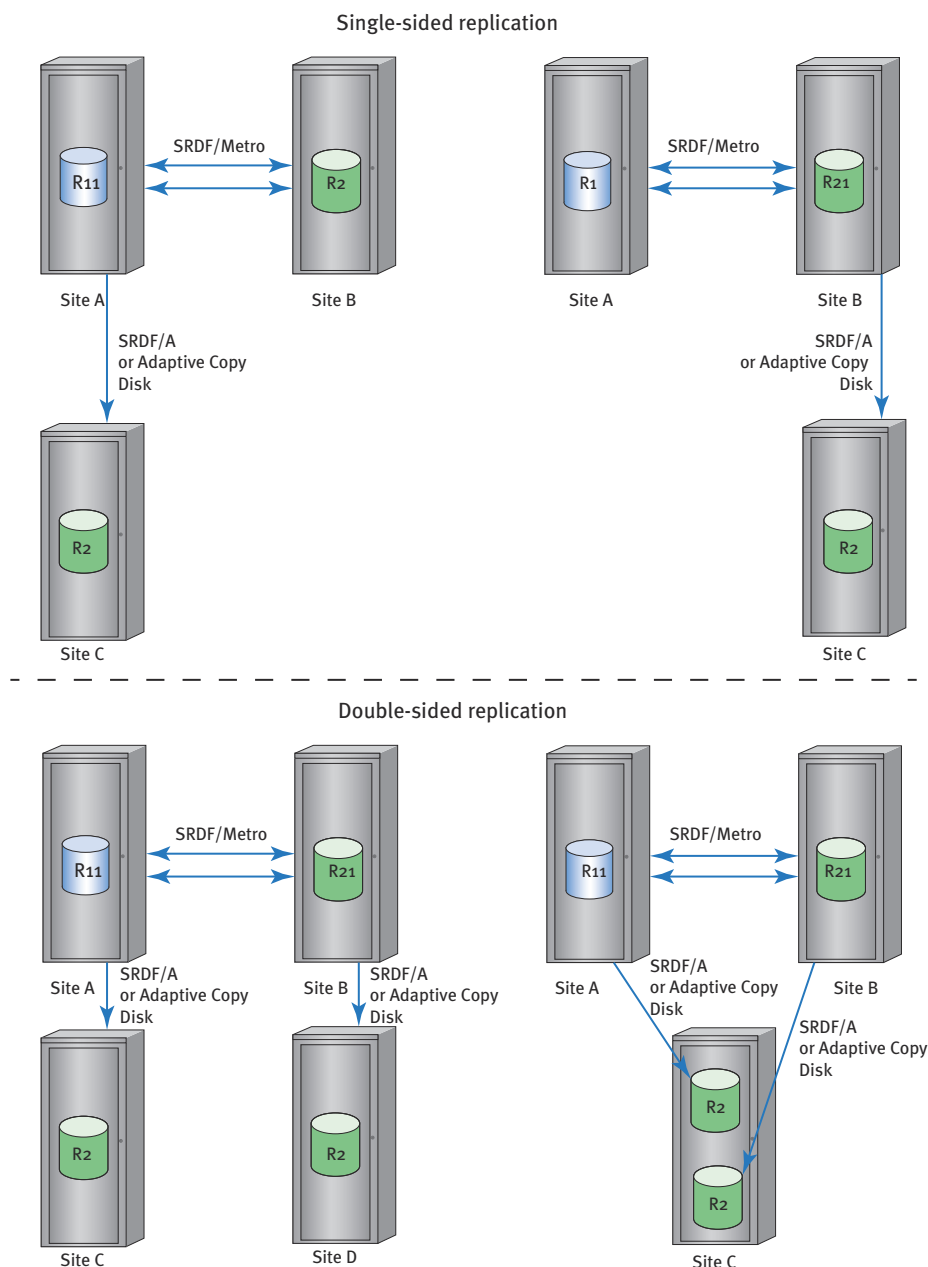


Figure 31. Disaster recovery for SRDF/Metro

The device names differ from a stand-alone SRDF/Metro configuration. This difference reflects the change in the devices' function when disaster recovery facilities are in place. For instance, when the R2 side is replicated to a disaster recovery site, its name changes to R21 because it is both the:

- R2 device in the SRDF/Metro configuration
- R1 device in the disaster-recovery configuration

When an SRDF/Metro uses a witness for resilience protection, the two sides periodically renegotiate the winning and losing sides. If the winning and losing sides switch as a result of renegotiation:

- An R11 device becomes an R21 device. That device was the R1 device for both the SRDF/Metro and disaster recovery configurations. Now the device is the R2 device of the SRDF/Metro configuration but it remains the R1 device of the disaster recovery configuration.
- An R21 device becomes an R11 device. That device was the R2 device in the SRDF/Metro configuration and the R1 device of the disaster recovery configuration. Now the device is the R1 device of both the SRDF/Metro and disaster recovery configurations.

Replication modes

As the diagram shows, the links to the disaster-recovery site use either SRDF/Asynchronous (SRDF/A) or Adaptive Copy Disk. In a double-sided configuration, each of the SRDF/Metro arrays can use either replication mode.

There are several criteria a witness takes into account when selecting the winner side. For example, a witness might take DR configuration into account.

Operating environment

In a HYPERMAX OS environment, both SRDF/Metro arrays must run HYPERMAX OS 5977.945.890 or later. The disaster-recovery arrays can run Enginuity 5876 and later or HYPERMAX OS 5977.691.684 and later.

In a PowerMaxOS environment, both SRDF/Metro arrays must run PowerMaxOS 5978.144.144 or later. The disaster recovery arrays can run PowerMaxOS 5978.144.144 and later, HYPERMAX OS 5977.952.892 and later, or Enginuity 5876.288.195 and later.

Deactivate SRDF/Metro

The administrator can delete or move a subset of device pairs from an SRDF/Metro group without changing the group's status as a SRDF/Metro configuration. To terminate a SRDF/Metro configuration, the administrator removes all the device pairs from the SRDF/Metro group.

NOTE:

- Only a delete operation can remove the final device pair or set of devices from an SRDF/Metro group.
- To delete devices in a configuration that runs HYPERMAX OS, the devices cannot be Ready on the SRDF link.
- To delete the final devices in a configuration that runs PowerMaxOS those devices must be Not Ready on the SRDF link.

When all the devices in the SRDF/Metro group have been deleted, the group is no longer part of an SRDF/Metro configuration. Also, the SRDF/Metro configuration has terminated.

SRDF/Metro restrictions

Some restrictions and dependencies apply to SRDF/Metro configurations:

- Online Device Expansion (ODE) is available when both sides run PowerMaxOS 5978.444.444 or later.
- The R1 and R2 must be identical in size.
- In an SRDF/Metro group, all the R1 devices must be on one side of the SRDF link and all the R2 devices on the other side.
- Devices can have Geometry Compatibility Mode (GCM) set only if the configuration consists of arrays that run PowerMaxOS 5978 or later.
- Devices can have user Geometry set only if GCM is also set.
- These operations apply to all devices in an SRDF/Metro group:
 - Setting devices Ready or Not Ready on the SRDF/Metro link.
For example, there is no provision to apply an establish or suspend operation to a subset of devices in an SRDF/Metro group.
 - Changing bias.
- An SRDF/Metro configuration contains FBA or IBM i D910² devices only. It cannot contain CKD (mainframe) devices.
- Mobility IDs are allowed in SRDF/Metro configurations only when both sides run PowerMaxOS. If either side runs HYPERMAX OS, Mobility ID is not available. Also, both sides must use native IDs or Mobility IDs. An SRDF/Metro group cannot contain devices that use a mix of native IDs and Mobility IDs.

Interaction restrictions

Some restrictions apply to SRDF device pairs in an SRDF/Metro configuration with TimeFinder and Open Replicator (ORS):

- Open Replicator is not supported.
- Devices cannot be BCVs.
- Devices cannot be used as the target of a TimeFinder data copy when the SRDF devices are both:
 - RW on the SRDF link
 - In the SyncInProg or ActiveActive SRDF, or Active Bias pair state

Devices can, however, be used as the source of a TimeFinder data copy.

² IBM i D910 requires PowerMaxOS 5978.444.444 or later.

Data migration

This chapter has more detail on the data migration facilities of SRDF.

Topics:

- [Introduction to data migration using SRDF](#)
- [Non-disruptive migration](#)
- [Migrating data with concurrent SRDF](#)
- [Migration-only SRDF](#)
- [Device Migration operations requirements](#)

Introduction to data migration using SRDF

Data migration is a one-time movement of data from one array (the source) to another array (the target). Typical examples are data center refreshes where data is moved from an old array after which that array is retired or re-purposed. Data migration is *not* data movement due to replication (where the source data is accessible after the target is created) or data mobility (where the target is continually updated).

After a data migration operation, applications that access the data reference it at the new location.

This chapter introduces the migration capabilities that are based on SRDF. These capabilities are available for open host (FBA) systems only. Mainframe systems have data migration capabilities but they are based on other technologies.

Non-disruptive migration

Non-disruptive migration (NDM) is a method for migrating data from one array to another without application downtime. The migration typically takes place within a data center.

This table shows the migration facilities that NDM provides:

Source	Target
VMAX running Enginuity 5876	PowerMax running PowerMaxOS 5978
VMAX running Enginuity 5876	VMAX All Flash or VMAX3 running HYPERMAX OS 5977
VMAX All Flash or VMAX3 running HYPERMAX OS 5977	PowerMax or VMAX All Flash running PowerMaxOS 5978
PowerMax running PowerMaxOS 5978	PowerMax running PowerMaxOS 5978

NDM does not affect the operation of the application host, enabling applications to continue to run while the migration takes place. Once the migration is complete, the application host switches to using the target array. A typical use of NDM is when a data center has a technology refresh and replaces an existing array.

The nondisruptive nature of NDM is heavily dependent on the pathing software that manages the connections between the host and the two arrays. So NDM is not available in environments that do not use PowerPath or another supported product. NDM Updates is the variant of NDM that is used in those environments. The major difference from NDM is that for some time during the migration the application that uses the migrated data has to be stopped. Besides that, the migration process and the valid configurations are the same as NDM.

Migration from VMAX array

Migrating from a VMAX array uses SRDF in Pass-through mode. In this mode, the application host can access data on both source and target devices while the migration is in progress. PowerMaxOS or HYPERMAX OS on the target ensures that the source processes all I/O operations sent to the target.

Process

The steps in the migration process are:

1. Set up the environment – configure the infrastructure of the source and target array, in preparation for data migration.
2. On the source array, select a storage group to migrate.
3. If using NDM Updates, shut down the application associated with the storage group.
4. Create the migration session – copy the content of the storage group to the target array using SRDF.

When creating the session, optionally specify whether to move the identity of the LUNs in the storage group to the target array.

5. When the data copy is complete:
 - a. If the migration session did not move the identity of the LUNs, reconfigure the application to access the new LUNs on the target array.
 - b. Cutover the storage group to the PowerMax, VMAX All Flash, or VMAX3 array.
 - c. Commit the migration session – remove resources from the source array and those used in the migration itself. The application now uses the target array only.

6. If using NDM Updates, restart the application.
7. To migrate further storage groups, repeat steps 2 on page 77 to 6 on page 78.
8. After migrating all the required storage groups, remove the migration environment.

Other features

Other features of migrating from VMAX to PowerMax, VMAX All Flash, or VMAX3 are:

- Data can be compressed during migration to the PowerMax, VMAX All Flash, or VMAX3 array
- Allows for nondisruptive revert to the source array
- There can be up to 50 migration sessions in progress simultaneously
- NDM does not require an additional license as it is part of PowerMaxOS or HYPERMAX OS
- The connections between the application host and the arrays use FC; the SRDF connection between the arrays uses FC or GigE

Devices and components that cannot be part of an NDM process are:

- CKD devices
- eNAS data
- Storage Direct and FAST.X relationships along with their associated data

Migration from VMAX All Flash or VMAX3

Migrating from a VMAX All Flash or VMAX3 array running HYPERMAX OS 5977 to an array running PowerMaxOS 5978 uses a modified form of SRDF/Metro. So both the source and target arrays are visible and accessible to the application host while the migration takes place.

Process

Normal flow

The steps in the migration process that is normally followed are:

1. Set up the migration environment – configure the infrastructure of the source and target array, in preparation for data migration.
2. On the source array, select a storage group to migrate.
3. If using NDM Updates, shut down the application associated with the storage group.
4. Create the migration session optionally specifying whether to move the identity of the LUNs in the storage group to the target array – copy the content of the storage group to the target array using SRDF/Metro.

During this time the source and target arrays are both accessible to the application host.

5. When the data copy is complete:
 - a. If the migration session did not move the identity of the LUNs, reconfigure the application to access the new LUNs on the target array.
 - b. Commit the migration session – remove resources from the source array and those used in the migration itself.
6. If using NDM Updates, restart the application.
7. To migrate further storage groups, repeat steps 2 on page 78 to 6 on page 78.
8. After migrating all the required storage groups, remove the migration environment.

Alternate flow

There is an alternative process that pre-copies the data to the target array before making it available to the application host. The steps in this process are:

1. Set up the migration environment – configure the infrastructure of the source and target array, in preparation for data migration.
2. On the source array, select a storage group to migrate.
3. Use the precopy facility of NDM to copy the selected data to the target array.

Optionally, specify whether to move the identity off the LUNs in the storage group to the target array.

While the data copy takes place, the source array is available to the application host, but the target array is unavailable.

4. When the copying of the data is complete: use the Ready Target facility in NDM to make the target array available to the application host also.
 - a. If the migration session did not move the identity of the LUNs, reconfigure the application to access the new LUNs on the target array.
 - b. If using NDM Updates, restart the application.
 - c. Commit the migration session – remove resources from the source array and those used in the migration itself. The application now uses the target array only.
5. To migrate further storage groups, repeat steps 2 on page 79 to 4 on page 79.
6. After migrating all the required storage groups, remove the migration environment.

Other functions

Other NDM facilities that are available for exceptional circumstances are:

- Cancel – to cancel a migration that has not yet been committed.
- Sync – to stop or start the synchronization of writes to the target array back to source array. When stopped, the application runs on the target array only. Used for testing.
- Recover – to recover a migration process following an error.

Other features

Other features of migrating from VMAX3, VMAX All Flash, or PowerMax to PowerMax are:

- Data can be compressed during migration to the PowerMax array
- Allows for nondisruptive revert to the source array
- There can be up to 50 migration sessions in progress simultaneously
- Does not require an additional license as NDM is part of PowerMaxOS
- The connections between the application host and the arrays use FC; the SRDF connection between the arrays uses FC or GigE

Devices and components that cannot be part of an NDM process are:

- CKD devices
- eNAS data
- Storage Direct and FAST.X relationships along with their associated data

Migrating data with concurrent SRDF

In concurrent SRDF topologies, you can non-disruptively migrate data between arrays along one SRDF leg while remote mirroring for protection along the other leg.

Once the migration process completes, the concurrent SRDF topology is removed, resulting in a 2-site SRDF topology.

Replacing R1 devices with new R1 devices

This diagram shows the use of migration to replace the R1 devices in a two-site configuration. The diagram shows the:

- Initial two-site topology
- Interim three-site migration topology
- Final two-site topology

After migration, the new primary array is mirrored to the original secondary array.

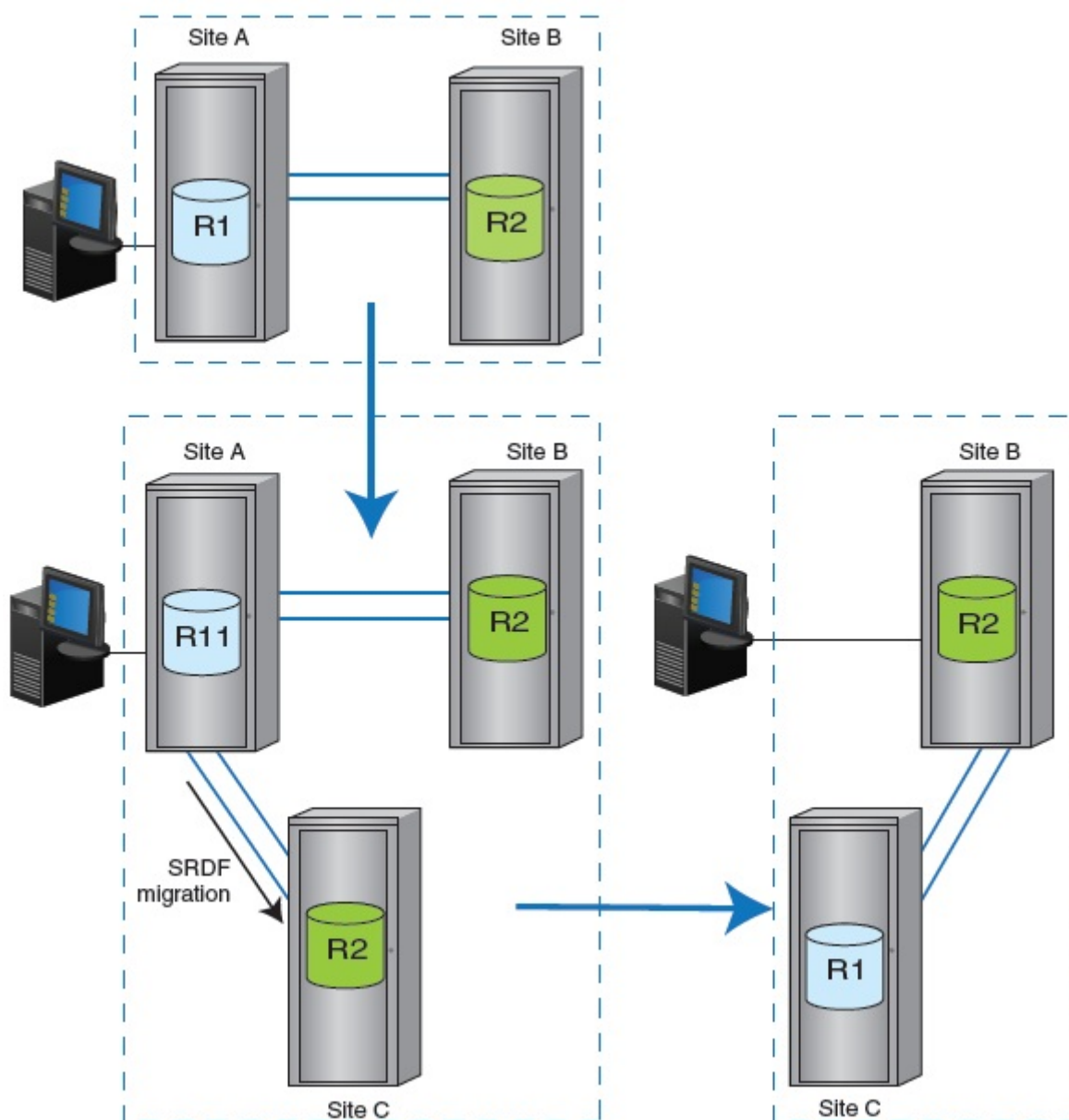


Figure 32. Migrating data and replacing the original primary array (R1)

Replacing R2 devices with new R2 devices

This diagram shows the use of migration to replace the R2 devices in a two-site configuration. The diagram shows the:

- Initial two-site topology
- Interim three-site migration topology
- Final two-site topology

After migration, the original primary array is mirrored to a new secondary array.

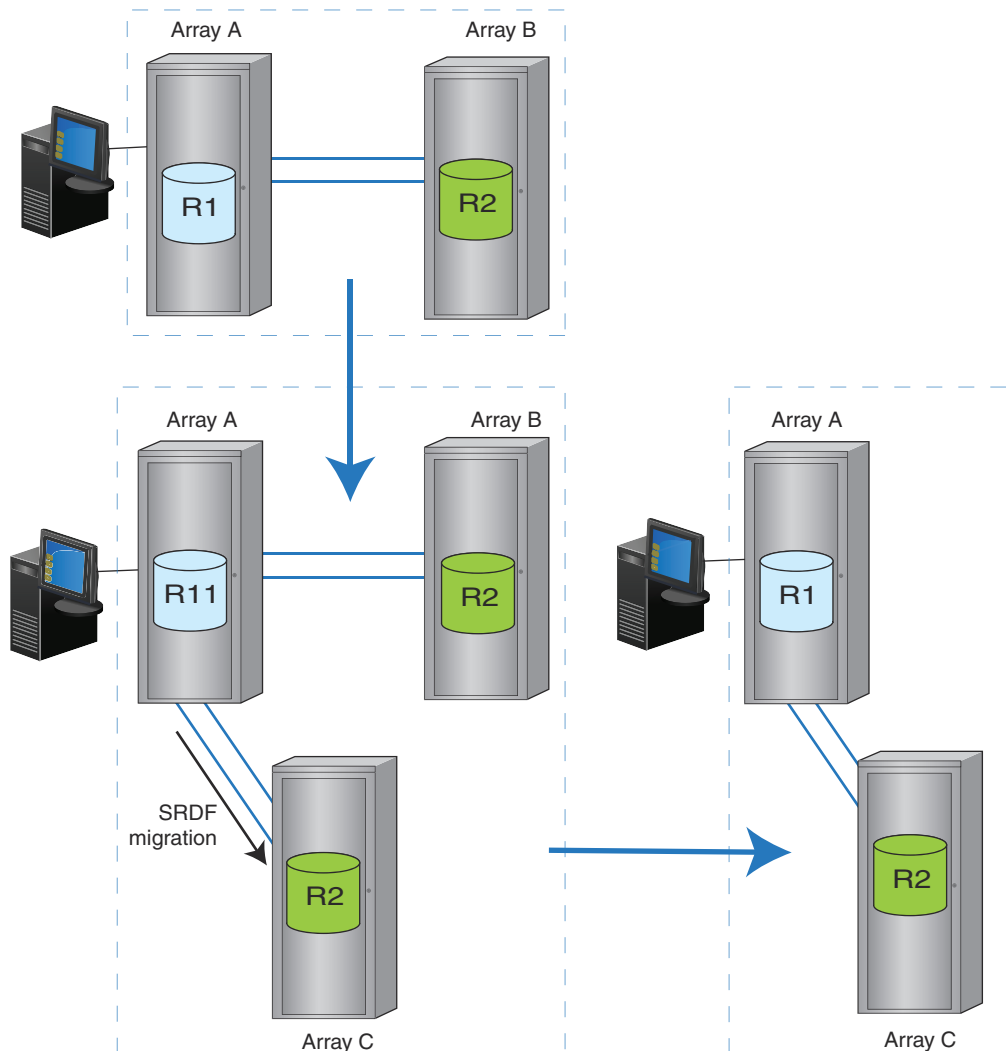


Figure 33. Migrating data and removing the original secondary array (R2)

Replacing R1 and R2 devices with new R1 and R2 devices

This diagram shows the use of migration to replace both the R1 and R2 arrays in a two-site configuration. The diagram shows the:

- Initial two-site topology
- Migration topology using a combination of concurrent and cascaded SRDF
- Final two-site topology

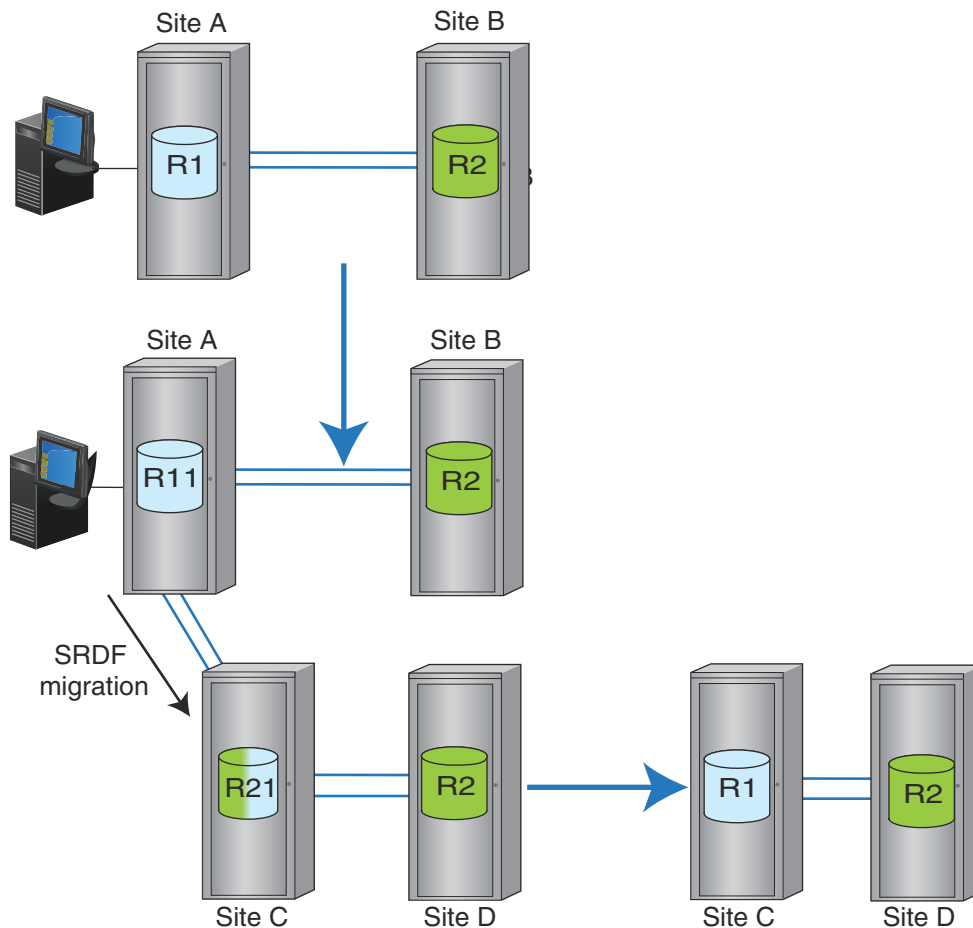


Figure 34. Migrating data and replacing the original primary (R1) and secondary (R2) arrays

Migration-only SRDF

In some cases, you can migrate data with full SRDF functionality, including disaster recovery and other advanced SRDF features.

In cases where full SRDF functionality is not available, you can move the data using migration-only SRDF.

This table lists SRDF common operations and features and whether they are available in SRDF groups during SRDF migration-only environments.

Table 8. Limitations of the migration-only mode

SRDF operations or features	Whether supported during migration
R2 to R1 copy	Only for device rebuild from un-rebuildable RAID group failures.
Failover, failback, domino	Not available
SRDF/Star	Not available
SRDF/A features (DSE, Consistency Group, ECA, MSC)	Not available
Dynamic SRDF operations (create, delete, and move SRDF pairs; R1/R2 personality swap)	Not available
TimeFinder operations	Only on R1
Online configuration change or upgrade	<ul style="list-style-type: none">• If the changes affect the group or devices being migrated, migration must be suspended before and while the upgrade or configuration changes take place.• If the changes do not affect the group or devices being migrated, migration can continue while the upgrade or configuration changes take place.
Out-of-family Non-Disruptive Upgrade (NDU)	Not available

Device Migration operations requirements

- Each array must have a unique ID (sid).
- The existing SRDF device and the new devices must be dynamic R1 or R2 capable.

PowerMaxOS and HYPERMAX OS

- Devices that are part of an SRDF/Metro configuration cannot be migrated.
- Adaptive copy write pending mode is not supported when the R1 side of the RDF pair is on an array running PowerMaxOS or HYPERMAX OS.

For configurations where the R1 side is on an array running PowerMaxOS or HYPERMAX OS, and the R2 side is running Enginuity 5876, the mode of the new device pair is set to the RDF mode of the R1 device being replaced.

- The Geometry Compatibility Mode (GCM) attribute allows devices on arrays running PowerMaxOS or HYPERMAX OS to be paired with devices on arrays running Enginuity 5876 that have an odd number of cylinders. When GCM is set, migration operations are subject to the following restrictions:
 - If the new device is on an array running PowerMaxOS or HYPERMAX OS:
 - If the R1 device is being replaced:

If the existing R2 device is on an array running Enginuity 5876 with an odd number of cylinders, the migration is allowed if the new device can be made the same size using the GCM attribute.

If the existing R2 device is on an array running PowerMaxOS or HYPERMAX OS with GCM set, the migration is allowed if the new device can be made the same size by setting the GCM attribute.
 - If the R2 is being replaced:

If the existing R1 device is on an array running Enginuity 5876 with an odd number of cylinders, then the migration is allowed if the new device can be made the same size by setting the GCM attribute.

If the existing R1 device is on an array running PowerMaxOS or HYPERMAX OS with GCM set, the migration is allowed if the new device can be made the same size by setting the GCM attribute.
 - If the new device is on an array running Enginuity 5876 and has an odd number of cylinders:
 - If the R1 is being replaced:

If the existing R2 device is on an array running Enginuity 5876, the new device must be the same configured size.

If the existing R2 device is on an array running PowerMaxOS or HYPERMAX OS with GCM set, the migration is allowed if the new device has the same GCM size as the R2 device.
 - If the R2 is being replaced:

If the existing R1 device is on an array running Enginuity 5876, the new device must be the same configured size.

If the existing R1 device is on an array running PowerMaxOS or HYPERMAX OS with GCM set, the migration is allowed if the new device has the same GCM size as the R1.

SRDF I/O operations

This chapter shows how SRDF handles write and read operations. In addition, there is information on the performance and resilience features of SRDF/A.

Topics:

- [SRDF write operations](#)
- [SRDF read operations](#)
- [SRDF/A resilience and performance features](#)

SRDF write operations

This section describes SRDF write operations:

- Write operations in synchronous mode
- Write operations in asynchronous mode
- Cycle switching in asynchronous mode
- Write operations in cascaded SRDF

Write operations in synchronous mode

In synchronous mode, data must be successfully written to cache at the secondary site before a positive command completion status is returned to the application host that issued the write command.

The following diagram shows the steps in a synchronous write operation:

1. The application host sends a write command to the local array.
The host emulations write data to cache and create a write request.
2. SRDF emulations frame updated data in cache according to the SRDF protocol, and transmit it across the SRDF links.
3. The SRDF emulations in the remote array receive data from the SRDF links, write it to cache and return an acknowledgment to SRDF emulations in the local array.
4. The SRDF emulations in the local array forward the acknowledgment to host emulations which pass it on to the application host.

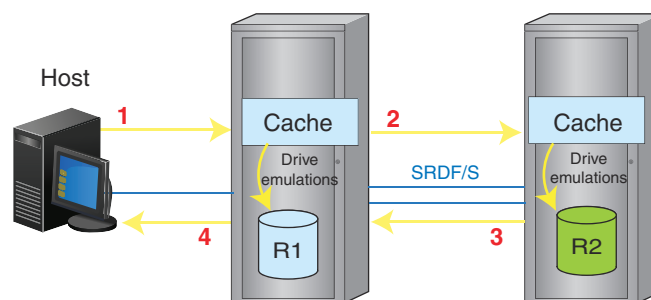


Figure 35. Write I/O flow: simple synchronous SRDF

Write operations in asynchronous mode

When SRDF/A is in use, the primary array collects host write operations into delta sets and transfers them in cycles to the secondary array. The primary array acknowledges the host write operations as soon as they are written to its cache.

SRDF/A sessions behave differently depending on:

- Whether they are managed individually (Single Session Consistency (SSC)) or as a consistency group (Multi Session Consistency (MSC)).
 - With SSC, the SRDF group is managed individually. The primary array's operating environment controls cycle switching. SRDF/A cycles are switched independently of any other SRDF groups on any array in the solution. [Cycle switching in asynchronous mode](#) on page 87 has more details.
 - With MSC, the SRDF group is part of a consistency group spanning all associated SRDF/A sessions. SRDF host software coordinates cycle switching to provide dependent-write consistency across multiple sessions, which may also span arrays. The host software switches SRDF/A cycles for all SRDF groups in the consistency group simultaneously. [SRDF/A MSC cycle switching](#) on page 89 has more details.
- The number of transmit cycles supported at the R1 side. Engenuity 5876 supports only a single cycle. PowerMaxOS and HYPERMAX OS support multiple cycles queued to be transferred.

Data in a delta set is processed using four cycle types:

- **Capture cycle**—Incoming I/O is buffered in the capture cycle on the R1 side. The host receives immediate acknowledgment.
- **Transmit cycle**—Data collected during the capture cycle is moved to the transmit cycle on the R1 side.
- **Receive cycle**—Data is received on the R2 side.

- **Apply cycle**—Changed blocks in the delta set are marked as invalid tracks and destaging to disk begins.
A new receive cycle is started.

The operating environment running on the R1 side determines when the next capture cycle can begin. It also determines the number of cycles that can be in progress simultaneously.

PowerMaxOS 5978 or HYPERMAX OS 5977 – Multicycle mode

If both arrays in the configuration run PowerMaxOS or HYPERMAX OS, SRDF/A operates in multicycle mode. There can be two or more cycles on the R1 side, but only two cycles on the R2 side:

- On the R1 side:
 - One Capture
 - One or more Transmit
- On the R2 side:
 - One Receive
 - One Apply

Cycle switches are decoupled from committing delta sets to the next cycle.

When the preset Minimum Cycle Time is reached, the R1 data collected during the capture cycle is added to the transmit queue. Then a new R1 capture cycle begins. There is no wait for the commit on the R2 side before starting a new capture cycle.

The transmit queue holds cycles waiting to be transmitted to the R2 side. Data in the transmit queue is committed to the R2 receive cycle when the current transmit cycle and apply cycle are empty.

Queueing enables smaller cycles of data to be buffered on the R1 side and reduces the size of delta sets transferred to the R2 side.

The SRDF/A session can adjust to accommodate changes in the solution. If the SRDF link speed decreases or the apply rate on the R2 side increases, more SRDF/A cycles can be queued on the R1 side.

Multicycle mode increases the robustness of the SRDF/A session and reduces spillover into the DSE storage pool.

Enginuity 5876

If either array in the solution is running Enginuity 5876, SRDF/A operates in legacy mode. There are two cycles on the R1 side, and two cycles on the R2 side:

- On the R1 side:
 - One Capture
 - One Transmit
- On the R2 side:
 - One Receive
 - One Apply

Each cycle switch moves the delta set to the next cycle in the process.

A new capture cycle cannot start until both the transmit cycle on the R1 side and the apply cycle on the R2 side are complete.

Cycle switching can occur within the preset Minimum Cycle Time. However, it can also take longer since it depends on both:

- The time taken to transfer the data from the R1 transmit cycle to the R2 receive cycle
- The time taken to destage the R2 apply cycle

Cycle switching in asynchronous mode

The number of capture cycles supported at the R1 side varies depending on whether one or both the arrays in the solution are running PowerMaxOS or HYPERMAX OS.

PowerMaxOS or HYPERMAX OS

SRDF/A SSC sessions where both arrays are running PowerMaxOS or HYPERMAX OS have one or more Transmit cycles on the R1 side (multicycle mode).

The following diagram shows multicycle mode:

- Multiple cycles (one capture cycle and multiple transmit cycles) on the R1 side, and
- Two cycles (receive and apply) on the R2 side.

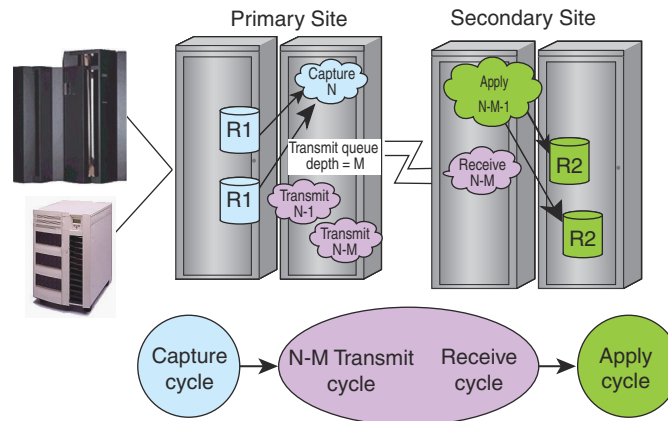


Figure 36. SRDF/A SSC cycle switching – multi-cycle mode

In multicycle mode, each cycle switch creates a new capture cycle (N) and the existing capture cycle (N-1) is added to the queue of cycles (N-1 through N-M cycles) to be transmitted to the R2 side by a separate commit action.

Only the data in the last transmit cycle (N-M) is transferred to the R2 side during a single commit.

Enginuity 5876

SRDF/A SSC sessions that include an array running Enginuity 5876 have one Capture cycle and one Transmit cycle on the R1 side (legacy mode).

The following diagram shows legacy mode:

- 2 cycles (capture and transmit) on the R1 side, and
- 2 cycles (receive and apply) on the R2 side

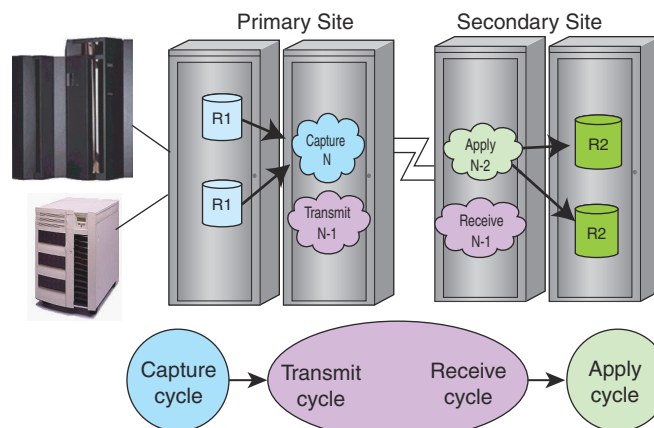


Figure 37. SRDF/A SSC cycle switching – legacy mode

In legacy mode, there are conditions must be met before an SSC cycle switch can take place:

- The previous cycle's transmit delta set (N-1 copy of the data) must have completed transfer to the receive delta set on the secondary array.
- On the secondary array, the previous apply delta set (N-2 copy of the data) is written to cache, and data is marked write pending for the R2 devices.

SSC cycle switching in concurrent SRDF/A

In single session mode, cycle switching on both legs of the concurrent SRDF topology typically occurs at different times. Data in the Capture and Transmit cycles may differ between the two SRDF/A sessions.

SRDF/A MSC cycle switching

SRDF/A MSC:

- Coordinates the cycle switching for all SRDF/A sessions in the SRDF/A MSC solution.
- Monitors for any failure to propagate data to the secondary array devices and drops all SRDF/A sessions together to maintain dependent-write consistency.
- Performs MSC cleanup operations (if possible).

PowerMaxOS 5978 or HYPERMAX OS 5977

SRDF/A MSC sessions, where both arrays are running PowerMaxOS or HYPERMAX OS, have two or more cycles on the R1 side (multicycle mode).

NOTE: If either the R1 side or R2 side of an SRDF/A session is running PowerMaxOS or HYPERMAX OS, Solutions Enabler 8.x or later (for HYPERMAX OS) or Solutions Enabler 9.0 or later (for PowerMaxOS) is required to monitor and manage MSC groups.

The following diagram shows the cycles on the R1 side (one capture cycle and multiple transmit cycles) and 2 cycles on the R2 side (receive and apply) for an SRDF/A MSC session when both of the arrays in the SRDF/A solution are running PowerMaxOS or HYPERMAX OS.

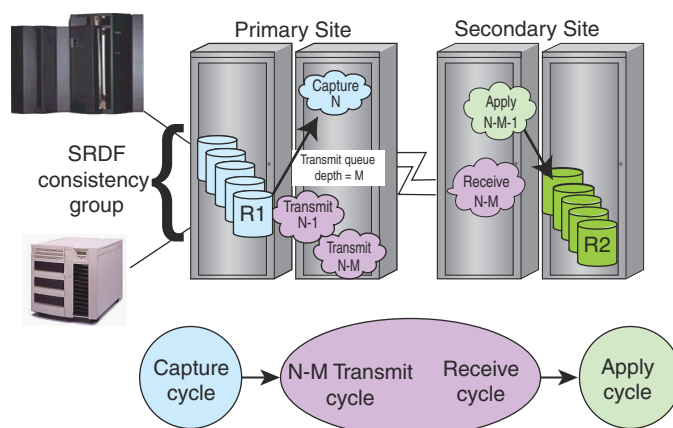


Figure 38. SRDF/A MSC cycle switching – multicycle mode

SRDF cycle switches all SRDF/A sessions in the MSC group at the same time. All sessions in the MSC group have the same:

- Number of cycles outstanding on the R1 side
- Transmit queue depth (M)

In SRDF/A MSC sessions, the array's operating environment performs a coordinated cycle switch during a window of time when no host writes are being completed.

So that it can establish consistency, MSC temporarily suspends write operations from the application host across all SRDF/A sessions. MSC resumes those write operations once there is consistency. There is a timeout of 12 seconds associated with the suspension of write operations to protect against failure of MSC. Should the timeout expire, write operations from the application host resume.

Enginuity 5876

SRDF/A MSC sessions that include an array running Enginuity 5876 have only two cycles on the R1 side (legacy mode).

In legacy mode, the following conditions must be met before an MSC cycle switch can take place:

- The primary array's transmit delta set must be empty.
- The secondary array's apply delta set must have completed. The N-2 data must be marked write pending for the R2 devices.

To achieve consistency through cycle switching, MSC suspends write operations from the application host in the same way as it does when both arrays run PowerMaxOS 5978 or HYPERMAX OS 5977. It also uses the 12-second timeout to protect against the failure of MSC while synchronization is in progress.

Write operations in cascaded SRDF

In cascaded configurations, R21 devices operate as:

- R2 devices to devices in the R1 array
- R1 devices to devices in the R2 array

I/O to R21 devices includes:

- Synchronous I/O between the production site (R1) and the closest (R21) remote site.
- Asynchronous or adaptive copy I/O between the synchronous remote site (R21) and the tertiary (R2) site.
- The storage administrator can Write Enable the R21 to a host so that the R21 behaves like an R2 device. This allows the R21 -> R2 connection to operate as R1 -> R2, while the R1 -> R21 connection is automatically suspended. The R21 begins tracking changes against the R1.

This diagram shows the synchronous I/O flow in a cascaded SRDF topology.

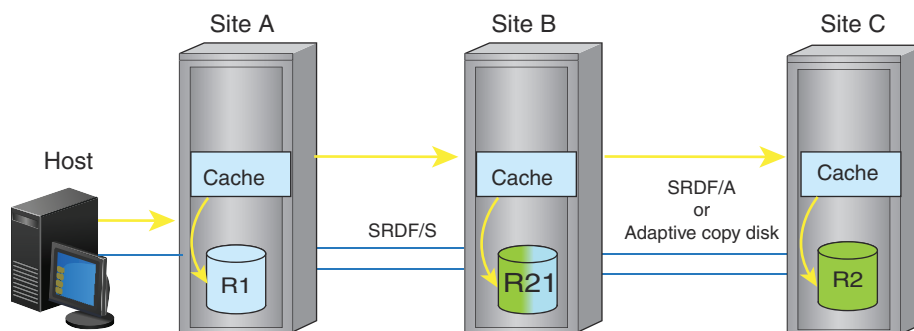


Figure 39. Write commands to R21 devices

When a write command arrives in cache at Site B:

- The SRDF emulation at Site B sends a positive status back across the SRDF links to Site A (synchronous operations), and
- Creates a request for SRDF emulations at Site B to send data across the SRDF links to Site C.

SRDF read operations

Read operations from the R1 device do not usually involve the SRDF emulations:

- For read “hits” (the production host issues a read to the R1 device, and the data is in local cache), the host emulation reads data from cache and sends it to the host.
- For read “misses” (the requested data is not in cache), the drive emulation reads the requested data from local drives to cache.

Read operations if R1 local copy fails

In SRDF/S, SRDF/A, and adaptive copy configurations, SRDF devices can process read I/Os that cannot be processed by regular logical devices. If the R1 local copy fails, the R1 device can still service the request as long as its SRDF state is Ready and the R2 device has good data.

SRDF emulations help service the host read requests when the R1 local copy is not available:

- The SRDF emulations bring data from the R2 device to the host site.
- The host perceives this as an ordinary read from the R1 device, although the data was read from the R2 device acting as if it was a local copy.

PowerMaxOS or HYPERMAX OS

Arrays running PowerMaxOS or HYPERMAX OS cannot service SRDF/A read I/Os with Delta Set Extension (DSE). So, spillover is not invoked during a SRDF/A restore operation until that restore operation is complete. [SRDF/A cache data offloading](#) on page 92 contains more information about DSE.

Read operations from R2 devices

Reading data from R2 devices directly from a host connected to the R2 is not recommended, because:

- SRDF/S relies on the application’s ability to determine if the data image is the most current. The array at the R2 side may not yet know that data currently in transmission on the SRDF links has been sent.
- If the remote host reads data from the R2 device while a write I/O is in transit on the SRDF links, the host is not reading the most current data.

Dell Technologies strongly recommends that you allow the remote host to read data from the R2 devices while in Read Only mode only when:

- Related applications on the production host are stopped.
- The SRDF writes to the R2 devices are blocked due to a temporary suspension/split of the SRDF relationship.

In a mainframe environment, however, PPRC does allow secondary devices to be defined as read only. Dell Technologies supports this with extra controls in SCF, AutoSwap, and ConGroup software. Such a configuration is known as Host Read Only (HRD).

SRDF/A resilience and performance features

Operational problems that can occur in a SRDF/A configuration include:

- Unbalanced SRDF/A configurations or I/O spikes can cause SRDF/A solutions to use large amounts of cache.
- Transient network outages can interrupt SRDF sessions.
- An application may write to the same record repeatedly.

This section describes the SRDF/A features that address these problems.

Tunable cache

The storage administrator can set the SRDF/A maximum cache utilization threshold to a percentage of the system write pending limit for an individual SRDF/A session in single session mode and multiple SRDF/A sessions in single or MSC mode.

When the SRDF/A maximum cache utilization threshold or the system write pending limit is exceeded, the array exhausts its cache.

By default, the SRDF/A session drops if array cache is exhausted. However, the SRDF/A session can continue to run for a user-defined period. The storage administrator can assign priorities to sessions, keeping SRDF/A active for as long as cache resources allow. If the condition is not resolved at the expiration of the user-defined period, the SRDF/A session drops.

The features described below help to prevent SRDF/A from exceeding its maximum cache utilization threshold.

SRDF/A cache data offloading

If the system approaches the maximum SRDF/A cache utilization threshold, Delta Set Extension (DSE) offloads some or all of the delta set data. DSE can be configured/enabled/disabled independently on the R1 and R2 sides. However, Dell Technologies recommends that both sides use the same configuration of DSE.

DSE works in tandem with group-level write pacing to prevent cache over-utilization during spikes in I/O or network slowdowns.

Resources to support offloading vary depending on the operating environment running on the array.

PowerMaxOS or HYPERMAX OS

PowerMaxOS and HYPERMAX OS offload data into a Storage Resource Pool. One or more Storage Resource Pools are pre-configured before installation and used by a variety of functions. DSE can use a Storage Resource Pool pre-configured specifically for DSE. If no such pool exists, DSE can use the default Storage Resource Pool. All SRDF groups on the array use the same Storage Resource Pool for DSE. DSE requests allocations from the Storage Resource Pool only when DSE is activated.

The Storage Resource Pool used by DSE is sized based on your SRDF/A cache requirements. DSE is automatically enabled.

Enginuity 5876

Enginuity 5876 offloads data to a user-configured DSE pool. There must be a separate DSE pool for each device emulation type (FBA, IBM i, CKD3380 or CKD3390).

- In order to use DSE, each SRDF group must be explicitly associated with a DSE pool.
- By default, DSE is disabled.
- When TimeFinder/Snap sessions are used to replicate either R1 or R2 devices, there must be two separate preconfigured storage pools: DSE and Snap pools.

Mixed configurations: PowerMaxOS or HYPERMAX OS and Enginuity 5876

If the array on one side of an SRDF device pair is running PowerMaxOS or HYPERMAX OS and the other side is running a Enginuity 5876, the SRDF/A session runs in Legacy mode.

- DSE is disabled by default on both arrays.
- Dell Technologies recommends that you enable DSE on both sides.

Transmit Idle

During short-term network interruptions, the transmit idle state indicates that SRDF/A is still tracking changes but is unable to transmit data to the remote side.

Write folding

Write folding improves the efficiency of your SRDF links.

When multiple updates to the same location arrive in the same delta set, the SRDF emulations send the only most current data across the SRDF links.

Write folding decreases network bandwidth consumption and the number of I/Os processed by the SRDF emulations.

Write pacing

SRDF/A write pacing reduces the likelihood that an active SRDF/A session drops due to cache exhaustion. Write pacing dynamically paces the host I/O rate so it does not exceed the SRDF/A session's service rate. This prevents cache overflow on both the R1 and R2 sides.

Use write pacing to maintain SRDF/A replication with reduced resources when replication is more important for the application than minimizing write response time.

You can apply write pacing to groups, or devices for individual RDF device pairs that have TimeFinder/Snap or TimeFinder/Clone sessions off the R2 device.

Group pacing

SRDF/A group pacing adjusts the pace of host writes to match the SRDF/A session's link transfer rate. When host I/O rates spike, or slowdowns make transmit or apply cycle times longer, group pacing extends the host write I/O response time to match slower SRDF/A service rates.

When DSE is activated for an SRDF/A session, host-issued write I/Os are paced so their rate does not exceed the rate at which DSE can offload the SRDF/A session's cycle data to the DSE Storage Resource Pool.

Group pacing behavior varies depending on whether the maximum pacing delay is specified:


- If the maximum write pacing delay is not specified, SRDF adds up to 50 ms to the host write I/O response time to match the speed of either the SRDF links or the apply operation on the R2 side, whichever is slower.
- If the maximum write pacing delay is specified, SRDF adds up to the user-specified maximum write pacing delay to keep the SRDF/A session running.

Group pacing balances the incoming host I/O rates with the SRDF link bandwidth and throughput capabilities when:

- The host I/O rate exceeds the SRDF link throughput.
- Some SRDF links that belong to the SRDF/A group are lost.
- Reduced throughput on the SRDF links.
- **Enginuity 5876 only:** The write-pending level on an R2 device in an active SRDF/A session reaches the device write-pending limit.
- **Enginuity 5876 only:** The apply cycle time on the R2 side is longer than 30 s and the R1 capture cycle time (or in MSC, the capture cycle target).

Group pacing can be activated by configurations or activities that result in slow R2 operations, such as:

- Slow R2 physical drives resulting in longer apply cycle times.
- Director sparing operations that slow restore operations.
- I/O to the R2 array that slows restore operations.

 **NOTE:** On arrays running Enginuity 5876, if the space in the DSE pool runs low, DSE drops and group SRDF/A write pacing falls back to pacing host writes to match the SRDF/A session's link transfer rate.

Device (TimeFinder) pacing

PowerMaxOS or HYPERMAX OS

SRDF/A device write pacing is not supported or required for asynchronous R2 devices in TimeFinder or TimeFinder SnapVX sessions when either array in the configuration is running PowerMaxOS or HYPERMAX OS, including:

- R1 PowerMaxOS or HYPERMAX OS - R2 PowerMaxOS or HYPERMAX OS
- R1 PowerMaxOS or HYPERMAX OS - R2 Enginuity 5876
- R1 Enginuity 5876 - R2 PowerMaxOS or HYPERMAX OS

Enginuity 5876

SRDF/A device pacing applies a write pacing delay for individual SRDF/A R1 devices whose R2 counterparts participate in TimeFinder copy sessions.

SRDF/A group pacing avoids high SRDF/A cache utilization levels when the R2 devices servicing both the SRDF/A and TimeFinder copy requests experience slowdowns.

Device pacing avoids high SRDF/A cache utilization when the R2 devices servicing both the SRDF/A and TimeFinder copy requests experience slowdowns.

Device pacing behavior varies depending on whether the maximum pacing delay is specified:

- If the maximum write pacing delay is not specified, SRDF adds up to 50 milliseconds to the overall host write response time to keep the SRDF/A session active.
- If the maximum write pacing delay is specified, SRDF adds up to the user-defined maximum write pacing delay to keep the SRDF/A session active.

Device pacing can be activated on the second hop (R21 -> R2) of a cascaded SRDF and cascaded SRDF/Star, topologies.

Device pacing may not take effect if all SRDF/A links are lost.

Write pacing and Transmit Idle

Host writes continue to be paced when:

- All SRDF links are lost, and
- Cache conditions require write pacing, and
- Transmit Idle is in effect.

Pacing during the outage is the same as the transfer rate prior to the outage.

Interfamily compatibility

This chapter has more detail on the compatibility between different families of Dell EMC storage arrays.

Topics:

- [Overview](#)
- [SRDF supported features](#)

Overview

SRDF can operate between different operating environments and arrays. Arrays running PowerMaxOS or HYPERMAX OS can connect to arrays running older operating environments. In mixed configurations where arrays are running different versions, SRDF features of the lowest version are supported.

PowerMax, VMAX All Flash, and VMAX3 arrays can connect to:

- PowerMax arrays running PowerMaxOS
- VMAX 250F, 450F, 850F, and 950F arrays running HYPERMAX OS
- VMAX 100K, 200K, and 400K arrays running HYPERMAX OS
- VMAX 10K, 20K, and 40K arrays running Enginuity 5876 with an Enginuity ePack

Interfamily connectivity allows you to add the latest hardware platform/operating environment to an existing SRDF solution, enabling technology refreshes.

NOTE: When you connect between arrays running different operating environments, limitations may apply. Information about which SRDF features are supported, and applicable limitations for 2-site and 3-site solutions is available in the *SRDF Interfamily Connectivity Information*.

SRDF supported features

The SRDF features supported on each hardware platform and operating environment are:

Table 9. SRDF features by hardware platform/operating environment

Feature	Enginuity 5876		HYPERMAX OS 5977	PowerMaxOS
	VMAX 40K, VMAX 20K	VMAX 10K	VMAX3, VMAX 250F, 450F, 850F, 950F	PowerMax 2000, PowerMax 8000
Max. SRDF devices/SRDF emulation (either Fibre Channel or GigE)	64K	8K	64K	64K
Max. SRDF groups/array	250	32	250	250
Max. SRDF groups/SRDF emulation instance (either Fibre Channel or GigE)	64	32	250 ^{ab}	250 ^{cd}
Max. remote targets/port	64	64	16K/SRDF emulation (either Fibre Channel or GigE)	16K/SRDF emulation (either Fibre Channel or GigE)
Max. remote targets/SRDF group	N/A	N/A	512	512
Fibre Channel port speed	2/4/8 Gb/s 16 Gb/s on 40K	2/4/8/16 Gb/s	16 Gb/s	16 Gb/s
GbE port speed	1 /10 Gb/s	1 /10 Gb/s	1 /10 Gb/s	1 /10 Gb/s
Min. SRDF/A Cycle Time	1 sec, 3 secs with MSC	1 sec, 3 secs with MSC	1 sec, 3 secs with MSC	1 sec, 3 secs with MSC
SRDF Delta Set Extension	Supported	Supported	Supported	Supported
Transmit Idle	Enabled	Enabled	Enabled	Enabled
Fibre Channel Single Round Trip (SiRT)	Enabled	Enabled	Enabled	Enabled
GigE SRDF Compression				
Software	Supported VMAX 20K VMAX 40K: Enginuity 5876.82.57 or higher	Supported	Supported	Supported

Table 9. SRDF features by hardware platform/operating environment (continued)

Feature	Enginuity 5876		HYPERMAX OS 5977	PowerMaxOS
	VMAX 40K, VMAX 20K	VMAX 10K	VMAX3, VMAX 250F, 450F, 850F, 950F	PowerMax 2000, PowerMax 8000
Hardware	Supported VMAX 20K VMAX 40K: Enginuity 5876.82.57 or higher	N/A	Supported	Supported
Fibre Channel SRDF Compression				
Software	Supported VMAX 20K VMAX 40K: Enginuity 5876.82.57 or higher	Supported	Supported	Supported
Hardware	Supported VMAX 20K: N/A VMAX 40K: Enginuity 5876.82.57 or higher	N/A	Supported	Supported
IPv6 and IPsec				
IPv6 feature on 10 GbE	Supported	Supported	Supported	Supported
IPsec encryption on 1 GbE ports	Supported	Supported	N/A	N/A

- If both arrays are running HYPERMAX OS or PowerMaxOS, up to 250 RDF groups can be defined across all of the ports on a specific RDF director, or up to 250 RDF groups can be defined on 1 port on a specific RDF director.
- A port on the array running HYPERMAX OS or PowerMaxOS connected to an array running Enginuity 5876 supports a maximum of 64 RDF groups. The director on the HYPERMAX OS or PowerMaxOS side associated with that port supports a maximum of 186 (250 – 64) RDF groups.
- If both arrays are running HYPERMAX OS or PowerMaxOS, up to 250 RDF groups can be defined across all of the ports on a specific RDF director, or up to 250 RDF groups can be defined on 1 port on a specific RDF director.
- A port on the array running HYPERMAX OS or PowerMaxOS connected to an array running Enginuity 5876 supports a maximum of 64 RDF groups. The director on the HYPERMAX OS or PowerMaxOS side associated with that port supports a maximum of 186 (250 – 64) RDF groups.

Management tools

This chapter contains an overview of the tools that enable you to manage an SRDF environment.

Topics:

- [Solutions Enabler](#)
- [Unisphere](#)
- [SRDF/TimeFinder Manager for IBM i](#)
- [Mainframe management tools](#)

Solutions Enabler

SYMCLI commands are invoked from a management host, either interactively on the command line, or using scripts.

SYMCLI is built on functions that use system calls to generate low-level I/O SCSI commands. Configuration and status information is maintained in a host database file, reducing the number of enquiries from the host to the arrays.

Use SYMCLI to:

- Configure array software (for example, TimeFinder, SRDF, Open Replicator)
- Monitor device configuration and status
- Perform control operations on devices and data objects

Solutions Enabler also has a Representational State Transfer (REST) API. Use this API to access performance and configuration information, and provision storage arrays. It can be used in any programming environments that supports standard REST clients, such as web browsers and programming platforms that can issue HTTP requests.

Unisphere

Unisphere is a web-based application that provides provisioning, management, and monitoring of arrays.

With Unisphere you can perform the following tasks:

Table 10. Unisphere tasks

Section	Allows you to:
Home	View and manage functions such as array usage, alert settings, authentication options, system preferences, user authorizations, and link and launch client registrations.
Storage	View and manage storage groups and storage tiers.
Hosts	View and manage initiators, masking views, initiator groups, array host aliases, and port groups.
Data Protection	View and manage local replication, monitor and manage replication pools, create and view device groups, and monitor and manage migration sessions.
Performance	Monitor and manage array dashboards, perform trend analysis for future capacity planning, and analyze data. Set preferences, such as, general, dashboards, charts, reports, data imports, and alerts for performance management tasks.
Databases	Troubleshoot database and storage issues, and launch Database Storage Analyzer.
System	View and display dashboards, active jobs, alerts, array attributes, and licenses.
Events	View alerts, the job list, and the audit log.
Support	View online help for Unisphere tasks.

Unisphere also has a Representational State Transfer (REST) API. With this API you can access performance and configuration information, and provision storage arrays. You can use the API in any programming environment that supports standard REST clients, such as web browsers and programming platforms that can issue HTTP requests.

SRDF/TimeFinder Manager for IBM i

Dell EMC SRDF/TimeFinder Manager for IBM i is a set of host-based utilities that provides an IBM i interface to SRDF and TimeFinder.

This feature allows you to configure and control SRDF or TimeFinder operations on arrays attached to IBM i hosts, including:

- SRDF: Configure, establish, and split SRDF devices, including:
 - SRDF/A
 - SRDF/S
 - Concurrent SRDF/A
 - Concurrent SRDF/S
- TimeFinder:
 - Create point-in-time copies of full volumes or individual data sets.
 - Create point-in-time snapshots of images.

Extended features

SRDF/TimeFinder Manager for IBM i extended features provide support for the IBM independent ASP (IASP) functionality.

IASPs are sets of switchable or private auxiliary disk pools (up to 223) that can be brought online/offline on an IBM i host without affecting the rest of the system.

When combined with SRDF/TimeFinder Manager for IBM i, IASPs let you control SRDF or TimeFinder operations on arrays attached to IBM i hosts, including:

- Display and assign TimeFinder SnapVX devices.
- Execute SRDF or TimeFinder commands to establish and split SRDF or TimeFinder devices.
- Present one or more target devices containing an IASP image to another host for business continuance (BC) processes.

Access to extended features control operations include:

- From the SRDF/TimeFinder Manager menu-driven interface.
- From the command line using SRDF/TimeFinder Manager commands and associated IBM i commands.

Mainframe management tools

There are tools for managing SRDF configurations in a mainframe environment:

- Mainframe Enablers
- GDDR

Mainframe Enablers

Mainframe Enablers (MFE) is a suite of products for managing and monitoring Dell EMC storage systems in a mainframe environment. The entire suite consists of:

- SRDF Host Component for z/OS
- ResourcePak Base for z/OS
- Autoswap for z/OS
- Consistency Groups for z/OS
- TimeFinder SnapVX
- Data Protector for z/Systems(zDP)
- TimeFinder/Clone Mainframe Snap Facility
- TimeFinder/Mirror for z/OS
- TimeFinder Utility

In the context of SRDF, only the SRDF Host Component for z/OS, TimeFinder/Mirror for z/OS, plus these components of the ResourcePak for z/OS are relevant:

- SRDF/A Monitor
- WPA Monitor
- SRDF/AR

SRDF Host Component for z/OS

SRDF Host Component for z/OS is a z/OS subsystem for controlling SRDF processes and monitoring SRDF status using commands issued from a host. With the SRDF Host Component you can manage these SRDF variants:

- SRDF/S
- SRDF/A
- SRDF/DM
- SRDF/AR
- SRDF/CG
- SRDF/Star
- SRDF/SQAR

You can issue SRDF Host Component commands to both local and remote storage systems. Commands destined for remote storage systems are transmitted through local storage systems using SRDF links. Configuration and status information can be viewed for each device on each storage system that contains SRDF devices.

There are user interfaces for the SRDF Host Component for the batch commands and through the system console.

SRDF/A Monitor

SRDF/A Monitor is a facility for managing and monitoring SRDF/A operations. It is a component of the ResourcePak Base for z/OS. SRDF/A Monitor:

- Discovers storage systems that are running SRDF/A and monitors the state of the SRDF/A groups
- Collects and writes System Management Facility (SMF) data about the SRDF/A groups
- Optionally, calls a user exit to perform user-defined actions when it detects a change in the state of a SRDF/A group
- Optionally, invokes SRDF/A automatic recovery procedures to recover a dropped SRDF/A session

WPA Monitor

SRDF/A Write Pacing extends the availability of SRDF/A by enabling you to prevent conditions that can result in cache overflow. The SRDF/A Write Pacing Monitor, a component of the ResourcePak Base for z/OS, gathers information about write pacing activities in a storage system. The data is collected for each:

- SRDF/A group by the storage system
- SRDF device by the SRDF group and the storage system

The data includes:

- Changes in the ARMED state by device
- Total paced delay by device
- Total paced track count by device
- Changes in the ENABLED/SUPPORTED/ARMED/PACED state for the SRDF/A group
- Total paced delay for the SRDF/A group
- Total paced track count for the SRDF/A group

The WPA Monitor writes the collected information as SMF records.

SRDF/AR process management

SRDF/AR automates data copying across SRDF links to provide a logically consistent, restartable image of data at a remote (recovery) site. That image can be used should a disaster occur at the production site.

SRDF/AR automatically propagates the restartable image to the recovery site in a way that is transparent to the host application or database. The result is a series of consecutive data *consistency points* that you use as the basis for restarting host applications at the recover site.

The ResourcePak Base for z/OS and TimeFinder/Mirror for z/OS components of Mainframe Enablers provide commands to configure, manage, monitor, start, pause, restart, and stop SRDF/AR processes.

Geographically Dispersed Disaster Restart (GDDR)

GDDR automates business recovery following both planned outages and disaster situations, including the total loss of a data center. Using the PowerMax, VMAX All Flash, or VMAX architecture and the foundation of SRDF and TimeFinder replication families, GDDR eliminates any single point of failure for disaster restart plans in mainframe environments. GDDR intelligence automatically adjusts disaster restart plans based on triggered events.

GDDR does not provide replication and recovery services itself. Rather GDDR monitors and automates the services that other Dell EMC products and third-party products provide that are required for continuous operations or business restart. GDDR facilitates business continuity by generating scripts that can be run on demand. For example, scripts to restart business applications following a major data center incident, or resume replication following unplanned link outages.

Scripts are customized when invoked by an expert system that tailors the steps based on the configuration and the event that GDDR is managing. Through automatic event detection and end-to-end automation of managed technologies, GDDR removes human error from the recovery process and allows it to complete in the shortest time possible.

The GDDR expert system is also invoked to automatically generate planned procedures, such as moving compute operations from one data center to another. This is the gold standard for high availability compute operations, to be able to move from scheduled DR test weekend activities to regularly scheduled data center swaps without disrupting application workloads.

More information

This chapter shows where there is further information available on some of the subjects mentioned in other chapters. All documents are available from the Dell Technologies support web site (<https://www.dell.com/support/home>).

Topics:

- [Solutions Enabler CLI](#)
- [Unisphere](#)
- [Mainframe Enablers](#)
- [GDDR](#)
- [SRDF/TimeFinder Manager for IBM i](#)
- [SRDF/Metro vWitness](#)
- [SRDF Interfamily Compatibility](#)
- [Storage arrays](#)

Solutions Enabler CLI

Solutions Enabler SRDF Family CLI User Guide

Unisphere

Unisphere for PowerMax Product Guide

Unisphere for PowerMax Online Help

Unisphere for VMAX Online help

Unisphere for PowerMax REST API Concepts and Programmer's Guide

Unisphere for VMAX REST API Concepts and Programmer's Guide

Mainframe Enablers

SRDF Host Component for z/OS Product Guide

ResourcePak Base for z/OS Product Guide (contains information about SRDF/A Monitor, WPA Monitor, and SRDF/AR process management)

TimeFinder/Mirror for z/OS Product Guide (contains information about configuring, managing, and monitoring SRDF/AR)

AutoSwap for z/OS Product Guide

Consistency Groups for z/OS Product Guide

GDDR

GDDR for SRDF/Star Product Guide

GDDR for SRDF/Star with AutoSwap Product Guide

GDDR for SRDF/Star-A Product Guide

GDDR for SRDF/SQAR with AutoSwap Product Guide

GDDR for SRDF/A Product Guide

GDDR for SRDF/S with AutoSwap Product Guide

GDDR for SRDF/S with ConGroup Product Guide

SRDF/TimeFinder Manager for IBM i

SRDF/TimeFinder Manager for IBM i Product Guide

SRDF/Metro vWitness

SRDF/Metro vWitness Configuration Guide

SRDF Interfamily Compatibility

SRDF and NDM Interfamily Connectivity Information

Storage arrays

PowerMax Family Product Guide

VMAX All Flash Product Guide

VMAX 3 Product Guide

Symmetrix VMAX Family with Enginuity Product Guide